PROJECTS ■ THEORY ■ APPLICATIONS ■ CIRCUITS ■ TECHNOLOGY

NUTS AND VOLTS

www.nutsvolts.com
March 2011

EVERYTHING FOR ELECTRONICS

Green Power With A

COMPUTER CONTROLLED AIR/STEAM ENGINE

YOU CAN BUILD

✦ **Introducing CHIPINO**
The bridge between PICs and Arduinos.

✦ **Indoor Lightning**
Build a miniature
Van de Graaff Generator.

U.S. $6.50 CANADA $7.50

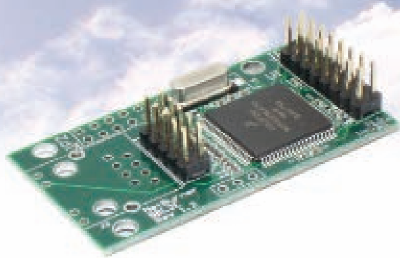
Low-cost Industrial
Serial to Ethernet Solutions

NetBurner
Networking in One Day!

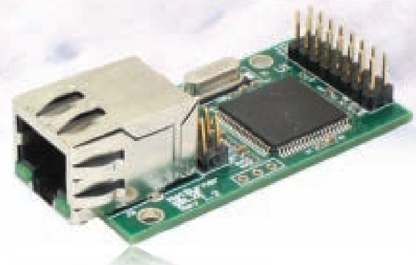
- Instantly network-enable any serial device
- No programming is required for serial to Ethernet application
- Customize to suit any application with a development kit

SBL2e Chip
2-port serial-to-Ethernet server with eight A/D converter inputs and optional SPI, I2C peripheral device support

SBL2e 200
2-port serial-to-Ethernet server with four A/D converter inputs, optional I2C peripheral support and 10-pin header



SBL2e 100
2-port serial-to-Ethernet server with four A/D converter inputs, optional I2C peripheral support and RJ45 connector



SBL2e
Serial-to-Ethernet server with RS-232 support

SBL2e XA
External server with up to four A/D converter inputs, up to eight digital I/O and up to two UARTS (one RS-232)

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

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

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

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

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

Integrate Touch Sensing Quickly and Easily

With Microchip's Range of Low Power, Low Cost Solutions

Microchip's mTouch™ Sensing Solutions allow designers to integrate touch sensing with application code in a single microcontroller, reducing total system cost.

Microchip Technology offers a broad portfolio of low power, low cost & flexible solutions for keys/sliders and touch screen controllers. Get to market faster using our easy GUI-based tools, free source code and low-cost development tools.

Capacitive Touch Keys and Sliders

- Extend battery life with eXtreme Low Power MCUs
  - Proximity sensing in less than 1 µA
- High noise immunity and low emissions
- Broad portfolio of MCUs lowers system cost
  - 8, 16 & 32-bit PIC® MCUs for Capacitive Touch
  - Integrated USB, Graphics, LCD, IrDA, CAN,
  - No external components
- With **Metal Over Cap technology** you can:
  - Use polished or brushed metal surfaces including stainless steel and aluminum
  - Sense through gloves
  - Create waterproof designs
  - Deploy Braille-friendly interfaces

Touch Screen Controllers

- Fully processed touch coordinates
- **Projected Capacitive technology**
  - Multi-touch enabling gestures
  - Low cost MCU implementation
  - Wide operating voltage: 1.8-5.5V
  - Low operating current 1.5 mA at 5V typical
- **Analog Resistive technology**
  - Lowest system cost, easy integration
  - Universal 4, 5 & 8-wire solution with on-chip calibration
  - I²C™, SPI, UART or USB interfaces
  - Low power "touch to wake-up" feature

GET STARTED IN 3 EASY STEPS

- Learn more at **www.microchip.com/mtouch**
- Download App Notes & royalty-free source code
- Order a development tool

Enhanced mTouch Capacitive Evaluation Kit - DM183026-2 (For keys & sliders)

Projected Capacitive Development Kit - DM160211

Analog Resistive Touch Screen Development Kit - DV102011

Equip your bench with LeCroy

Performance you need, at a price you can afford

Oscilloscopes Starting at $695
Logic Analyzer at $990
ArbStudio – Waveform Generator from $2,490

WaveAce™ Oscilloscopes
40 MHz – 300 MHz

WaveJet™ Oscilloscopes
100 MHz – 500 MHz

WaveSurfer® & Mixed Signal Oscilloscopes
200 MHz – 1 GHz

ArbStudio
Arbitrary and Digital Pattern Generators

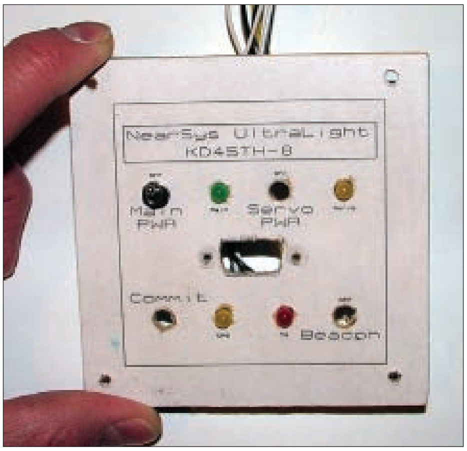
LogicStudio™
USB Logic Analyzer

LeCroy

www.lecroy.com • 1-800-5-LeCroy

Nuts & Volts
March 2011
www.nutsvolts.com

Page 67



Page 44

Page 10

Projects & Features

Departments

Columns

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.

DEVELOPING PERSPECTIVES

Sound Control

Designing and building circuits may be a solitary, silent endeavor, but once things are plugged in, it tends to get noisy. Electronic devices typically emit a variety of beeps, buzzes, and clicks. If you're working with amps and sound systems, you may be working with hundreds of watts of audio. Unfortunately, your family and even neighbors may not be as pleased with the noise your circuits created. Whether you're helping a friend configure a home theater environment or trying to keep your electronics experimentation from bothering your family, you need to know a little something about sound control.

The first thing you need to appreciate is the difference between sound shaping and sound isolation. The goal of sound shaping is to tune the acoustics of a room to minimize reflections and absorb certain ranges of frequencies to produce an acoustically flat environment free of echoes. The goal of sound isolation — in contrast — is to prevent sound from entering or leaving an area.

Full sound isolation – that is, soundproofing – is expensive and difficult. I've experienced only two fully soundproof rooms. One is a recording studio in Atlanta that floats on a bed of liquid mercury so that the low frequency vibrations from trucks on a nearby highway don't make it to the microphones. The second is a room used for audiology research in Boston. The 20′ x 30′ room — in the center of the top floor of a high-rise — is suspended by a specially dampened cable and pulley system.

If sound isolation is your goal, you don't have to float your room on a bed of mercury or suspend it in mid-air if you're willing to put up with less than perfect results. If you have the luxury of building from scratch, you can use thick fiberglass insulation and double wall construction, along with seamless sheets of rubber between the floor and foundation, and on the walls.

Note that not just any insulation or rubber sheeting will do — use materials rated for sound absorption. A commonly used acoustic metric is the Noise Reduction Coefficient (NRC). A higher number indicates greater absorption. For example, ordinary drywall has an NRC of about 0.05, compared with 1.00 for 2″ thick Owens-Corning 703 fiberglass boards.

In addition to insulation, you can also use special shock mounts to attach gypsum board or plywood to wall studs. Double-paned windows will reduce acoustic transfer to or from your neighbors or the street. Regardless of what you use, you'll find it most difficult to block the lower frequency sounds which are coupled through the foundation and frame of a building.

Sound shaping is less arduous and more easily achieved than sound isolation. If you're renting or otherwise can't tear down walls to add insulation and sheet rubber sound barriers, you can mount cotton or fiberglass blankets in doorways, along walls, and on the inside of windows. Thick, heavy curtains will help absorb the high frequency audio transmitted through your windows. Add a few throw rugs to minimize reflections from the floor. Thick moving blankets with grommets are affordable, easily removed sound absorbers that you can use to cover a doorway, window, or a problematic (paper-thin) wall. I've had good luck with 'acoustic' moving blankets from Movers Supplies ($16; **www.moverssupplies.com**).

If you have a modest budget and don't like the blanket wall approach, you can step up to sound panels which are simply framed fiberglass and fiber wool. The fiberglass or fiber wool attenuates audio and therefore reflections. Standard panels are 2″ or 4″ thick, and 2′ x 4′ or 2′ x 6′ in size. I've purchased fiberglass sound panels from ATS Acoustics (**www.atsacoustics.com**) which sells fully assembled fiberglass panels and DIY supplies. A 2′ x 2′ panel with 2″ of fiberglass and a burlap covering sells for about $30. Of course, you can go nuts with coverings. The same panel in microsuede is double the price. Moreover, other vendors sell similar panels with even fancier coverings in the $1,000 range.

A popular choice for sound control materials among musicians and audiophiles is acoustic foam. The most popular – and most expensive – option for acoustic foam is Auralex Acoustics (**www.auralex.com**). In certain settings — such as a music studio — acoustic foam looks nicer than panels. It's important to note that acoustic foam isn't egg carton foam. Acoustic foam absorbs much more sound – that is, has a higher NRC – than the dimpled foam used to pack eggs.

One problem with acoustic foam is that it's normally attached to walls and ceilings with glue. Auralex foam ships with a permanent adhesive – not necessarily what you want to use if you're renting. A better option is DAP 18354 Seal 'N Peel Removable Caulk which is available in a 10.1 ounce caulking gun tube. I used the caulk to mount a few dozen sheets of Auralex foam in my workshop. Unfortunately, the fumes from the curing caulk were so strong — even with good ventilation — that

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
Jon Williams
Paul Verhage
John Molnar
David Williams
The Chipaxe Team

CIRCULATION DIRECTOR
Tracy Kerley
subscribe@nutsvolts.com

SHOW COORDINATOR
Audrey Lemieux

MARKETING COORDINATOR WEBSTORE
Brian Kirkpatrick
sales@nutsvolts.com

WEB CONTENT
Michael Kaudze
website@nutsvolts.com

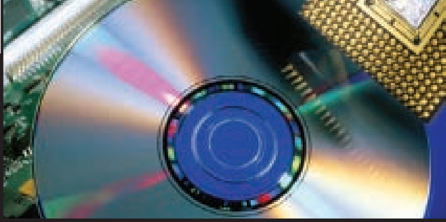
ADMINISTRATIVE ASSISTANT
Debbie Stauffacher

PRODUCTION/GRAPHICS
Shannon Christensen

I couldn't use the area for two days.

Finally, don't forget about lining your doors with closed cell foam tape. Consider adding weather stripping, or at least a door sweep, to the bottoms of doors leading to your work area. An added benefit of all this insulation and sealing is that you may save money on your heating and cooling bills.

Of course, there are also downsides to sound proofing and shaping. The first is fire hazard issues. Covering your walls with flammable cotton quilting may not be the safest approach to sound shaping. If fire is an issue, consider using all-fiberglass quilts instead of cotton products. They're considerably more expensive than the cotton variety ($240 for a 4 x 6' sheet, Amazon), but you won't have to worry about a spark from your workbench setting off a fire. **NV**



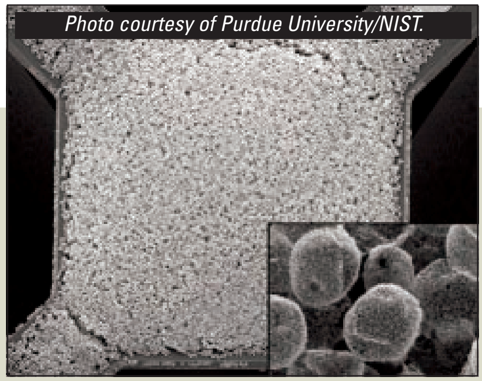
■ BY JEFF ECKERT

ADVANCED TECHNOLOGY

MEDICAL BREATHALYZER DEVELOPED

Let's say the Dudley Do-Right of the state Highway Patrol spots you pulling away from your favorite watering hole and decides to find out if you've had a couple too many. He makes you blow into a breathalyzer and subsequently announces the findings: "Well, sir, your blood alcohol level is 0.07 percent, so you're okay. But your acetone levels are above normal, so you need to be checked for diabetes."

Photo courtesy of Purdue University/NIST.



■ New breath-analysis sensor enables medical diagnosis.

Unlikely, perhaps, but not impossible given a breakthrough in breath-analysis technology recently announced by Purdue University (**www.purdue.edu**) and the National Institute of Standards and Technology (NIST, **www.nist.gov**). The researchers have developed a sensor that is capable of rapidly detecting "biomarkers" in a person's respiration at levels as low as parts per billion — 100 times better than previous technologies. According to Purdue's Prof. Carlos Martinez, "People have been working in this area for about 30 years but have not been able to detect low enough concentrations in real time. We solved that problem with the materials we developed, and we are now focusing on how to be very specific, how to distinguish particular biomarkers."

In early research, the technology has shown an ability to detect tiny amounts of acetone which is a biomarker for diabetes. "We are talking about creating an inexpensive, rapid way of collecting diagnostic information about a patient," Martinez said. "It might say, 'there is a certain percentage that you are metabolizing a specific compound indicative of this type of cancer,' and then additional, more complex tests could be conducted to confirm the diagnosis."

The sensor was fabricated by coating a template of micron-size polymer particles with a surface of much smaller metal oxide nanoparticles, thereby increasing the porosity of the sensor films and hence their sensitivity. Unfortunately, it may be as much as 10 years before such breathalyzers become a reality, "in part because precise standards have not yet been developed to manufacture devices based on the approach." But it's at least a first step. ▲

DIAMONDS ARE A SWITCH'S BEST FRIEND

Most existing nano/microelectromechanical (NEMS/MEMS) switches are based on silicon or metal materials which involves such drawbacks as poor mechanical, chemical, and thermal stability, plus unsatisfactory levels of reliability and durability. By contrast, diamond is said to be the "ideal material" for NEMS/MEMS because of its high elastic modulus, mechanical hardness, thermal conductivity, and variable electrical conductivity from insulator to conductor. Until lately, it has been extremely difficult to fabricate the required structures in single-crystal diamond. However, researchers at Japan's National Institute for Materials Science (**www.nims.go.jp**) recently announced development of a technique for batch fabrication of suspended structures of single-crystal diamond for use in such systems. Based on the process, they created what is said to be the world's first single-crystal NEMS switch which displays improved reliability, life span, switching speed, and other characteristics.

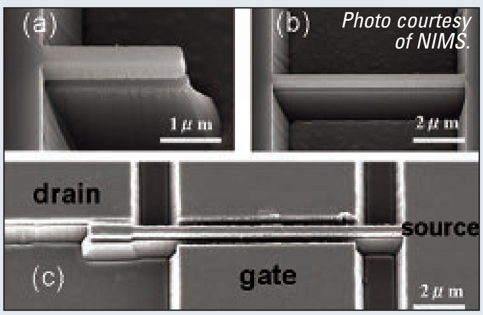


Photo courtesy of NIMS.

■ SEM images of single-crystal diamond structures: (a) cantilever; (b) bridge; and (c) three-terminal NEMS switch.

The switch — a transistor-like structure with three electrodes — offers low leakage current and power consumption of less than 10 pW. It also has exhibited stable operation at temperatures as high as 250°C. According to NIMS, GigaHertz switching operation can be expected, and further developments should lead to the development of "various chemical, physical, and mechanical sensors." ▲

NEW SOLAR SPEED RECORD SET

It's not such a new or highly advanced technology per se, but it is worth noting that on January 7, 2011, the Sunswift IVy — a project of Australia's University of New South Wales — set a new speed record for a solar-powered vehicle, averaging 88.8 km/h (55.2 mph) over a 500 m (640 ft) course, in both directions. The car used about 1,050W in the achievement, which earned it a spot in the *Guiness Book of World Records*. For this run, its batteries were removed so as to compete purely on the basis of energy efficiency. The previous record of 78 km/h (48.5 mph) was set way back in 1988 by GM's Sunraycer, which traveled on about 1,500W. At one point, the IVy's solar array was generating 1,340W, but clouds prevented the car from operating at peak power long enough to complete any faster runs. For details, more pics, vids, etc., visit **www.sunswift.com**. ▲

■ Sunswift IVy on the road during the World Solar Challenge.

COMPUTERS AND NETWORKING

FIRST ANDROID 3.0 TABLET

Rumors of the death of the traditional PC may be premature, but you wouldn't know it judging by this year's Consumer Electronics Show. There, the total number of desktop and laptop computers receiving the 2011 Best of Innovations honor totalled zero, and PC introductions were pretty much ho-hum. However, more than 80 tablet computers were displayed at the four-day show, many of which are expected to bite the dust before the end of the year.

The hot operating system these days is Google's Android 3.0 (a.k.a., Honeycomb) — the first system designed from the ground up for tablets, and one of the hot new tablets is Motorola's XOOM. The machine is based on a NVIDIA Tegra-2 dual-core processor with each core running at 1 GHz, and a 10.1 inch widescreen HD display. It also features a front-facing 2 Mp camera for video chats over Wi-Fi or 3G/4G LTE, plus a rear-facing 5 Mp camera that captures video in 720p HD.

■ Motorola's Android 3.0-based XOOM tablet, on the Verizon Wireless network.

It promises console-like gaming performance on a 1280 x 800 display and features a built-in gyroscope, barometer, e-compass, accelerometer, and adaptive lighting. It also includes Google Maps 5.0 with 3D interaction and delivers access to Google eBooks and apps from Android Market®.

The 3G version is already on the market, with an upgrade to 4G LTE coming soon. (In case you're wondering, the LTE designation — meaning "long term evolution" — means that Verizon can call its network 4G even though it doesn't meet the International Telecommunication Union's definition.) As of this writing, the official price tag has not been announced, but U.K. vendor Handtec is taking preorders at £599.99 ($935.00) excluding VAT. Info is available at **http://phones.verizonwireless.com/xoom/**. ▲

ONE TOUGH DRIVE

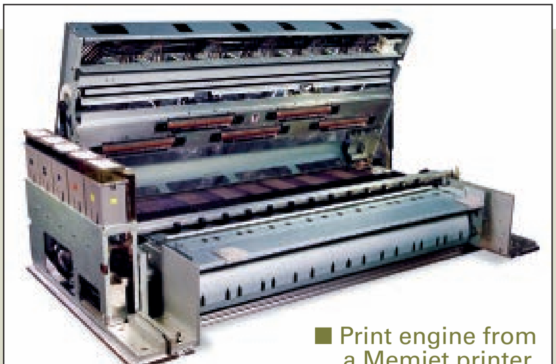
It may look pretty much like any other external drive, but the ioSafe Rugged Portable (**www.iosafe.com**) may be the most disaster-proof storage device you'll ever see. In terms of the enclosure, it features "Full Metal Jacket™" technology, meaning a CNC machined box made from a solid billet of aluminum or titanium alloy that is crush resistant up to 5,000 lb (Ti) or 2,500 lb (Al). The drive is suspended in all six axes of motion to withstand drops of 20 ft (SSD version) or 10 ft (HDD) on to concrete per MIL-STD-810G Method 516.5. Plus, it offers data loss protection in up to 10 ft (Al) or 30 ft (Ti) of water for three days. You also get protection against immersion in things like diesel fuel and hydraulic fluids, and a barrier against sand, dust, salt fog, and more.

■ Kick it, dunk it, shoot it; the ioSafe drive doesn't care.

To prove its invincibility, at the 2011 CES show, the company actually blasted one with five shotgun rounds, and it survived. So, if you work in extreme environments (or are just a world-class klutz), this may be for you. Prices start at a relatively modest $149.99 for a 250 GB HDD but stretch up to $3,499.99 for a 512 GB titanium-encrusted SSD version. ▲

CIRCUITS AND DEVICES

THIRD OPTION FOR COLOR PRINTING



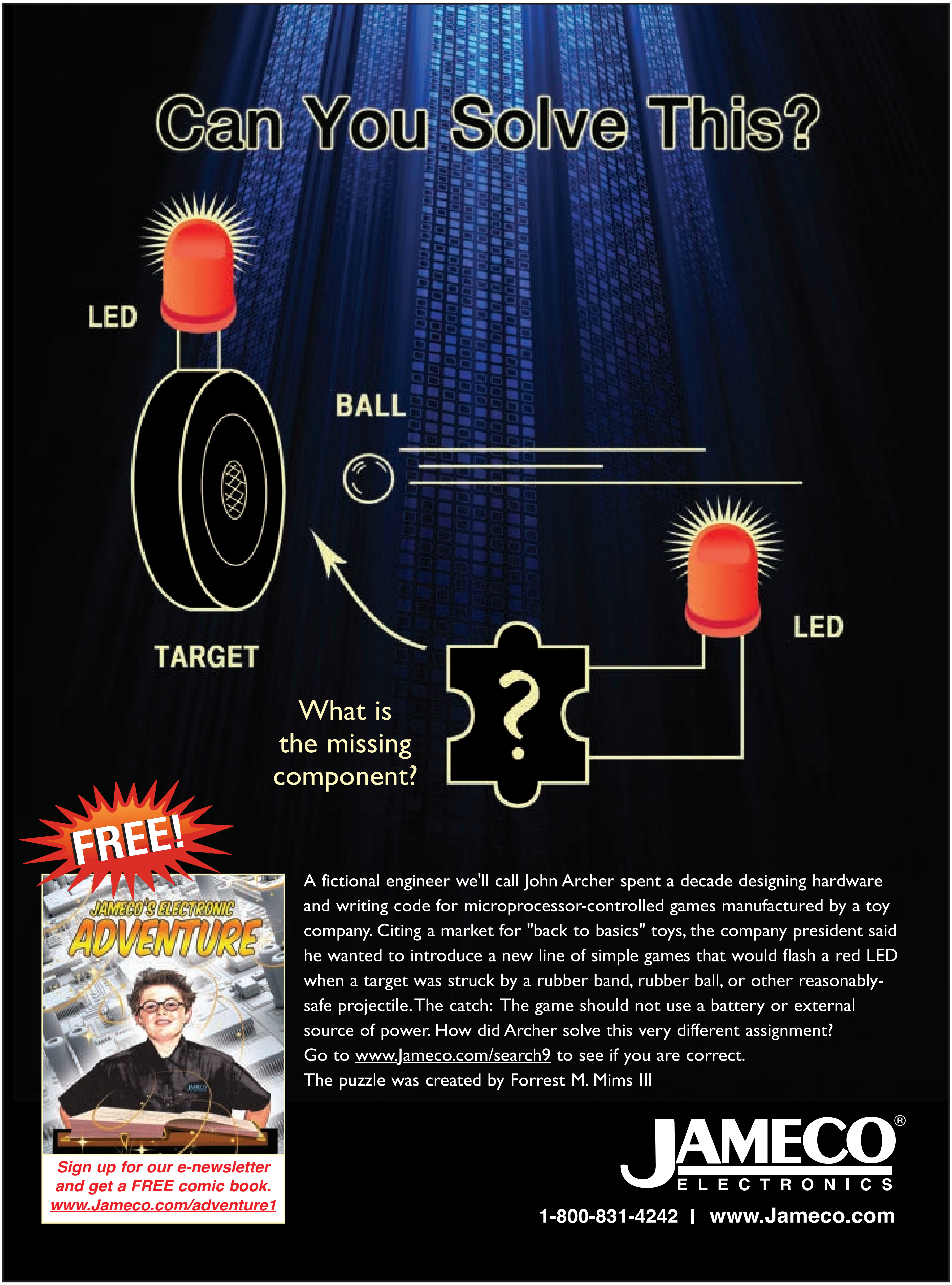
■ Print engine from a Memjet printer.

For years, your only choice for color printing has been between laser and inkjet (well, okay, there are dye-sub printers, but you probably don't want to go there). However, Memjet (**www.memjet.com**) is now offering a third path, providing the "world's fastest color printers" at a reasonable price. A Memjet printer works pretty much like an inkjet, but instead of having a printhead zip back and forth over the paper, it puts 70,000 ink nozzles on a stationary print head that's 8.77 inches wide. This translates into 60 pages per minute at 1600 x 800 resolution. Indications are that the company won't be marketing office printers itself but rather will license partners to develop and market their own Memjet-powered products. The first one (from Lenovo) is slated to hit the markets in China soon, with other partners jumping into the fray in India and Taiwan. A US partner has not yet been named, but stay tuned. Prices are expected to be between $500 and $600, with ink running about five cents per page. ▲

YOU MAKE ME SICK

So, there's this woman named Limor, an MIT engineering graduate who creates things, "some of which are beautiful and interesting." She also calls herself Lady Ada for some unspecified reason, but that seems to be unimportant. One of her creations that falls into the "interesting" category is the Bedazzler, a "do-it-yourself handheld LED-incapacitator." It seems that Limor attended a conference where the geniuses at the US Department of Homeland Security demonstrated the Dazzler "seasick flashlight," developed at a cost of $1 million. Concluding that the price seemed a little steep, she did some research into the concept and came up with a version you can assemble yourself for less than $250. Best of all, you can download all of the documentation at **www.ladyada.net/make/bedazzler**/. But beware, because she warns: "Yes, this project does indeed cause nausea, dizziness, headache, flash blindness, eye pain, and (occasional?) vomiting. So, don't use it on your friends or pets." It seems odd that the admonition doesn't include family members or coworkers, but maybe you already make them sick. ▲

■ The Bedazzler LED nausea light.


Can You Solve This?
LED
BALL
TARGET
LED
What is the missing component?
FREE!
JAMECO'S ELECTRONIC ADVENTURE
Sign up for our e-newsletter and get a FREE comic book.
www.Jameco.com/adventure1
A fictional engineer we'll call John Archer spent a decade designing hardware and writing code for microprocessor-controlled games manufactured by a toy company. Citing a market for "back to basics" toys, the company president said he wanted to introduce a new line of simple games that would flash a red LED when a target was struck by a rubber band, rubber ball, or other reasonably-safe projectile. The catch: The game should not use a battery or external source of power. How did Archer solve this very different assignment?
Go to www.Jameco.com/search9 to see if you are correct.
The puzzle was created by Forrest M. Mims III
JAMECO®
ELECTRONICS
1-800-831-4242 | www.Jameco.com

■ BY JON WILLIAMS

Wii WILL ROCK YOU

The design and development of effective man-machine interface devices is an incredibly difficult task. And, yes, I'm speaking from experience — a long time ago in a galaxy far away, I used to help design sprinkler timers that were supposed to be simple enough for anyone to use. The good news for those of us that are hardware and firmware junkies is that the multi-billion dollar gaming industry has the resources to create really cool widgets, and with a snip snip here and a bit of code there, we can take advantage of their efforts.

For me, one of the coolest and most cleverly designed gaming accessories on the market today is the Wii Nunchuk controller. This little beauty fits very comfortably in your hand (see **Figure 1**) and has an analog X/Y joystick (eight-bit values), two active-low buttons, and a three-axis accelerometer (10-bit values). The best part? The Wii Nunchuk costs about $20 and is easy to connect to any microcontroller using I²C. Of course, I'm going to show you how to connect it to the Propeller.

If you do an Internet search on "Wii Nunchuk," you will find dozens of sites with instructions on how to connect it to the Arduino — I guess this is what Fred Eady meant by the "Arduino effect." As I'm not an Arduino user, I was initially confused when adapting code samples written for that platform; it all came down to the way Arduino does I²C (versus the rest of the world): the Arduino requires the slave ID to be shifted right by one bit. No worries, with a little reading on how the Arduino does things I got it figured out and using the Nunchuk is, in fact, every bit as easy as everyone says. Let's jump in.

GETTING CONNECTED: HARDWARE

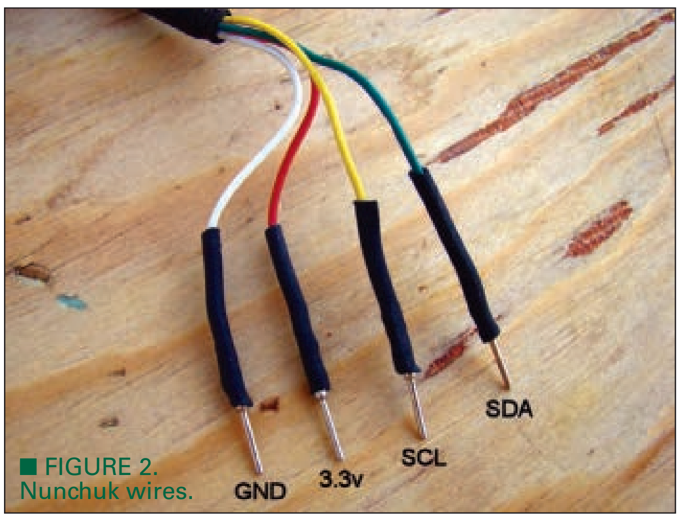
I don't actually own a Wii console; again, I bought the Nunchuk when I learned it uses I²C communications. I've done lots of work with I²C. For my projects, then, I simply hacked off the connector and soldered male pins to the wires, protecting each with heat shrink tubing. If you do have a console and don't want to take this action, you can find many companies offering adapters.

The Nunchuk has four wires for connecting to a host:

White: Ground
Red: 3.3V
Yellow: SCL (pin 28 on the Propeller)
Green: SDA (pin 29 on the Propeller)

Figure 2 shows my hacked Nunchuk cable and labels for each of the wires. The Propeller uses P28 and P29 to connect to the boot EEPROM, so that's what I'm using with my Nunchuk. These pins are free after the Propeller boots up so it makes


■ FIGURE 1. Nunchuk in hand.



■ FIGURE 2. Nunchuk wires.

good sense to use them and not consume additional I/O pins. If you don't use pins 28 and 29, then you'll need to add pull-ups on the SCL and SDA lines as shown in **Figure 3**.

GETTING CONNECTED: SOFTWARE

As I stated above, the Nunchuk uses I²C which is a very simple protocol to implement and use on the Propeller. You'll find a few I²C objects in the ObEx; for this project, I used one of my own — jm_i2c_basic.spin — as it handles the essentials of I²C and that's all that is required by the Nunchuk. Once there is a connection to the I²C buss, the Nunchuk is initialized with a simple sequence:

```
Start
$A4
$40
$00
Stop
```

The sequence only needs to be sent once. From this point, we can request an update from the Nunchuk with the following sequence:

```
Start
$A4
$00
Stop
```

The request sequence is followed by a short delay. I've seen suggestions for 200-300 microseconds; in my program, I bumped this up to 500 microseconds. After the delay, the Nunchuk registers can be read into your program:

```
Start
$A5
Byte 0 (read)
Byte 1 (read)
Byte 2 (read)
Byte 3 (read)
Byte 4 (read)
Byte 5 (read)
Stop
```

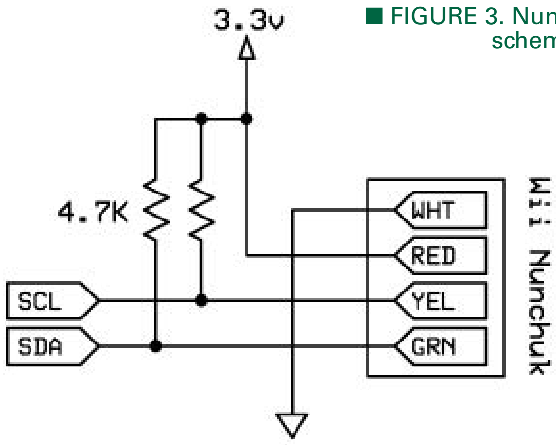
At this point, we have the Nunchuk registers but they're not ready for use. Each has been encoded and must be decoded as follows:

```
wiibyte := wiibyte ^ $17 + $17
```

Figure 4 shows the location of each element in the Nunchuk packet. Note that the lower bits of each accelerometer channel are encoded into the button's byte (byte 5). Okay, let's convert these Nunchuk access sequences to Spin. In order to make the code easy to use across many projects, it should be its own object. The first method of the object will connect to the I²C buss pins of our choice and initialize the Nunchuk:

```
pub init(sclpin, sdapin)

  i2c.init(sclpin, sdapin)

  i2c.start
  i2c.write($A4)
  i2c.write($40)
  i2c.write($00)
  i2c.stop
```

As you can see, this follows the sequence described



■ FIGURE 3. Nunchuk schematic.

above to the letter.

Before moving on, though, let me cover a detail about objects that is a little easy to miss: If a program declares the same object in multiple places, the compiler will only load one copy of the code, saving valuable EEPROM and RAM space. As the hub allows only one cog access to the hub RAM at any given time, the [Spin] code can be shared across multiple objects and cogs. With the Nunchuk initialized, we can scan it at will. This is a little more involved, but not any more difficult:

```
pub scan | idx

  waitcnt(clkfreq / 2_000 + cnt)
  i2c.start
  i2c.write($A4)
  i2c.write($00)
  i2c.stop

  waitcnt(clkfreq / 2_000 + cnt)
  i2c.start
  i2c.write($A5)
  repeat idx from 0 to 4
    raw[idx] := i2c.read(i2c#ACK)
  raw[5] := i2c.read(i2c#NAK)
  i2c.stop

  repeat idx from 0 to 5
    raw[idx] := (raw[idx] ^ $17) + $17
```

To prevent "over clocking" the Nunchuk, there is a 500 µs delay before each of the two major sections. This

■ FIGURE 4. Nunchuk data packet.

	7	6	5	4	3	2	1	0
Byte 0	Joystick X (7:0)							
Byte 1	Joystick Y (7:0)							
Byte 2	Accelerometer X (9:2)							
Byte 3	Accelerometer Y (9:2)							
Byte 4	Accelerometer Z (9:2)							
Byte 5	Acc Z (1:0)		Acc Y (1:0)		Acc X (1:0)		Z_b	C_b

delay is derived from the system clock frequency so that it works at any crystal and PLL setting. The top section moves the address pointer to $00 which commands the Nunchuk to update the registers with current readings from its sensors. In most I²C devices, there would simply be a restart sent before the values are read, but the Nunchuk needs a little time to read the sensors; hence, the full stop and the short delay.

After the second delay, the read command ($A5) is issued and six bytes are read from the Nunchuk. As with other I²C devices, all bytes except the last are followed by an ACK from the master; after the final byte, the master responds with NAK. The last step is to decode the packet bytes as described earlier.

To make using the Nunchuk object painless, I created routines for each of the elements. For example, this method returns the X joystick value:

```
pub joyx

  return raw[0]
```

Simple, right? Yes! Too much so? I don't think so. I'm a big believer of writing code so that it's easy for others to read and put to use in their own projects. Let's say we wanted to use the Nunchuk joystick to steer a robot; we might do something like this:

```
  wii.scan
  steering := wii.joyx
```

You see, code like this is easy for just about anyone to understand and modify. Again, this should be our goal — especially as it helps us when going back to code we wrote ages ago. (Admit it, you've opened code you wrote a long time ago and scratched you head! We all have.)

A 10-bit accelerometer value is achieved by using the channel byte (bits 9:2) and two bits (1:0) from byte 5 (button's byte) of the packet. Here's the method that returns the X accelerometer reading:

```
pub accx

  return (raw[2] << 2) | ((raw[5] >> 2) & %11)
```

To resolve the 10-bit accelerometer value, the channel byte is shifted left by two bits, and then bits 1 and 0 are extracted from byte 5 with shifting and masking. The same approach — shifting and masking — is used to extract button status:

```
pub btnz

  return (!raw[5] & %10) >> 1
```

Note that the bits in byte 5 are inverted before masking to convert the active-low pressed status returned by the Nunchuk to active-high. This lets us write code like this:

```
  if wii.btnz
    ' do something for button press
```

By inverting the status, we can treat a button press as true or false in Spin. And there we have it: Just a tiny bit of code nets us a super cool controller for the Propeller. That said, let's not get ahead of ourselves; we really need to test an object before deploying it.

To be a bit more serious than usual, this point cannot be made strongly enough. I frequently see code posted in the ObEx that has not been tested, or is "land locked" to specific I/O pins rendering it almost useless as a general-purpose object. Be kind to yourself and fellow ObEx users. Make sure your code works before posting it, include a demo that shows off your hard work, and make sure that others can use your code without having to "fix" it in any way. Okay, I'll climb down off the soap box now ....

Putting the Nunchuk object to work is very easy; the first step — as always — is to declare an object of Nunchuk type:

```
obj

  wii   : "jm_nunchuk_ez"
```

At the top of the program, we need to call the init() method to assign the I²C buss and activate the Nunchuk. In my programs, SCL and SDA are constants for the EEPROM pins 28 and 29:

```
  wii.init(SCL, SDA)
```

As we've seen above, a call to the scan() method gets fresh data from the Nunchuk which we can access through the individual method calls. **Figure 5** shows the terminal output of my test program: jm_nunchuk_ez_test.spin. Nothing to write home about here, we're simply scanning the Nunchuk every 50 ms and updating a terminal window with the new values.

Now, I'm betting more than a few of you are thinking, "Why is his Nunchuk dead-on center and mine is off by a few counts?" I confess: I cheated. That screenshot is actually from a second version of the program that fixes the joystick offset.

This problem with inexpensive joysticks has been irritating me for a while now and I finally decided to sit down and do something sensible about it. After a few minutes at my white board, the solution turned out to be unbelievably simple. It's a matter of scaling based on which side of the idle point our new input falls. At start-up, we need to read the idle values from each joystick axis:

```
  wii.scan
  xidle := wii.joyx
  yidle := wii.joyy
```

To correct for the offset, we will call the calibrate() method:

```
pub calibrate(in, idle, range) | mid

  mid := range >> 1

  if (in == idle)
    result := in
  elseif (in < idle)
    result := mid * in / idle
  else
    result := mid * (in-idle) / (range-idle)
+ mid
```

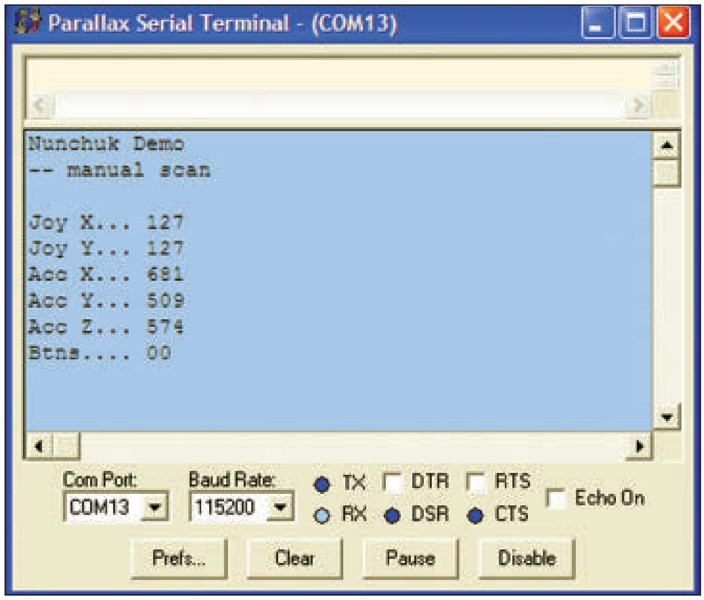
We need to pass the new (uncalibrated) reading, the idle position from start-up, and the range of the joystick. The desired midpoint is calculated by dividing the range in half. The new input is compared to the original idle position and scaled accordingly. The net effect is that values from the "short" side of the joystick are stretched back to the normal, while inputs on the "long" side of the joystick are compressed to normal. This method is much better at auto-centering than my previous efforts and there is no loss of range. Yes, we do lose a little resolution on the "short" side but that cannot be helped.

What will you control with the Wii Nunchuk? I know it's only March, but I'm already thinking about Halloween and using the Nunchuk to program movements into animatronic props. Are games your thing? The Nunchuk was designed for a game console so that makes perfect sense.

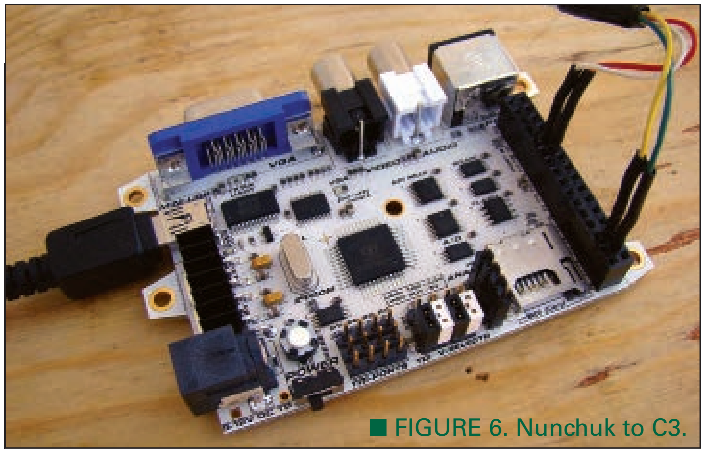
In **Figure 6**, I've connected my Nunchuk to the new Parallax C3 (Credit Card Computer) and started — just barely started, that is — experimenting with the Propeller graphics object. The C3 is a really cool little board that is packed with an incredible list of features. You can expect to see me put it to use in future columns. At the moment, part of my crawling up the learning curve of DipTrace is creating a default "shield" layout for the C3.

One caveat for game creators: The Nunchuk has a fixed slave ID ($A4), so you cannot put two of them on the same SDA pin. If you're using the jm_nunchuk_ez.spin object that only accesses the Nunchuk when commanded, you can share the clock line (SCL) without problems; the data (SDA) pins, however, will have to be separated.

Please download the source files because I've included more code than I could cover here. You'll find a simple servo control demo, a "move the dot on the TV" graphics demo, and a fully automated version of the Wii driver that will sample at the rate you specify. You'll see a few of the techniques discussed in my January article put to work in that version. Until next time, keep spinning and winning with the Propeller! **NV**



■ FIGURE 5. Nunchuk demo output.



■ FIGURE 6. Nunchuk to C3.

JON "JONNYMAC" WILLIAMS
jwilliams@efx-tek.com

PARALLAX, INC.
Propeller C3
www.parallax.com

GADGET GANGSTER
Propeller Platform kits and accessories
www.gadgetgangster.com

NUNCHUCK ADAPTER
www.solarbotics.com/products/31040

■ BY FRED EADY

FLY HIGH WITH THE NEW SPECTRUM ACE ECS

My first serious embedded control program was hosted by an Intel 8048. In those days, the addition of external EPROM and SRAM ICs was not an option if one wanted to effectively use the 8048's CPU. There was no C compiler for it. I had no choice but to resort to programming my 8048 using native assembler mnemonics. There were no commercially-available third-party debug tools either. With that, I was forced to find innovative ways to debug my 8048 programs as an Intel-manufactured 8048 emulator went for about $10,000. As my embedded control program needs grew, so did my preference of microcontrollers. It didn't take long before the 8751 became my primary embedded core.

Like the Boeing B-52 Stratofortress, the classic architecture of the original 8051 is still flying in a number of variants to this day. For instance, the 8751 I used was an 8051 core with a built-in UV erasable EPROM. These days, the EPROM has been replaced by Flash on the new parts, but their core is still modeled around the original 8051.

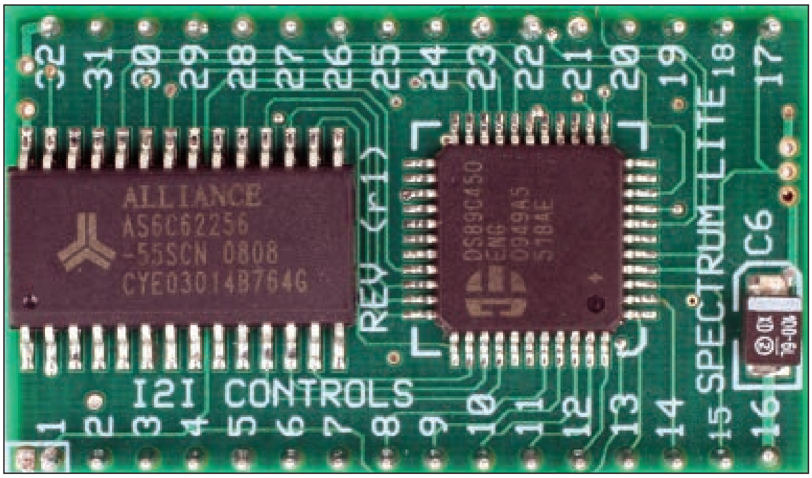
The embedded control system we're about to discuss does not require an IDE (Integrated Development Environment), a compiler, or a hardware programmer/debugger. Just as today's B-52s fly on upgraded aero platforms and instrumentation, the unique standalone SPECTRUM ACE embedded control system we're about to explore is based on an updated version of the original 8051 microcontroller core.

THIS AIN'T THE APRIL ISSUE

I'm not fooling. Deploying any of the SPECTRUM ACE single-board computer modules does not require the services of an IDE, compiler, or external hardware programmer/debugger. All you need to initially seed, debug, and deploy a SPECTRUM ACE-based application is a HyperTerminal session. The compilation, hardware programming, and debugging duties fall into the realm of ALEC. ALEC is not a lab technician or one of the go-to guys in the programming pool. ALEC — which is short for Advanced Language for Embedded Control — is a complete operating system that is embedded within every SPECTRUM ACE single-board computer module.

I know what you're thinking. Here we go with yet another so called "operating system" that under the covers is nothing more than a fancy interpreter. ALEC is not an execution unit. Nor is it a language interpreter that simply runs programs. Together, any SPECTRUM ACE single-board computer module coupled with ALEC form a complete embedded computing system. HyperTerminal is the default console. However, if you can read it or enter data with it, that can also act as an ALEC console.

In addition to overseeing and executing the application, the SPECTRUM ACE and ALEC team up to provide communications and mass storage services. Application debugging is also an integral part of the SPECTRUM ACE/ALEC system package. ALEC is as easy to learn and use as it is powerful. In a nutshell, source code that is passed to the SPECTRUM ACE/ALEC-based embedded system via a HyperTerminal text transfer is compiled and loaded into the DS89C450 microcontroller's Flash or RAM by ALEC for execution. ALEC allows the downloaded source code to be edited, debugged, and monitored for performance within the SPECTRUM ACE/ALEC system environment. Thus, there is no need for an external compiler or hardware programmer/debugger. As you get to know ALEC, you'll find that you can also employ its services to eliminate having to load your application code using a



■ PHOTO 1. The DS89C450 microcontroller and the 32K x 8 SRAM IC are shown here. The RTCC/microcontroller crystals, 74HC573 D-type octal transparent latch, and DS28DG02 real time clock are on the opposite side of the printed circuit board.

HyperTerminal session.

Just like the DS89C450 microcontroller, ALEC is based on familiar concepts. If you've ever written a C or Basic program, you can write a program using ALEC. As a matter of fact, if you've ever written a program using any programming language, you can write a program using ALEC. We'll be using ALEC to exercise the SPECTRUM ACE hardware as we move through this discussion. So, let's get ready to do some programming by examining the smallest of the SPECTRUM ACE family: the SPECTRUM LITE.

THE SPECTRUM LITE

The I2I Controls' SPECTRUM ACE family is based on the Dallas/Maxim DS89C450 microcontroller. There is a really good reason for that. In terms of MIPS (Millions of Instructions per Second), it is a stretch for the original 8051 to crank out one MIPS with a 10 MHz clock. The 8051 MIPS bottleneck is due to the fact that to perform a single machine cycle, it takes 12 clock cycles. The DS89C450 can perform one machine cycle per clock cycle. Thus, the DS89C450 can execute instructions up to 12 times faster. Typically, the 8051 pin-compatible DS89C450 microcontroller will run 10 times faster than the original 8051 core, which equates to one MIPS per MegaHertz for the DS89C450.

Another key design point supporting the DS89C450 is its open address and data busses. Access to the DS89C450 microcontroller's address and data lines enable it to be attached to multiple banks of external memory devices and memory mapped peripherals.

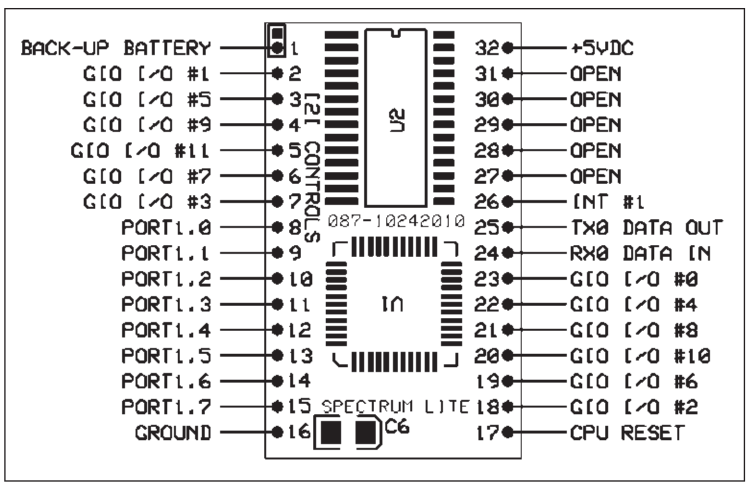
The SPECTRUM LITE posing in **Photo 1** is a 32-pin single-board computer module that is equipped with a DS89C450, 32K of eight-bit SRAM, and 32K of eight-bit program Flash. The program Flash is integral to the DS89C450. **Figure 1** exposes the SPECTRUM LITE's I/O layout.

The key to efficient embedded design is to tailor the microcontroller and its I/O subsystem to the job. The SPECTRUM LITE is small in stature and is intended for smaller embedded applications that require a fair amount of compute power.

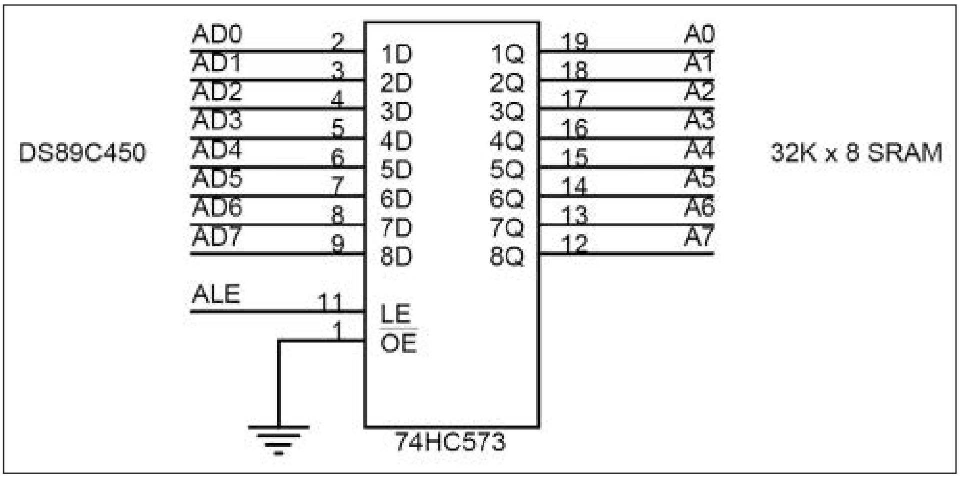
As far as the SPECTRUM LITE is concerned, the programmatic things you will do with ALEC will be more easily understood if you understand the fundamentals that stand behind the SPECTRUM LITE hardware. There is no mystery behind the basic operation of its hardware as it is vintage 8051. At the core of every SPECTRUM ACE variant is a DS89C450 microcontroller, a 74HC573 D-type octal transparent latch, and a DS28DG02 real time clock. The SPECTRUM LITE's DS89C450 address, data, and control lines are dedicated to accessing the SPECTRUM LITE's 32K x 8 SRAM IC with some help from the 74HC573.

Port 0 of the DS89C450 doubles as the least significant byte of the SRAM address and the eight bits of SRAM data. The 74HC573 octal D-type transparent latch's D inputs electrically parallel the DS89C450 microcontroller's port 0 and are responsible for latching the least significant byte of address information for use by the 32K x 8 SRAM. The capture of the address information is controlled by the status of the 74HC573's ALE (Address Latch Enable) input which is under the control of the DS89C450. Once the SRAM has all of the address information on its address pins, the microcontroller can read or write that particular SRAM memory location via its data bus. The classic 74HC573 configuration I've just described is shown in **Figure 2**.

The DS89C450 microcontroller's port 1 is an eight-bit bidirectional I/O port that is multiplexed with a set of alternate functions. In the case of the SPECTRUM LITE, the ALEC port 1 operator allows the ALEC programmer to use port 1 as an I/O port. PWM bit streams can be generated on any of the port 1 pins using the ALEC PWM operator.



■ FIGURE 1. There's more to the DS28DG02 real time clock than its name implies. Its RTC I/O pins are really hefty and can directly drive 20 mA loads.



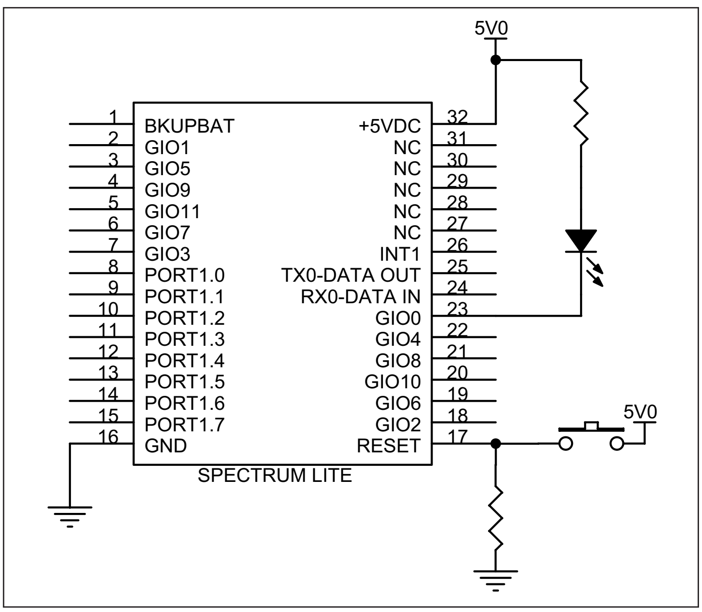
■ FIGURE 2. This is a classic 74HC573 implementation. The DS89C450 microcontroller throws out the address information first and latches it for the SRAM's address bus using the ALE control line.

Nope. It's not a Basic program. ALEC uses the line numbers as an internal pointer mechanism to locate the code associated with the line number. Note that lines 10 and 20 contain multiple ALEC statements delimited by colons. If you take a look at **Figure 4**, you'll see that I assigned the variable *iodata* to the least significant byte of the real time clock's I/O subsystem. The I/O pin's input/output directions are controlled using the value of variable *iodir*. A zero in an *iodir* bit position associates the *iodir* bit with an output pin. The really cool thing about the *iodata* and *iodir* variables is that I simply declared them as I went along without the need for a C #define or Basic DIM statement. That's ALEC at work.

The DS28DG02 real time clock is the SPECTRUM LITE's primary I/O subsystem. The I/O pin direction and I/O pin logical state are controlled by reading and writing to I/O pin registers inside of the real time clock IC. Rather than spout words that you can read for yourself in the datasheet, let's use ALEC to blink an LED according to **Figure 3**:

```
;   iodata = 0x120
;   iodata.7 - 0x80 - PIO7
;   iodata.6 - 0x40 - PIO6
;   iodata.5 - 0x20 - PIO5
;   iodata.4 - 0x10 - PIO4
;   iodata.3 - 0x08 - PIO3
;   iodata.2 - 0x04 - PIO2
;   iodata.1 - 0x02 - PIO1
;   iodata.0 - 0x01 - PIO0 - toggle this one
10 iodata = 0x120 : iodir = 0x122
20 rtc(iodata)=0  : rtc(iodir)=0
30 rtc(iodata)= 0x01
40 delay(1000)
50 rtc(iodata)  = 0x00
60 delay(1000)
70 jump 30
```

■ FIGURE 3. This is extremely simplified. However, all we're interested in is where the LED we want to blink fits into this circuit.



The *rtc* operator is part of ALEC and is used to manipulate the real time clock's internal registers. After I assigned variable names to the lower eight bits of the clock's I/O subsystem in line 10, I zeroed the associated DS28DG02 clock registers in line 20.

The actual LED blinking begins with line 30 which sets the least significant bit of the eight-bit PIO I/O register. Each bit in the PIO I/O register is directly related to a physical I/O pin. Thus, setting the least significant bit causes I/O pin GIO0 to assume a logically high state.

Delay is yet another ALEC operator. Each unit of the delay argument is equivalent to 100 µS. So, that means that line 40 and line 60 are 0.100 second delays. I think you can find your way through the rest of the LED blinker code. You may be asking how do I blink an LED with the other four-bit DS28DG02 real time clock I/O port. No worries. Let's blink an LED with ALEC using I/O pin PIO11:

```
;   iodata = 0x121
;   iodata.4-7 – N/A
;   iodata.3 - 0x08 - PIO11 - toggle this one
;   iodata.2 - 0x04 - PIO10
;   iodata.1 - 0x02 - PIO9
;   iodata.0 - 0x01 - PIO8
10 iodata = 0x121  : iodir = 0x123
20 rtc(iodata)=0  : rtc(iodir)=0
30 rtc(iodata)= 0x08
40 delay(1000)
50 rtc(iodata)  = 0x00
60 delay(1000)
70 jump 30
```

Once you're able to blink an LED, you know how to manipulate the microcontroller module's I/O pins. When you know how to manipulate the SPECTRUM LITE's I/O pins, you can do things much more complex than blinking an LED. For instance, you can do things like driving an industry standard HD44780-based LCD.

THE SETUP

120h	SRAM	R/W	PIO output state (PIO0 to PIO7).
121h	SRAM	R/W	PIO output state (PIO8 to PIO11).
122h	SRAM	R/W	PIO direction (PIO0 to PIO7).
123h	SRAM	R/W	PIO direction (PIO8 to PIO11).

■ FIGURE 4. This is an excerpt from the DS28DG02 real time clock memory map. These are its I/O port addresses.

We're going to use the same clock I/O pins we flashed the LED with to build our HD44780-based LCD interface. The circuit is graphically depicted in **Figure 5**. There are rules we need to follow to initialize the LCD. So, let's begin our ALEC program just as we would with any other programming language. We'll write some code that follows the rules necessary to bring the LCD online:

```
; LCD SETUP
100 lcd_cntl = 0x120 : lcd_data = 0x121
102 rtc(lcd_cntl)=0 : rtc(lcd_data)=0 :
     rtc(0x122)=0 : rtc(0x123)=0
104 E = 0x80 : RS = 0x40
106 rtc(lcd_cntl)=rtc(lcd_cntl)|E
              ; set E high
108 cmd=0x02   : gosub 400
              ; dummy write 4 bit
110 cmd=0x28   : gosub 400
              ; 4 bit / 2 line
112 cmd=0x0C   : gosub 400
              ; display on w/cursor
114 cmd=0x06   : gosub 400
              ; increment display
116 cmd=0x01   : gosub 400
              ; clear display
118 cmd=0x00   : gosub 400
              ; dummy write 4 bit
```

Line 100 can be verified against the table and graphical data contained within **Figures 4** and **5**. We have assigned the LCD data bus to the four-bit real time clock I/O port. All 12 of the clock's I/O pins are set as outputs and initialized to zero in line 102. To make the code a bit easier to follow, I've declared bit map variables for the LCD's E and RS pins in line 104. Line 104 is optional as we can simply toggle the E and RS bits using AND/OR operators without having to identify them. The remaining lines of code are native HD44780 commands which are outlined in the comments.

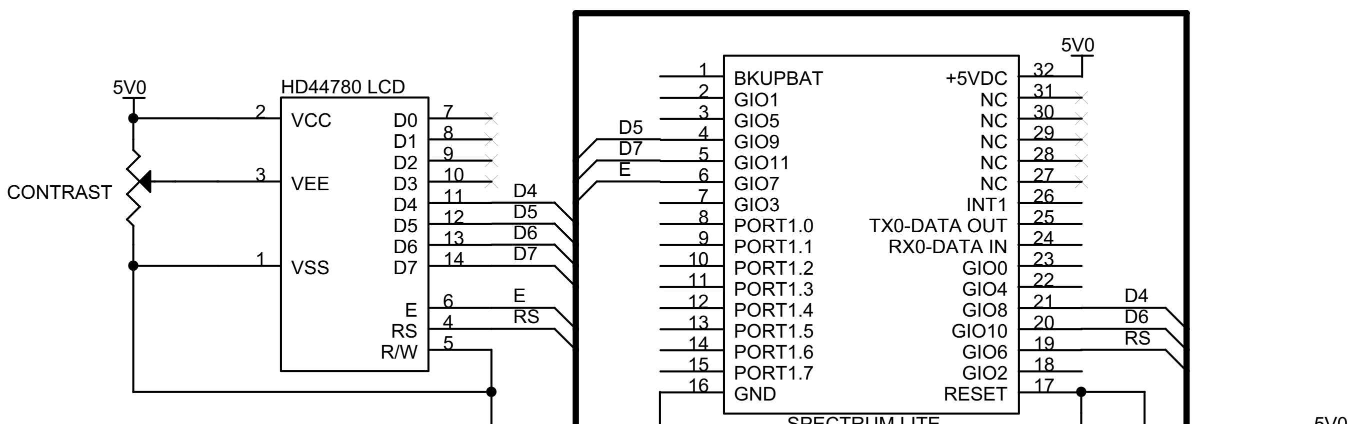
Note that each HD44780 command is followed by a GOSUB operator. The subroutine at line 400 performs a portion of the actual command write operation:

```
; SEND COMMAND SUBROUTINE
400 delay 100
402 rtc(lcd_cntl)=rtc(lcd_cntl) & 0xBF
       ;binary 10111111-RS is LOW for command
404 rtc(lcd_data)=(cmd & 0xF0)/0x10 : gosub 500
406 rtc(lcd_data)=cmd & 0x0F : gosub 500
408 return
```

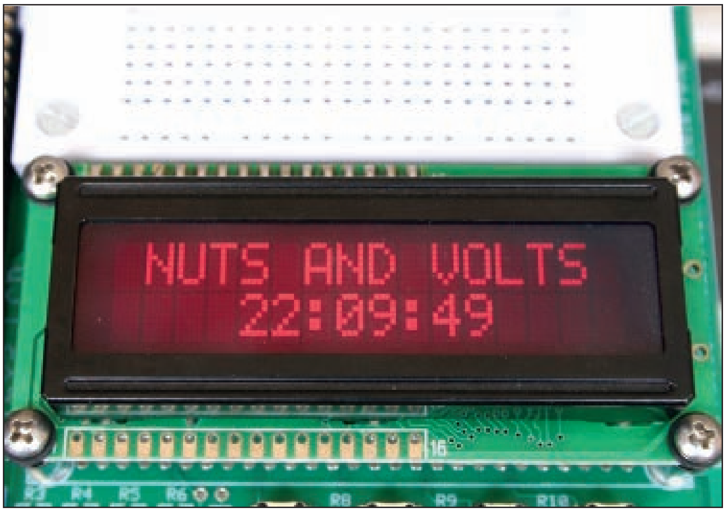
The command send routine does no more than send a byte of command data in two four-bit transfers. The HD44780 chipset transfers data on the state of the E pin. The subroutine at line 500 majors in toggling the E pin:

```
; TOGGLE E SUBROUTINE
500 rtc(lcd_cntl)=rtc(lcd_cntl) | E
       ; set E logically HI
502 rtc(lcd_cntl)=rtc(lcd_cntl) & NOT(E)
       ; take E logically LOW
504 rtc(lcd_cntl)=rtc(lcd_cntl) | E
       ; set E logically HI
506 return
```

My logical use of ALEC line numbers would be approved by Mr. Spock and the ship's computer. If you've been keeping tabs as Spock would, you have mentally noted that the 200 and 300 series of line numbers have not be discussed as of yet. Keeping with logic, the 100 series of line numbers physically preceed the 200 series of line numbers. Look back at the LCD SETUP code and



■ FIGURE 5. Learning to toggle those DS28DG02 real time clock I/O pins using a simple LED has opened the door to more complex bit twiddling with an LCD.



■ PHOTO 2. This particular red LCD is part of the I2I SPECTRUM ACE evaluation board. The evaluation board also has a home port for the SPECTRUM LITE module. Don't worry. Before we're done, we'll visit every home port on the evaluation board.

you'll find that the LCD initialization routine falls into the main application code which resides within the confines of the 200 series of line numbers:

```
; MAIN LOOP
200 num_chars=16 : $(1)=" NUTS AND VOLTS " :
    gosub 300
; Set cursor line 2
202 cmd=0xC0 : gosub 400
204 p. [s]  u.(s,2),"    ",!H,":",!M,"
    :",!S,"     "
206 num_chars=16 : $(1)=$O : gosub 300
208 jump 202
```

The main loop prints a 16 character message on line 1 of the LCD, followed by the time of day on the second line of the LCD. Line 200 sets up the subroutine at line 300 to print a total of 16 characters from string $(1) to the first line of a 2 x 16 LCD. Once the cursor is set to the second line of the LCD in line 202, the code in line 204 loads the special ALEC output string $O with the current time.

The $O string is a port of ALEC and is there to eliminate having to manually allocate a buffer area for the output string. Here's how to translate line 204:

```
p. [s]        - output to special string $O
u.(s,2)       - output using supressed leading
                zeros and 2 characters
!H            - current real time clock hours
!M            - current real time clock minutes
!S            - current real time clock seconds
```

If you include the spaces in the text area of line 204, you will count a total of 16 text characters which is passed to the subroutine at line 300 at the beginning of line 206.

As you can see, the subroutine at line 300 displays the contents of the string $(1):

```
; 4 BIT LCD DATA DRIVER SUBROUTINE
300 rtc(lcd_cntl)=rtc(lcd_cntl) | RS
     ; set RS logically high
302 for char_ptr=1 to num_chars
     ; loop num_chars times
304 char_hi=(asc($(1),char_ptr) & 0xF0) / 0x10
     ; get high nibble of char
306 char_lo=(asc($(1),char_ptr) & 0x0F)
     ; get low nibble of char
308 rtc(lcd_data)=char_hi : gosub 500
     ; send high nibble to display
310 rtc(lcd_data)=char_lo : gosub 500
     ; send low nibble to display
312 next char_ptr
     ; do it again
314 return
```

ALEC can be self-commenting if you choose variable names that humans can easily relate to. Offhand, I would say the only lines of code that need additional explanation are 304 and 306. Actually, we need only address line 304 to understand line 306. The ALEC code in line 304 pulls the numeric value of an ASCII character from the $(1) array that is being pointed to by the *char_ptr* variable. The rest of the four-bit LCD driver subroutine is obvious to the most casual observer as are the contents of **Photo 2**.

ALEC IN MOTION

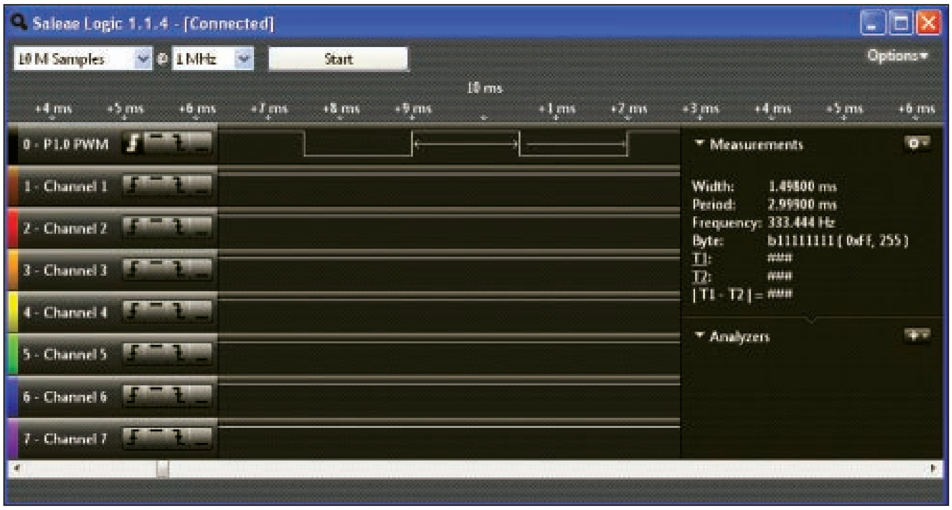
Recall that the SPECTRUM LITE's port 1 has the ability to host PWM signals. Well, this is another area in which ALEC excels. The ALEC PWM instruction can operate in both console and execution modes. Console mode involves using HyperTerminal to issue the PWM instruction and associated arguments. In console mode, the PWM command is carried out in real time.

Placing the PWM instruction inside of an ALEC program constitutes issuing the instruction in execution mode. There are many ALEC operators and instructions that can be issued in both console and execution modes. I use console mode to tweak instructions and variables that will ultimately be inclued in an ALEC application.

The ALEC PWM instruction is very easy to understand and use. Here's that instruction syntax:

```
PWM A,B,C,D
Where:
       A = Selected Port 1 I/O Pin (0-7)
       B = Number of PWM clock cycles HIGH
       C = Number of PWM clock cycles LOW
       D = Number of  PWM cycles to generate
       Each PWM clock cycle is equivalent to
       8.138 µS.
```

If you desire to drive a hobby servo to the center of its rotation, you must apply a high-going 1.5 mS PWM pulse followed by no more than about 40 mS of low-going PWM pulse. Doing the PWM cycle math (1.5 mS / 8.138 µS) yields 184 PWM clock cycles for every 1.5 mS of PWM pulse width. Here's the PWM instruction to issue in console or execution mode to obtain a set of 1.5 mS pulses:



■ SCREENSHOT 1. Equal amounts of time logically high and logically low will produce a square wave. As you can see in the shot, we're within a couple of frog hairs of our target of 1.5 mS.

```
PWM 0,184,184,1000
```

In that the number of logically high PWM cycles is equal to the number of logically low PWM cycles, our PWM instruction has generated a square wave. I tied my Saleae Logic analyzer to the SPECTRUM LITE's P1.0 pin and kicked off the PWM instruction in console mode. As you can see in **Screenshot 1**, our resultant PWM cycle is close enough for government work.

Now that we have a handle on the ALEC PWM instruction, consider this bit of ALEC PWM prose:

```
10 a = 184 : b = 184
20 pwm 0,a,b,1
30 a– : b++
31 delay(130)
40 if(a = 123)then 60
50 j.20
60 pwm 0,a,b,1
70 a++ : b–
71 delay(130)
80 if(a = 246) then 20
90 j.60
```

Line 10 contains a pair of ALEC fast variables. Fast variables (A-Z) are preallocated within ALEC and as their name implies, they are accessed and rendered faster than a user-generated variable. Line 20 should be of no surprise. The gist of the PWM program is to drive a hobby servo between the 1.0 mS minimum and 2.0 mS maximum pulse widths.

The code is really easy to understand. Just follow the logic and do the PWM cycle math. You'll see that the high-going pulse is swept up and down between 1.0 mS (123) and 2.0 mS (246).

ALEC embodies a bunch of neat shortcuts. The *j.* shortcut stands in for *jump* and *goto*. Note also that ALEC supports auto increment (++) and auto decrement (–) operators.

COOL STUFF

The SPECTRUM LITE is just one of the available SPECTRUM ACE single-board computer modules. There are more capable SPECTRUM ACE family members and we will explore them all. ALEC is an adventure all by itself. I'll cover the appropriate ALEC instructions and operators as we uncover them with our study of the SPECTRUM ACE hardware.

So, what do you think? Can you handle writing some really serious embedded control code by simply applying power and connecting to your pc's serial port? Let's find out. **NV**

Fred Eady can be reached at fred@edtp.com.

SOURCE

SPECTRUM ACE Single Board Computer Modules
SPECTRUM ACE Evaluation Board
ALEC
I2I Controls
www.i2icontrols.com

IT'S SPRINGTIME AT RAMSEY!

Four-Mode Keyless Entry Test Set NEW!

- ✔ **Troubleshoot vehicular keyless entry and wireless remote control systems!**
- ✔ **Detects and verifies key fob to vehicle signals as well as vehicle to key fob signals!**
- ✔ **Separate visual indicators for the presence of 315/433MHz, 125kHz, 20kHz and IR signals!**
- ✔ **Can also test virtually any wireless IR/RF control and building access systems!**
- ✔ **Can even test household and home entertainment IR remote controls for the presence of IR signal output!**

Ahh!!... the conveniences of today's technology in our modern world! Voice recognition, LED's instead of incandescent bulbs, on-board computers, on-board hard drives, automatic parallel parking, automatic radar cruise control, and of course, wireless remote controls! They make it so simple, just have the "key" (called a key fob) somewhere in your pocket or purse, get near the vehicle, it knows that you are there! Touch the door handle and the vehicle unlocks. Get in and touch the start button and the vehicle starts. You have yet to use a key through the whole process! And don't forget all the wireless controls for your house lights, building access and entertainment systems. They're so great… until they don't work!

Just like the days of "plugs, points, and condenser" are over, so are the days of having the hardware store grind out a spare key for your car! Now when your keyless access system doesn't work, you need to accurately detect what part of the system is malfunctioning. This could be anything from a dead battery in the key fob, a "brain-dead" key fob, to malfunctioning sensors, antennas, or other system components in the vehicle. The WCT3 is designed for both the car dealer service shops as well as the consumer. Until now there was no way to determine where the system was failing. Please note that the WCT3 simply verifies the generation of the control signals. Indication of signal presence is not an indication the encoded data is valid, nor is it a reader of that code, so don't worry, this will not help anyone steal your car!

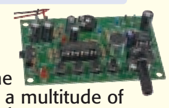
First, let's cover a few basics about vehicular keyless entry. In general, (not all systems are created equal), the vehicle itself generates a signal at 125 kHz or 20kHz. This is the signal that is used to "talk" to your individual key fob. Upon receiving the signal, your key fob "returns" a 315MHz signal uniquely encoded with an identification code and unlock command. If the embedded codes of the vehicle and your key fob match, you're in! Once you have "unlocked" the vehicle, and are inside the vehicle, the presence of your key fob is detected in the same way when the "start" button is pressed. If the codes match, the vehicle can be started. Some manufacturers also use Infrared (IR) signals in their key fobs to add additional user control functions to the vehicle. In that case, the key fob generates a modulated IR signal that is received by the vehicle's IR detectors placed throughout the perimeter of the vehicle.

Testing your system is easy. To test the complete 125 kHz/315 MHz communications path just stand close to the vehicle with the WCT3 and your key fob in hand. Press the test button and the WCT3 will detect and display the presence of the vehicle's 125kHz/20KHz signal and, if they "handshake", will also detect and display the presence of your key fob's 315MHz return signal. You can independently test key fob only signals (panic, lock, trunk, etc.) by holding the key fob near the WCT3, pressing the test button, and pushing the function button on the key fob. The same functionality testing can be done with IR key fobs. The modulated IR signal is detected and will illuminate the IR test LED on the test set. If you know a few "secrets" you can also see if the tire pressure sensors/transmitters are generating signals or the built-in garage door opener in your rear view mirror is transmitting a signal! But the WCT3's uses go beyond the automotive world. The majority of building wireless access systems also utilize 125 kHz. Just hold the test set near the building access sensor and the WCT3 will detect the 125 kHz signal. That will help you troubleshoot door access locations that are not working. It gets even better... you can use the WCT3 to test virtually any other 315 MHz, 433 MHz, 125kHz, 20kHz and IR wireless control system to verify generation of a signal. We should rename this "the handy-dandy, universal, wireless remote control tester"!

The WCT3 test set is housed in a compact 2.25" x 4.6" x 9" case and is powered by a standard 9VDC battery. The test set is available as a do-it-yourself hobby kit or factory assembled and tested. For the kit builder, the WCT3 contains both SMT and through-hole components, with 170 solder points. If you're a car dealer, independent service shop, or simply an owner of a newer vehicle with keyless entry, or have wireless entertainment controls you can't afford not to have a WCT3!

WCT3	**Four-Mode Keyless Entry Test Set Kit**	**$59.95**
CCT3WT	**Four-Mode Keyless Entry Test Set, Factory Assembled & Tested**	**$99.95**

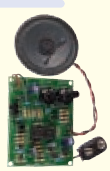
Digital Voice Changer



This voice changer kit is a riot! Just like the expensive units you hear the DJ's use, it changes your voice with a multitude of effects! You can sound just like a robot, you can even ad vibrato to your voice! 1.5W speaker output plus a line level output! Runs on a standard 9V battery.

MK171 Voice Changer Kit $14.95

Steam Engine & Whistle



Simulates the sound of a vintage steam engine locomotive and whistle! Also provides variable "engine speed" as well as volume, and at the touch of a button the steam whistle blows! Includes speaker. Runs on a standard 9V battery.

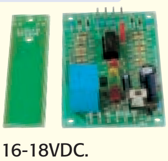
MK134 Steam Engine & Whistle Kit $11.95

Laser Trip Senser Alarm

True laser protects over 500 yards! At last within the reach of the hobbyist, this neat kit uses a standard laser pointer (included) to provide both audible and visual alert of a broken path. 5A relay makes it simple to interface! Breakaway board to separate sections.

LTS1 Laser Trip Sensor Alarm Kit $29.95

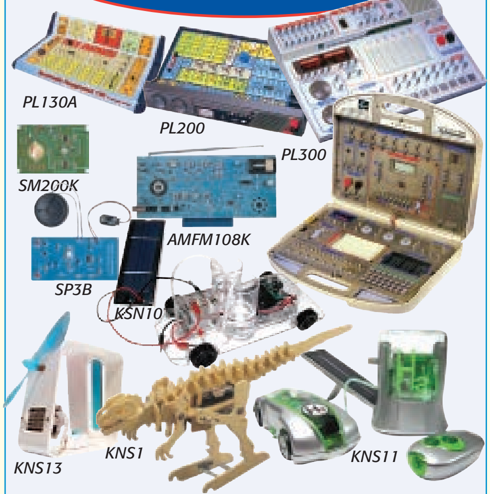
Liquid Level Controller



Not just an alarm, but gives you a LED display of low, middle, or high levels! You can also set it to sound an alarm at the high or low condition. Provides a 2A 240VAC rated relay output. Runs on 12-14VAC or 16-18VDC.

K2639 Liquid Level Controller Kit $21.95

The Learning Center!



Beginners To Advanced... *It's Fun!*

- ✔ **Learn and build!**
- ✔ **130, 200, 300, & 500 in one electronic labs!**
- ✔ **Practical through hole and SMT soldering labs!**
- ✔ **Integrated circuit AM/FM radio lab!**
- ✔ **Fuel Cell, Solar Hydrogen, and Bio-Energy labs!**
- ✔ **Beginner's non-soldering kits!**

For over 3 decades we've become famous for making electronics fun, while at the same time making it a great learning experience. As technology has changed over these years, we have continued that goal!

PL130A Gives you 130 different electronic projects together with a comprehensive learning manual describing the theory behind all the projects.

PL200 Includes 200 very creative fun projects and includes a neat interactive front panel with 2 controls, speaker, LED display and a meter.

PL300 Jump up to 300 separate projects that start walking you through the learning phase of digital electronics.

PL500 The ultimate electronics lab that includes 500 separate projects that cover it all, from the basics all the way to digital programming.

SP3B Whether young or old, there's always a need to hone your soldering skills. Either learn from scratch or consider it a refresher, and end up with a neat little project when you're done!

SM200K Move up to Surface Mount Technology (SMT) soldering, and learn exactly how to solder those tiny little components to a board!

AMFM108K We not only take you through AM and FM radio theory but we guide you through IC's. When you're done you've built yourself an IC based AM/FM radio that works great!

KNS10 With a reversible PEM fuel cell that combines electrolysis and power conversion into a single device you end up building your own fuel cell car! Learn tomorrows technology today!

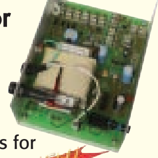
KNS11 Learn alternative fuel technology while you build your own H-Racer car and refueling station!

KNS13 Convert ethanol alcohol to run a PEM fuel cell and watch it all work in front of your eyes!

KNS1 A great beginner's kit for the dinosaur enthusiast in the family, young and old! A wooden hobby kit that teaches motor and gear driven operation that requires no soldering.

PL130A	**130-In-One Lab Kit**	**$39.95**
PL200	**200-In-One Lab Kit**	**$69.95**
PL300	**300-In-One Lab Kit**	**$89.95**
PL500	**500-In-One Lab Kit**	**$219.95**
SP1A	**Through Hole Soldering Lab**	**$9.95**
SM200K	**SMT Practical Soldering Lab**	**$22.95**
AMFM108K	**AM/FM IC Lab Kit & Course**	**$34.95**
KNS10	**Fuel Cell Car Science Kit**	**$82.95**
KNS11	**H-Racer & Refueling Station Kit**	**$144.95**
KNS13	**Bio-Energy Fuel Cell Kit**	**$129.95**
KNS1	**Tyrannomech Motorized Kit**	**$17.95**

Vintage Battery Eliminator



Collectors come across some great deals on antique battery-powered radios, but how to power them is a real problem. Many classic radios operated on batteries only, and in many cases a series of three batteries for each radio were required!

The new ABCE1 Battery Eliminator gives you an easy way to replace all these batteries with a simple household AC power connection and resurrect your vintage antique radios! Provides "A" filiment, "B" plate, and "C" control grid supplies, which are all isolated from each other. Complete with aluminum case. Runs on 110-240VAC.

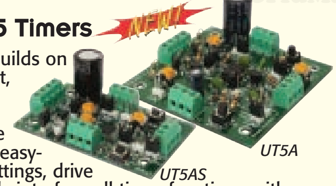
ABCE1 Vintage Radio Battery Elim Kit $199.95

Digital Message System

The third generation of Ramsey digital voice storage kits! We started with the latest digital voice storage technology. It provides up to 8 minutes of digital storage at a frequency response up to 3.5 KHz. (Total message time and frequency response is dependant on selected internal sampling rate.) Once recorded, messages are available for playback on-demand or automatic continuous looping. Standard RCA unbalanced line level output is provided for easy connection to any amplifier, amplified speaker, mixer, or sound system. In addition, a standard 4-8 ohm speaker output is provided to directly drive a monitor speaker. Can be remote controlled via 3-wire BCD with our interface options. Check www.ramseykits.com for all options!

DVMS8 Digital Voice Message 8Ch Kit $99.95
DVMS8WT Assembled DVMS8 $149.95

Ultimate 555 Timers



This new series builds on the classic UT5 kit, but takes it to a whole new level! You can configure it on the fly with easy-to-use jumper settings, drive relays, and directly interface all timer functions with onboard controls or external signals.

All connections are easily made though terminal blocks. Plus, we've replaced the ceramic capacitor of other timer kits with a Mylar capacitor which keeps your timings stable over a much wider range of voltages! Available in through hole or surface mount versions! Visit www.ramseykits.com for version details.

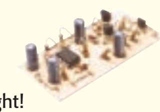
UT5A Through Hole 555 Timer/Osc Kit $24.95
UT5AS SMT 555 Timer/Osc Kit $26.95

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 Receiver Kit $89.95

Voice Activated Switch



Voice activated (VOX) provides a switched output when it hears a sound. Great for a hands free PTT switch or to turn on a recorder or light! Directly switches relays or low voltage loads up to 100mA. Runs on 6-12 VDC.

VS1 Voice Switch Kit $9.95

OBDII *CarChip Pro*

The incredible OBDII plug-in monitor that has everyone talking! Once plugged into your vehicle it monitors up to 300 hours of trip data, from speed, braking, acceleration, RPM and a whole lot more. Reads and resets your check engine light, and more!

8226 CarChip Pro OBDII Monitor-Asmb $79.00

RF Preamplifier

The famous RF preamp that's been written up in the radio & electronics magazines! This super broadband preamp covers 100 KHz to 1000 MHz! Unconditionally stable gain is greater than 16dB while noise is less than 4dB! 50-75 ohm input. Runs on 12-15 VDC.

SA7 RF Preamp Kit $19.95

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

Doppler Direction Finder

Track down jammers and hidden transmitters with ease! 22.5 degree bearing indicator with adjustable damping, phase inversion, scan and more. Includes 5 piece antenna kit. Runs on 12VDC vehicle or battery power.

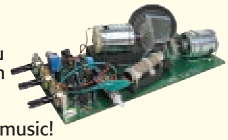
DDF1 Doppler Direction Finder Kit $169.95

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

Laser Light Show



Just like the big concerts, you can impress your friends with your own laser light show! Audio input modulates the laser display to your favorite music! Adjustable pattern & speed. Runs on 6-12VDC.

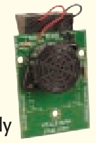
LLS1 Laser Light Show Kit $49.95

Retro Nixie Tube Clock

Genuine Nixie tubes popular in the 50's brought back in one of the neatest digital clocks around today! Hand made teak maple base, 12/24 hour format, soft fade-out, auto-dim, and a crystal time base at 20ppm! Tube kits also available.

IN14TM Teak Maple Nixie Clock Kit $329.95

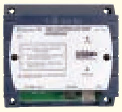
Water Sensor Alarm



This little $7 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

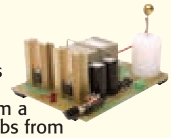
USB DMX Interface



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

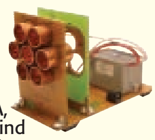
HV Plasma Generator



Generate 2" sparks to a handheld screwdriver! Light fluorescent tubes without wires! This plasma generator creates up to 25kV at 20kHz from a solid state circuit! Build plasma bulbs from regular bulbs and more! Runs on 16VAC or 5-24VDC.

PG13 HV Plasma Generator Kit $64.95

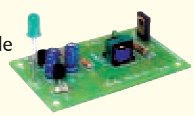
Air Blasting Ion Generator



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

Tickle-Stick Shocker



The kit has a pulsing 80 volt tickle output and a mischievous blinking LED. And who can resist a blinking light and an unlabeled switch! Great fun for your desk, "Hey, I told you not to touch!" Runs on 3-6 VDC.

TS4 Tickle Stick Kit $12.95

Speedy Speed Radar Gun

Our famous *Speedy* radar gun teaches you doppler effect the fun way! Digital readout displays in MPH, KPH, or FPS. You supply two coffee cans! Runs on 12VDC or our AC125 supply.

SG7 Speed Radar Gun Kit $69.95

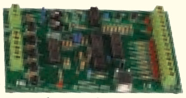
Tri-Field Meter Kit

THE GHOST DETECTOR!

"See" electrical, magnetic, and RF fields as a graphical LED display on the front panel! Use it to detect these fields in your house, find RF sources, you name it. Featured on CBS's Ghost Whisperer to detect the presence of ghosts! Req's 4 AAA batteries.

TFM3C Tri-Field Meter Kit $74.95

USB Experimenter's Kit



Get hands-on experience developing USB interfaces! 5 digital inputs, 8 digital outputs, 2 analog I/O's! Includes diagnostic software and DLL for use with Windows based systems. The mystery is solved with this kit!

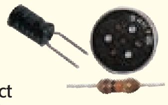
K8055 USB Experimenter's Kit $49.95

Broadband RF Preamp

Need to "perk-up" your counter or other equipment to read weak signals? This preamp has low noise and yet provides 25dB gain from 1MHz to well over 1GHz. Output can reach 100mW! Runs on 12 volts AC or DC or the included 110VAC PS. Assmb.

PR2 Broadband RF Preamp $69.95

Electret Condenser Mic



This extremely sensitive 3/8" mic has a built-in FET preamplifier! It's a great replacement mic, or a perfect answer to add a mic to your project. Powered by 3-15VDC, and we even include coupling cap and a current limiting resistor! Extremely popular!

MC1 Mini Electret Condenser Mic Kit $3.95

Sniff-It RF Detector Probe

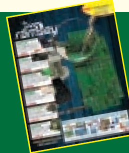
Measure RF with your standard DMM or VOM! This extremely sensitive RF detector probe connects to any voltmeter and allows you to measure RF from 100kHz to over 1GHz! So sensitive it can be used as a RF field strength meter!

RF1 Sniff-It RF Detector Probe Kit $27.95

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 latest 2011 Ramsey Hobby Catalog today! Filled with some of the neatest goodies around! Order yours today... Or be green and download the PDF at *www.ramseykits.com/catalog*!

800-446-2295
www.ramseykits.com

RAMSEY ELECTRONICS, LLC
590 Fishers Station Drive
Victor, NY 14564
(800) 446-2295
(585) 924-4560

Q&A

WHAT'S UP:

Join us as we delve into the basics of electronics as applied to every day problems, like:

- ✔ Car Finder
- ✔ Circuit for Pulsating LEDs
- ✔ Solar Panel Manager

■ WITH RUSSELL KINCAID

In this column, I answer 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**

POWER SUPPLY

Q I recently purchased a compact DC blower motor that runs on 53 VDC with a max current draw of four amps. It was cheap and the dimensions with the CFM output made it perfect. I thought I could skirt around the power issue, but I was wrong. The motor is variable speed using a 0 to 10 volt DC input. I am currently using a nine volt battery for this.

I am using a transformer from a home stereo amplifier; it is rather large and heavy, and has a 90 volt winding with a 45 volt center tap. I am currently using the 45 VCT winding with a full wave rectifier and large capacitor.

With nine volts on the speed input, the motor runs at 90% and the voltage is about 50 volts. Anything more and the voltage falls too low. I need it to be at 100% to get the 400+ CFM I need. I'm almost sure it will cook the transformer on continuous use because I am demanding too much from the 45 volt winding. I have run the setup for about 60 seconds with no noticeable temperature rise but I am still concerned.

I've looked everywhere for a 53 volt regulator schematic with no success (can't imagine why). I was even considering building two identical 45 volt supplies and use a few heavy-duty diodes to tie them together to double the amps. Common sense then compelled me to stop before bad things happened and write this email ... perhaps a little fear, as well! I've seen previous Q&A solutions and was hoping your brilliant engineering team could come up with an amicable schematic for my situation.

My brother gifted me a subscription last year and I intend to ask him for a renewal. Count on that like a calculator.

I'm hoping my invention will benefit a lot of people in a lot of places in a lot of ways. So, even if my problem is not worthy of publication (I understand there can only be a chosen few), I would appreciate any help your team could offer with the prototype.

— Jeff Shearer

POWER SUPPLY PARTS LIST

PART	DESCRIPTION	MOUSER PART #	PRICE
FUSE CLIP, 2 NEEDED	PC MOUNT, 5X20	534-3517	0.14
D7, BRIDGE RECTIFIER	1,000V, 6A	625-GBPC610-E4	1.43
R11, SURGE LIMITER	5 OHMS, 2.8A	527-CL160	0.84
C2	1 µF, 35V, TANTALUM	581-TAP105K035SCS	0.40
C3	3.3 µF, 16V, TANTALUM	581-TAP335K016SCS	0.40
C4, C5	0.01 µF, 250V, CERMIC	810-FK14X7R2E103K	0.18
C6, C8, C10, C18	1,000 µF, 200V, LOW ESR	871-B43231B2108M000	7.86
C7, C9, C11, C14	150 µF, 200V, LOW ESR	667-EEU-ED2D151	1.55
C12	0.1 µF, 250V, CERAMIC	810-FK20X7R2E104K	0.54
C13, C15	10 µF, 25V, TANTALUM	80-T350E106K025AT	0.67
C16, C1	1,000 pF, 50V, CERAMIC	81-RPER71102K2P1A03B	0.13
C17	.01 µF, 50V, CERAMIC	81-RPER71K103K2P1A03	0.12
ALL RESISTORS	1/4 WATT, METAL FILM	271-VALUE-RC	0.09
R9	10K, 1/2W, POT	31VC410-F	1.40
R13	1K, 1/2W, POT	652-3006P-1-102-LF	1.55
Q1, Q2 (TO-220)	N-MOS, 600V, 7A	512-FCP7N60	1.62
Q3 (TO-225)	NPN, 250V, 0.5A	863-2N5655G	0.58
D1, D2	600V, 1A, FAST	821-1N4937	0.06
D3, D4, D5, D6, D13, D14, D15, D16	75V, 200 mA	512-1N4148	0.09
D9, D10, D11, D12 (TO-220-2)	600V, 15A, 25 nS	511-STTH5L06FP	1.82
D8 (DO-41)	ZENER, 12V, 1W	512-1N4742A	0.06
IC1 DRIVER	15V, 2A	595-TPS2812P	2.17
IC2 REGULATOR	PWM, PUSH-PULL	595-UC2525AN	3.43
IC3 DC REG (TO-92)	10V, 100 mA	512-KA78L10AZTA	0.39
IC4 OP-AMP	DUAL, GEN-PURP	511-LM358N	0.50
OK1 OPTO COUPLER	600% TRANSFER	551-PS2501A-1-A	0.32
ALL MOUSER PART NUMBERS EXCEPT			
T1 POWER TRANSFORMER	FERRITE CORE	RADIOSHACK 273-104	8.99

A

This design (**Figure 1**) is modeled after successful power supplies that I have built, but I did not build this one, so no guarantee. I am not sure that the feedback compensation (R18, C16, and C17) is optimum but if you have problems with oscillation, reduce the gain by increasing R17.

The supply is designed to produce 50 to 60 VDC output at six amps. The input DC voltage will be about 160 volts due to the full wave bridge rectifier. The power output is 6 x 60 = 360 watts. The power input will be more due to losses; about 400 watts. The input current is therefore: I = P/E = 400/160 = 2.5 amps. The operating frequency is 100 kHz and the period is 10 microseconds. The ripple frequency at the input is 120 Hz (period = 8.3 mS). In order to limit the ripple voltage to one volt, the input capacitance should be C = I x dT/dV = 2.5 * 8.3/1 mF = 21,000 µF. Well, maybe we don't need to limit the ripple so much.

If the control loop bandwidth is 120 Hz or more, it can regulate out the ripple. Let the ripple voltage be five volts, then C = 4,000 µF. I will parallel four 1,000 µF caps for lower equivalent series resistance (ESR). The transistor Q3 and diode D8 supplies start up power to the PWM IC and driver.

As soon as the output voltage is up, the bootstrap feedback will be 12 to 15 volts and will shut off Q3. If the bootstrap voltage exceeds 17 volts, the emitter-base junction could zener, which could damage Q3. I chose the RadioShack choke core (273-104) for the ferrite core of the transformer. I used Micrometals catalog and handbook to calculate the parameters.

First, I need to find the inductance coefficient of the core in mH per 1,000 turns (Al). I wound 26 turns on the core and resonated it with .01 µF. The frequency was 109.96 kHz which calculates the inductance to be 209.5 microhenries. Since the inductance is proportional to the square of the turns, Al = .3099 mH/1000T.

Since I want the leakage current of the transformer to be small relative to the primary current of 2.5 amps, I will set the leakage at 200 mA and see what inductance is needed.

Xl = 2*pi*F*L and Xl = E/I = 160/.2 = 800 ohms then L = 800/2/pi/100KHz = 1.27 mH

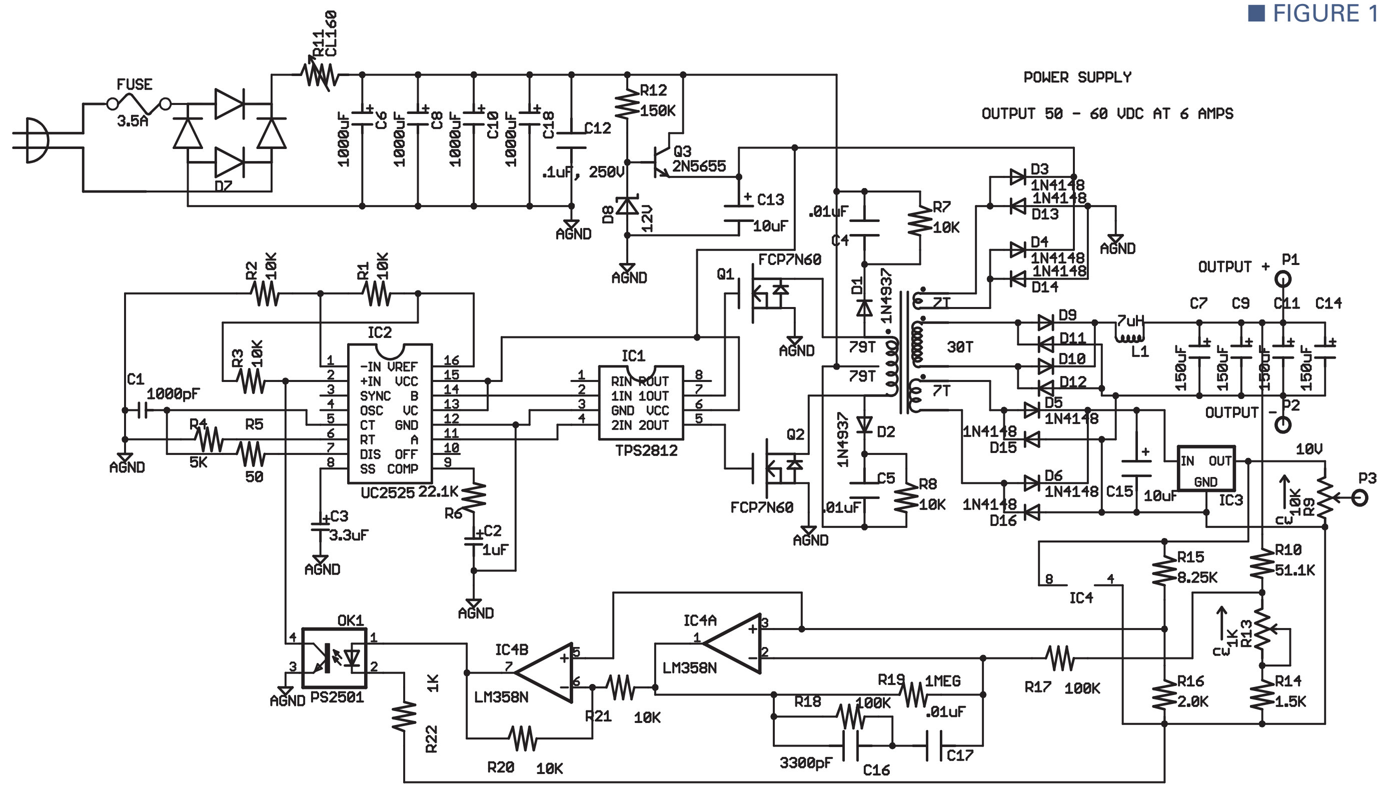
Calculating the turns from N = (L*10^6/Al)^1/2, I get N = 64 turns but I have been using 79 turns which is better and fits, so I will continue with that. The secondary turns are just the voltage ratio, so N2 = N1*V2/V1 = 79*60/160 = 29.6; call it 30 turns. The 15 volt bootstrap winding will be 79*15/160 = 7.4; call it seven turns.

Before I get too far down this road, I should find out if the core will handle the power. Micrometals defines a parameter WaAc which is the product of the window area and the cross-section of the core. In this case, the window is 2.2 cm*1 cm and the core cross-section is .6 cm*.6 cm. So, WaAc = 2.64 cm^4. The window is .337 in^2.

The formula — from the Micrometals catalog — is:

WaAc =Po*Dcma/K1/Bmax/f

■ FIGURE 1



where Po = power output in watts
Dcma = current density in cir. mils/amp
Bmax = flux density in gauss
f = frequency in hertz
K1 = .001 for push pull topology

Solving for Po and using Dcma = 400 cir. mils/amp, Bmax = 2,000 gauss; f = 100e3 Hz; then Po = 1,320 watts. I don't believe this; in fact, a chart on page 4.7 (Figure 7) of the catalog indicated a power output of 600 watts which I think is pushing it. However, the core should handle 400 watts okay.

Now to find the wire size. The formula in the catalog is:

AWG = -4.31*ln(1.889*I/Cd)

where ln is the natural logarithm
I is the current in amps (2.5)
Cd is the current density in amps/sq cm (400)
Solving, I get AWG = 20.

Since high frequency current travels on the surface of the conductor, it is better to use multiple small wires rather than one large one. Referring to wire tables, I could use two #23 wires or four #26. Choosing four #26, the total primary turns is 79*2*4 = 632. The secondary current is six amps, so the AWG is #15. If we again use four parallel wires, #21 could be used or #20 is more available. The secondary total turns is 30 * 4 = 120.

Now let's see if all these turns will fit in the window which is 0.337 sq in. The cross-section of #26 wire is 1.996*10^-4 sq in and the cross-section of #20 is 8.023*10^-4 sq in. Multiplying 632*1.996*10^-4 and adding 120*8.023*10^-4, I get 0.222 sq in. That fits, but is going to be snug considering the two seven-turn windings of #26 wire for the bootstrap and 10 volt output, plus insulation.

You will need to make a bobbin to wind the wire on; it could be two bobbins — primary on one side and secondary on the other — or my choice would be one bobbin and put all the wires on it.

Back to the circuit design. I chose push-pull because it is easier; you don't have to think about which way the windings go. I chose bridge rectification for the secondary because it requires fewer turns. However, a full wave with a center tap would allow half the wire size and double the turns, so it's six of one and a half dozen of the other.

The feedback is arranged such that when the output voltage drops, the pulse width increases to hold the voltage constant. The output filter bandwidth is about 1 kHz, so if the closed loop bandwidth is less than that, it should be stable.

If the loop should oscillate, reducing the feedback gain will reduce the bandwidth and make it stable. Increase R17 to lower the gain. The 7 µH inductor in the output filter is something you will have to make because I could not find any through hole part that was suitable. To make it, wind 10 turns on half of the RadioShack toroid. Be sure to tape the core; the corners are sharp. You can use four #20 wires in parallel or whatever will fit.

The diode R-C network at the primary is to damp spiking that could exceed the 600 volt rating of the switching transistors. I don't think it is likely but if the DC voltage at D1 cathode exceeds 500 volts, reduce the value of R7 and R8.

R13 is a trim pot for output voltage setting. R9 is the control voltage output pot. You should be aware that because the motor has a tachometer output, it is possible to make a servo and have positive control of motor RPM. Good luck with your project.

CAR FINDER

Q I frequently lose my car in large parking lots. I would like to use a key chain switch that I have for another car with a light alarm. It could be a sound or flashing alarm as both cars are never that close when I need the function. The frequency is unknown, of course, as all cars are apparently different. The car with the built-in alarm has one button that opens the doors; the other one opens the trunk.

— Charles Forman

A As you pointed out, all cars are different, so you will have to purchase the same receiver as used in the other car if you want to use the same transmitter. Another possibility is to find the Velleman kit #660-2N-310 (transmitter) and #AM6621-310 (compatible receiver).

LINUX AND PIC

Q After years of enslavement to the Microsoft conglomerate, I was finally able to break free and find shelter in Linux. Ubuntu to be specific. Sadly, PIC programming is not something I have found easy to accomplish in Linux. For some reason, PIC programmers — who you would think of as intelligent, outside-the-box type people — seem to gravitate towards Windows.

Anyway, if need be, I can always write the code in a text editor, but writing that code to the chip is a little more complicated. Further complicating matters is that the laptop I use only has USB ports, and the PIC programmer I built a few years ago is a parallel programmer based on the DonTronics Kit 96. I'll either have to get a USB-to-parallel converter or build/buy a USB programmer, but I am still left with the problem of programming it. Do you know of any software that allows you to program PIC chips from within Linux?

— Anonymous

A The short answer is no, but I seem to remember that I found a solution to your problem with a Google search. I can't find it now, however, or any record of it so I must have been dreaming.

These URLs will give you some information but no real solution:

www.microchipc.com/

www.linuxjournal.com/article/3045

www.micahcarrick.com/pic-programming-linux.html

I use the microEngineering Labs' U2 USB programmer with a ZIF adapter (**www.melabs.com**).

HOW TO SET UP A CUSTOM WEBSITE

Q You have helped me before and I frequently get ideas from your column, so please keep up the good work.

Today's question is more of a request for a direction rather than for a specific schematic. I belong to an RC model aircraft club and we fly on a sod farm. The farm has an automatic watering system that consists of a huge, long revolving arm that pivots at the center point so there is power available at this point.

Discussions with both the club members and the farm owner have suggested that they would both find the following information useful. I would like to mount a camera, wind gauge, and thermometer at the pivot point of the watering system that would be accessible on a dedicated member's only website.

What I am looking for is maybe a block diagram of what is needed with some ideas as to if this sort of thing is available complete off-the-shelf, or would it be a case of wiring several black boxes together. Or, do I have to get out the soldering iron? As we are definitely a non-profit club, some idea of the cost would be appreciated.

— Geff Waite

A I asked my oldest son about your problem. Here is his reply:

I agree, he would need an RF link from the remote location with a PC, router, and broadband Internet connection within 1,000 feet.

Here is an RF camera ($400) good for 1,000': **www.brickhousesecurity.com/outdoor-camera-surveillance.html**.

I found a remote weather station without a camera that ran around $355. Good for up to 1,000' under ideal conditions: **www.weathershack.com/davis-instruments/davis-vantage-vue.html**.

Software to connect it to a PC: **www.weathershack.com/davis-instruments/6555.html**.

The weathershack system can put the weather data on the Internet as part of their website but you cannot add anything to it. You could put the camera on any website that allows streaming video. You would need custom software to combine the weather data and video on a dedicated website. I have no idea how easy or difficult that would be.

QUESTION ABOUT PC BOARD SOFTWARE

Q I have used WinBoard — a good program, now out of business — and Eagle, which is too restrictive and mostly for European designers. Who makes a reasonably priced schematic capture and layout software package?

— Jay Harford

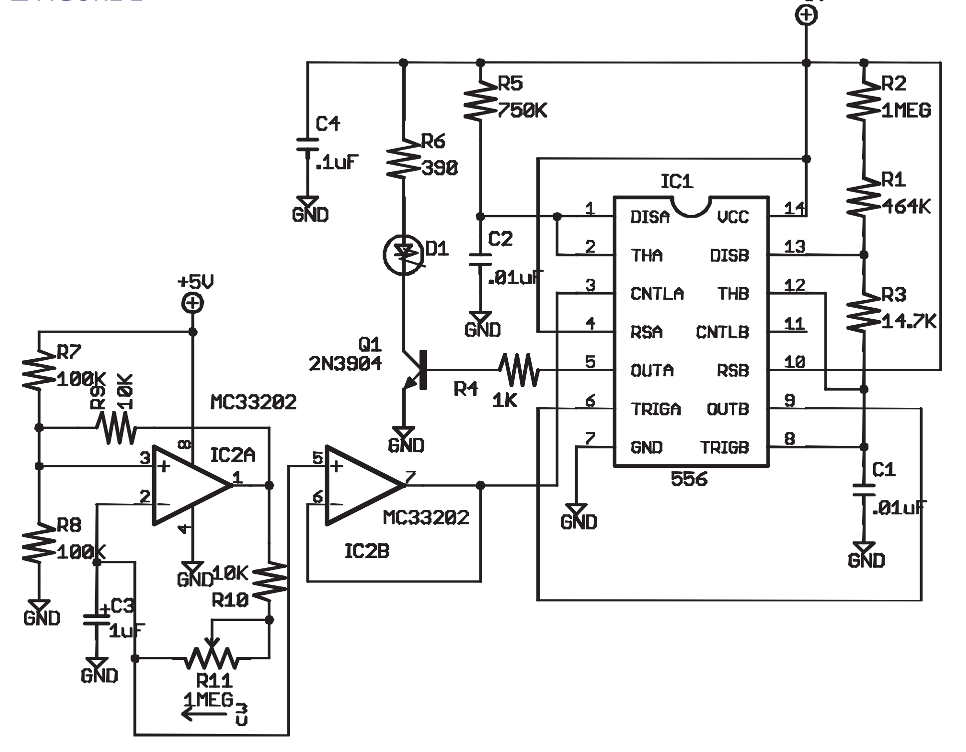
A Just because the program originator is out of business does not make the program bad. I use several programs that are so old they run in DOS, but they are still useful. I use Eagle which has both European and US symbols and does everything I need; although the inability to group and copy is frustrating. You might consider PCB123 from Sunstone Circuits. It is free but you have to have the boards made by them. I have had boards built by Sunstone and find their quality and price to be good.

CIRCUIT FOR PULSATING LEDS

Q I am looking for a simple circuit which would make a set of LEDs slowly pulsate from low to high illumination, preferably with an adjustable rate of pulsation. I would greatly appreciate any info you can provide.

— Bob Christopher

■ FIGURE 2



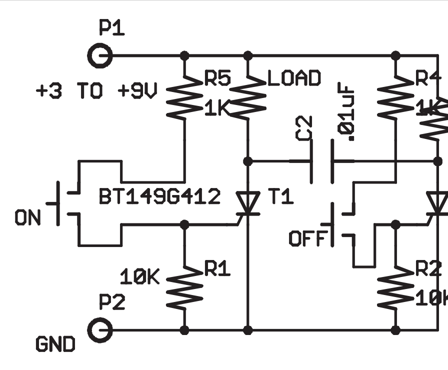


FIGURE 3

A microprocessor solution is straightforward but I prefer a 555 solution. In **Figure 2**, the B side of the 556 is a 100 Hz astable which triggers the A side. The A side is a monostable with nominal 5 mS pulse width, but the pulse width is varied from near zero to near 10 mS by the ramp voltage on the control input.

U2 is a rail-to-rail op-amp which is necessary to get the maximum variation in pulse width. The ramp rate is set by R11 and C3; you may want to increase one or both to get the rate you want. You can parallel 10 or more LEDs and series resistors at Q1.

ON/OFF SWITCH NEEDED

Q I am assisting on a local theater play and need to purchase a remote control. I just found out that the LED headset that the 'Spirit of Christmas Past' uses needs an RF remote to turn it on and off. I don't have time to learn the BASIC Stamp and the LINX eval board is nearly too big. I am leaning towards an SCR for the switch for the LEDs.

The play starts Thursday. I was hoping for one button; no matter how many times you push it, it turns on. Then, another button; no matter how many times you push it, it turns off. Is there anything out there that I can purchase overnight to make this happen? Nov '10, pg 40 is close but would take weeks to design, test, and build. Thanks for any help you can provide.

— Thomas Prohaska

A This is too late to help you I am afraid, but the **Figure 3** schematic shows how to make an SCR turn on and off.

The "on" pushbutton turns on T1 and energizes the load; the "off" pushbutton turns on T2 which dumps the charge on C2 into T1, turning it off. The size of C2 will depend on the load current, but .01 µF should work for your application.

SOLAR PANEL MANAGER

Q I want to power a water fountain (12 VDC bilge pump) with a 45W solar panel. I need a low voltage (current?) cut-off circuit that will protect the motor when the panel is shaded and at sunset.

— Pat McNeill

A The voltage and current of the solar panel is reduced when it is shaded. If there is not enough current to turn the motor, there is not enough to damage it either, so no protection is needed.

TURN SIGNAL REMINDER

Q I often don't hear the clicking of a turn signal or see the indicator on a dashboard. How can I make a visual indicator to be mounted in the driver's line-of-sight, or perhaps an escalating tone that would start after a reasonable amount of time?

— Pat McNeill

A If your car has a two-pin or three-pin "clicker" that plugs into the fuse panel, you can buy a loud one at the local parts store. If you want something louder, I suggest Mouser part number 539-PK26N04P12AQ ($3.50). It is 400 Hz which even we old timers can hear, and has two flanges for mounting. You could solder some small wire (#22 hookup) to the prongs of the clicker such that you can still plug it in, and then connect it to the audio alarm, making sure the polarity is correct. Alternately, you could use a 12 volt LED lamp; Mouser part number 607-1090a1-12v.

If you don't have a clicker, then you will have to get behind the dash and connect a pair of diodes (1n4004) to the right and left lamps. Connect the cathodes together and to the audio alarm or visual indicator. The negative side of the indicator goes to chassis ground. **NV**

MAILBAG

Dear Russell: Re: Lightning Protection Circuit, Dec '10, page 25.

In the Dec issue, you stated: "The PIC12F675 has two timers but neither will count for five seconds, so an external clock is needed (LM555)."

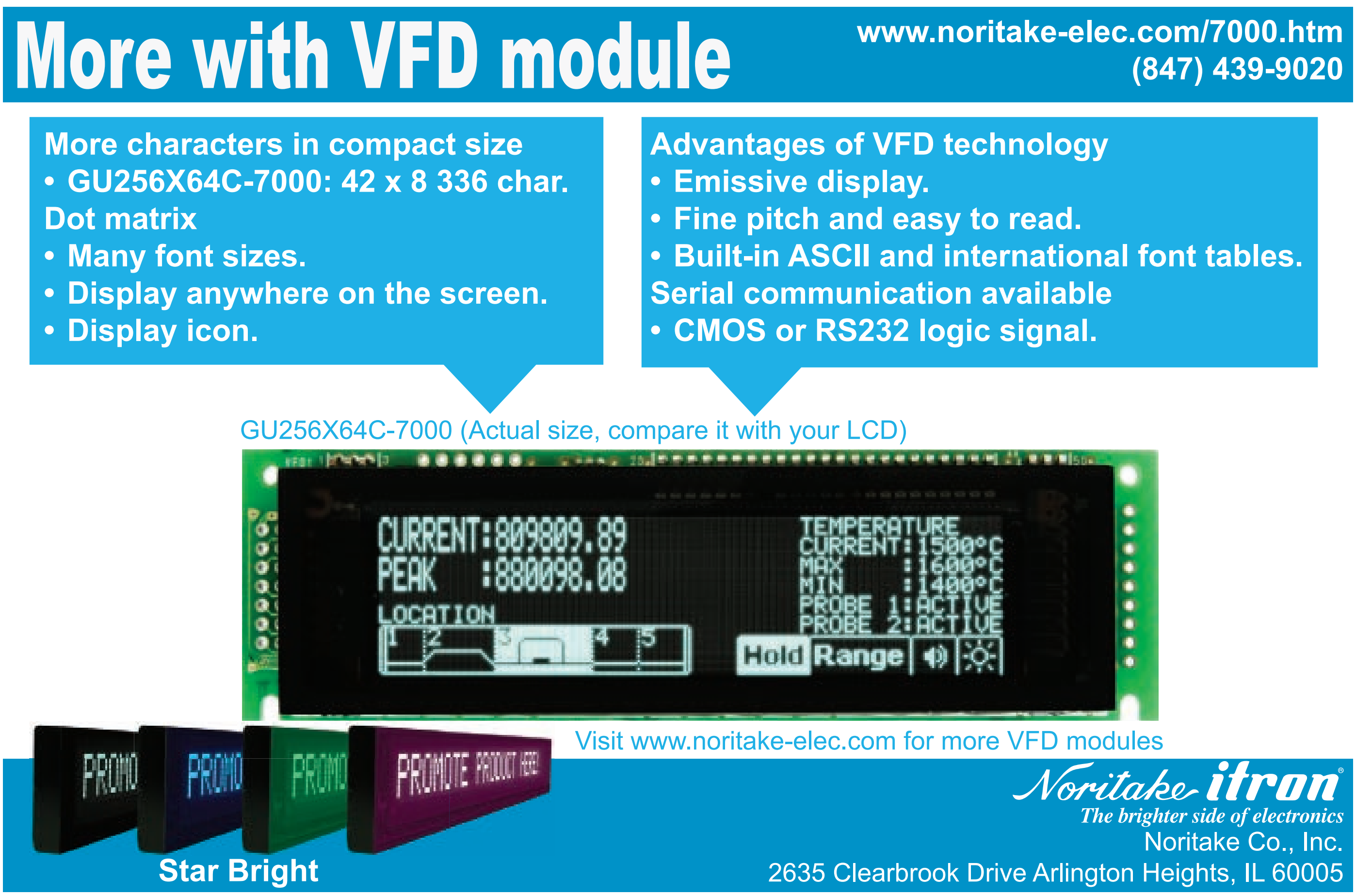
I use the internal RAM registers as counters to increase the timer. A few lines of code are needed to load a predetermined value into a register. Then, when the timer interrupts, have the code decrement the register value. When the value reaches zero, you now have a much longer timeout. I have even used two and three registers that when the first one reaches zero, the code decrements a second register, and so on, to get timers in the minute and even hour ranges. Also, I had one application where I had 10 registers all counting down for different timers, each with a different timer range.

— Gary Emmich

Response: Thanks for the tip, Gary. I am sure that will come in handy some time.


V-MODULE
Make Design Easier
Class D Audio Amplifiers
High Quality, High Effeciency
DC/DC Converters
High Effeciency, No Heat Sink
A2015
P3201
FM Audio Transmitters
HiFi Sound Quality
Evaluation Boards
Low Cost, Ready-to-Use
R6512
Features:
All integrated design
Ultra compact
Almost no external parts
Soldered by SMT or by hand
Low cost
High Reliability
Many ready-to-use EVB
Applications:
Consumer
Auto accessories
Instruments
Industry
Toys
Communication Equipments
And many more
Dealer Wanted!
URL: http://www.v-module.com
Email: sales@v-module.com
Tel: +8610 82562708
Fax: +8610 82562707

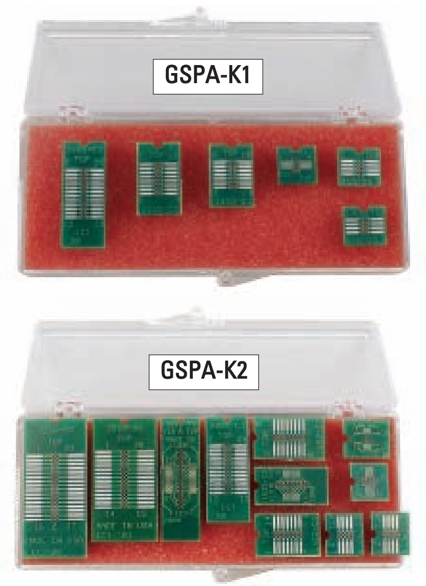

Web-based Control and Monitoring
Get your project online... in minutes!
> No Programming Required
> No Port Forwarding or Dynamic DNS
> No Network Configuration Necessary
> No Need to Host a Web Server
IO-204 Ethernet Module
Real-time, secure control and monitoring...
> Servo and Relay Control
> X10 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
Open | Close
Outside
Outside Light
On | Off
Inside
Driveway Cam
Left | Right
Feed Sparky
Feed Fish
Solar Panel Output
WHAT WILL YOU DO?
ioBridge
www.iobridge.com


More with VFD module
www.noritake-elec.com/7000.htm
(847) 439-9020
More characters in compact size
• GU256X64C-7000: 42 x 8 336 char.
Dot matrix
• Many font sizes.
• Display anywhere on the screen.
• Display icon.
Advantages of VFD technology
• Emissive display.
• Fine pitch and easy to read.
• Built-in ASCII and international font tables.
Serial communication available
• CMOS or RS232 logic signal.
GU256X64C-7000 (Actual size, compare it with your LCD)
CURRENT:809809.89
PEAK :880098.08
LOCATION
TEMPERATURE
CURRENT:1500°C
MAX :1600°C
MIN :1400°C
PROBE 1:ACTIVE
PROBE 2:ACTIVE
Hold
Range
Visit www.noritake-elec.com for more VFD modules
PROMO
PROMOTE PRODUCT HERE
Star Bright
Noritake itron
The brighter side of electronics
Noritake Co., Inc.
2635 Clearbrook Drive Arlington Heights, IL 60005

NEW PRODUCTS

- HARDWARE
- SOFTWARE
- GADGETS
- TOOLS

SURFACE-MOUNT ADAPTERS



Global Specialties announces the release of two surface-mount to through hole adapter kits — Models GSPA-K1 and GSPA-K2.

These surface-mount adapter kits are a collection of SOIC, SSOP, and MSOP to DIP adapter sockets which allow the user to solder popular pin-configured surface-mount integrated circuits onto a platform that is easily adaptable to solderless breadboards or existing circuit layouts. Once the surface-mount IC is mounted to the DIP adapter, it becomes re-usable in prototyping applications, similar to other through hole ICs, saving time and money. Each board can be used in conjunction with all of Global Specialties' electronic trainers and solderless breadboard systems.

Each kit is perfect for educational prototyping and experiment applications, as well as production retrofits and daughter board configurations. Individual boards are available for large quantity orders.

GSPA-K1 is a six piece set with a retail cost of $70 and GSPA-K2 is an 11 piece set with a retail cost of $160. Each is housed in a plastic case with a snap enclosure.

For more information, contact:
Global Specialties, LLC
22820 Savi Ranch Parkway
Yorba Linda, CA 92887
Web:
www.globalspecialties.com
Contact Frank Menichello:
Email: **frankm@globalspecialties.com**
Tel: **800-572-1028**

NEW PROTOMAT LINE

LPKF Laser & Electronics announces the release of four new circuit board plotters. This is a revision of their best known product group: the ProtoMat® S-Series circuit board plotters, which were reengineered to work faster and more efficient. Also being introduced is the new ProtoMat® E33 entry-level unit. At first glance, you'll notice all S-Series ProtoMat's come inside a new modern looking acoustic cabinet and include available upgrade options. With the new ProtoMat line, LPKF continues its innovation efforts in circuit board prototyping.

All new systems come with completely redesigned CircuitPro software by LPKF which is aimed at a wider range of users. While inexperienced users are guided step-by-step through the production processes by finely tuned wizards, proficient users can continue using menu functions.

Britta Schulz, Vice President of the Rapid PCB Prototyping Division notes, "We have attained a superior level in affordable and easy-to-use PCB prototyping equipment and offer the widest range in the market. Advancement in technology allows each system to be upgraded and we will still continue working on new and existing prototyping equipment to meet ever changing needs."

For more information, contact:
LPKF Laser & Electronics
Web: **www.lpkfusa.com**

EDUCATIONAL SCOPE/LOGGER

Saelig Company, Inc., introduces the GP-24116 and GP-24132 which combine a high-speed digital pattern generator, logic analyzer, and bi-directional protocol (SPI/I²C) host adapter with either 16 MB or 32 MB of internal memory. GP-241xx devices offer high speed programmable bidirectional interfaces that allow the stimulation and analysis of digital electronic systems and ICs, providing flexible and powerful access to electronic devices under test.

With an internal 16 or 32 MB embedded memory buffer and 100

MHz operation on all 16 digital lines, GP-241xx devices provide powerful general-purpose and expandable PC-based tools that ideally complement existing system debug equipment such as logic analyzers and high speed oscilloscopes. The GP-241xx can be used as an arbitrary digital pattern generator, logic analyzer, or serial protocol master/analyzer, with 16 address/data lines and six control lines, offering up to 100 MHz (200 MB/s burst) performance. The ADWG/pattern generator mode of operation allows the generation of extended depths of arbitrary digital stimuli directly from a PC via a USB connection, while the analyzer mode transforms the GP-241xx into a convenient logic analyzer to collect and examine digital data.

While GP-241xx devices are able to sustain a theoretical continuous throughput of 48 MByte/s, actual throughput for continuous transfer will depend on the host PC's performance, and can be expected between 11 MB/s and 30 MB/s. For 3.3V I/O standards, GP-241xx is powered directly via the host PC's USB port, enabling very quick device setup. Alternatively, an external power supply can be connected when higher current or another I/O voltage is required.

8PI Control Panel software provided with GP-241xx devices offers free graphical, scripting (TCL/tk), and programming (C/C++) interfaces to automate

continued on page 51

SHOWCASE

COMPUTER CONTROLLED AIR/STEAM ENGINE

By John Molnar

For the past several years, I've been investigating various sources of alternative energy for home use and in my ham radio shack. Some solar panels are in place for charging batteries and I'm producing a quantity of hot water from some old homemade copper collectors, but I started itching for yet another source of green power. I discovered that there is a large undercurrent of experimenters working with small steam powered engines directly coupled to efficient generators capable of producing meaningful power. I decided to see what this was all about. We're not talking about those massive multi-ton Industrial Revolution engines here. There are many modern 1–10 HP engines available that are realistic for domestic power generation … but where to start?

The Concept

I had just replaced my old gasoline powered weed-eater with a new propane powered four-stroke, and decided to develop an air/steam powered engine from the old remains. However, I refused to completely regress to the old days of mechanical linkages and leaking valves, so I determined that my 21st century engine would have to be computer controlled.

What follows is my effort to convert a conventional two-cycle engine into one that is powered externally by compressed air for testing, then ultimately steam produced by a small, safe "monotube" boiler (see the **sidebar**).

This engine is completely controlled by a PIC18F2525 based controller that allows complete control of inlet and exhaust timing and duration, with real time RPM and parameter display. The engine piston position is relayed to the processor via Hall-effect magnetic sensors.

This project is very easily reproduced by any experimenter with average mechanical skills; the controller is an easy breadboard; and — best of all — it's cheap and green! I found it to be a great platform for learning and applying computer engine control. **Figure 1** is the completed engine and controller on the bench.

So, let's get started!

Find An Engine

Any two-cycle engine can be adapted to my controller. I chose the 25 cc weedwacker engine because I had it. However, they can be obtained usually for free by checking your landfill or the trash bin at small engine repair shops. Often people pitch them just because the recoil starter rope broke. Look for one that turns over freely; you won't need the carb, rope start, or magneto.

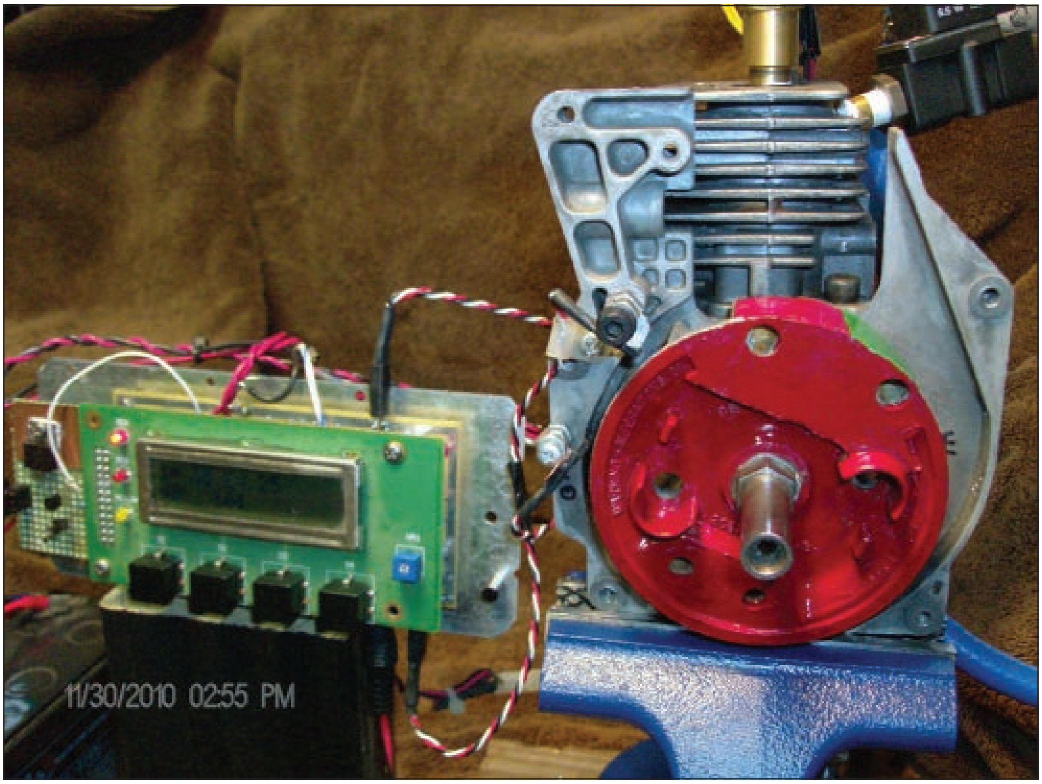
Strip it down to the essentials: crankcase and head. Remove the old plastic shroud, exhaust can, recoil starter, and ignition parts. Be sure to remove the rear crankcase "stuffing" plate to expose the crankshaft and connecting rod. If this plate is not removed, the modified engine will exhibit poor performance due to backpressure in the crankcase.

So, How Does This Work?

What we're building is an "external combustion engine." The motive power (steam, air) is developed OUTSIDE the engine, not inside as when burning gasoline in a confined cylinder. Our pressurized gas is injected INTO the spark plug hole for a controlled time period, forcing the piston down. At the bottom of the power stroke, the piston uncovers the exhaust port, allowing pressure to escape. Flywheel inertia carries the piston back up for another power cycle.

Our modified engine uses a 12V air solenoid to admit the gas for a specific duration. A second identical solenoid opens on the piston upstroke to release the backpressure that makes the engine difficult to start/run at very low RPM. The timing of the solenoids is based on the engine's Top Dead Center (TDC) position where the piston is at its uppermost point of travel.

Conveniently, the original spark ignition system functioned by placing two powerful permanent magnets in the flywheel at TDC to induce the high voltage spark current in the magneto coil. We use those



■ **FIGURE 1.** Modified engine and controller. The exhaust solenoid is visible in the upper right, and the two Hall sensors are mounted on the frame to the left of the engine flywheel. The red/green zone on the flywheel show the magnets – the demarcation in color is the exact engine Top Dead Center (TDC).

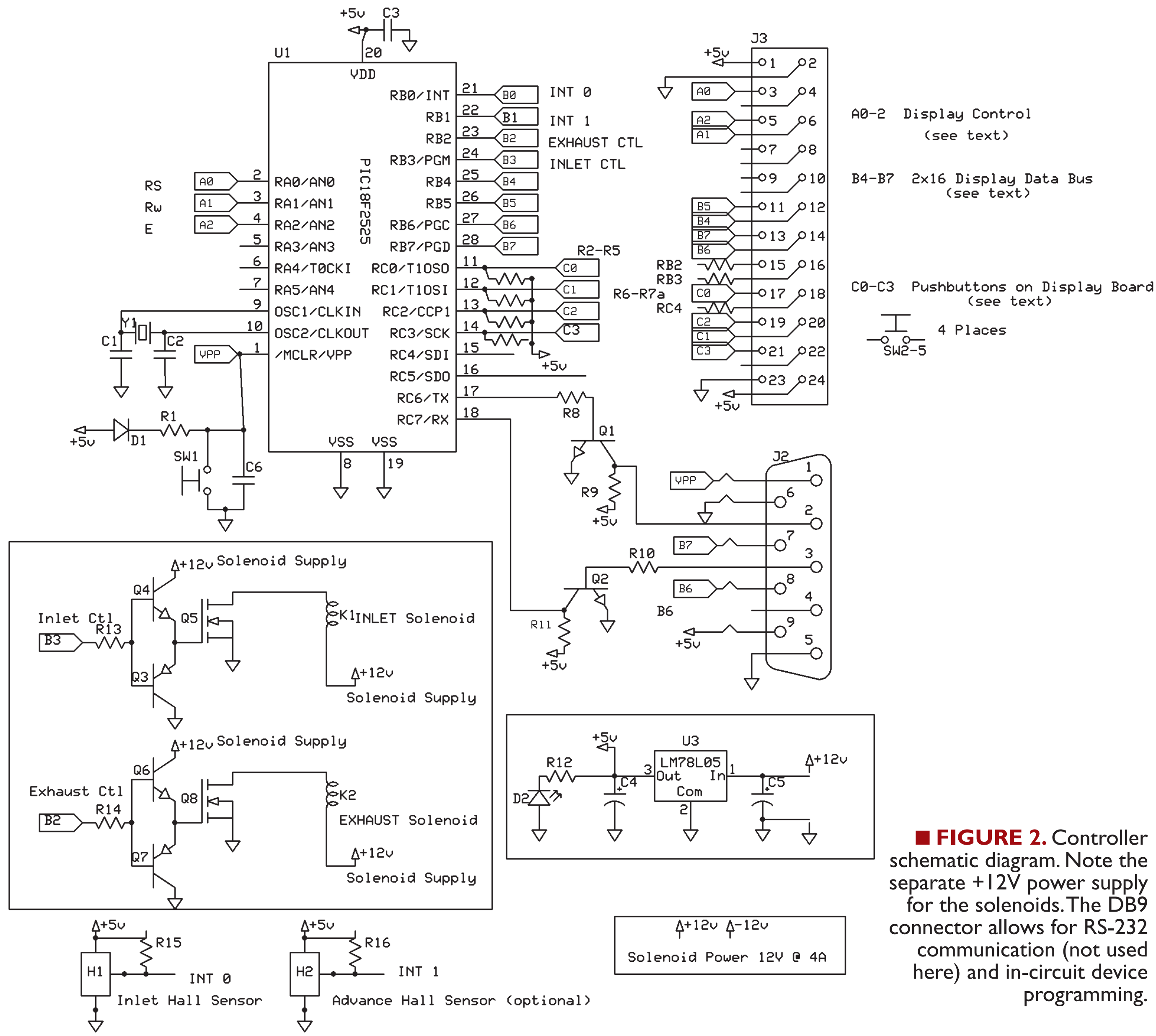
WHY COMPUTER ENGINE CONTROL

Piston engines produce power based on the force of a gas (burning air/gasoline mix, steam, compressed air, etc.) acting on a piston(s) within a closed cylinder(s). The up/down piston motion is linkage coupled to a crank that produces the more familiar circular motion. Timing and moving the gas in and out of the cylinder usually requires a series of valves, pushrods, and eccentrics (cams) mechanically connected to the crank. The spark that fires the cylinder is also timed by yet more linkages — the end result being the engine usually runs most efficiently within a very narrow RPM/power range. Operation outside that range results in pollution, wasted fuel, and poor performance.

The automobile and small engines we are so wedded to in today's society would have been legislated into oblivion years ago if not for the introduction of computerized engine control – the "module" controlling every aspect of engine operation. Spark timing, pollution control, air-fuel mixture, and even linkages are now controlled "off engine" based upon RPM, load, temperature, fuel quality, and altitude. The computer allows modification of parameters that used to be fixed by rigid mechanical linkages.

The engine described here is a "modified uniflow single-acting two stroke". Unlike the four-stroke automobile engine described above, the uniflow engine accepts compressed gas above the piston, and drives it down until an exhaust port milled into the lower cylinder wall is uncovered, allowing used gas to escape. The exhaust does not have to be forced out by an additional piston stroke, thus you have "uniflow" — in the top, out the bottom. Single-acting means the engine only produces power on the down stroke, not when returning. However, the timing of the external power (gas) is based on engine RPM, desired torque, pipe diameter, and solenoid latency — once again, enter computer control! The simple platform described here allows the builder to apply real computer control to an engine and actually observe how tuning a mechanical device can radically improve (or degrade) performance. As with most things, knowledge is power!

When operating with steam, DO NOT be tempted to construct a boiler out of a coffee can or the like and place it on a fire! Compressed steam has massive potential energy and must be treated with total respect. I use a "mono-tube" boiler that consists of a continuous coil of tubing — water in one end, and steam out the other. There is no captive container of steam to explode. Feel free to contact me for details and, as usual, there is a wealth of information on the Web – search for "mono-tube steam boiler."



■ **FIGURE 2.** Controller schematic diagram. Note the separate +12V power supply for the solenoids. The DB9 connector allows for RS-232 communication (not used here) and in-circuit device programming.

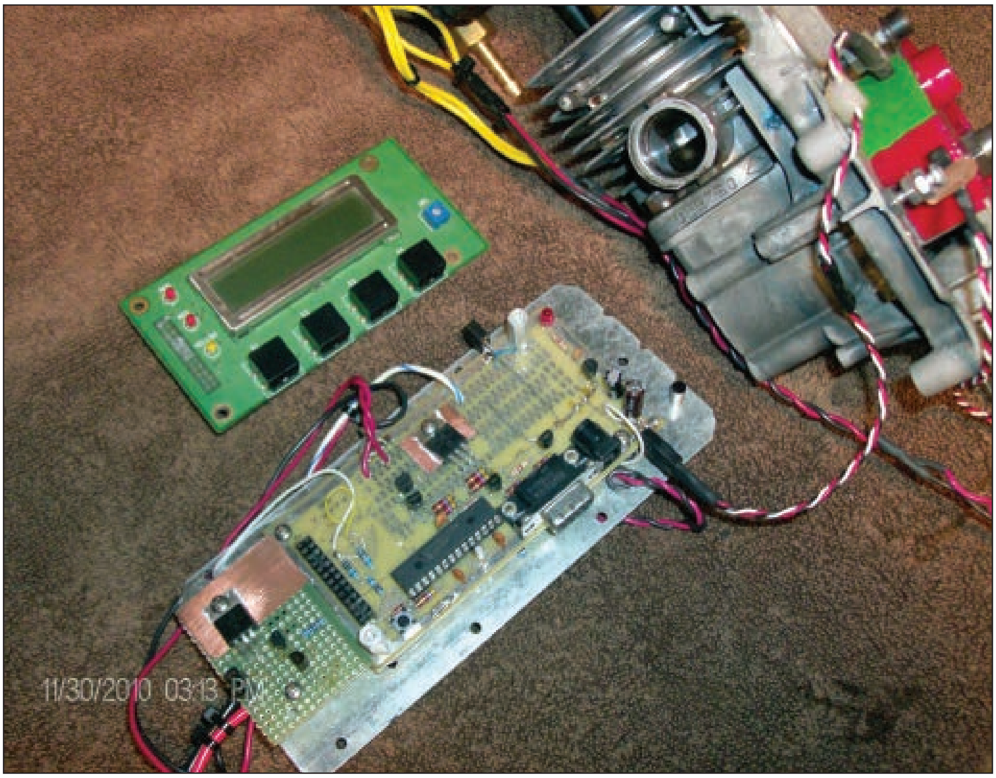
magnets to trigger a Hall-effect electronic sensor that indicates the TDC position to the controller — no mechanical interface exists between the engine and controller. The Hall sensor we will employ is an open-collector device used to produce an interrupt to the timing software.

Engine Controller Primer — Timing Is Everything

As mentioned, the controller for the engine is based upon a PIC18F2525 programmed in CCS "C." Check out the schematic in **Figure 2.** Nothing exotic here, for sure! The entire breadboard requires no more than an evening's work. I used an existing board/display from another project with modifications for the power FET solenoid drivers. The display uses the upper four bits of port B and three port A bits for its function. Two falling edge external interrupts on port B are triggered by the Hall-effect sensors positioned around the engine flywheel. Multiple 16-bit timers in the PIC are employed to generate the required timing cycles for the engine. Four pushbuttons on port C allow the inlet and exhaust timing to be modified in real time while the engine is running. In **Figure 3**, the display module is removed, showing the PCB. The inlet MOSFET subassembly is the breadboard on the left; the exhaust driver is in the center of the main board. Note the Hall sensors on the frame of the engine. (Engine TDC is the red/green demarcation line on the flywheel.)

Here's how it works: When the engine passes TDC, the Hall interrupt triggers a 16-bit timer. Ticking at a 1.6 µS rate, the timer is preloaded with the variable inlet duration time; in my case, about 30 mS for 600 RPM. The inlet solenoid is opened, admitting pressurized gas to power the piston down. Simultaneously, another counter is started to control the opening of the exhaust solenoid based on RPM and inlet timing. The timer 0 (RTCC) clock is started, and on the NEXT TDC its value is used to derive

■ **FIGURE 3.** Controller details with display board removed. The leftmost breadboard is the INLET logic. The EXHAUST logic is in the center of the breadboard area on the main board. The PIC18F2525 and logic are toward the bottom of the main board. The Hall sensor mounting brackets and adjusting nuts are clearly visible on the engine frame. The engine runs clockwise viewed from the flywheel front. Thus, when the red/green line passes under the upper sensor, the engine firing sequence will commence.



the actual RPM of the engine. The RPM, as well as the inlet/exhaust durations in milliseconds, are displayed.

Finally, the four pushbuttons are checked for activity; two buttons each allow the inlet timing and exhaust position to be modified in real time — up or down — in 0.5 mS increments. Thus, the performance of the engine can be tuned on-the-fly based on the observed RPM and gas inlet pressure. Since there was display space left over, I also show an average RPM value based on 16 engine revolutions for trend analysis.

The running code is split into a "major" and "minor" cycle; each triggered by a TDC interrupt event. I attempted to balance the display and button functions across each cycle to maximize free time in each cycle, allowing for future feature creep.

A second (not required) Hall sensor is positioned at the 20 degree advance point over the flywheel. For the non-motorheads out there, this means the piston is 20 degrees away from the top of the cycle, still going up. The cheap air solenoids I used have considerable latency. After power is applied to the solenoid, it takes several milliseconds (about four) to open, and much longer (about 20) to close when power is removed. These times are factored into the software after much analysis of the solenoids.

Consequently, "firing" the solenoids in advance of the actual desired mechanical effect really increases engine performance. Right now, after starting the engine, the software uses the TDC sensor and transitions to the advance sensor after a user-defined number of revolutions. Again, this feature is a user option.

For interest's sake, I trigger an LED on the display board to show "idle time" — that is, when the controller is waiting for a TDC interrupt. It is on most of the time. Consider this: At 600 RPM, one revolution takes 100 mS — a lifetime for a PIC running at 20 MHz! There is a ton of processing time for additional features as the code uses only a small fraction of the processor. Most of the time is spent waiting for various interrupts. The source code for this project is on the *N&V* website and is heavily commented and pretty straightforward. Feel free to modify timing values as you like.

The bipolar drivers for the power MOSFETs are necessary to provide the hard drive required to overcome the high gate capacitance of the devices. As indicated in the schematic, a separate +12V supply at least 4A is needed for the solenoids. Don't use the logic supply as high switching currents may cause the PIC to reset. The solenoid circuits must not use the logic ground trace. I use a 12V gel cell battery to power the solenoids.

Component Selection

Although I mentioned how straightforward this project is, some comments about the circuit and components should be mentioned. If you follow the **Parts List** suggestions, you can't go wrong. However, we all like to substitute and scrounge — that's what this is all about! Anyway, I used an Internet Special surplus display board with built-in switches and LEDs. Any 2x16 LCD can be used as long as it is based on the Hitachi HD44708A style controller. The LEDs and switches can be discrete, as noted in the **schematic**. I highly recommend the display board indicated in the **Parts List**. It is a nice package and will directly mate to your logic board using connector J3.

Hall-effect sensors come in different flavors. The ones to use are the "6852" series — under $1 each. They respond to a south-pointing magnet. Some Hall sensors

Plumber's Helper — So Let's Build One

Okay, you found an old engine and stripped it down. **Figure 4** shows an example of suggested plumbing for the solenoids. Use your imagination here — pretty much anything works. The air solenoids I used are cheap plastic body units designed to control the processes in soda vending machines (see the **Parts List**). They have 1/4" NPT female connections. Adapting 1/4 NPT pipe to the spark plug hole was the most difficult issue with this project. Initially, I broke the ceramic insulator off an old spark plug, and drilled out the center conductor. I brazed a 1/4" NPT short male nipple to the old plug body. Then, I found an old V-8 compression tester in an ancient toolbox, which just happened to have a brass spark plug adaptor to 1/4" NPT threads!

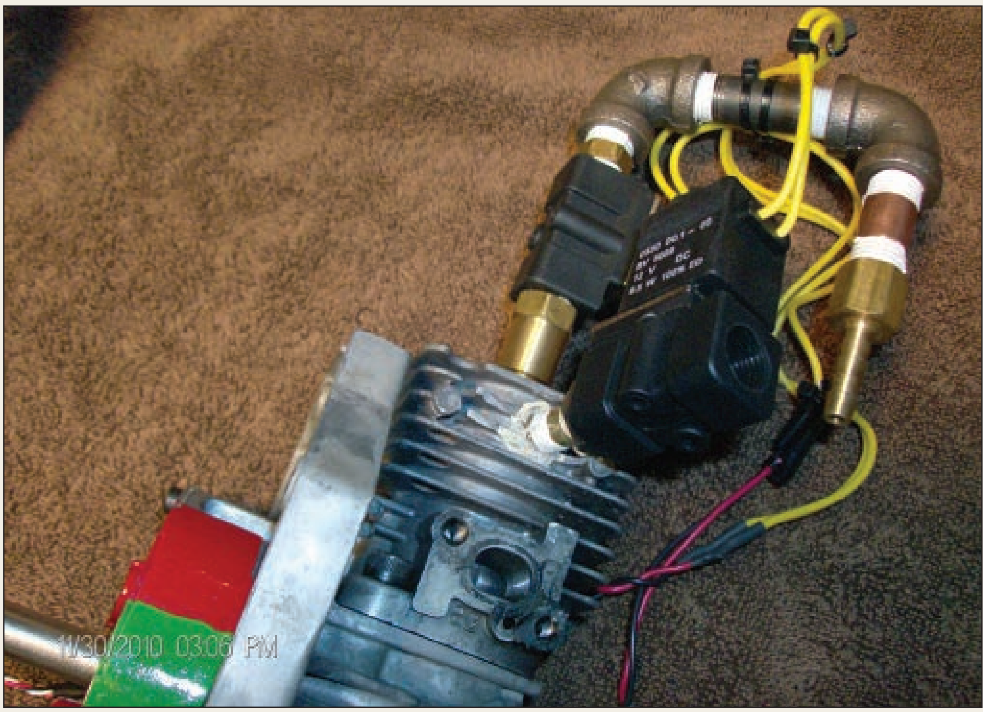
I also tested attaching the pipe nipple to the reamed out spark plug with " JB Weld " epoxy. It's amazing stuff and it was good at 150 psig – much easier! Keep the engine side of the pipe nipples as short as possible, as this is space that must be pressurized and exhausted along with the cylinder as the engine cycles, affecting engine performance. An alternative method is to remove the cylinder from the crankcase and fill the spark plug hole with JB Weld epoxy. After hardening, the epoxy can be drilled and tapped to directly accept the pipe nipple.

The inlet side of the solenoid can be however you want it; use whatever fittings you desire to match your air/steam source. **Figure 4** is worth the 1,000 words on that subject.

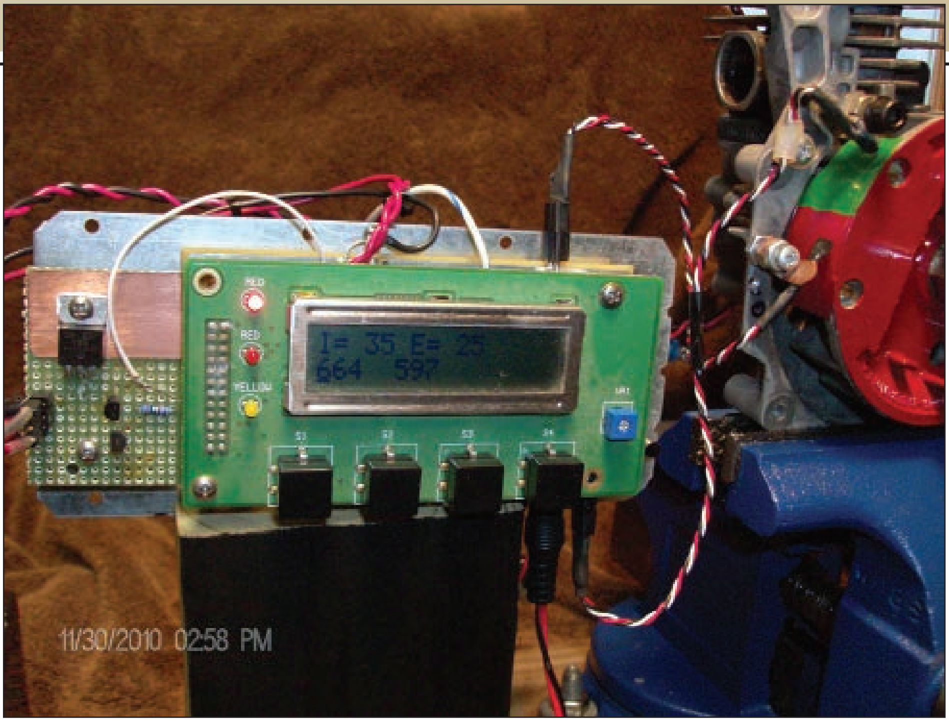
There are two ways of plumbing up the exhaust solenoid. I drilled and tapped a 1/8" NPT hole in the side of the cylinder just above the maximum piston travel, and adapted it to the solenoid with a 1/8" to 1/4" bushing. The exit side of the exhaust solenoid is left open to the atmosphere. An alternative method — which I used initially — is to use the spark plug hole for BOTH inlet and exhaust ... the short nipple from the plug hole mates to an NPT "tee" fitting. The second hole in the tee mates the exhaust solenoid, and the third to the inlet solenoid. The engine performance is not quite as good with this scheme, due to the inlet and exhaust bucking each for access to the spark plug hole during gas flow transitions.

However, I found that by tuning the engine with the control buttons, a sweet spot could be found based on the gas dynamics and pipe length resonance resulting in big performance increases at specific RPMs. Remember to use Teflon pipe tape or sealant as NPT fittings by their nature leak under any pressure. Use black iron pipe — while those bronze pipes look nice, they are expensive. Pipe nipples and fittings are common at all hardware stores.

The tiny Hall-effect magnetic sensors are mounted to the frame of the engine. The soft aluminum engine was super easy to drill and tap for a 6-32 machine screw. I superglued the sensor to a small copper tab, then bent it 90 degrees. The TDC point of the engine is easily determined by watching the crankpin or piston as you rotate the flywheel. At TDC, the original magneto magnets will probably be at the top of the flywheel. My engine has a small, 1/8" gap between the magnets — the actual TDC position. Position the Hall sensor over that gap, about 1/4" inch from the flywheel. Mark the spot on the frame for your copper bracket, and drill/tap a 6-32 or 4-40 hole (again, epoxy/superglue is fine). I used two extra nuts on the bolt to allow additional freedom in positioning the sensor. TRUST ME – the actual position is not critical as long as you are within a couple degrees of TDC. Also, the Hall sensors are shockingly sensitive against the massive magnets in the flywheel. I was able to reliably test my engine holding the sensor 2.5" away from the flywheel.



■ **FIGURE 4.** Inlet and exhaust solenoid plumbing. The exhaust solenoid is in the foreground – note the 1/8" NPT nipple angled into the upper cylinder wall. The inlet solenoid is directly behind the exhaust device. The brass "spike" allows air/steam to be admitted to the inlet solenoid. With the exception of the exhaust, all plumbing is standard 1/4" NPT hardware store material. Seal all joints with Teflon tape or pipe cement.



■ **FIGURE 5.** Controller and engine in operation. Note inlet and exhaust timing in milliseconds on line 1 of the display, and the real time and average RPM on line 2. The four pushbuttons control gas timing. The leftmost two buttons INCREASE and DECREASE control inlet (power) timing respectively, and the rightmost buttons INCREASE and DECREASE control exhaust timing. The relationship between inlet and exhaust cutoff is controlled by the software. The main exhaust port is the round opening on the left side of the cylinder.

I investigated latch up when energized and require an opposite magnetic field to turn them off — no good here.

Also, when mounting the Hall sensor, the "hot" surface is the side of the package with the numbers. That surface should face the flywheel magnets. The back of the package — while not as sensitive — will respond to a north-seeking magnet.

While looking down at the sensor with the pins down and part numbers facing up, the left pin is ground, the center pin is +5, and the rightmost pin is the open collector output.

Finally, you may have noticed that I "forgot" flyback diode protection on the solenoid coils in the **schematic**. Well, I originally used them with the result of adding over 15 mS of close time to the solenoid due to the flyback current when the device is shut off. Tough on the timing analysis, for sure.

The N-channel IRF Z30 or equivalent power FETs I used employ internal kickback protection diodes and are designed as solenoid drivers.

Bottom line, if you use the parts I specify, your engine will run with few issues — but don't be afraid to substitute and experiment within reason! I used small copper tabs as heatsinks for the power FETs — probably not necessary considering the solenoid duty cycle.

PARTS LIST AND RESOURCES

All resistors are 1/8 watt film or wire. All capacitors are disc ceramic 15V unless noted.

COMPONENT	DESCRIPTION	SOURCE
D1	1N914 or equiv. diode	
D2	Red LED	
C1-C2	18 pF	
C3-C4	01@15V	Mouser TKR15V.01
C5	1.0 µF@50V	Mouser TKR50V1.0
H1, H2 Hall	DN6852-A	Panasonic, Digi-Key DN6852-A-ND
J2	DB9, right angle if used	Jameco DB9S590
J3	24-pin female header (display)	Jameco 24SMLP
K1-K2	Process air solenoid, 1/4" NPT (STC 2V025-1/4-1-G) $18.81 ea, no minimum order required	Order Direct 650-856-8833 **Sales@STCVALVE.com**
R1, R12-R16	4.7K	
R2-R7a	1K	
R8, R10	10K	
R9, R11	2.7K	
Q1-Q2, Q4, Q6	2N2222A transistor or small signal NPN	
Q3, Q7	Small signal PNP transistor	RadioShack 276-2023 or equiv.
Q5, Q8	IRF-Z30 or equiv. N-channel power MOSFET	
Y1	Crystal, 20 MHz, HC-49/U 18 pF	Digi-Key X-439-ND

The LCD front panel set consists of a 2x16 display with HD44780 controller, three indicator LEDs, and four NO pushbuttons. It mates to J3 above. Discrete display, buttons, and LEDs may be used if desired. Available in no minimum quantities from **www.piclist.com/techref/io/lcd/panel1.htm**.

Thanks to Peter Nicola N1DYL for designing the PCB. It was fabricated by ExpressPCB using their free design tools. Given interest, this board can be made available at our cost plus shipping. Contact the author at **WA3ETD@arrl.net**.

The PIC code and ExpressPCB file are available on the *Nuts & Volts* website.

Running Up!

I initially mounted the engine in a small bench vise for testing. Connect the Hall sensor(s) to your controller breadboard and a 12V source to the solenoid circuit. If you used my **schematic** and software as shown, rotating the engine slowly clockwise should produce satisfying clicks of the solenoids as the flywheel passes over the Hall sensor(s). The LEDs will flash as the solenoids fire; after one revolution, the display will indicate RPM and timing parameters. If you manually turn the engine over for a bit, you will observe the actual and average RPM display changing.

Now, connect a compressed air source to the input side of the inlet solenoid. Start with 10 to 20 psig air at most. Manually position the flywheel such that the magnets are about 45 degrees away from TDC. A hard clockwise flip should start the engine at about 400-600 RPM. Fine-tuning of the inlet/exhaust timing will produce dramatic increases in RPM and power. I'm working on a second release of the software that will auto-tune the engine for maximum RPM at a given air/steam pressure. I will post it on the server when available, so stay tuned!

You'll find the engine will require retuning with differing gas pressure and load. The engine is running in **Figure 5.** The inlet and exhaust timing in mS is indicated on the top display line, and the actual and average RPM on the second line. The engine is accelerating in this example.

When tuning the engine, start by increasing the duration of the inlet pulse (left button). You will get to a point where the exhaust note becomes noticeably sharper and higher pitched. This indicates that the inlet is still open when the exhaust port on the cylinder is uncovered by the piston, thus wasting power. Decrease the inlet pulse a bit, and watch the RPM increase. The exhaust assist solenoid timing is critical only at higher RPM – above 1,200 RPM or so.

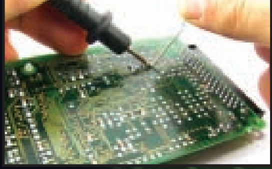
Most of the weedwacker engines out there are about 25 cc in displacement; mine has a 1" bore and a stroke of 1.125". The inlet duration is set for about 60% of the power stroke, as steam will continue to expand and provide power after timing cutoff. The built-in start timings for a 600 RPM start-up are based on those dimensions, and are obvious in the source comments.

Have fun, and stay green with steam! **NV**


WIRELESS MADE SIMPLE®
QUICKLY AND COST-EFFECTIVELY MAKE YOUR PRODUCT WIRELESS
RF MODULES
LOW COST • LONG RANGE
TRANSMITTERS • RECEIVERS •TRANSCEIVERS
INTEGRATED CIRCUITS
ENCODERS • DECODERS • TRANSCODERS
COMMAND • CONTROL •KEYLESS ENTRY
OEM PRODUCTS
EASILY CUSTOMIZED
ANTENNAS
FROM WHIPS TO CHIPS
Featured Products:
GPS Modules
GPS MADE SIMPLE ™
RXM-GPS-SR
RXM-GPS-SG
• High Sensitivity
• Easy Serial Interface
• Low Power Consumption
• High Performance
• Ultra Compact
• Low Cost
These easily applied GPS modules blend performance, and cost effectiveness into a single, compact, SMD package. The low-power chipset minimizes power consumption and provides exceptional sensitivity. Choose either an integrated or external antenna.
Development Systems
THE FAST TRACK TO WIRELESS SUCCESS
Linx development systems contain everything needed to evaluate the Linx product of you choice and implement it in record time.
LINX TECHNOLOGIES
800-736-6677
159 Ort Lane
Merlin, OR 97532
linxtechnologies.com


Saelig.com
unique electronics
FREE CASE!
7-in-1 Scope
2-ch 10-bit 2MHz scope & spectrum analyzer & wfm gen. CGR-101 $199
50MHz Scope
50MHz 2-ch 2GS/s scope w/ 2000 wfm/s refresh rate. DS1052E $399
Mixed Sig. Scope
100MHz scope, spectrum and logic analyzer. 4MSa storage. CS328A $1259
USBscope50
Thumb-drive-size PC based 50MSa/s scope - connects via USB port. $199.95
Custom Meter
High contrast 2.4" TFT color LCD w/ built-in touch screen. SGD 24M $90.25
SPI Xpress
USB-based SPI host adapter for standard & non-standard interfaces <50Mbps. $699
.NET Boards
Tiny open source boards w/ Microsoft's .NET. FEZ Mini & FEZ Domino $98
Windows Touchpanel
Touch-input 10.2" LCD 12V powered Windows PC 4GB CUPC-80/90/100 $849+
25MHz Scope
2-ch 25MHz color LCD - FFT, autoscale, & trigger delay functions. PDS5022S $287
Camera Board
Compact compressed-serial output module for any host system. uCAM $58
I/O Controllers
Simple-to-use universal I/O controllers for USB interface. No drivers req'd. $10.35+
OLED Displays
Compact smart OLEDs w/ graphics for stand-alone functionality. $36.90+
888-772-3544 www.saelig.com info@saelig.com


* Quality Quick Turn
Sheet Metal Products
&
Precision Machined Parts
* No Minimum Quantity
Restrictions
* Quote in 24 Hours
or Less!
* Several Metallic &
Non-Metallic
Materials in Stock
Visit www.iitmetalfab.com today for your free, no obligation quote!
Integrated Ideas & Technologies, Inc.
ITAR Compliant
ISO 9001:2008
AS 9100 Certified
Precision Laser Cut Stencils
Sheet Metal Fabrication
Precision Parts Manufacturing
Defense
Aerospace
Medical
Aviation
Communications
Maritime
Electronics
Serving All Industries
Metal Fabrication: www.iitmetalfab.com
SMT Stencils: www.integratedideas.com

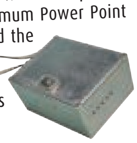


Power / Solar Kits
Electronic Enthusiasts

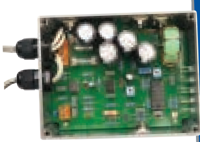
KIT OF THE MONTH

Solar Charge Controller Kit

KC-5500 $64.50 plus postage & packing
Charge controllers are essential for solar setups, although commercial units can run into several hundred dollars. Designed for use with 40W to 120W 12V solar panels and lead acid batteries, this solar charger provides 3-stage charging with the option of equalisation and with MPPT (Maximum Power Point Tracking). Operation is for 12V and the kit configured for this voltage, a 24V upgrade will be available in future. Kit includes PCB, all components and case.

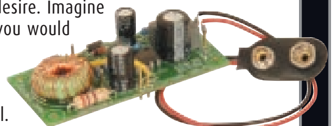


Features
- Suitable for 40W to 120W 12V solar panels
- 3-step charging
- MPPT (maximum power point tracking) charging
- Charge indicator LEDs
- Temperature compensation for charge voltage
- Optional equalisation cycle
- Optional 24V 80W to 240W operation upgrade



3V TO 9V DC TO DC CONVERTER KIT

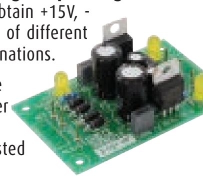
KC-5391 $9.50 plus postage & packing
This great little converter allows you to use regular Ni-Cd or Ni-MH 1.2V cells, or Alkaline 1.5V cells for 9V applications. Using low cost, high capacity rechargeable cells, the kit will pay for itself in no-time! You can use any 1.2-1.5V cells you desire. Imagine the extra capacity you would have using two 9000mAh D cells in replacement of a low capacity 9V cell.



- Kit supplied with PCB, and all electronic components

UNIVERSAL POWER SUPPLY

KC-5038 $11.00 plus postage & packing
This small kit enables you to obtain +15V, -15V or ±15V DC from a number of different transformer and rectifier combinations.



- ±15V rails from 30V AC centre tapped (MM-2007) transformer
- Kit includes PCB and all components for all options listed above
- Transformer not included.
  Use Cat. MM-2006 $4.00 / MM-2007 $5.50

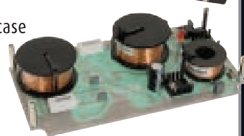
BATTERY KITS

Battery Zapper Mk III

KC-5479 $46.50 plus postage & packing
The popular battery zapper kit has gone through a couple of upgrades and this is the latest easier-to-build version. Like the original project from 2005, it attacks a common cause of failure in lead acid batteries: sulphation, which can send a battery to an early grave. The circuit produces short bursts of high levels of energy to reverse the sulphation effect. The battery condition checker is no longer included and the circuit has been updated and revamped to provide more reliable, long-term operation. It still includes test points for a DMM and binding posts for a battery charger.

- PCB with solder mask and overlay
- Components
- Screen printed machined case
- 6, 12 & 24VDC

Note: Not recommended for use with gel batteries



Battery Checker

KC-5482 $46.50 plus postage & packing
The first versions of the battery zapper included a checker circuit. The Mk III battery zapper (KC-5479) has a separate checker circuit - and this is it. It checks the health of SLA batteries prior to charging or zapping with a simple LED condition indication of fair, poor, good etc.

- Overlay PCB and electronic components
- Case with machined and silk-screened front panel
- PCB: 185 x 101mm

Dont just sit there BUILD SOMETHING

VOLTAGE REGULATORS

Improved Low Voltage Regulator

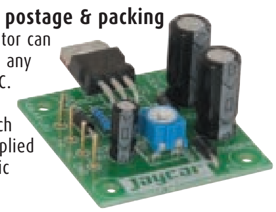
KC-5463 $10.50 plus postage & packing
This handy regulator will let you run a variety of devices such as CD, DVD or MP3 players from your car cigarette lighter sockets or even a digital camera or powered speakers from the power supply inside your PC. This unit can will supply either 3V, 5V, 6V, 9V, 12V or 15V) from a higher input voltage at up to four amps (with suitable heatsink). Kit includes screen printed PCB and all specified components. Heatsink not included.

- PCB: 108 x 37mm

Please note: that to ensure trouble free 4 amp output, a heatsink with a thermal resistance of 1.4 degrees C per watt, and an input voltage 3VDC above the output voltage is required.

Voltage Regulator Kit

KC-5446 $10.00 plus postage & packing
This handy voltage regulator can provide up to 1,000mA at any voltage from 1.3 to 22VDC. Ideal for experimental projects or as a mini bench power supply etc. Kit supplied with PCB and all electronic components.

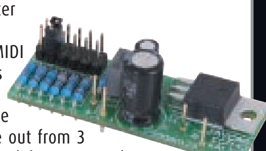


DC RELAY SWITCH KIT

KC-5434 $10.00 plus postage & packing
An extremely useful and versatile kit that enables you to use a tiny trigger current - as low as 400µA at 12V to switch up to 30A at 50VDC. It has an isolated input, and is suitable for a variety of triggering options. The kit includes PCB with overlay and all electronic components with clear instructions.

Voltage Regulator

KA-1797 $4.75 plus postage & packing
A low-powered DC converter suited for many applications such as a peripheral computer power supply, powered speakers, modems, music/MIDI keyboards, etc. Just plug it's input into your PC's internal power supply cable and have selectable regulated voltage out from 3 to 15VDC. Output current capability is around 1.5 amps depending on the size of heatsink used (heat sink sold separately). PCB plus electronic components included.



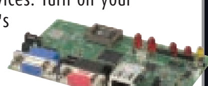
LED BATTERY VOLTAGE INDICATOR

KA-1778 $6.00 plus postage & packing
This tiny circuit measures just 25mm x 25mm and will provide power indication and low voltage indication using a bi-colour LED, and can be used in just about any piece of battery operated equipment. Currentconsumption is only 3mA at 6V and 8mA at 10V and the circuit is suitable for equipment powered from about 6-30VDC. With a simple circuit change, the bi-colour LED will produce a red glow to indicate that the voltage has exceeded a preset value.

- PCB, bi-colour LED and all specified electronic components supplied

IP POWER RELAY BOARD

KV-3595 $58.00 plus postage & packing
This 4 input switcher can be given an IP address which allows you to switch up to 4 devices. Turn on your security cameras and view what's going on at your premises by switching your cameras on from anywhere in the world! TVs, air conditioners, lighting, or other home appliances can be switched on or off by utilising this pasword protected module.



- Polarity protection
- Supports; HTTP, IP, DDNS and DHCP
- Embedded Web Server

POST & PACKING CHARGES

Order Value	Cost
$25 - $49.99	$7.50
$50 - $99.99	$20
$100 - $199.99	$40
$200 - $499.99	$60
$500+	$75

Max weight 12lb (5kg).
Heavier parcels POA.
Minimum order $25.

Note: Products are dispatched from Australia, so local customs duty & taxes may apply.
Prices valid until 31/3/2011

HOW TO ORDER

WEB:	www.jaycar.com
PHONE:	1-800-784-0263*
FAX:	+61 2 8832 3118*
EMAIL:	techstore@jaycar.com
POST:	P.O. Box 107, Rydalmere NSW 2116 Australia

- **ALL PRICING IN US DOLLARS**
- **MINIMUM ORDER ONLY $25**

*Australian Eastern Standard Time
(Monday - Friday 09.00 to 17.30 GMT + 10 hours)
Note: 10-14 days for air parcel delivery

Order online: www.jaycar.com

Jaycar Electronics

CREATE INDOOR LIGHTNING WITH THIS MINIATURE VAN de GRAAFF GENERATOR

By David Williams

A Van de Graaff generator is an electrostatic machine which uses a moving belt to accumulate very high voltages on a hollow metal dome. The American physicist Robert Jemison Van de Graaff invented this generator in 1929. Modern day applications for these high voltage generators include accelerating electrons to sterilize food and process materials, accelerating protons for nuclear physics experiments, and driving x-ray tubes.

This miniature version is designed for having fun, but it can also be used to perform electrostatic experiments. Under optimal conditions, it can produce over 25,000 volts and sparks as long as 3/4 inches. Though startling, discharges from the Van de Graaff generator do not represent a serious shock hazard since the currents attainable are so small.

What can you do with this amazing device? You can make paper streamers float up and stand on end as the like charges repel each other. Or, you can watch a spinner whirl around in circles from just the force of an "ion wind." You can impress both family and friends by making a fluorescent lamp glow by simply holding it in your hand! It makes a great science project and will provide hours of fascinating demonstrations that teach electrostatic concepts.

How the Generator Works

The process of creating static charges when dissimilar insulating materials are rubbed together is known as the triboelectric effect. This project creates its charge by passing a rubber belt over two plastic rollers. Refer to the simplified diagram in **Figure 1**. The belt is moved on the rollers by means of a small DC motor. The moving contact between the rubber and the lower nylon roller causes the roller to acquire a positive charge. This positive charge then begins attracting an opposing charge from the lower brush. The attraction causes the metal brush to deposit negative charges onto the belt which carries these charges upward.

When the negatively charged belt reaches the upper Teflon® roller, the opposite process occurs. The upper roller acquires a negative charge which repels the similar charges riding on the belt, forcing them onto the upper brush. At the same time, the upper brush deposits positive charges onto the belt as it travels downward.

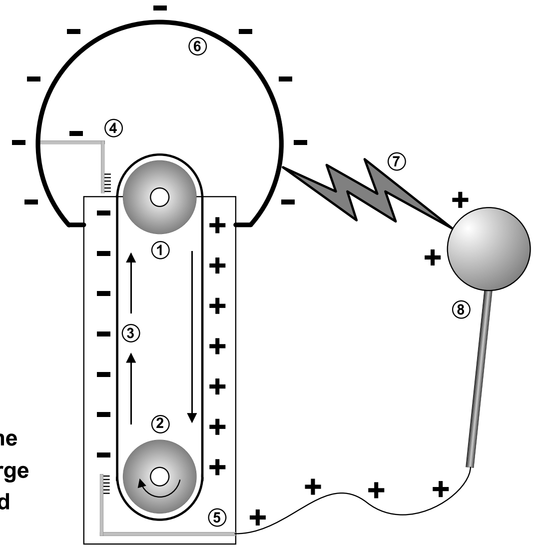
The upper brush conducts the negative charges to the metal collector dome. They collect and spread out over the surface of the dome until the voltage is great enough to jump to a nearby object, such as the discharge rod shown in **Figure 1**. This can be seen as a large spark similar to that of a lightning discharge, only on a much smaller scale.

Construction

As you look through the **Parts List**, you will see that this project uses fairly common materials, but it does involve a lot of fabrication. As an option, you can purchase a complete kit of fabricated parts. Details are shown in the **Parts List**. You can also view a short video of this amazing generator in action at **www.techkits.com/#vdg**.

Start construction with the phenolic plastic case described in the **Parts List**. This will become the generator's base and will house the motor, lower puller, and lower brush. Carefully drill the five holes at the locations shown in **Figure 2**. Then, file or cut a notch at the rear for the wall transformer cord to exit. To make the supporting column, cut the two

FIGURE 1.



Safety

Small Van de Graaff (VDG) machines are actually very safe. The voltage potential developed is no more dangerous than the shock you get from rubbing your feet across the carpet and touching a doorknob. They produce a maximum continuous current of less than 10 micro-amps and the energy is limited by the small capacitance of the charge collector dome. However, the sparks can create a sudden surprise. The immediate danger is more from sudden motion due to overreaction than from the electric discharges. When operating a VDG, the following rules should **ALWAYS** be followed:

- **Do not allow children to use the Van de Graaff generator unsupervised.**
- **Never use your VDG to electrify a Leyden jar or other capacitors. They can store extremely dangerous amounts of energy and create an electrocution hazard.**
- **People with heart conditions, cardiac pacemakers, or other medical implants should never operate the generator or come in contact with it. Discharge of static electricity could cause the electronic device to be damaged or to malfunction.**
- **Do not operate a VDG next to any electronic devices. The ionized air around the VDG and the discharges can permanently damage or destroy electronics such as cell phones, computers, MP3 players, and watches.**
- **Do not operate a VDG around flammable liquids or gasses. A spark from a VDG might trigger a fire.**

PARTS LIST

ITEM (PART#)	DESCRIPTION	SOURCE
Base	Keystone 705 Phenolic Enclosure	Digi-Key Corp. (705K-ND)
Tip Jack	Tip Jack, Female, Panel Mount	Digi-Key Corp. (J103-ND)
Inner Column	Acrylic tube, 1" OD, 1/16" Wall	Tap Plastics
Outer Column	Acrylic tube 1-1/4" OD, 1/8" Wall	Tap Plastics
O-ring	O-ring, 13/16" ID, 1" OD, 3/32' Wall	McMaster Carr (1879T34)
Belt	Rubber Band, Size 74, 3-1/2" L X 3/8" W	McMaster Carr (12205T81)
Lower Roller	Nylon Rod, 3/8" OD	McMaster Carr (8682K821)
Upper Roller	Teflon Rod, 1/4" OD	McMaster Carr (8547K231)
Upper Shaft	Steel Rod, Size 31, 0.120" Dia.	McMaster Carr (88725K54)
Upper and Lower Brush	Brass Strip, .016" Thick, 1/4" Width	McMaster Carr (8859K661)
Motor	Motor, 12V DC	Jameco (232013)
Motor Clamp	Plastic 3/4" Pipe Clamp, For 1-1/16" OD	
Collector Dome	Metal Globe, two piece, 1-3/4" Dia.	
Power Supply	Wall Transformer, 12V DC	

Miscellaneous
Aluminum cover panel, #4-40 screws and nuts, #10-32 screws and nuts, terminal lug, 16 gauge wire, silicon tubing.

Parts Suppliers

Digi-Key Corp.	**www.digikey.com**
Jameco Electronics	**www.jameco.com**
Tap Plastics	**www.tapplastics.com**
McMaster Carr	**www.mcmaster.com**

A complete kit of pre-fabricated parts can be purchased for $49 from:
LNS Technologies
PO Box 501
Vacaville, CA 95696
www.techkits.com
Email: LNSTECH@TECHKITS.COM

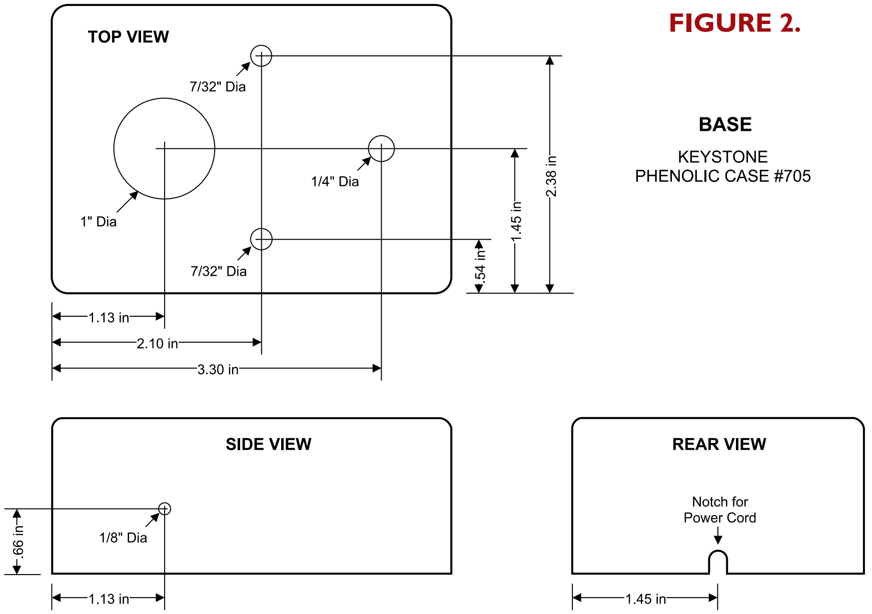


FIGURE 2.

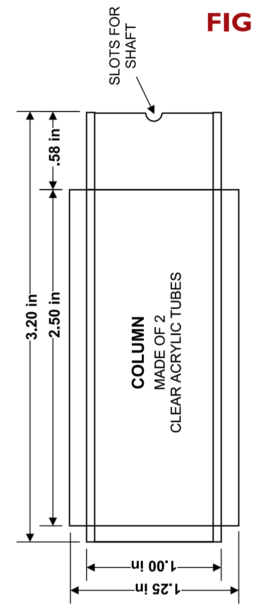


FIGURE 3.

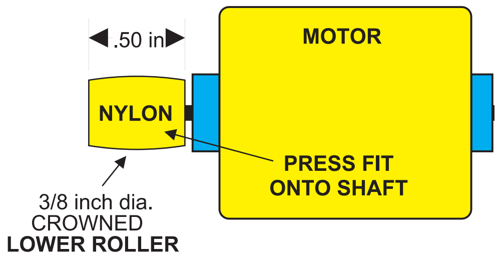


FIGURE 4.

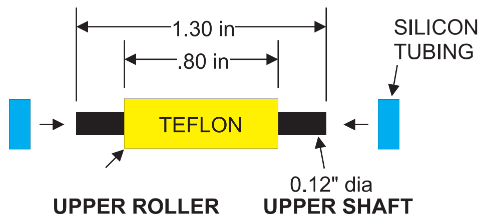


FIGURE 5.

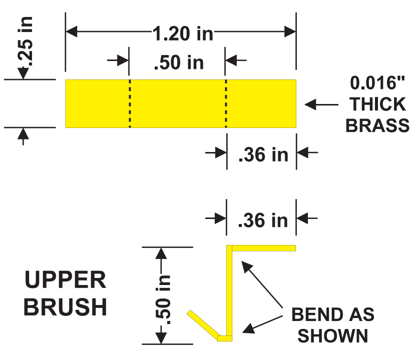


FIGURE 6.

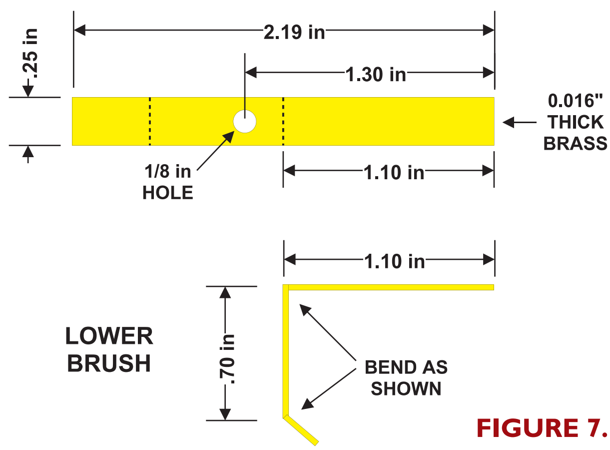


FIGURE 7.

acrylic tubes to the lengths shown in **Figure 3**. Cut two equal notches in the end of the smaller tube that will hold the upper roller shaft. Slide the smaller tube inside the larger tube to the position shown in **Figure 3**.

To make the lower roller, start with a 3/8 inch diameter nylon rod and cut it to the length shown in **Figure 4**. Measure the diameter of your motor's shaft and drill a slightly smaller center hole in the roller so it will

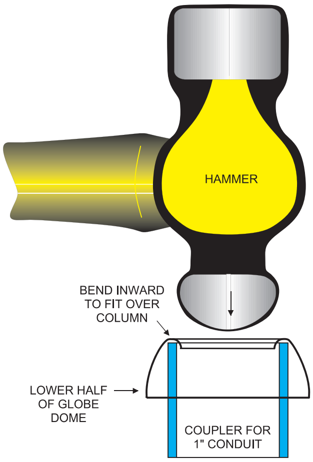


FIGURE 8.

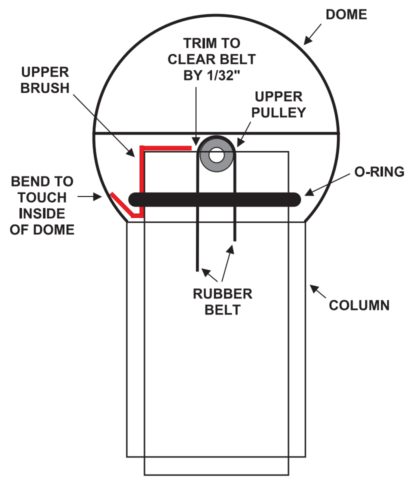
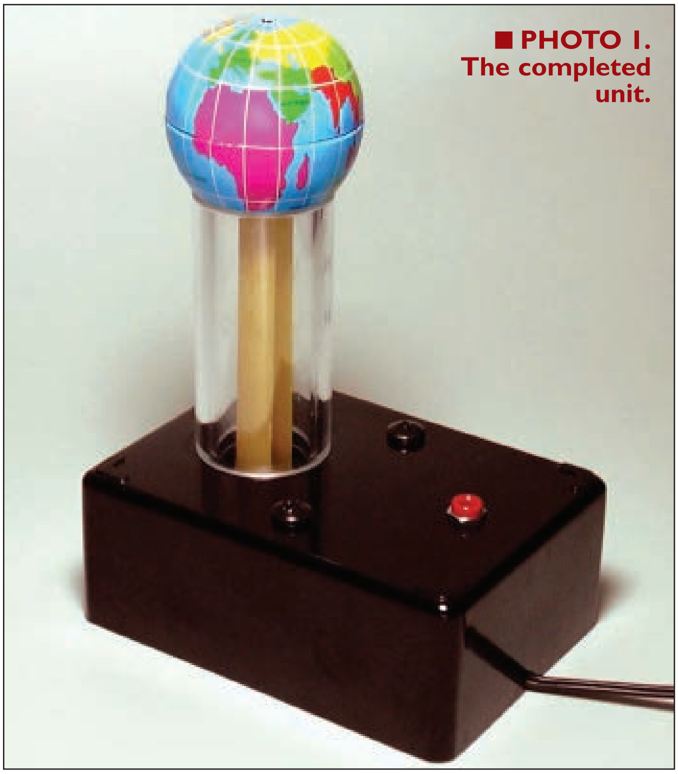


FIGURE 9.



■ PHOTO 1. The completed unit.

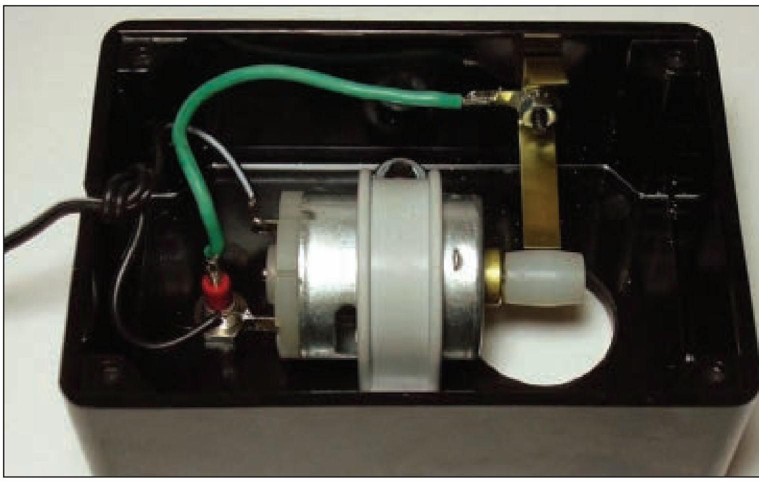
press snugly onto the shaft. The lower roller needs to be crowned in order to keep the belt centered when the motor runs. If you don't have access to a lathe, you can press the roller onto the motor, then run the motor and form the crown with a file.

The upper roller doesn't need a crown, so simply cut a 1/4 inch diameter Teflon rod to the length shown in **Figure 5**. Center-drill a 1/8 inch hole in the upper roller. That will allow it to spin freely on the 0.12 inch upper steel shaft. Refer again to **Figure 5** and cut the steel shaft to the length shown. The upper and lower brushes are made from 1/4 inch wide strips of 0.016 inch thick brass. The brass strips are available in most hobby shops or can be cut from shim stock. Trim the brass strips to the proper lengths and bend as shown in **Figures 6** and **7**. Be sure to drill the hole in the lower brush before bending to the final shape.

The collector dome is a metal globe from a pencil sharpener. Remove the globe from its plastic base and separate the two halves with a bit of prying. One half of the dome needs to slide down over the one inch OD acrylic column. One approach would be to drill or punch a one inch hole, but that would leave a sharp metal edge which can cause excessive corona leakage. Make a 15/16 inch hole, then utilize a ball-peen hammer and a metal conduit coupler as shown in **Figure 8**. Use the rounded end of the hammer to gently bend the edge of the hole inward to form a rounded opening that will fit over the column. It is not necessary to remove the paint from the globe unless you desire the classic bare metal look. Operating with or without paint on the dome showed no significant difference in performance.

Assembly

Once all of the important pieces have been fabricated, it is time to assemble everything together. First, press the lower roller



■ PHOTO 2.

onto the motor shaft and mount the motor inside the base using the motor bracket and two #10-32 x 5/8 inch screws and nuts. Position the motor so that the lower roller is centered below the one inch hole in the base. Install the tip jack in the 1/4 inch hole of the base and secure it with the hex nut.

The tip jack provides a convenient place to attach a grounding wire or a discharge rod. Then, mount the lower brush inside the base. Use a #4-40 x 3/8 inch screw and install a terminal lug on top of the brush and under the hex nut.

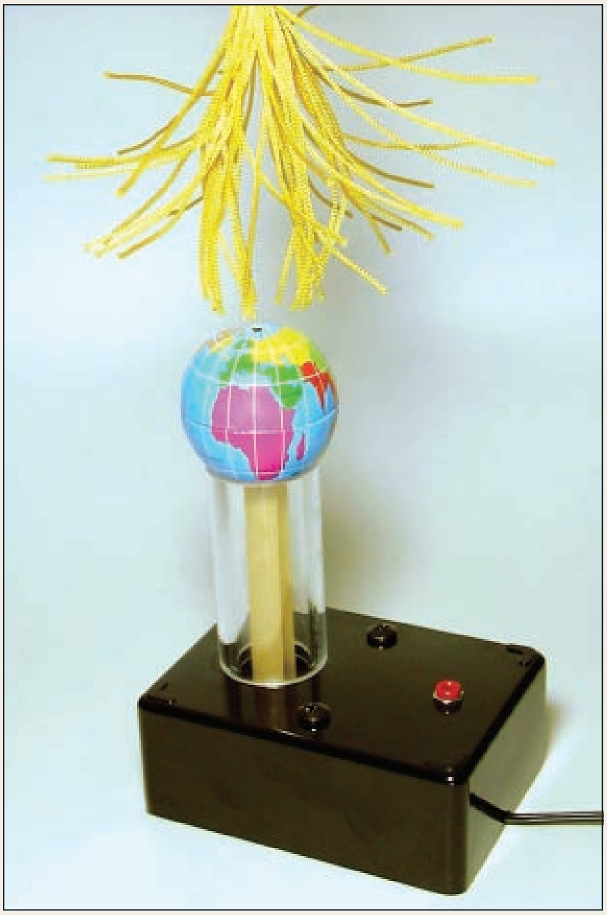
Next, make an electrical connection between the brush and the tip jack by soldering a three inch piece of wire from the jack to the terminal lug. Finally, attach the power wires from the 12 volt DC wall transformer to the two motor tabs. Before soldering, verify that the lower roller is turning in the correct direction. Refer again to **Figure 1** and observe that the belt must travel towards the lower brush, then up the column. If the direction is wrong, simply swap the two wires, then solder. The completed base assembly is shown in **Photo 2**.

Move to the column and upper assembly as shown in

Experiments and Demonstrations

To create lightning-like sparks, construct a discharge rod similar to what is shown in **Figure 1**. Solder a smaller metal globe onto a three inch piece of 12 gauge solid copper wire. The 12 gauge wire fits perfectly into the tip jack and can be bent to adjust the gap from the collector dome. If the belt and rollers are clean and the room's humidity is 30% or less, your Van de Graaff generator should be able to create continuous sparks of up to 3/4 inches long.

You can illuminate fluorescent bulbs or tiny neon tubes. For best results, do these experiments in a darkened room or at night. Bring the bulb toward the dome as the



generator is operating. You may wish to make a non-conducting holder for the light bulb to avoid receiving a shock as you approach the dome.

Another fun demonstration involves bouncing balls. You can purchase small fuzzy fabric balls called pom-poms at most craft or sewing stores. Tie one of the balls on the end of some thread and suspend it to touch the collector dome. As soon as it acquires a negative charge from the dome, it will immediately swing away with a strong repelling force. Or, better yet, try two of the balls at once and watch them dance as they interact with each other.

You can also cut several thin strips of tissue paper and attach them to the dome with a small piece of cellophane tape. With the motor on, they will become charged and will stand on end since like charges repel each other. They will be attracted to your finger if you bring it close. They will even remain upright for a while after the generator is turned off, as the charge slowly leaks away.

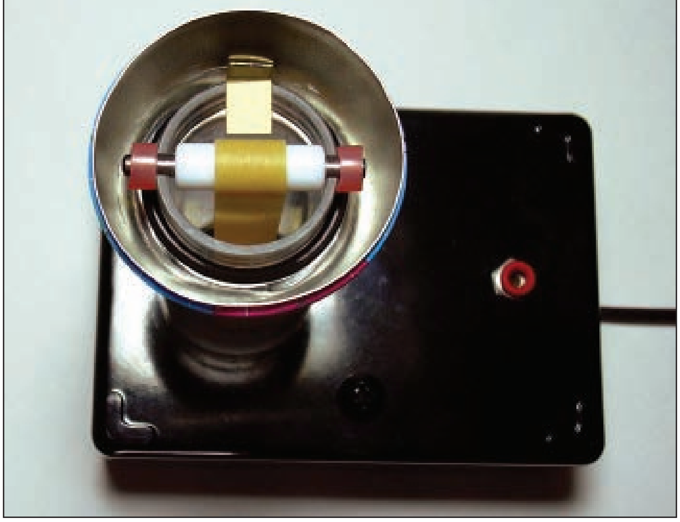
Figure 9. Slide the half of the dome with the hole onto the top of the inner column and press it all the way down to rest against the outer column. Position the upper brush between the column and the dome as seen in the **figure**. Stretch the o-ring to go around the brush and column to hold the upper brush in position. Insert the upper shaft into the upper roller and put the rubber belt around the upper roller.

Set the upper shaft into the two notches at the top of the column with the belt hanging down inside the column. Then, slip two small pieces of silicon tubing over the ends to keep it positioned in the column notches. Adjust the upper brush position to make it exactly perpendicular to the upper roller. The completed upper assembly is shown in **Photo 3**.

Now, place the column and upper assembly into the hole on the base and loop the belt around the lower roller. Be sure that the upper brush is directly above the lower brush on the same side of the rubber belt. If needed, rotate the column to make sure the upper roller is parallel to the lower roller so the belt is not twisted when the motor runs. Once the Van de Graaff generator is working well, you can apply a drop of glue to keep the column from rotating in the base.

This is a good time to check the gap between the brushes and the belt. They are not supposed to touch the belt. Conversely, if the gap is too large, the generator will perform poorly or not at all. The recommended gap for each is 1/32 inches. Fasten the aluminum plate and the four rubber feet to the bottom of the base using four #4-40 x 3/8 inch screws. Set the generator upright and complete the assembly by rejoining the two halves of the globe dome.

■ **PHOTO 3.**



Maintenance

A belt or rollers that have become dirty can reduce the performance of the generator. If necessary, wash the parts with soap, and then rinse and dry thoroughly. Also check for too much clearance between the belt and upper and lower brushes. Another common cause of poor performance is high humidity. Dry the inside of the column and globe gently with a hair dryer to remove excess humidity inside the machine. Be careful, because too much heat can crack or melt the plastic column.

Any build-up of dust or lint on the collector dome or on the column can cause the high voltage charge to leak away. Clean the dome, column, and base with a lint-free cloth. Lastly, the rubber belt should be replaced if it becomes brittle, cracked, or oxidized.

Wrap-Up

This project will surely be a hit with audiences of all ages and it is guaranteed that everyone will get a *charge* out of watching it! **NV**

continued from page 33

tasks and connect programming environments to real-world hardware prototypes. GP-241xx supports many input formats from binary to ASCII file, and also features an advanced C/C++ interface (through DLL function call) which allows easy reuse of existing tabular input patterns.

For designers, the GP-241xx is an essential stimulus source and a potentially enormous time-saver for almost any type of digital device or application: digital and mixed-signal ASIC, FPGA, microprocessors, and microcontrollers. GP-241xx can be used early in the design cycle to substitute for system components that are not yet available. For example, it might be programmed to send interrupts and data to a newly developed bus or circuit when the chip is still being designed. Or, when a radio frequency chip is received from the fab after manufacturing, its analog front-end can be tested and characterized by programming parameters through a digital interface and measuring the resulting behavior of the chip. Applications for the GP-241xx include: versatile ADWG (Arbitrary Digital Waveform Generator), digital pattern generator, and Logic Analyzer; board access through serial protocols SPI, I²C, I²S, and RS-232; 8/16 bit microcontroller interfacing; bus master emulation; SRAM and Flash memory interfacing; system and prototype debug; IP evaluation; digital recorder/player tools; JTAG (IEEE 1149.1) access; repetitive DAC control; exercising custom IP in an FPGA or sending specific bit-patterns in real-time to ASIC designs at up to 100 MHz and capturing output responses; testing and characterizing RF IC's analog circuitry through sequential SPI programming; validation of SoC image processing functionality by custom interfacing to system buses; intercepting microcontroller/FPGA data traffic during problematic code execution for faithfully reproducing faults; extracting SPI data for debug, modifying functional system parameters in real-time; running automatic tests and collecting trace values, etc. Made in Europe by Byte Paradigm, GP-24116 and GP-24132 are available now at $1,399 and $1,499, respectively from Saelig Co., Inc.

For more information, contact:
Saelig Company, Inc.
Tel: **888-7-SAELIG**
Email: **info@saelig.com**
www.saelig.com

continued on page 59

Introducing CHIPINO

the Bridge Between PICs and Arduinos

By The CHIPAXE Team

There are so many different options for building an electronic project, it would be nice if they all worked together in some way with some kind of standard connection scheme. The popularity of the Arduino module has essentially created one popular standard with their plug-in modules called shields. Unfortunately, that standard has a unique pin-out spacing that makes it difficult to directly plug a shield into a breadboard. Because of this, users are essentially locked into using them with an Arduino base, and most of them use the Atmel microcontroller. This is fine unless you are a Microchip PIC user and have your favorite set of tools to program your PICs. You probably don't want to switch to the Arduino module. This is where the new CHIPINO module comes to the rescue to help expand the use of this connection standard, and get more people from different micro choices to share shields.

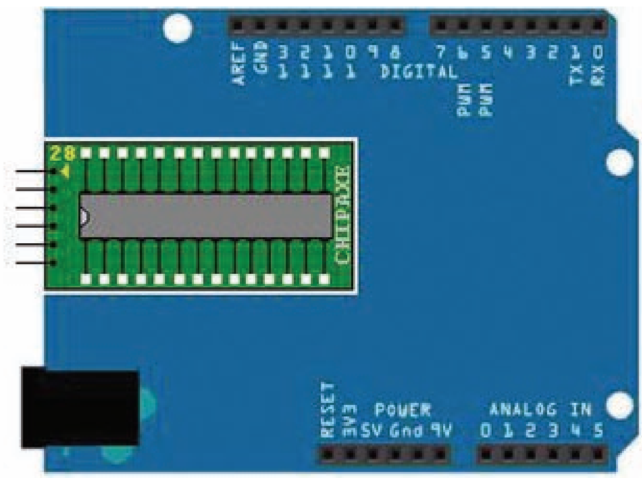


FIGURE 1. CHIPAXE adapter board sketch.

The CHIPINO is a PIC-based module with the Arduino connection scheme. The CHIPINO was developed by the Chipaxe team of **chipaxe.com**. It was designed specifically to match the board outline, mounting holes, connector spacing, and most of the microcontroller I/O functions found on the popular Arduino module. It offers a PIC user the opportunity to use their existing compiler and programming tools with all the great shields available to the open source world of Arduino.

The original idea we had was to create an adapter board for our popular CHIPAXE modules, similar to the concept sketch shown in **Figure 1**. This was quickly thrown out as the height of the CHIPAXE module would hit the bottom of the shield being stacked on top. We also liked the idea of having a built-in voltage regulator on the base, so we designed the CHIPINO module shown in **Figure 2**. The other choice we made was to eliminate the bootloader programming method used in Arduino. At first, you might think this is a mistake since many people tend to think using a programmer is expensive and difficult.

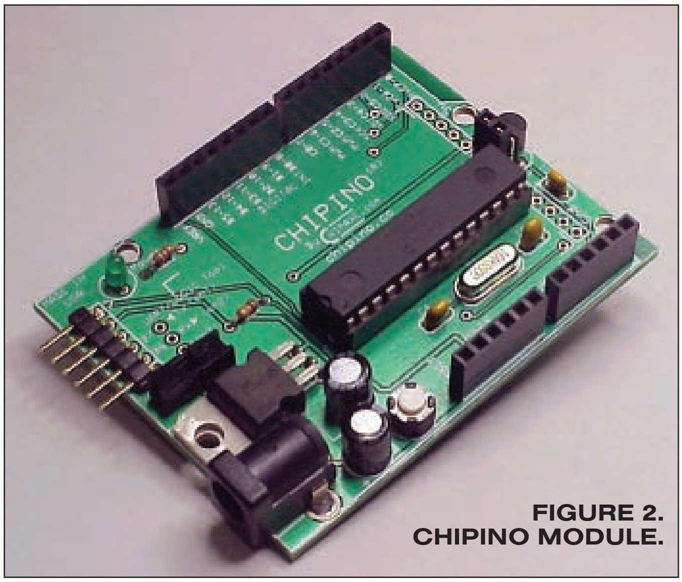


FIGURE 2. CHIPINO MODULE.

Truth is, a PICkit 2 programmer is open source so there are many clones out there at very affordable pricing. You can typically get a PICkit 2 clone programmer for around the same price as a USB to RS-232 converter cable (which some bootloader schemes require). And, using a programmer actually has advantages over a bootloader which is why we chose this method of programming for the CHIPINO. Because you don't need a micro with the custom bootloader software pre-programmed in, you can plug any blank PIC16F or PIC18F 28-pin 0.300" pitch microcontroller into the CHIPINO and program it in Assembly, BASIC, C, Pascal, Flowcode, or any other compiler that supports the PIC. This means your choices are nearly endless for how you want to use this module with an Arduino shield.

The PICkit 2 programmer also allows you to debug code with single stepping and setting breakpoints when you use it with Microchip's MPLAB IDE. The PICkit 2 also has features such as an RS-232 terminal and logic tool, so using the two programming pins as feedback connections, the CHIPINO can help debug itself. Finally, to show we just want to do our part for the electronics hobbyist community, we decided to open source the design by releasing the board layout, schematic, and parts list.

The openness of this concept also lets you choose whatever IDE you want that works with your programming tools. The programming connection will work with many different Microchip programming tools. For the Integrated Development Environment (IDE), we like the MPLAB IDE which is free to download and will automatically install the HI-TECH C compiler that supports the 16F micros, including the 16F886 we use in our version of the CHIPINO. We also received an early look at a library of C functions developed for the PIC16F886 using the HI-TECH C language compiler, by Chuck Hellebuyck for his soon to be released *Beginner's Guide to Embedded C Programming – Volume 3* book.

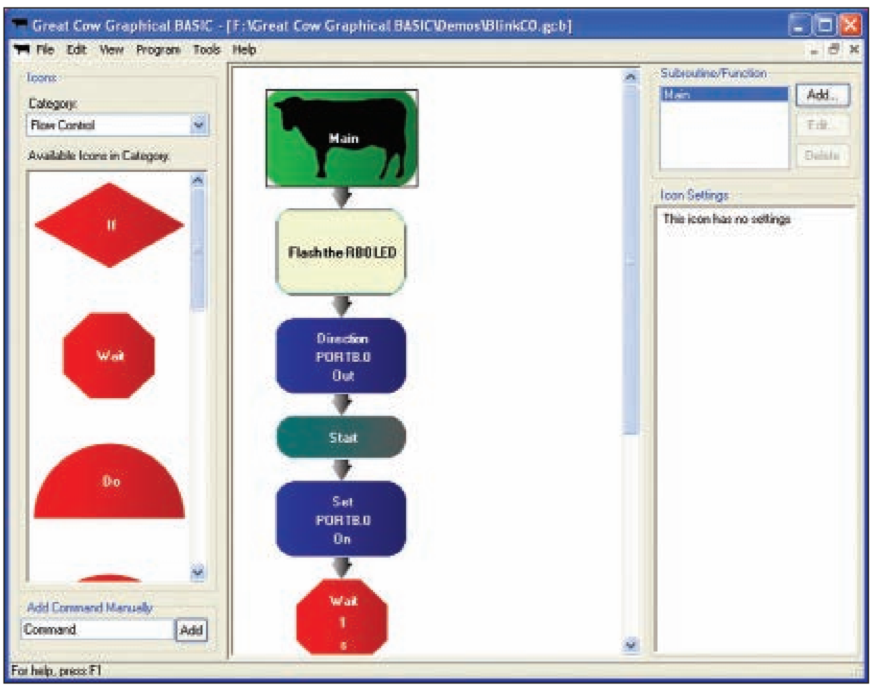
The CHIPINO name comes from an Italian fishing town in San Francisco where fishermen were asked to "chip in" at the end of the day from their daily catch. The result was a community soup (Chipino soup) everyone could share. We thought CHIPINO — the electronic module — does the same thing for the PIC community by offering an open source platform to build upon and 'chip in' their own application ideas. Users can submit their own application code and shield designs for the CHIPINO community to share. We've also created a website dedicated to the CHIPINO project at **chipino.cc**.

Great Cow Graphical BASIC

We wanted to find an open source Basic compiler for CHIPINO. Great Cow Basic was our first choice since we were already working on converting to it for our CHIPAXE modules. This open source compiler has really impressed us in a favorable way. It started out as a text-based compiler and continues with that theme today, but added another optional graphics layer on top of that. The documentation is limited, but after a little bit of trial and error, we were able to get many projects running on a CHIPINO board using this software.

You can easily switch between graphical mode and text mode. There is a great forum on this compiler with many users, so help is out there. The best part is this compiler is completely free to use. No code limitations or up-sell to a pro version. Great Cow Basic will allow you to program most eight-bit PICs including 10F, 12F, 16F, and

FIGURE 3. GREAT COW GRAPHICAL BASIC SCREENSHOT.



18F chips. Great Cow's development software application allows you to easily use a PICkit 2 programmer for one click compile and program, so using this with our CHIPAXE or CHIPINO modules is a natural fit. Here are some of the features listed on their website for this compiler:

- Standard Basic flow control statements — If, Select Case, Do, For, Goto.
- Support for multiply, divide, add, subtract, Boolean operations, and comparisons.
- Bit, byte, word, and string data types, in addition to byte arrays.
- Subroutines and functions.
- Inline assembly; in most cases, without any special directives.
- Data tables.
- Automatically recalculates all delay commands, depending on the clock speed of the chip.
- Generates standard MPASM compatible assembly code for PICs.

I/O Capabilities:

- Standard 2x16 LCD routines.
- Routines for on-chip A/D, PWM, SPI, USART, EEPROM, and timers.
- RS-232 communications — rates between 300 and 19200 bps with user configurable parity, start, and stop bits.
- PS/2 keyboard reading.
- 4x4 keypad.

Notice the feature that allows you to write functions. We really like this one because it allows you to customize the compiler for the CHIPINO with your own unique commands. For more info or to download it, visit the Great Cow Basic website at **http://gcbasic.sourceforge.net/index.html**.

Simple C Library

One of the features people like about Arduino is the free C code compiler that makes programming the module easy. PIC users have had this type of easy programming language for many years. The CCS and BoostC compilers offer these simplified commands or functions that make programming easier. If you like graphical programming, Matrix Multimedia's Flowcode gives you that option. Unfortunately, none of these options were free or open source like the Arduino.

The HI-TECH C compiler included with MPLAB is free and also not code limited, but it lacks a library of pre-written functions for controlling LCDs, ADC, and other popular options. Chuck's Simple C library (mentioned previously) appears to be a great start at what we wanted for the CHIPINO. He uses the PIC16F886 in his book, and shows how he developed and how to use this set of common functions to control ADC, LCDs, SPI, and I^2C, to name a few. As a courtesy from Chuck, here is the list of functions and compiler features. His idea is to open source the library so anybody — including the CHIPINO community — can help expand this library to offer more features everyone can use.

Structure

main( )

Control Structures

if
if...else
for
switch case
while
do... while
break
continue
return

Further Syntax

; (semicolon)
{} (curly braces)
// (single line comment)
/* */ (multi-line comment)

Arithmetic Operators

= (assignment)
+ (addition)
- (subtraction)
* (multiplication)
/ (division)
% (modulo)

Comparison Operators

== (test for equal)
!= (not equal to)
< (less than)
> (greater than)
<= (less than or equal to)
>= (greater than or equal to)

Boolean Operators

&& (and)
|| (or)
! (not)

Compound Operators

++ (increment)
— (decrement)
+= (compound addition)
-= (compound subtraction)
*= (compound multiplication)
/= (compound division)

Variables

Constants

#define

Data Types

bit
char
byte

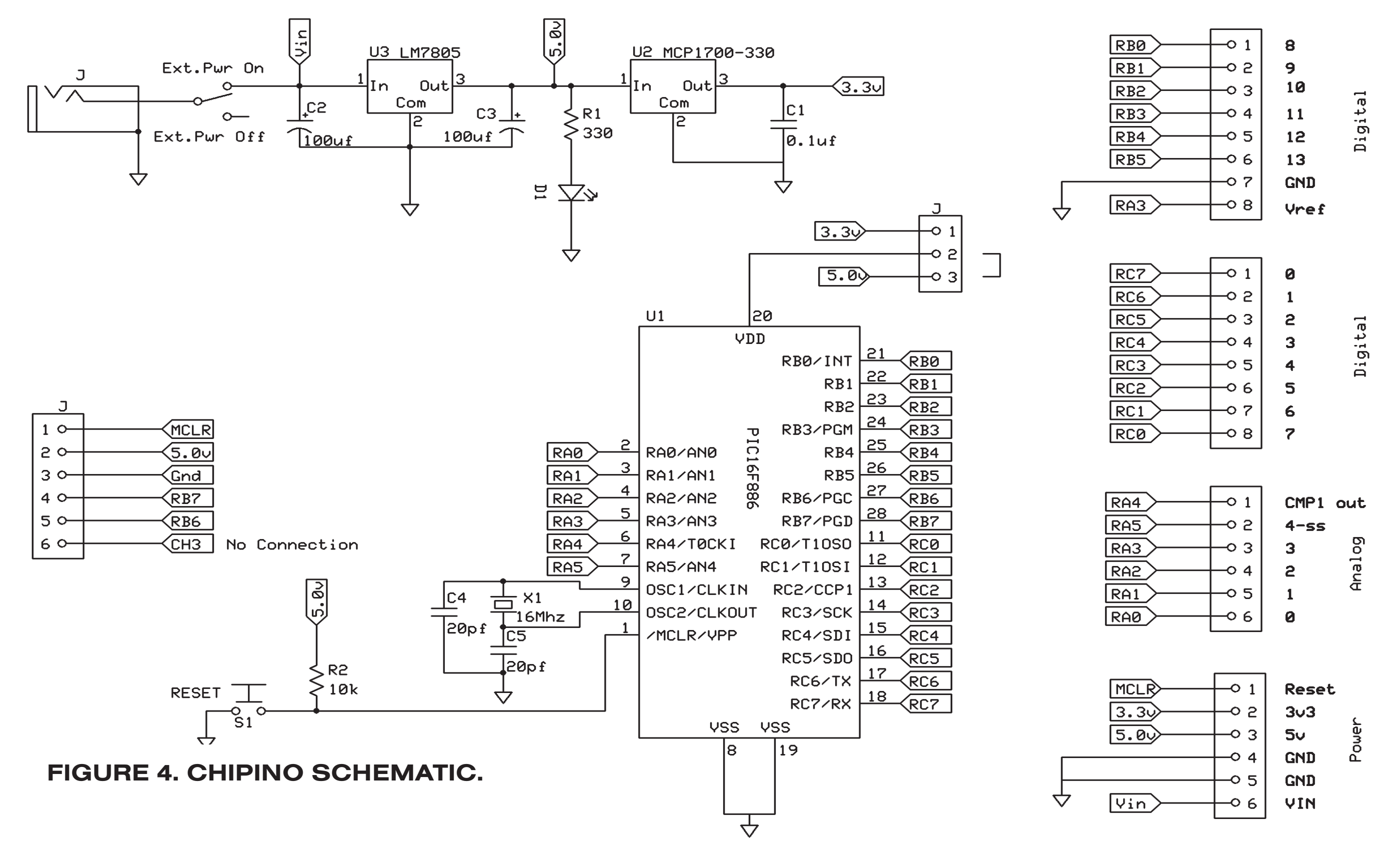


FIGURE 4. CHIPINO SCHEMATIC.

int
unsigned int
long
unsigned long
float
double
string
array
void

Functions

Digital I/O
- high(*pin no.*)
- low(*pin no.*)
- *var* = input(*pin no.*)

Analog I/O
- adcin(*pin no.*)
- pulseout(*pin no., period*)

Delay (val = 1 to 65535)
- pause (*val*)
- pauseus (*val*)

Hardware PWM (1 kHz to 32 kHz)
- pwmout1(*duty cycle,freq*)
- pwmout2(*duty cycle,freq*)

Hardware SPI
- spi_init( )
- spi_out(*byte*)
- wr_byte(*address,data*)
- var = rd_byte(*address*)
- wr_MCP23S08(*register,data*)
- *var* = rd_MCP23S08

Hardware I²C
- i2c_init( )
- i2c_start( )
- i2c_repstart( )
- i2c_stop( )
- *var* = i2c_read( )
- i2c_waitforidle( )
- i2c_write(*data*)
- i2c_eeout(*address,data*)
- *var* = i2c_eein(*address*)

Hardware USART (2400 or 9600 baud)
- init_serial(*baud*)
- putch(*char*)
- *var* = getch( )

LCD (2x16)
- lcd_init( )
- lcd_clear( )
- lcd_goto(*position*)
- lcd_text(“*characters*”)
- lcd_symbol(*char*)
- lcd_number(*3digit int*)

Hardware

We went through every pin and tried to match the exact same pin-out as Arduino, but we ran short of a few PWMs and the I²C pins are in a different location on the

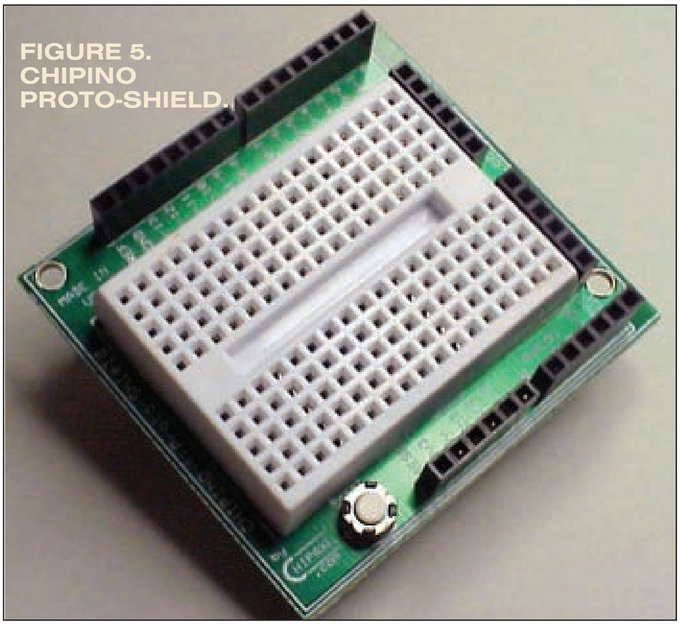


FIGURE 5. CHIPINO PROTO-SHIELD.

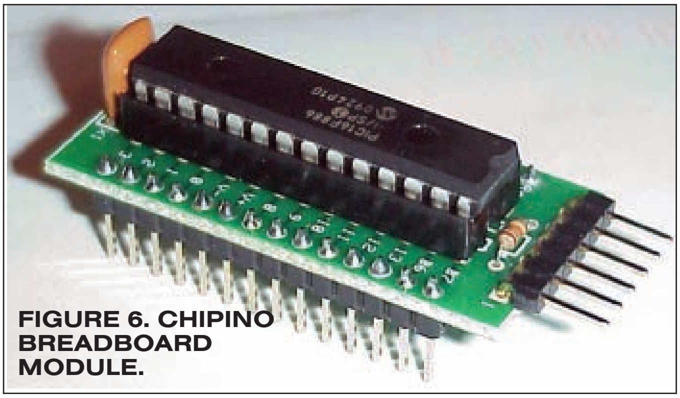


FIGURE 6. CHIPINO BREADBOARD MODULE.

PIC16F886. This may be fixed in the future as Microchip has released some new parts that offer more PWMs and extra I²C ports, so look for a possible version 2.0 CHIPINO in the future. The schematic for the CHIPINO is shown in **Figure 4**. The CHIPINO was designed to use all leaded parts, so it's easy to build for any electronic hobbyist (even a caveman could do it). In fact, we offer it in kit form for that purpose. The kit can be used in schools to teach soldering and then programming. You can also get just the blank circuit board and build it the way you want. Because the module can be programmed in Basic, it can be used at the entry programmer level. When your project is ready for mass production, CHIPINO makes that transition painless.

CHIPINO Shields

We have plans to create our own set of shields with the features we don't see today. This is the fun part since we really enjoy creating circuit boards. We also open sourced the CHIPINO board files, so a user can create their own shields with less effort. The CHIPINO board files are available at the **chipino.cc** site. The proto-shield shown in **Figure 5** is the first official CHIPINO shield.

CHIPINO Breadboard Module

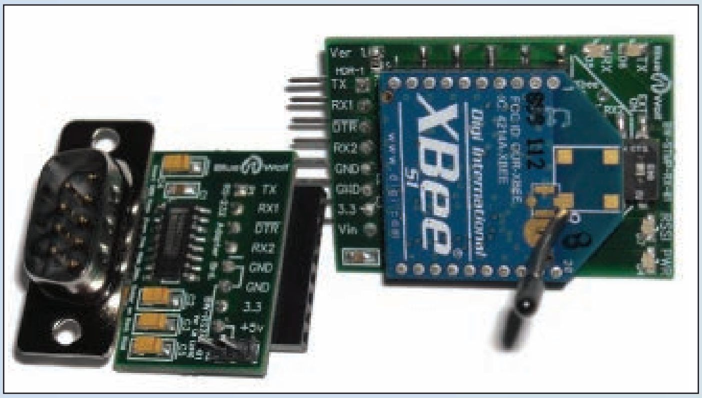
We didn't forget the true CHIPAXE users. If you like to use a breadboard rather than these shields but want to share code with a CHIPINO user, we modified our 28-pin CHIPAXE module to add a 16 MHz resonator and 10K pull-up resistor (**Figure 6**). We also relabeled the pins to match those of the CHIPINO/Arduino.

Conclusion

Getting Microchip users on board with Arduino shields will help expand this defacto standard and get more people sharing ideas and projects with each other. That is the CHIPINO's ultimate goal. **NV**

continued from page 51

BASIC STAMP WIRELESS **DOWNLOADING**



Blue Wolf, an engineering and design company, is introducing a new line of wireless robotic products. Their newest product is FlashFly which is an innovative system that allows a user to remotely download BASIC Stamp programs to Parallax's Stamp modules or interpreter chips. FlashFly consists of three interdependent modules: the remote receiver board; an optional RS-232 adapter board; and the XBee USB base transmitter board.

FlashFly's wireless capabilities — in coordination with common Series 1 XBee modules — will allow a user with a BASIC Stamp mobile robot or stationary platform to modify his or her program remotely. FlashFly eliminates the tedious task of having to connect a robot to a computer before any programming changes can be made. FlashFly also enriches a user's learning experience by providing instant feedback data to the DEBUG terminal screen which allows a user to evaluate his or her program flow and I/O data remotely. FlashFly can also be used in conjunction with any other microprocessor platform that needs to transmit data wirelessly.

Blue Wolf designed FlashFly with versatility and convenience in mind for both hobbyists and educational users. The (1x8) 0.1″ inline header makes breadboard use simplistic. Alternatively, an RS-232 converter board can be plugged in for direct connection to a DB9 female connector on a remote or mobile robot platform. Furthermore, FlashFly can be used for more advanced XBee experimentation by directly connecting additional wires or headers to the two rows of 0.1″ plated holes on FlashFly. Any existing Series 1 XBee modules can be easily reconfigured to work with the flexibility and modularity of FlashFly.

FlashFly features include:

- Onboard 3.3V regulator.
- Small footprint for mobile robotic applications or fixed platforms.
- LEDs for indication of TX, RX, RSSI, and POWER.
- Uses Parallax's BASIC Stamp Editor software for program downloading (no additional steps required).
- Two rows of 11 plated through holes with 0.1″ spacing allow the option of soldering jumper wires or headers (not included) for easy access to the remaining XBee module pins for more advanced designs.
- Designed for the option of using XBee Pro modules if a larger wireless range is required.
- Low cost, flexible use system that can be used on many different platforms.

FlashFly will begin shipping in March '11.

For more information, contact:
Blue Wolf
Email: **info@bluewolfinc.com**
Web: **www.bluewolfinc.com**

SHOW US WHAT YOU'VE GOT!

Is your product innovative, less expensive, more functional, or just plain cool? If you have a new product that you would like us to run in our New Products section, please email a short description (300-500 words) and a photo of your product to:

newproducts@nutsvolts.com

#32 SMILEY'S WORKSHOP

C PROGRAMMING - HARDWARE - PROJECTS

avrtoolbox — Organizing an Open Source Project

by Joe Pardue

Recap

Last month, we learned how to create an open source project on Google Code using avrtoolbox as an example [**http://code.google.com/p/avrtoolbox**]. We also learned how to manage the software versions using TortoiseSVN. This month, we will look at the design philosophy for avrtoolbox, and a project C coding and documenting style guide. One of the things we intend for avrtoolbox (remember this is a *we* project — 'collaborative' and 'open') is that it will be usable by everyone interested in AVR microcontrollers — from the greenest novice to the most experienced professional. Oh yes, we are ambitious!

We want an elementary library with tutorials designed for novices (someone who has never ever used a microcontroller or programmed in C) so they are easy to learn and use just like systems such as the Arduino or the BASIC Stamp. We want to build this elementary library from lower level libraries that can be easily expanded by advanced users who need professional quality AVR C programming tools. These goals may seem at odds — mixing novice and professional libraries in the same project — but they are actually just the ends of a programmer competence spectrum that doesn't have any formal dividers. All real people are somewhere in between. Oh, and a word of warning. Most of what follows this month is for avrtoolbox developers, so novices might want to skim a lot of this and consider it a sort of coming attractions featurette – but please do read the last section.

Design Philosophy Statement

avrtoolbox is a collection of open source **educational** tools for learning about and using Atmel AVR (eight-bit) microcontrollers with the standard Atmel AVR Toolchain: AVRStudio/WinAVR/avrdude.

Directories

- **libraries** – Useful C functions.
- **avr_applications** – Complete programs for the AVR.
- **pc_applications** – Terminals, IDEs, etc.
- **documentation** – avrtoolbox documents.
- **hardware** – Schematics and layouts.
- **miscellaneous** – Useful stuff that doesn't fit elsewhere.

Because the AVR architecture varies among sub-families, it would be very difficult to create compiled hardware libraries that could be run on all possible AVR variants. Our initial approach will be to create a single source code repository where all the code will be developed for and tested on three different AVRs available on inexpensive development boards. This will demonstrate concepts for writing code that can be compiled for multiple devices and leave it to the users to follow the methods shown to add additional devices as needed.

The three AVRs we will use are the ATmega169, ATmega328, and ATmega644; we will test our tools on three development boards: Butterfly (ATmega169), Arduino Duemilanove or Uno Boards (ATmega328), and the BeAVR (ATmega644)

For the Arduino board, this system does **not** use the Arduino IDE **nor** is it compatible with the existing Arduino libraries. The BeAVR – Breadboard enabled AVR – was shown in the May '10 *Smiley's Workshop.*

AVR Source Code Libraries

The lower level directory structures will evolve over time, but to begin, the libraries' directory will have five subdirectories.

libraries – Useful C functions.

- **elementary** – High level simple functions similar to those in Arduino or PBASIC.
- **general** – Useful embedded system software that is not AVR specific.
- **driver** – AVR specific software.
- **board** – Custom functions for development boards such as the Butterfly.
- **testers** – Test software for each library.

Library Organization

The term 'library' has a confused use with microcontrollers sometimes referring to source code listings and sometimes referring to object module archives. We hope to avoid confusion by using **source library** for the former and **archive library** for the latter.

Archive Library

Archive libraries are the simplest to use. The user needs only to include the library in a project and think of the individual library functions as black-boxes promised to work as described by the API (Application Programmer's Interface) document. These were discussed in the Jan '11 column.

Source Library

The source library is only intended for use by the avrtoolbox developers. However, the source code is made available to everyone since many folks will prefer to rip off, er, re-use the source in their own project rather than using the archive libraries.

AVR Libraries Directory Structure

The reader should be aware that it is early in the evolution of avrtoolbox and changes will take place over time. However, we have to start somewhere so the planned directory structure with the library names will be shown on the home page of the avrtoolbox project and updated as libraries are completed.

C Programming Style

The C programming guide is modified from the avrcore library suggested guide by Ruddick Lawrence at **http://www.nongnu.org/avr-libc/avr-corelib/style_guidelines.html**. We will try to use this guide as much as possible since we would like to have our software usable with the avr-corelib project. However, several good (or at least adamant) suggestions from AVRFreaks led to differences. [Link: **http://tinyurl.com/2uwjfrq**.]

Naming Conventions

- **lower_case**: Has all lower case letters, with words separated by underscores.
- **UPPER_CASE**: Has all upper case letters, with words separated by underscores.

Public Versus Private

We will conceptually divide our code into public and private modules, functions, and variables. The public materials are meant to be used in archive libraries and are documented in the API and the .h file supplied with the library. The private materials are those used to build the libraries and are not intended for casual users. However, the source code is available for advanced users.

Modules

Each module will have a short descriptive name such as spi or uart. Short module names is a subjective requirement with the example of spi_init() versus serial_peripheral_interface_initiation().

Files

Public files will use **lower_case** and will have the module name as a prefix. For example: spi.c and spi.h. Private files will have the first letter of the module in **UPPER_CASE**: Spi.c.

Functions

Public function names and arguments will all use **lower_case** and have the module name as a prefix: module_init(). Private functions will have the first letter of the module in **UPPER_CASE**: Module_init(). Module initialization functions will all have _init as a suffix. For example: uart_init(uint32_t baud_rate).

Variables

Variable names will all use **lower_case.** Keep global variables to a minimum and make sure they are well justified and documented.

Data Structures

Data typedefs use **lower_case** and end in _t (i.e., long_timer_t). Any member variables follow the guidelines for variables specified above.

Formatting

As a general guideline, code will use BSD Allman [**http://tinyurl.com/5tfhav**] formatting. Tab characters are allowed, and the suggested setting is four spaces for consistent appearance, especially in the documentation. Use braces for all statements that could have them, even for enclosing a single line. An example from Wikipedia:

```
while (x == y)
{
    something();
    somethingelse();
}
finalthing();
```

Constant Values

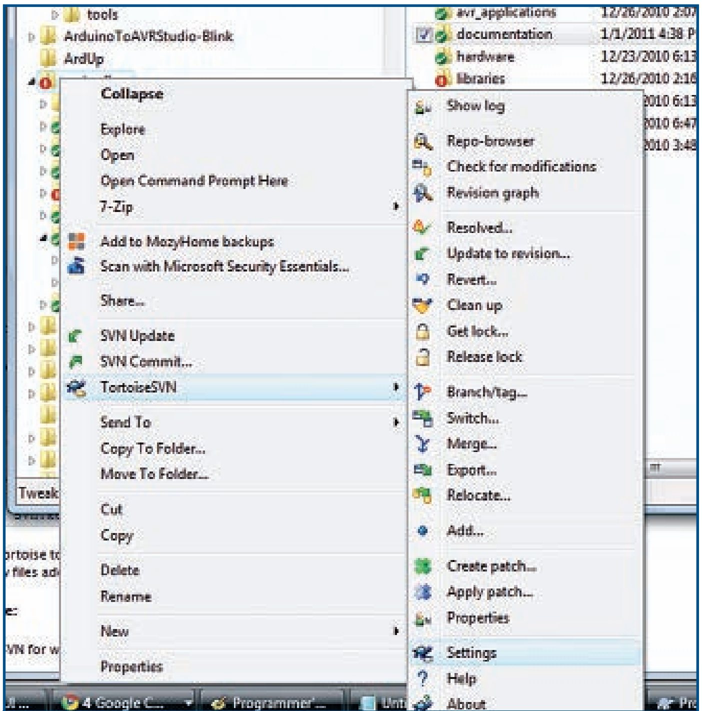
Public constants (in header files) use #define for single values because it will not reserve memory space. Private constants use static const values to limit the scope to the module. Constants will be **UPPER_CASE** and have the module name as a prefix. For example: #define UART_BAUD 19200.

Registers

Whenever possible, use the read/modify/write paradigm to change registers in order to avoid overwriting other parts of the register. This is best done by using standard bitwise operator techniques.

Scope

Although C has no "private" definition, any functions and otherwise global variables not meant to be used by client code should be declared static.



■ FIGURE 1. TortoiseSVN Settings.

API

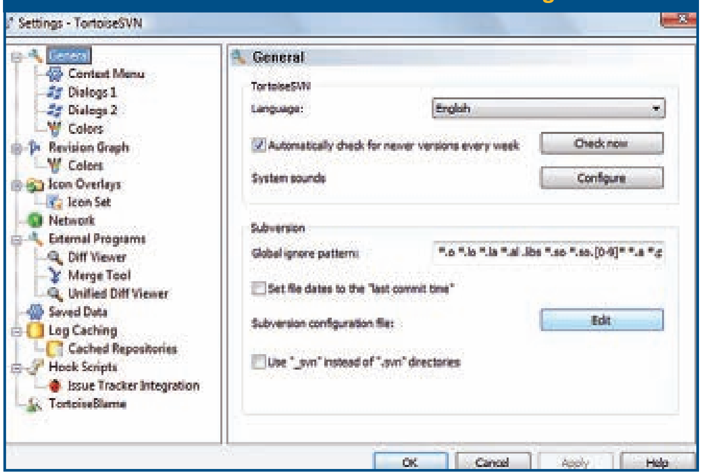
Each module's API is defined in a single header file, named after the module (i.e., uart.h).

C Program Documentation Style Guide

Documentation requirements vary with the type of user. A **developer** will be intimately familiar with the code and not need much other than the code itself. A **maintainer** should have the same level of programming skill as a developer, but will need information about the software that may not be obvious from reading the raw code. So, he will want in-line comments sufficient to help him quickly understand what the code is doing. A **user** may just want to use the functions and not care how they were generated, so all he'll need is an overview of the module and the specifics of how to use each function – what goes in and what comes out. Generally speaking, he doesn't care what happens in between.

This leads to two conceptually separable types of documentation: in-line comments in the C source for the developer/maintainer, and a separate API document for the casual user. The developer should just use common sense and let the obvious things be self-documenting, while adding comments on things that might not be immediately obvious. The casual user will need a bit more handholding, so we will use Doxygen comments in the public header file to generate the API document and will try to anticipate the real needs of an average user.

■ FIGURE 2. TortoiseSVN Select Edit Configuration.



Keeping Track of the Printed Documents

A big part of being organized is being able to identify what you've got. If you sort through the pile of papers on your desk and find three printed copies of widget.c, you'll probably want to know which one is the most current so you can toss the other two. SVN provides lots of information attached to the digital file on your PC and in the repository, but none of that information is in the text of the file unless you specifically tell SVN to add it (which is what we will learn in this section).

Is This the Latest Version of the File?

Select the avrtoolbox directory on your PC and right-click to open the TortoiseSVN menu. Then, select Settings as shown in **Figure 1**. Next, click the Edit button as shown in **Figure 2**. Finally, the configuration file will open in NotePad as shown in **Figure 3**. Scroll down to the ### Section for configuring automatic properties: #[auto-props]. Remove the '#' from the [auto-props] item. Next, scroll down and remove the '#' from the *.c and *.h item. Finally, copy the svn:keywords from the *.h item to the *.c item and you should have:

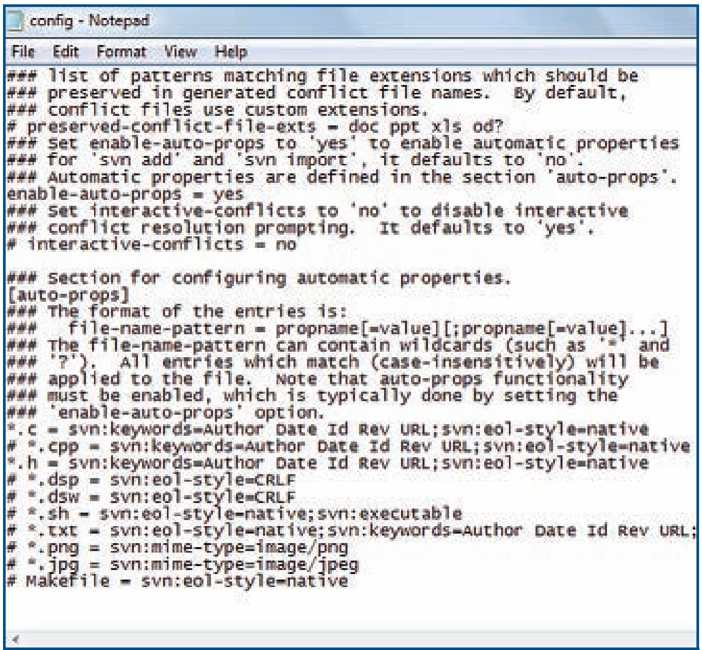
*.c = svn:keywords=Author Date Id Rev URL;svn:eol-style=native
*.h = svn:keywords=Author Date Id Rev URL;svn:eol-style=native

When you commit a *.c file or an *.h file with this block at the beginning of the file, such as:

```
<!--
Repository path:     $HeadURL$
Last committed:      $Revision$
Last changed by:     $Author$
Last changed date:     $Date$
ID:                $Id$
-->
```

TortoiseSVN will substitute something like this:

```
<!--
Repository path:     $HeadURL:
https://avrtoolbox.googlecode.com/svn/trunk/docum
entation/joe.c $
```



```
config - Notepad
File Edit Format View Help
### list of patterns matching file extensions which should be
### preserved in generated conflict file names.  By default,
### conflict files use custom extensions.
# preserved-conflict-file-exts = doc ppt xls od?
### Set enable-auto-props to 'yes' to enable automatic properties
### for 'svn add' and 'svn import', it defaults to 'no'.
### Automatic properties are defined in the section 'auto-props'.
enable-auto-props = yes
### Set interactive-conflicts to 'no' to disable interactive
### conflict resolution prompting.  It defaults to 'yes'.
# interactive-conflicts = no

### Section for configuring automatic properties.
[auto-props]
### The format of the entries is:
###   file-name-pattern = propname[=value][;propname[=value]...]
### The file-name-pattern can contain wildcards (such as '*' and
### '?').  All entries which match (case-insensitively) will be
### applied to the file.  Note that auto-props functionality
### must be enabled, which is typically done by setting the
### 'enable-auto-props' option.
*.c = svn:keywords=Author Date Id Rev URL;svn:eol-style=native
# *.cpp = svn:keywords=Author Date Id Rev URL;svn:eol-style=native
*.h = svn:keywords=Author Date Id Rev URL;svn:eol-style=native
# *.dsp = svn:eol-style=CRLF
# *.dsw = svn:eol-style=CRLF
# *.sh = svn:eol-style=native;svn:executable
# *.txt = svn:eol-style=native;svn:keywords=Author Date Id Rev URL;
# *.png = svn:mime-type=image/png
# *.jpg = svn:mime-type=image/jpeg
# Makefile = svn:eol-style=native
```

■ FIGURE 3. TortoiseSVN Configuration File.

```
Last committed:     $Revision: 54 $
Last changed by:    $Author:
smiley.micros@gmail.com $
Last changed date:    $Date: 2011-01-04 16:41:18
-0500 (Tue, 04 Jan 2011) $
ID:               $Id: joe.c 54 2011-01-04
21:41:18Z smiley.micros@gmail.com $
-->
```

Now when you look at two files with the same name, you can see which is the most recent.

License

This license block will follow each title block:

```
/*
 *  BSD License
 *  -----------
 *
 *  Copyright (c) YEAR, AUTHOR NAME, All rights
 *  reserved.
 *
 *  Redistribution and use in source and binary
 *  forms, with or without modification, are
 *  permitted provided that the following
 *  conditions are met:
 *
 *  - Redistributions of source code must retain
 *  the above copyright notice, this list of
 *  conditions and the following disclaimer.
 *
 *  - Redistributions in binary form must
 *    reproduce the above copyright notice,
 *    this list of conditions and the following
 *    disclaimer in the documentation and/or
 *    other materials provided with the
 *    distribution.
 *
 *  - Neither the name of the AUTHOR NAME nor
 *    the names of its contributors may be used
 *    to endorse or promote products derived
 *    from this software without specific prior
 *    written permission.
 *
 *  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT
 *  HOLDERS AND CONTRIBUTORS "AS IS" AND ANY
 *  EXPRESS OR IMPLIED WARRANTIES, INCLUDING,
 *  BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 *  OF MERCHANTABILITY AND FITNESS FOR A
 *  PARTICULAR PURPOSE ARE DISCLAIMED.  IN NO
 *  EVENT SHALL THE COPYRIGHT OWNER OR
 *  CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
 *  INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
 *  CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 *  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 *  OR SERVICES; LOSS OF USE, DATA, OR PROFITS;
 *  OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 *  ON ANY THEORY OF LIABILITY, WHETHER IN
 *  CONTRACT, STRICT LIABILITY, OR TORT
 *  (INCLUDING NEGLIGENCE OR OTHERWISE)
 *  ARISING IN ANY WAY OUT OF THE USE OF THIS
 *  SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
 *  OF SUCH DAMAGE.
 */
```

where the **YEAR** has the year the code was first released and the **AUTHOR NAME** has the initial author's name in both locations.

In-Function Comments

Comments in the source code are there for the experienced programmer, not the casual user who should stick with the API document. Comments that explain the source code are as idiosyncratic as the individual programmer. Too many comments just wastes time; too few jeopardize future maintenance. We will assume that the reader of a C function is a skilled programmer and just wants to quickly be told about anything that isn't obvious. Novice level comments are to be reserved for the API Doxygen comments in the header file. However, since too few comments are more likely to cause future harm than too many, we recommend that the programmer opt for being verbose. The following function is provided as an example of reasonable in-function comments:

```
void spi_init_master(void)
{
     // Set pullups with output high
     PORTB |= (1<< MISO_HARDWARE_PIN) \
          | (1<< MOSI_HARDWARE_PIN)  \
          | (1<< SCLK_HARDWARE_PIN)  \
          | (1<< SS_HARDWARE_PIN);

     // Set MOSI, SCK AND SS to outputs
     DDRB |= (1<< MOSI_HARDWARE_DDR)  \
          | (1<< SCLK_HARDWARE_DDR)  \
          | (1<< SS_HARDWARE_DDR);

     // Set Miso to input
     DDRB &= ~(1<< MISO_HARDWARE_DDR);

     // Enable SPI, Set as master, set clock to
     // fosc/16
     SPCR = ( 1 << SPE ) | ( 1 << MSTR ) |
     ( 1 << SPR0 );
}
```

Using Doxygen to Generate the API Document

Doxygen comments will only be used in the public header file that will contain complete Application Programmers Interface information for each public function. This is the only document that the library user needs.

Module Title Block

This block contains a general description of the module, along with whatever API style information seems appropriate for ease of use. This block will always include:

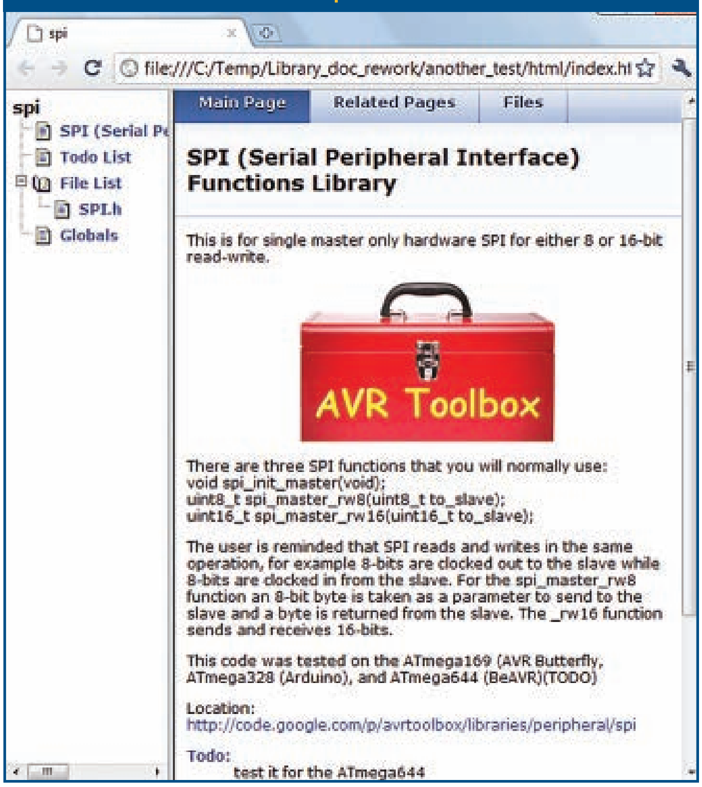
- **\image** wave the flag for our team.
- **\mainpage** with module name.
- **\brief** - that 'brief'ly describes what the module does.
- Location: This is not a Doxygen command but will be included at the end of the \brief section to show the specific project location for this module.
- **\todo** notes on what remains to be done.
- **\author** – tells who wrote this function.
- **\license** New BSD.
- **\date** – when it was first released.
- The /* ***...*** */ dividers to make the file more readable.

The following example block is from SPI.h:

```
/* ******************************************* */
/*!
   \image html avrtoolbox.gif

   \mainpage SPI (Serial Peripheral Interface)
   Functions Library

   \brief This is for single master only
   hardware SPI for either 8 or 16-bit
   read-write.

   There are three SPI functions that you will
   normally use:\n void spi_init_master(void);\n
   uint8_t spi_master_rw8(uint8_t to_slave);\n
   uint16_t spi_master_rw16(uint16_t to_slave);\n

   The user is reminded that SPI reads and
   writes in the same operation, for example
   8-bits are clocked out to the slave while
   8-bits are clocked in from the slave. For the
   spi_master_rw8 function an 8-bit byte is
   taken as a parameter to send to the slave and
   a byte is returned from the slave. The _rw16
   function sends and receives 16-bits.

   This code was tested on the ATmega169 (AVR
   Butterfly, ATmega328 (Arduino), and ATmega644
   (BeAVR)(TODO)

   Location:
   http://code.google.com/p/avrtoolbox/libraries
/peripheral/spi

   \todo test it for the ATmega644

   \author Joe Pardue
   \license New BSD
   \date December 1, 2010
*/
/* ******************************************* */
```

This generates a Doxygen page as shown in **Figure 4**.

■ FIGURE 4. spi API document.



Documenting Each API Function

Each function declaration in the *.h file will have a preceding block of Doxygen comments. This block will always include:

- **\brief —** that 'brief'ly describes what the function does. The final line will contain the avrtoolbox directory location for the module.
- **\param —** name data type and use for each parameter.
- **\return —** name data type and use for return value.
- The /* ***...*** */ dividers to make the file more readable.

This block may include:

- **\note —** provides extra information that might prove useful.
- **\todo —** items that remain to be done.

The following example block is from SPI.h:

```
/* ******************************************* */
/*!
   \brief Writes and reads an 8-bit byte via
   software SPI.

   \return 8-bit byte from the slave device.

   \param to_slave - 8-bit byte to send to the
   slave device.

   \author Joe Pardue
   \date October 29, 2010
   \version 1.00
*/
/* ******************************************* */
uint8_t spi_master_rw8(uint8_t to_slave)
```

This lets Doxygen create the API text shown in **Figure 5.** In the above case, stating the obvious that the \param and \return are eight-bit may be overkill, but it is used to differentiate between this function and the nearly identical 16-bit version of the function. These comments will be used by Doxygen to generate the avrtoolbox user manual. That process is explained elsewhere.

Help!

I intend for the avrtoolbox to grow into a community-based tool that is not dependent on any individual. That is why I keep emphasizing *we*. I am taking the lead at the moment by writing this Workshop series and supplying the base code to get things going. Other folks have done similar projects in the past and then disappeared, leaving some really good tools laying around unsupported and with restrictive licenses. I want this project to be so useful that folks will continue to use it and keep it up-to-date even if I get run over by a truck. This will take a community that sees the value of the project and commits to helping out. If you are interested, please start a thread on **www.avrfreaks.net** and let's start discussing what avrtoolbox can be and what you can contribute. Also, AVRFreaks is the place to go for questions about avrtoolbox and AVRs in general.

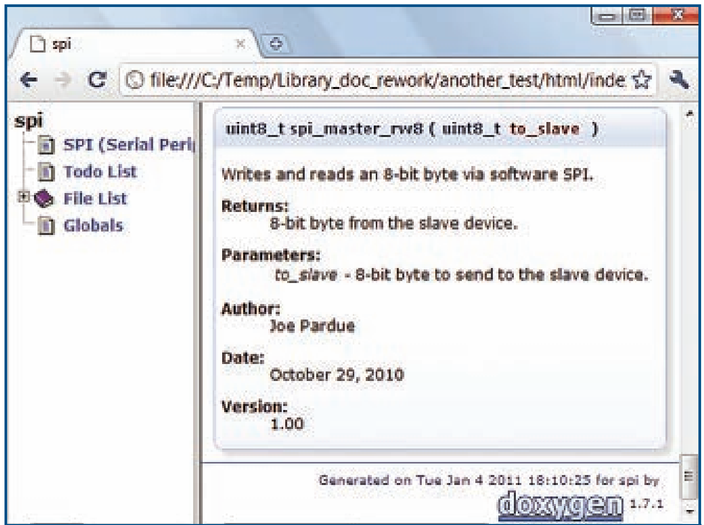
But AVRFreaks.net is a Sewer

Well, you'll get no argument from me on that point, since it is on the Internet that IMHO is also a sewer. Nonetheless, the Internet is the best learning tool ever invented despite the stench. Some guy who I suspect was just looking for trouble recently started a thread on AVRFreaks asking if folks would laugh at him if he posted pictures of his project. I responded:

Let's assume for a moment that you are sincere with your question. My answer is ***OF COURSE PEOPLE WILL LAUGH AT YOU — THIS IS THE INTERNET!*** *To make matters worse, AVRFreaks is a nearly unmoderated forum. We have folks here who will not only laugh at you, if you give them the opportunity they will give you advice that will kill you if you take it.*

The best way to succeed with microcontrollers is to grow a pair, and use the Internet and sites like AVRFreaks using the good and ignoring the bad. At times, it is like panning for gold in a stream of sewage but it is also the best way to get the gold. If you really are afraid of being laughed at on the Internet, you'll just make your own life harder and the [expletive deleted]*s win.*

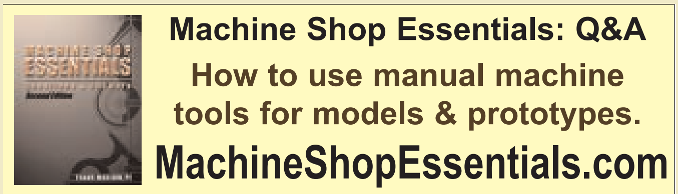
Maybe that is harsh, but the Internet is just too good of a tool to give up over dainty sensitivity. So, I'm hoping some folks will put on their waders and gas masks, and join in the fun. I will generally look at posts on AVRFreaks when the title has avrtoolbox, Butterfly, or Smiley's Workshop. I may not respond since AVRFreaks has members all around the world and someone usually beats me to providing a good answer. Also, don't PM me. I participate in a community and public discussion is available to everyone who might have the same question, so I only need to answer it once.

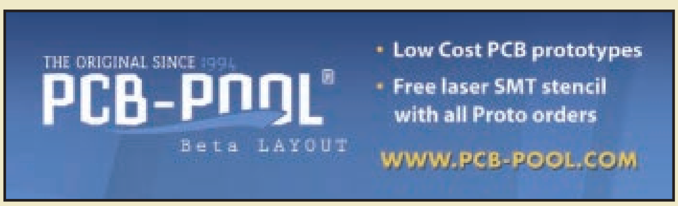


■ FIGURE 5. spi API for a function.

If you just can't wait and want to get a good leg up on C and the AVR (while helping support your favorite magazine and technical writer), then buy my C Programming book and Butterfly projects kit through the *Nuts & Volts* webstore. Next month, if all goes well, we will continue with avrtoolbox by writing an elementary serial function library. **NV**

ELECTRONET





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

BY L. PAUL VERHAGE

CABLING THE NEARSPACE ULTRALIGHT – PART 2

Perhaps you've built an UltraLight PCB and soldered the components. If so, you're ready to make the wiring harness. The wiring harness permits the programming and operation of the UltraLight flight computer. So read on and learn all about the wiring harness, plus the camera cables and control panel.

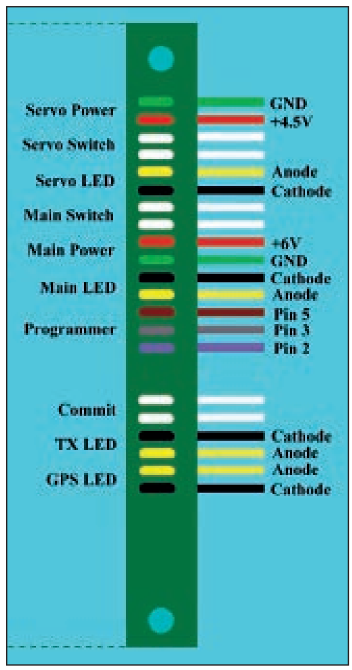
There are 21 wires in the wiring harness. There are wires to battery packs, indicator LEDs, switches, a 1/8" mono receptacle, and a DB-9 connector. A harness one foot long is sufficient for any near spacecraft; therefore, cut 21 feet of wire into one foot lengths. As long as you keep track of each wire as you work with it, you don't need a wiring color scheme. However, you may find using one is helpful if you complete the wiring harness over two or more days.

After cutting the 21 wires, strip 1/4" of insulation from one end of each one. Then, run each wire through its strain relief hole and into its solder pad on the PCB. Solder and trim. In the next step, we'll terminate each wire in the harness according to its function. Use the **figure** provided as a reference as you complete the harness.

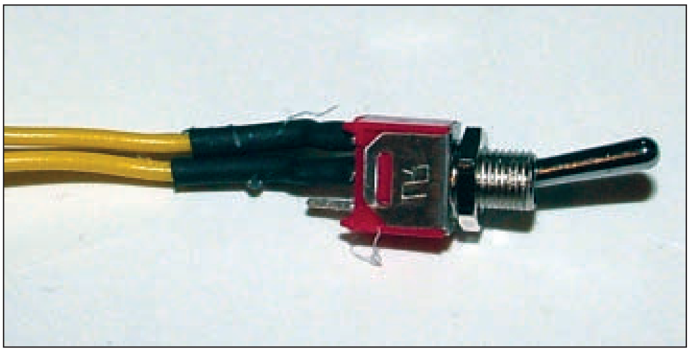
First, connect two power sources to the harness. Since servos operate well with only 4.5 volts, attach a three AAA cell holder to the two servo power wires. Do this by stripping about an inch of wire from the ends of the two servo power wires. The battery holder already has stripped wires, so you don't need to prep those wires any further. Slide thin heat shrink over the servo wires and use a simple splice to connect the battery holder's wires (watch their polarity). Slide the heat shrink over the splice after the solder cools, and shrink.

A four AA cell holder works for main power; however,

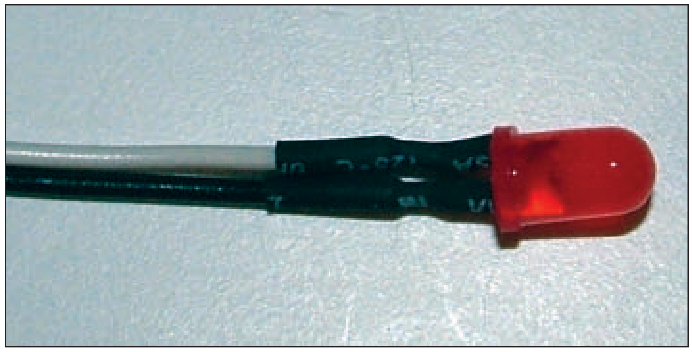
■ "Affirmative Dave, I read you." Unlike HAL 9000 of the spacecraft Discovery, the NearSpace UltraLight is always a friendly flight computer. This is an example of the control panel you'll make for your near spacecraft. It gives access to the PICAXE and indicates the status of the flight computer, without having to open up the near spacecraft.



■ The UltraLight PCB has been turned on its side to place the wiring harness on the right side of it (only the top of the PCB is shown; the rest of the PCB is located to the left). The wires are color coded in this illustration. Red and green wires are for power (main power and servo power), white wires are for the non-polarized switches (including Commit), yellow and black are for LEDs, and the brown, gray, and violet wires are for the DB-9 programming header.



■ I used a DPDT subminiature toggle switch on this flight computer, so ignore the third terminal. If you use a DPDT toggle, then solder a switch wire to the center terminal and either one of the end terminals (don't solder both wires to both outside terminals or else the switch will never work).



■ Don't mix your wires! The LED is polarized and will not function if soldered backwards.

I prefer using a rechargeable lithium-ion battery as the main power source. If you decide to use AA cells, be sure to use the lithium camera cells in the holder (lithium cells stands up to the cold of near space better than alkaline cells). Connect the four AA cell holder to the main power wires using the same technique as for the servo battery pack.

If, on other hand, you decide to use a rechargeable battery instead of a battery holder, then terminate the main power wires with the connectors of your choice. I recommend either an Anderson Power Pole or a Dean's Micro Plug. Both are available at hobby stores that sell R/C race cars.

Next are the switches. The main and servo switch wires connect to terminals on subminiature toggle switches (like Jameco part number 72161). Strip one inch of insulation from the ends of both main switch wires and slide 3/16″ diameter heat shrink over them. The terminals of each subminiature toggle switch have small holes drilled through them and the strands in the wires must be twisted tightly in order to fit through these holes.

After twisting, pass half of the stripped portion of a wire through the hole in a terminal, fold it back, and wrap the excess around the rest of the bare wire. Solder the connection well and slide heat shrink over it. Repeat with the second terminal of the toggle switch. Then, repeat the entire process once again for the servo switch and its wires.

Before connecting the LEDs to the main LED and servo LED wires, cut the LED leads to a length of 1/2″ and tin them. Then strip, twist, and tin the LED wires. After the solder cools, slide 3/32″ diameter heat shrink over each wire. Identify the anode lead of the LED (it's the lead next to the rounded side of the LED). Then, take the main LED anode wire and hold it against the tinned anode LED lead while heating both with a freshly tinned, hot soldering iron. The solder in both the lead and the LED will fuse together. After it cools, slide the heat shrink over the connection and shrink.

Repeat for the cathode wire and the cathode lead of the LED. Then, attach the servo LED wires to the second LED following the same procedure. While this connection is not strong, once inserted into the control panel, the LED leads are protected well enough for years of use.

We're over half way done — only the Commit and Programmer wires remain to be terminated. The Commit consists of two non-polarized wires. The first connects directly to the PICAXE and has a 4.7K resistor on the flight computer PCB, pulling it up to five volts. The second Commit wire connects directly to ground. Both Commit wires terminate with a 1/8″ mono receptacle so that a shorted 1/8″ mono jack can act as the Commit pin.

When the Commit pin is inserted into the mono jack, it shorts the pulled-up Commit wire to ground. The PICAXE is programmed to monitor this pin and while it detects this logic low, it waits within a programming loop. Remove the Commit pin and the pulled-up Commit wire goes high. Upon detection of this transition to high, the PICAXE jumps out of its programming loop to begin recording data. You'll notice the Commit pin prevents the flight computer from recording data until after the GPS has a position lock and the mission is ready to begin. There's no sense in recording a lot of boring data on the ground when it's near space data you're interested in.

Make the Commit pin first, as we'll need it to locate the proper terminals in the mono receptacle. Unscrew the plastic jacket from the back of the mono jack and you'll see there are two terminals. Cut at least a six inch length of wire and strip an inch (or more) of insulation from both ends. Push the ends of the wire through

■ The wire is just about ready to be soldered to the terminals of this 1/8″ mono jack, but first I'll crimp the top terminal.

the holes in the terminals and twist the wire around itself. Notice that there are two flanges on the larger terminal — crimp these around the wire. Now solder.

After crimping and soldering, fold the wire sharply in half so that it fits through the opening in the back of the plastic jacket. The wire will form a loop that protrudes from the opening. Now, remove the jacket and put a thin coating of hot glue around the terminals. However, do not get hot glue on the threads in the mono jack. Quickly — before the glue can cool — screw the plastic jacket back over the jack, letting the wire loop stick out of the opening. Squirt hot glue into the opening to fill the entire interior of the jacket. After the interior of the jacket is filled, the mono jack becomes a solid piece of plastic and metal — very durable. Use the wire loop at the end of the jack to tie a strip of brightly colored cloth or plastic. This flag is a reminder to remove the Commit pin before launch.

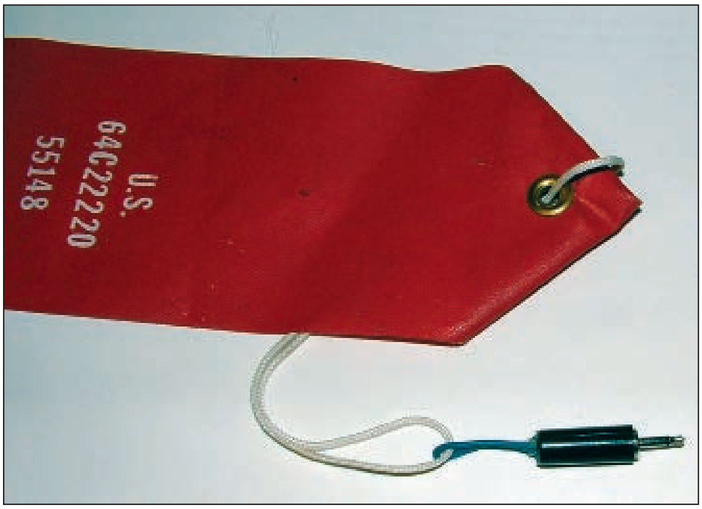
Now it's time to connect the 1/8″ mono receptacle to the flight computer harness (a stereo jack may also work — the following procedure will let you verify this). The 1/8″ mono receptacles I purchase often have a different number and arrangement of terminals. So, the task at hand is to identify the two terminals that the Commit pin shorts together when it's plugged in. When the Commit pin is removed, there must be no connection between these terminals. Identify the proper pins by inspection (not as reliable of a method) or by using a multimeter set to continuity. Test and measure between pairs of terminals, looking for the two terminals that are shorted when the Commit pin is inserted and opened when the Commit pin is removed.

After locating the proper terminals, strip 1/2″ of insulation from the ends of the harness' two Commit wires. Cut two pieces of 1/4″ diameter heat shrink (the terminals are wide) and slide them over the wires. Pass a wire through the hole in each terminal, twist tightly, and solder (recall the wires are not polarized, so it doesn't matter which terminal each wire connects to). After the solder cools, slide the heat shrink over the soldered terminals and shrink.

Now we're at the last item — a female DB-9. Only three wires connect to the back of the DB-9: the ground, data input, and data output. The **wiring harness diagram** shown earlier reveals that DB-9 pins 2, 3, and 5 are soldered to the harness. So, strip 1/4″ of insulation from the end of each wire, twist tightly, and tin. The back of the DB-9 has solder cups, or open cups that the wires can lie within. Tin the insides of solder cups 2, 3, and 5. You can identify each solder cup because each has a small number stamped next to it. If you don't see the number, then check on the front of the DB-9 for a little number stamped next to each opening.

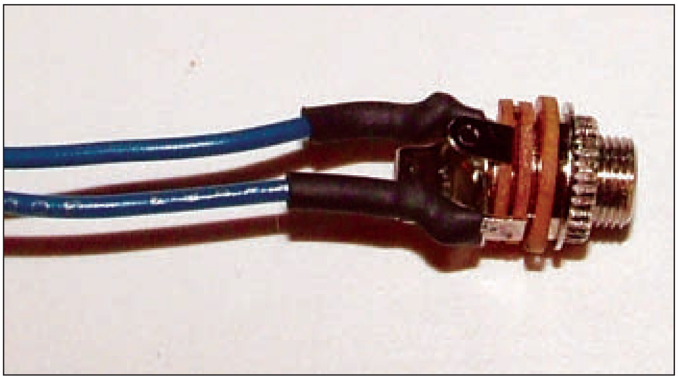
Slide thin heat shrink over each wire and then solder the wire to the DB-9. I find a third hand helps to steady the DB-9 while you press the tinned wire into the tinned solder cup. Heat both the wire and the cup with a soldering iron and they will melt together. After the solder cools, slide the heat shrink over the connection and shrink.

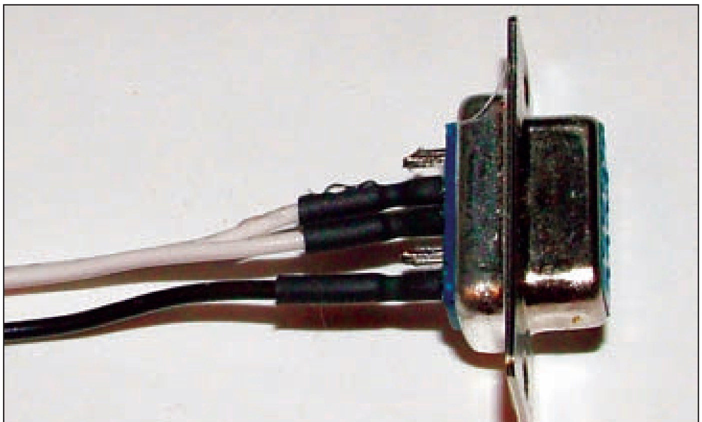
The UltraLight harness now has two battery connectors (these could be battery holders), two toggle



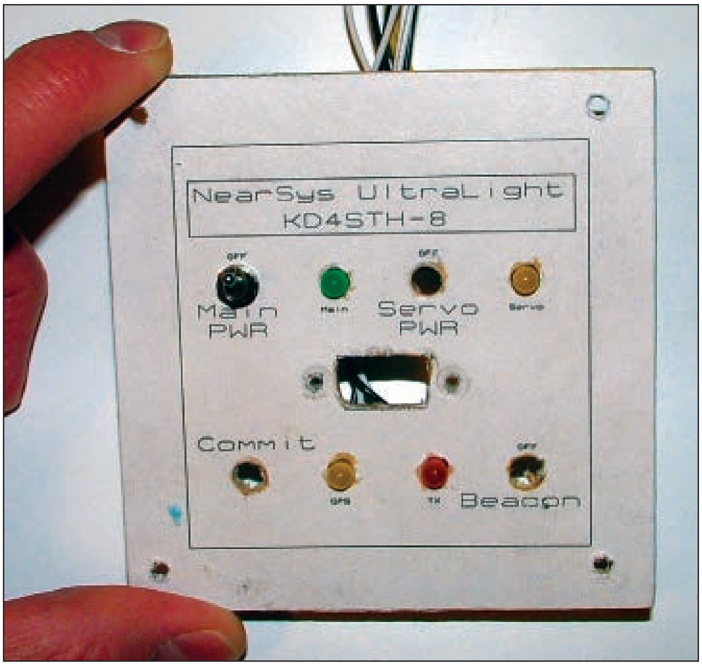
■ I was able to purchase a few "Remove Before Flight" tags from the Boeing surplus store in Kent, WA a few years back. These make great flags to remind me to start the flight computer's data recording before we release the near spacecraft. A flag like this also gives your near spacecraft a more professional look.

■ This mono receptacle is ready to be attached to the control panel. I find that the nut on these receptacles are shipped loose, so tighten it up before it has a chance to drop off and get lost.





■ This DB-9 is finished. Soon it will be bolted to the control panel.



■ I used a simple CAD program to design the face of this control panel. The four holes in the corner are where #6-32 bolts will attach the panel to the near spacecraft airframe. This layout works well because the Commit pin hangs down where it can't cover anything else on the control panel. Also, because each indicator LED is labeled and located next to its toggle switch.

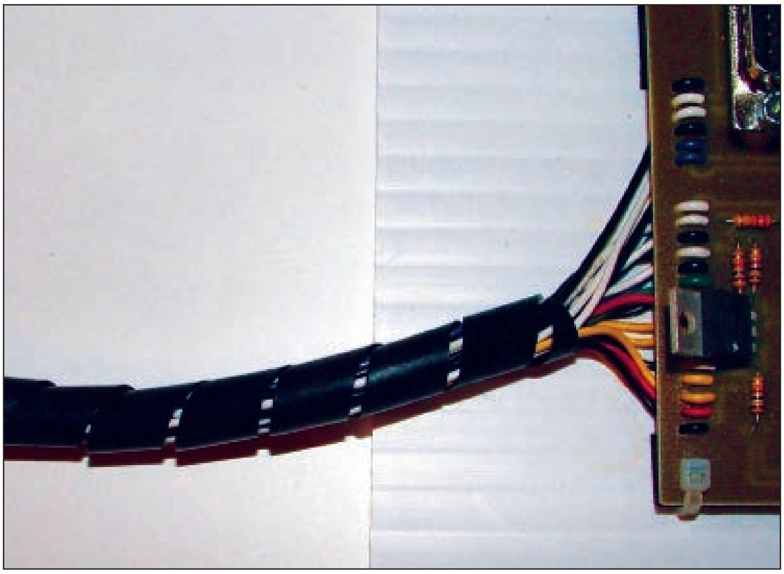
■ I used a sheet of 1/8" thick modeling plywood for this control panel. Notice there is a small amount of hot glue around the back of each LED to ensure they don't slip out.

switches, four LEDs, a 1/8" mono receptacle, and a female DB-9. The battery connectors remain inside the airframe, but the rest of the harness mounts to a control panel. I find a thick sheet of Styrene plastic or 1/8" modeling plywood works well for this control panel.

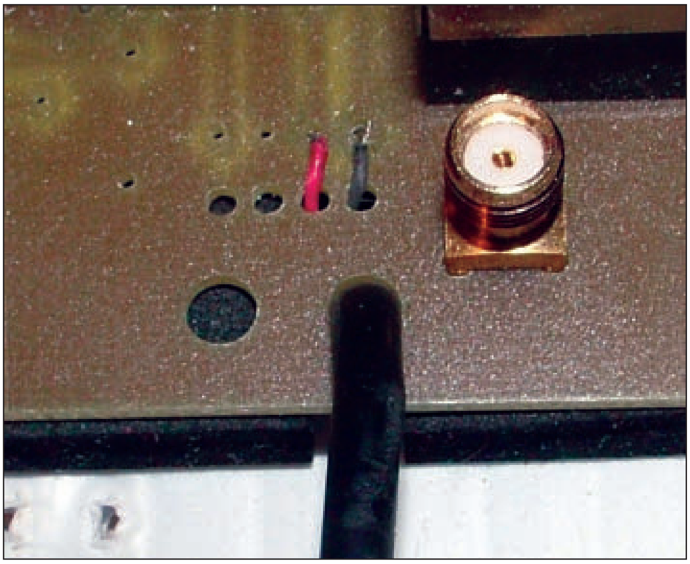
Before drilling holes into the control panel, design a smart layout for all the pieces. The design must provide adequate spacing between switches, LEDS, the mono receptacle, and DB-9. Adequate spacing means fingers must be able to reach individual toggle switches while the programming cable is plugged into the DB-9. A smart design also means the Commit pin shouldn't cover an LED, toggle switch, or the DB-9. I find a four inch square panel with a 1/2" border gives adequate spacing. I like to use plywood for the control panel so I can glue a sheet of paper to the face. On the paper, I mark out the locations for all the LEDS, etc., and their labels. If you decide to use a paper label on the face of the control panel, glue it to the plywood and give it a good coat of clear paint (dull coat) after the glue dries. A clear dull coat protects the ink on the paper from moisture.

Drill 1/4" holes in the plastic or plywood sheet for the toggle switches, LEDS, and mono receptacle (you may need to enlarge some holes slightly). The DB-9 needs a specially shaped cut-out. I use a Dremel to roughly cut out its outline and an Exacto to finish. Don't forget to also drill 1/8" holes for the two bolts that mount the DB-9 to the control panel. Use #2-56 bolts and nylocks to attach the DB-9. Before tightening each toggle switch, rotate them so they flip to the ON position in the same direction. The control panel makes a great deal more sense when both toggle switches operate in the same direction. To make sure the LEDs remain in place, squirt a little hot glue around them (but only on the back of the control panel). This completes the UltraLight control panel.

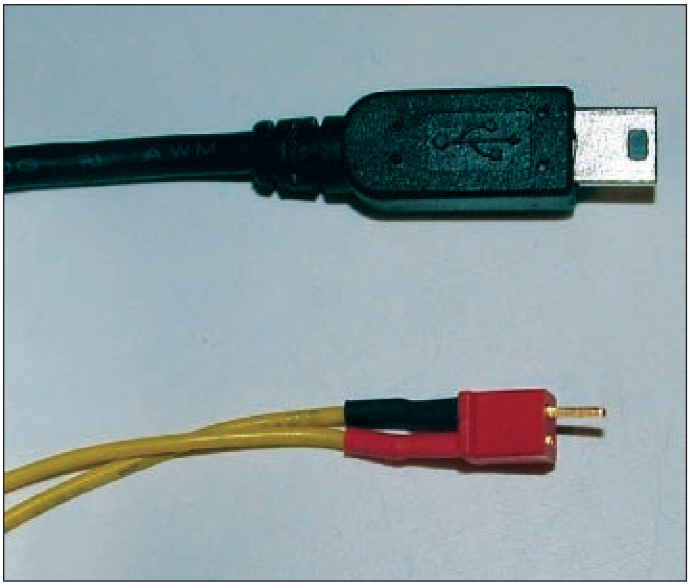
There are four last wires on the UltraLight; those for the two camera cables. If the UltraLight will control Canon cameras running the USB remote program (in CHDK), then the camera cables must be mini USB cables. Only two wires are needed in the USB cable: the +5V wire (red) and the ground wire (black). Use the least expensive USB to mini USB adapter cable you can find for this. Cut off the regular USB connector



■ After attaching the control panel to the flight computer's harness, wrap the wires in a spiral wrap. That gives the wires some protection and makes the harness easier to manage.



■ Because of the thickness of the USB cable compared to the two wires that are needed to operate a Canon camera, there's a large outer strain relief hole. The USB cable is too heavy for just two wires to act as the strain relief.



■ I soldered both kinds of camera cables to my UltraLight. That way, my flight computer can operate either my Canon A550 or my modified Canon Elph.

(not the mini USB) and strip back the outer insulation. Peel back the wire mesh or aluminum foil shield to expose about 3/4" of the inner four wires. Cut off the two unnecessary wires and strip about 1/4" of insulation from the red and black wires. Run the entire cable through the large strain relief hole at the bottom of the PCB (either of the ones marked USB 3 and USB 4 in the previous article), making sure the insulated USB cable passes through this large strain relief hole. Then, pass just the red and black wires through the second, smaller strain relief holes. Finally, solder the red and black wires to the PCB (watch the polarity). A properly strained relieve USB cable is shown in the **photo**.

If the UltraLight is not operating a Canon camera, then the camera must be modified by soldering two wires to its shutter switch. That way, when the two wires are shorted together by the relay on the UltraLight flight computer, the camera takes a picture. New camera designs are always coming out, so there's no sense giving directions in this article about modifying cameras. There are current directions on the Internet, so do a search (try searching under modifying cameras near space or wild life cameras) for directions with specific cameras.

After making the modification, terminate the wires with the connector of your choice. I like using the Dean's Micro Plug as my camera connector. Do not solder wires directly between the camera and the UltraLight PCB as the weight of the camera or flight computer will eventually pull the wires off the camera's shutter. On the UltraLight side, solder two #24 gauge wires, pass them through their strain relief holes, and terminate with the connector matching the one on the camera (use either strain relief holes marked Cam 3 and Cam 4 as in the previous article).

This completes the UltraLight flight computer. The next article will wrap this up by programming the UltraLight for a mission. That includes programming the APRS tracker and the PICAXE-28X1. I'll include examples of flight code that you can use for your missions.

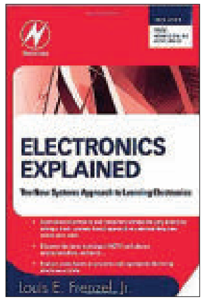
Onwards and Upwards,
Your near space guide **NV**

ELECTRONICS

HTML: A Beginner's Guide
by Wendy Willard
Essential HTML Skills Made Easy!

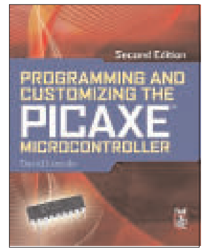
Create highly functional, impressive websites in no time. Fully updated and revised, *HTML: A Beginner's Guide, Fourth Edition* explains how to structure a page, place images, format text, create links, add color, work with multimedia, and use forms. You'll also go beyond the basics and learn how to 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

Electronics Explained
by Louis Frenzel
The New Systems Approach to Learning Electronics



Don't spend time reading about theory, components, and old ham radios - that's history! Industry veteran, Louis Frenzel, gives you the real scoop on electronic product fundamentals as they are today. Rather than tearing electronics apart and looking at every little piece, the author takes a systems-level view. For example, you will not learn how to make a circuit but how a signal flows from one integrated circuit (IC) to the next, and so on to the ultimate goal.
$29.95

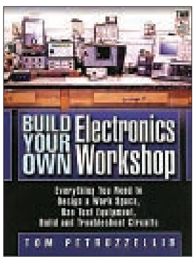
Programming and Customizing the PICAXE Microcontroller 2/E
by David Lincoln
UNLEASH THE POWER OF THE PICAXE!



The PICAXE is a powerful and easy-to-use processor, capable of highly sophisticated projects, without the complexities and high costs of alternative chips. Beginners can produce tangible results within minutes, and experienced users can achieve truly professional results. This Second Edition has been fully updated for the latest hardware and software upgrades, and shows you step-by-step how to take full advantage of all the capabilities of the PICAXE and build your own control projects.
$49.95*

Build Your Own Electronics Workshop
by Thomas Petruzzellis

BUILD YOUR OWN DREAM ELECTRONICS LAB!

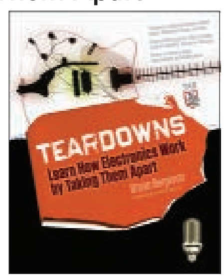


This value-packed resource provides everything needed to put together a fully functioning home electronics workshop! From finding space to stocking it with components to putting the shop into action -- building, testing, and troubleshooting systems. This great book has it all! And the best part is, it shows you how to build many pieces of equipment yourself and save money, big time!
Reg Price $29.95
Sale Price $24.95

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 (CCS) 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

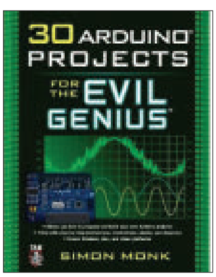
TEARDOWNS
by Bryan Bergeron
Learn How Electronics Work by Taking Them Apart



Amp up your knowledge of electronics by deconstructing common devices and analyzing the revealed components and circuitry. *Teardowns: Learn How Electronics Work by Taking Them Apart* contains 14 projects that expose the inner workings of household appliances, workbench measuring instruments, and musical equipment. Discover how resistors, capacitors, sensors, transducers, and transistors function in real circuitry.
Reg $24.95 **Sale Price $21.95**

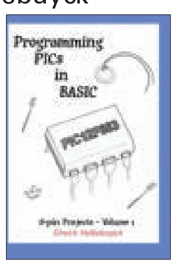
---

30 Arduino Projects for the Evil Genius
by Simon Monk
30 Ways to Have Some Computer-Controlled Evil Fun!



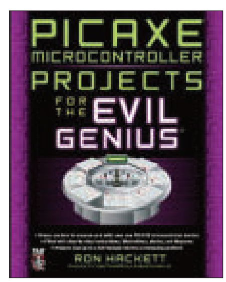
Using easy-to-find components and equipment, this do-it-yourself book explains how to attach an Arduino board to your computer, program it, and connect electronics to it to create fiendishly fun projects. The only limit is your imagination!
$24.95

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**

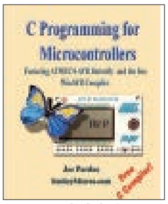
PICAXE Microcontroller Projects for the Evil Genius
by Ron Hackett



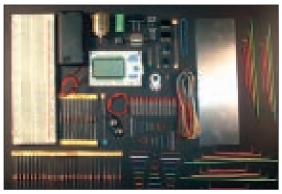
This wickedly inventive guide shows you how to program, build, and debug a variety of PICAXE microcontroller projects. *PICAXE Microcontroller Projects for the Evil Genius* gets you started with programming and I/O interfacing right away, and then shows you how to develop a master processor circuit.
$24.95

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

BOOK & KIT COMBOS



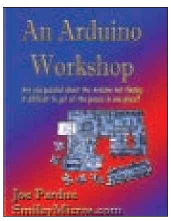
From the
Smiley Workshop
C Programming for
Microcontrollers
by Joe Pardue



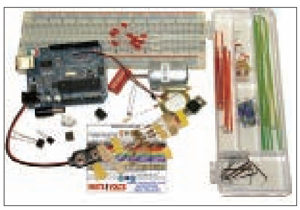
Book **$44.95**

Kit **$66.95**

Do you want a low cost way to learn C programming for microcontrollers? This 300 page book and software CD show you how to use ATMEL's AVR Butterfly board and the FREE WinAVR C compiler to make a very inexpensive system for using C to develop microcontroller projects.
Combo Price $99.95 Plus S/H



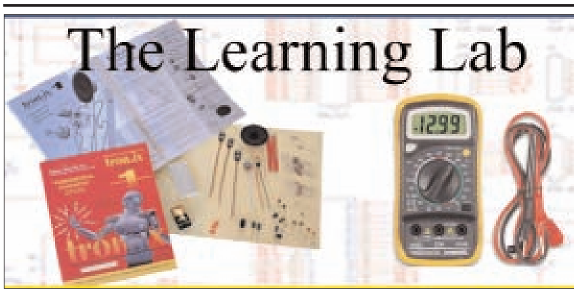
From the
Smiley Workshop
An Arduino
Workshop
by Joe Pardue



The book *An Arduino Workshop* and the associated hardware projects kit bring all the pieces of the puzzle together in one place. With this, you will learn to: blink eight LEDs (Cylon Eyes); read a pushbutton and 8-bit DIP switch; sense voltage, light, and temperature; make music on a piezo element; sense edges and gray levels; optically isolate voltages; fade an LED with PWM; control motor speed; and more!

An Arduino Workshop Combo
Reg Price $ 124.95 **Subscriber's Price $119.95 Plus S/H**

CD ROM SPECIALS



This lab — from the good people at GSS Tech Ed — will show you 40 of the most simple and interesting experiments and lessons you have ever seen on a solderless circuit board. As you do each experiment, you learn how basic components work in a circuit. Along with the purchase of the lab, you will receive a special password to access the fantastic online interactive software to help you fully understand all the electronic principles. For a complete product description and sample software, please visit our webstore.

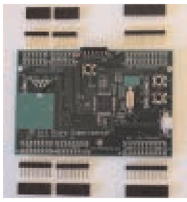
Reg Price $79.95 **Subscriber's Price $75.95**

This great little book is a concise all-purpose reference featuring hundreds of tables, maps, formulas, constants, and conversions. AND it still fits in your shirt pocket!

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

PROJECTS

32-Bit Micro Experimenter Board

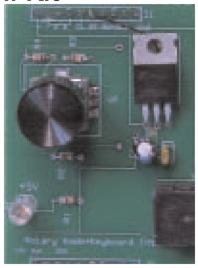


The new 32-Bit Micro Experimenter is the fastest way to learn 32-bit microcontrollers.

The kit includes onboard 46 programmable I/O and USB, free software, carefully documented step-by-step experiments for USB, embedded web server, graphics & 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. Also supports Arduino compatible interface.

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

The Mini Kit

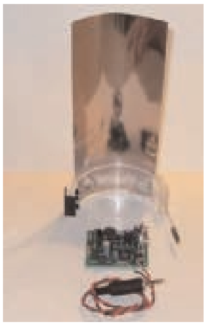


The best experiment yet for the 16-Bit Experimenter Board.

Adding this Mini Kit to your Experimenter Board will enhance the Experimenter. The Mini Kit is a user interface with a rotary encoder using the PIC24F timer peripheral set and its interrupt capability.
For more information, see the December 2010 issue
Assembled units also available.

Subscriber's Price **$36.95**
Non-Subscriber's Price **$39.95**

Piezoelectric Film Speaker Kit



As seen in the November 2010 issue, here is a great project to amaze your friends and to demonstrate a unique way of producing sound. Kit contains one piece of piezoelectric film, speaker film stand, PCB, components, audio input cable, and construction manual. All you'll need to add is a battery and a sound source.
For more info, please visit our website.

Subscriber's Price **$69.95**
Non-Subscriber's Price **$74.95**

Garage Door Alarm PCB & Chips

As seen in the November 2010 issue.
Is Your Garage Door Open?
This project uses the latest in wireless technology, and is a fun and easy project to build. We provide the difficult parts: the transmitter and receiver PCBs with their matching programmed MCUs. The other components can be found at your favorite parts house.
Includes an article reprint.

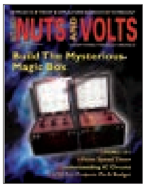
Subscriber's Price **$29.95**
Non-Subscriber's Price **$31.95**

Mini-Bench Supply Complete Kit

A small power supply with +5V, +12V, and -12V outputs is a handy thing to have around when you're breadboarding circuits with both op-amps and digital ICs.
Kit includes: Enclosure box, accessories, DC-to-DC converter kit, switching regulator kit, and article reprint.
For more information, please see the "feature article section" on the of the *Nuts & Volts* website.

Subscriber's Price **$76.95**
Non-Subscriber's Price **$83.95**

Magic Box Kit



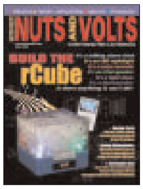
As seen on the April 2007 cover

This unique DIY construction project blends electronics technology with carefully planned handcraftsmanship. This clever trick has the observer remove one of six pawns while you are out of the room and upon re-entering you indicate the missing pawn without ever opening the box.
Includes an article reprint.

Subscriber's Price **$39.95**
Non-Subscriber's Price **$45.95**

rCube Talking Alarm Clock Kit

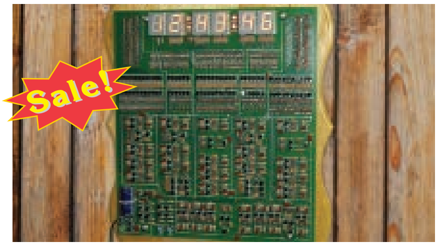
As seen on the May 2009 cover



Available in blue, black, red, and green. All components are pre-cut and pre-bent for easier assembly and the microcontrollers are pre-programmed with the software. Kits also include PCB, AC adapter, and instructions on CD-ROM.

Subscriber's Price **$49.95**
Non-Subscriber's Price **$54.00**

Transistor Clock Kit

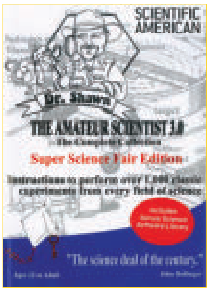


If you like electronic puzzles, then this kit is for you! There are no integrated circuits; all functionality is achieved using discrete transistor-diode logic. The PCB is 10"x11" and harbors more than 1,250 components! For more info, see the November 2009 issue.

Reg $225.95
Sale Price $199.95
PCBs can be bought separately.

The Amateur Scientist 3.0
The Complete Collection

by Bright Science, LLC



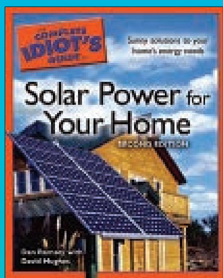
There are 1,000 projects on this **CD**, not to mention the additional technical info and bonus features. It doesn't matter if you're a complete novice looking to do their first science fair project or a super tech-head gadget freak; there are enough projects on the single CD-ROM to keep you and 50 of your friends busy for a lifetime!

Reg Price $26.95
Sale Price $23.95

WE ACCEPT VISA, MC, AMEX, and DISCOVER. Prices do not include shipping and may be subject to change.

ALTERNATIVE ENERGY SECTION

The Complete Idiot's Guide to Solar Power for Your Home
by Dan Ramsey / David Hughes

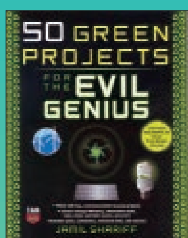


The perfect source for solar power — fully illustrated. This book helps readers understand the basics of solar power and other renewable energy sources, explore whether solar power makes sense for them, what their options are, and what's involved with installing various on- and off-grid systems.
$19.95

NEW! Green Lighting
by Brian Clark Howard, Seth Leitman, William Brinsky
Flip the switch to energy-efficient lighting!

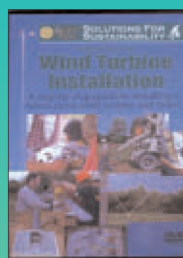
This do-it-yourself guide makes it easy to upgrade residential and commercial lighting to reduce costs and environmental impact while maintaining or even improving the quality of the lighting. Filled with step-by-step instructions and methods for calculating return on investment, plus recommended sources for energy-efficient products.
$24.95

50 Green Projects for the Evil Genius
by Jamil Shariff



Using easy-to-find parts and tools, this do-it-yourself guide offers a wide variety of environmentally focused projects you can accomplish on your own. Topics covered include transportation, alternative fuels, solar, wind, and hydro power, home insulation, construction, and more. The projects in this unique guide range from easy to more complex and are designed to optimize your time and simplify your life!
$24.95

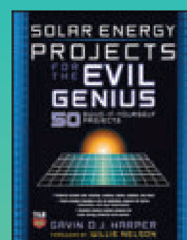
NEW! Wind Turbine Installation DVD



Manuals can sometimes be confusing, especially for people who learn by seeing how things work. So, this DVD takes the viewer step-by-step through the entire installation process, from choosing a site, to running wire, assembling the tower, and finally using a winch for the final lift. This is a must-watch for anyone planning on installing a wind turbine who wants to learn the process and the proper techniques for a safe and successful installation.

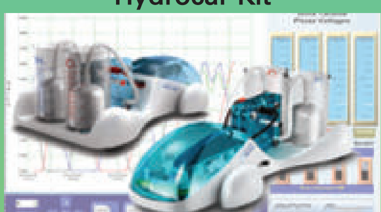
$19.95*

Solar Energy Projects for the Evil Genius
by Gavin D J Harper



Let the sun shine on your evil side — and have a wicked amount of fun on your way to becoming a solar energy master! In *Solar Energy Projects for the Evil Genius*, high-tech guru Gavin Harper gives you everything you need to build more than 30 thrilling solar energy projects. You'll find complete, easy-to-follow plans, with clear diagrams and schematics, so you know exactly what's involved before you begin. **$24.95**

Hydrocar Kit



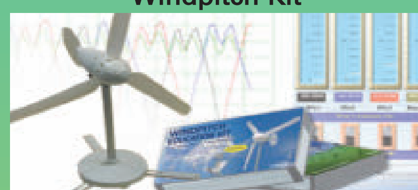
The Hydrocar is used in a couple of great projects from the series of articles by John Gavlik, "Experimenting with Alternative Energy." In Parts 10 and 11, he teaches you the operation of the Polymer Electrolyte Membrane "reversible" fuel cell.
For kit details and a demo video, please visit our webstore.

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

Solar Hydrogen Education Kit

The Solar Hydrogen Education Kit includes a solar cell, a PEM reversible fuel cell, oxygen and hydrogen gas containers, and more! The set only needs pure water to create hydrogen and produce electricity. Perfect for science labs, classroom use, or demonstration purposes.
Subscriber's Price **$56.95**
Non-Subscriber's Price **$59.95**

Windpitch Kit



The WindPitch Wind Turbine Kit is a miniature real-working wind turbine and is one of the great projects from the series of articles by John Gavlik, "Experimenting with Alternative Energy." In Parts 8 and 9, he teaches you how to produce the most power by evaluating the pitch (setting angle) of the profiled blades. For kit details, please visit our webstore.
Subscriber's Price **$94.95**
Non-Subscriber's Price **$99.95**

FUNDAMENTALS FOR THE BEGINNER

BUILD A TWO-TRANSISTOR OSCILLATOR

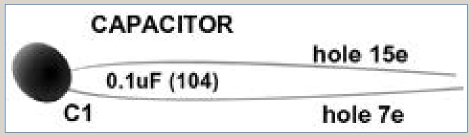
In this experiment, we will build a simple two-transistor oscillator that will generate an audio tone in a speaker.

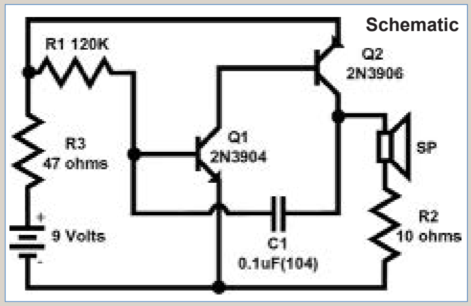
1. Build the Circuit.

Using the schematic along with the pictorial diagram, place the components on a solderless breadboard as shown. Verify that your wiring is correct.

2. Do the Experiment.

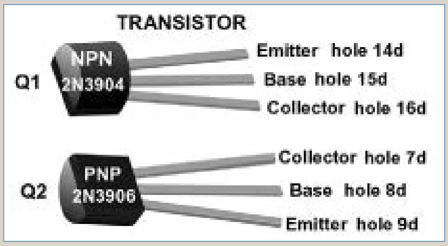
__Theory:__ A transistor oscillator is a circuit that generates a constantly changing current all by itself. The frequency of this varying current tells you how many times per second a complete cycle of change occurs. This oscillator generates a signal of a few volts (about three volts) and approx. 500 cycles per second. This two-transistor direct coupled oscillator uses one NPN and one PNP type transistor. The oscillation is maintained by the feedback from the output (speaker) to the input (base of Q1) through capacitor C1. The frequency of the oscillation is determined by the values of C1 and resistor R1. The larger the values of R1 and C1, the lower the frequency of oscillation. As the battery is connected, electrons begin to flow from the negative terminal of the battery through R2 and the speaker to charge C1. This small current flowing through the speaker while C1 is charging causes the speaker cone to move slightly. As C1 is charging, Q1 begins to conduct current which allows Q2 to begin to conduct current. When Q2 conducts, the electrons travel from the negative of the battery through R2, the speaker, and Q2, back to the positive of the battery. This current is larger and causes the speaker cone to move even more in the same direction. When Q2 conducts, it will discharge C1 which, in turn, causes Q1 to stop conducting, which causes Q2 to stop conducting. The speaker returns to its rest position.

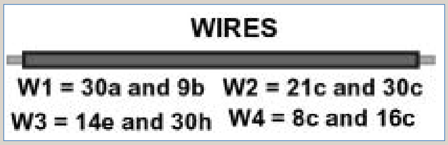


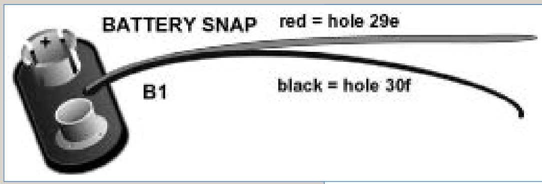


__Procedure:__
Connect a nine-volt battery to the battery snap and you will hear an approx. 500 Hz (500 cycles per second) tone from the speaker.

This tone can be used as an alarm sound to let you know someone has opened a door, for example.



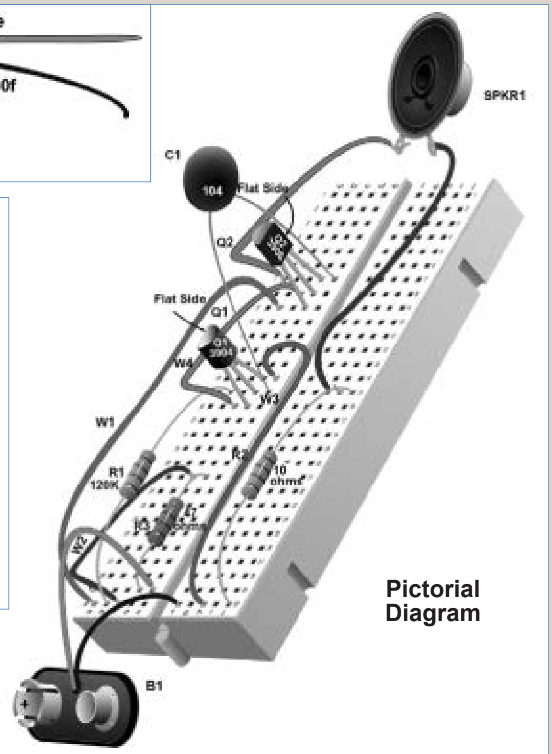




These experiments are provided by GSSTechEd at **www.gssteched.com**
You can order parts for this experiment from their website as follows:

GK01017	10 ohm resistor
GK01033	47 ohm resistor
GK01115	120K ohm resistor
GK02016	0.1 µF capacitor
GK18001	NPN Transistor
GK18002	PNP Transistor
GK35001	Nine-Volt Battery Snap
GK27001	Speaker
GK45011	Four Hookup Wires

A kit for these experiments can be purchased from the *Nuts & Volts* Webstore at **www.nutsvolts.com** or call us at **800-783-4624**.



Pictorial Diagram

>>> QUESTIONS

Phone Fails To Answer Via VOIP

Our phone service is by a cordless phone connected to a VOIP box, to a router, to a cable modem. Much of the time the phone works well, but sometimes when the phone rings, it is dead, no voice, no dial tone. The calling party continues to hear it ringing. I have experienced this intermittent problem from three VOIP companies and using different interface boxes. Customer support said this was a known problem with Uniden brand cordless phones. I just bought newer technology Panasonic DECT-6 phones and the problem persists! I suspect that answering a call may not always register at the VOIP-to-phone converter box as having lifted the handset. I am looking for a circuit to increase the load or settings that the VOIP company can make to their system or the interface box, or at least an explanation of what is going on.

#3111 **Barry Cole**
Camas, WA

Electronic Game Design

We have an idea to make a sequential logic circuit game where there are four defined sequential circuits which will appear at random, preferably on our eight LEDs. Now the problem is how to design our registers that would trap the sequence of the lights each turn, that would be compared with the eight buttons. Ideas?

#3112 **Niel Caday**
Manila, Philippines

AC to DC

Why do most laptops, cell phones, etc., have relatively large transformers to convert AC power to the proper DC voltage, but the Amazon Kindle only has a small adapter that seems to have no space for the traditional transformer?

#3113 **Peter**
Boston, MA

>>> ANSWERS

[#1111 - January 2011]
75 ohm Coax Connections

I have reliability issues with 75 ohm coax connections. I'm certain the problem is the center conductor.

I'm thinking about applying solder paste to the end, to maybe seal out oxygen.

#1 Instead of trying to use solder paste which will probably gum up the center conductor as it will attract dirt and dust, try applying some DeoxIT to the conductor before you make the connection. The DeoxIT will chemically improve the connections without causing any damage to the wire. If you go to **http://store.caig.com/s.nl/sc.2/category.188/.f** (their main homepage is **www.deoxit.com**), you can see what is available and order it directly through Caig. RadioShack sells DeoxIT as well, along with their ProGold product in a combination package. I have used both and can say that where connectivity is an issue, DeoxIT will improve matters and not change any electrical characteristics.

Ralph J. Kurtz
Old Forge, PA

#2 If you mean F-type connectors, then there definitely can be reliability issues with them. The female connector contacts the center conductor of the cable by the edges of two opposing metal strips. If the connector is used repeatedly, the metal strips can lose their contact tension and intermittent contact will occur. If large center conductor wire such as RG-11 cable is plugged in, the contacts will be bent back far enough that they may not recover to contact a smaller center conductor wire next time. Because the center conductor may be a softer metal, it can develop nicks if used repeatedly and contact may become intermittent. In outdoor use, the F-type connectors will frequently oxidize which will also cause intermittent contact. The F-type connector is usually very cheap, and made for convenience rather than long life and reliability. I have found better quality F-type connectors available at a supplier such as Milestek in Texas.

Bill Seabrook
Dunkirk, MD

[#1112 - January 2011]
Electric Fence

I need to show when an electric fence (6 KVa to 16 KVa) is active by flashing a simple LED with every pulse. Ideally, the circuit would simply attach between high V and ground strands of the fence. A perfect solution would be able to cope with an inadvertent reverse connection.

I would use an NE-2 or similar neon bulb. The advantages of this are numerous:

1. Neon bulbs don't care about polarity.
2. Neon bulbs draw very little current.
3. Neon bulbs will last for years.
4. Neon bulbs are much more resistant to transients and induced voltages that are common on electric fences.

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.*

From my General Electric glow lamp manual, a NE-2 neon bulb — which is a common glass bulb with two lead wires extending out the bottom — has a nominal current of 0.5 mA to operate. If the fence provides 6 to 16 kV, I'd use a 7.5 megohm 1/2 watt resistor in series with one of the leads. Enclose the lamp and resistor in a small plastic pill bottle or something similar, then run the leads out to each wire of the fence.

Assuming a 7.5 meg resistor, at 16 kV the current will be 2.1 mA. At 6 kV, the current will be 0.8 mA. You can experiment with the resistor value if needed to get a flash and the brightness you need, but I would not go below about two megohms.

Kirk Ellis, KI4RK
Pikeville, NC

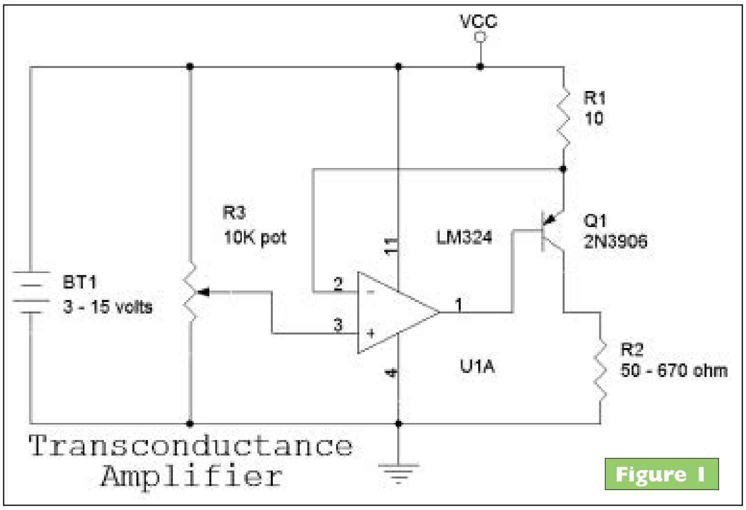


Figure 1

[#12101 - December 2010]
Circuit Design Assistance

I am trying to design a voltage monitoring circuit that will monitor a 12 VDC solar battery. The input voltage signal range needs to be linear, starting no lower than 3 VDC and up to 15 VDC. The output signal needs to be 4 to 20 mA ... 4 mA is 3 VDC and 20 mA is 15 VDC. On my schematic (see Feb '11 issue), I would like to vary the voltage input at V1 and have the output at R2, or something like that. I was not able to get the circuit to produce my desired output results without the use of a second voltage source set to 4 VDC. If at all possible, I prefer to have no power sources other than the primary source being monitored.

#1 The problem stated presents a few issues leaving me to assume several bits of information. It is stated that the output should be a linear current based on the input voltage (transconductance amplifier). However, the circuit presented shows only a load of 670 ohms and the circuit is not capable of being a transconductance amplifier; therefore, I will assume that providing the output into the 670 ohm load is sufficient. It seems, in this case, the forest has gotten in the way of the tree and what is needed is a simple solution.

The simple solution is just two resistors in a voltage divider setup. If we keep R2 as the 670 ohm load, then let's get rid of the rest of the circuit displayed and only add R1 as an 80 ohm resistor between the input voltage and R2. With this setup, when Vin is three volts and we wish 4 mA to pass through R2, we see that R2's voltage is 2.68 volts (670 * 0.004) and we have 0.32 volts across R1 (80 * 0.004) that works. For our 15 volts in, we again have the 20 mA passing through R2, so 13.4 volts (670 * 0.02), and our remainder 1.6 volts across R1 (80 * 0.02), so that works. Our circuit to drive a 670 load is just a voltage divider.

However, from an engineering perspective, if we really want a transconductance amp, then a little more effort and a few more parts are required. First, the circuit provides a slope of amplification of one (output of op-amp just follows the input). Our targeted slope is not one because we want our output to provide 4 mA @ three volts in; our intercept is 0 which makes things much easier than they could be. A true current amplifier would need to provide the 4 mA @ three volts to a wide range of load resistance (I will ignore the possibility of impedance with reactive possibilities to keep it a little simpler). Also, as the circuit is presented as attaching to a solar supply, I will assume that the bandwidth is very low. **Figure 1** shows in conjunction with R1 as a current source following the voltage applied at the base. The pot is adjusted to provide the correct current at the load, so if the load is 670, a voltage of 13.4 should be seen across the resistor. Once set, it should be fairly accurate for the range of load listed (50-670 ohms). The current supplied to R2 is approximately (Vin - Vbase - 0.65)/R1. The 0.65 is the base emitter diode behavior and the current is a little less due to the beta (hfe) of the transistor; a little less than 1% with a beta 100+ (the 2n3906 spec is 150-250 as typical). Because the output is specified as 20 mA @ 15 volts in, our R2 range is limited with 670 ohms being probably near the upper end of the scale, and 50 ohms being near the lower. The 670 is @ three volts is only 0.32 volts below the rail, not leaving much room to put the transistor in bias correctly, while the 50 ohms @ 15 volts necessitate Q1 dissipate near 250 mW (well within the listed range, but I would not want to push it much beyond this point).

Mitchell Hilger
Oklahoma City, OK

#2 The original circuit in the December issue has some serious flaws. The current through R2 will not track V1 in any predictable and

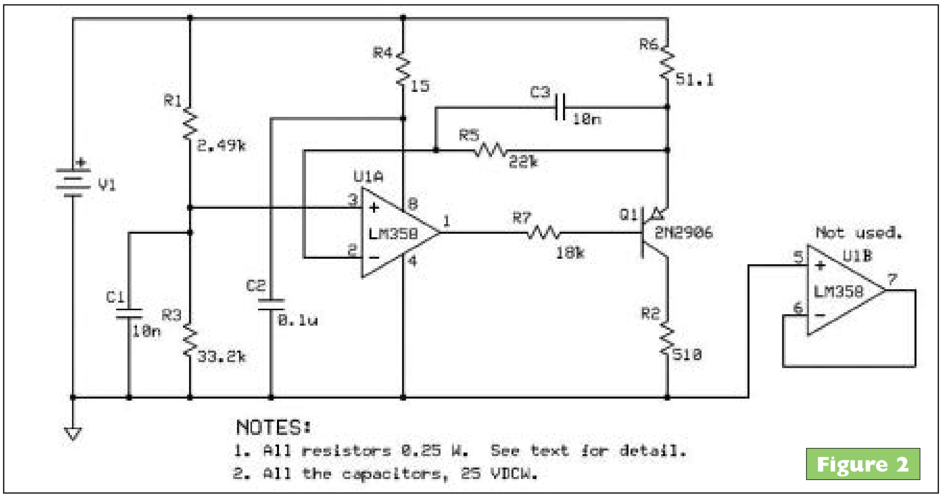


Figure 2

accurate manner. There is no feedback to correlate R2 current as a function of V1 in any manner. Paralleling transistors Q1 and Q2 is not necessary for the maximum current of 20 mA. Furthermore, the paralleling was wrong. When paralleling transistors, it is necessary to configure the circuit to share the current. The suggested schematic in **Figure 2** is based on the following:

1. V1 varies from three to 15 VDC (3 < V1< 15 VDC).

2. No external power supply.

3. Load resistor current is 4-20 mA, corresponding to V1 3.0-15 VDC.

Based on the question, when V1 = 15V the required current through R2 (670 ohms) = 20 mA and the voltage across the resistor will be 13.4V. This leaves only 1.6V head room for the control circuit. It states R2 is "something like that," which may suggest that the value can be changed from the 670 ohms. Therefore, in order to maintain reliable performance, the circuit in **Figure 2**, R2 is only 510 ohms which allows larger current sense resistor R6. This arrangement provides for a higher feedback signal at 4 mA and more head room at 20 mA. See **Figure 2** for the schematic details. Simulation shows that if R2 is 670 ohms, the circuit may still perform, but will not be reliable. **Figure 3** shows actual simulation results. Resistors R1, R3, and R6 are the critical components to maintain reasonable accuracy and are 1%. The other resistors are 5%. All the resistors are 0.25W. If necessary, the user can use the nearest 5% value tolerance

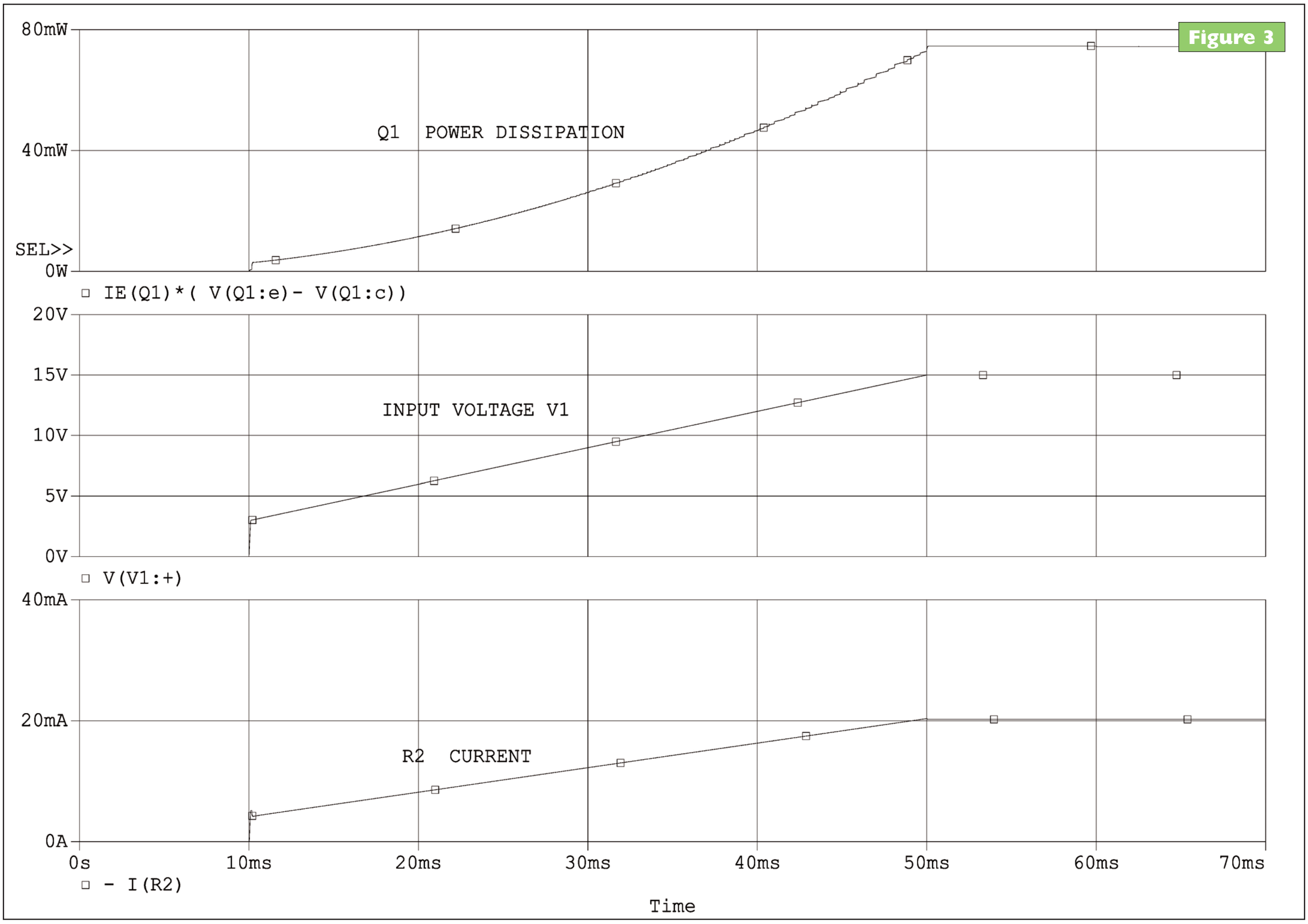


Figure 3

resistors instead of 1%. However, current accuracy as a function of input voltage will be degraded. C1 and C3 limit the circuit bandwidth, while R4 and C2 are optional, but recommended to insure 'clean' voltage for U1.

With values as shown, the typical current through R2 will be 4.05 to 20.33 mA for input voltage of three to 15 VDC. In the actual built hardware, the current will differ slightly due to U1A input characteristic (offset voltage) variations.

The basic simulation results in **Figure 3** show R2 current (lower trace) as a function of the input voltage (center trace) and Q1 power dissipation (upper trace). The simulation time scale was chosen arbitrarily. This is probably much faster that any changes from a solar panel. The steady state Q1 power dissipation at 15V and 20 mA will be approximately 80 mW. Not knowing the application temperature environment and being on the safe side, use 2N2906 for Q1; a 2N3906 transistor will do as well, if the application environment is not hot.

Good construction techniques are essential for good performance. Use short leads, point to point wiring, and attempt to place the components in a similar flow pattern as in the schematic. Note that U1 is a dual device. Only one section (U1A) is used. The second section (U1B) is not used here, but should be wired as shown.

Caution: *R2 should not be smaller than 250 ohms. With a smaller than 250 ohms resistor, the Q1 power dissipation will exceed the safe limits and it may be damaged. V1 can be increased safely to nearly 20 VDC without any problem. The current through R2 will increase linearly to approximately 26 mA.*

Note: U1 was selected for its suitable electrical characteristics; it is readily available at low cost and in a DIP 8 package which is simpler to construct using point-to-point wiring without the benefit of a printed circuit board. If the user wishes to create a printed circuit board and use a single operation amplifier, look at the LM7322 (National Semiconductor) which is an electrically suitable device but surface-mount. In this case, ignore U1B.

Mort Arditti, NA6MA
Los Angeles, CA

Triple Output Bench Power Supplies

The new CSI3303S & CSI5505S regulated DC power supplies are high reliability, variable DC Power Supplies with built in short circuit and thermal protection. These power supplies are suitable for the laboratory, electronics, communications equipment maintenance, production line, scientific research and educational institutions. Both units are equipped with protection circuits that protect the units from short circuits and over temperature by shutting the unit down for safety. Both units allow independent, serial and parallel mode operation.

Technical Specifications:

Independent mode: 2 independent 0-30V outputs

Series mode: CSI3303S Output from 0-60V & 0-3A
CSI5505S Output from 0-60V & 0-5A
Parallel mode: CSI3303S Output from 0-6A & 0-30V
CSI5505S Output from 0-10A & 0-30V

Both units also provide a 5V fixed output @ 3A

Load regulation: <0.1%+3mV (rating current <3A)
<0.2% +3mA
Ripple and noise: <1mVrms 5Hz-1MHz
<3mArms

Voltage accuracy: +/-0.5%rdg+2byte
Current accuracy: +/-.5%rdg+2byte

Display resolution: +/-0.5%rdg+2byte
Rated output: 5.0V +/-0.1V 3A

Tracking characteristics
Series specifications:
Load regulation: less than 50mV
Ripple and noise: (5Hz~1MHz) <=3mVRMS

Parallel characteristics:
Load regulation: less than 50mV
Ripple and noise: (5Hz-1MHz) CV less than 1mV<=6A), CV less than 1.5mV (I>6A)

Once again Circuit Specialists brings you a quality product at a great price!

Item#
0 to 3A **CSI3303S** **$169.00**
www.circuitspecialists.com/CSI3303S
4+ $155.00 ea.

Item#
0 to 5A **CSI5505S** **$194.00**
www.circuitspecialists.com/CSI5505S
4+ $179.00 ea.

3 Channel Programmable Regulated DC Power

Check out our new programmable hi preformance 3 channel power supplies. Featuring both USB & RS232 interfaces, Overload Protection, Auto Fan Control, and Series or Parallel Operation. Both units feature a Large LCD display panel with simultaneous output and parameter view and a keypad for control. They are ideal for applications requiring high resolution, multiple output, and automated operation such as in production testing. There are both fine and coarse controls via the shuttle knob and 90 memory settings. Software included.

Reliable, Highly Stable, Fan Cooled.

FEATURES

* Full digital programmable interface
* High resolution 10mV , 1mA
* Compact and lightweight package
* Alterable display mode
* Intelligent window interface Facilitate user operation
* Supports multiple immediate buttons
* Automatic parallel function
* High stability, low drift
* Over-voltage, over-current, over temperature protection
* Intelligent fan control (Changes with output power)
* Warning Alert Buzzer
* 100 memory locations
* RS-232 & USB interfaces standard

Model	CSIPPS33T	CSIPPS55ST
DC Voltage	0-32V x2 0-6V x1	0-32V x2 0-6V x1
DC Current	0-3A x3	0-5A x2 0-3A x1

Item # **CSIPPS33T** **$479.00**
www.circuitspecialists.com/CSIPPS33T

Item # **CSIPPS55T** **$619.00**
www.circuitspecialists.com/CSIPPS55T

USB Digital Storage Oscilloscopes

* High performance:
* USB connected: Uses USB and supports plug'n play, with 12Mbp communication speed.
* Best performance for your dollar: Thease units have many features that are comparable to the high speedah stand-alone DSOs. But costs a fraction of the price.
* No external power required: Bus-powered from the host computers USB port.
* Probes & USB cable included.
* Easy to use: Intuitive and easy to understand.
* Various data formats: Can save wavrfrom in the following formats: .txt .jpg .bmp & MS excel/word

40MHz **DSO-2090** **$169.00**
www.circuitspecialists.com/DSO-2090

60MHz **DSO-2150** **$209.00**
www.circuitspecialists.com/DSO-2150

100HMz **DSO-2250** **$259.00**
www.circuitspecialists.com/DSO-2250

200MHz **DSO-5200** **$299.00**
www.circuitspecialists.com/DSO-5200

200MHz **DSO-5200A** **$355.00**
www.circuitspecialists.com/DSO-5200A

Specifications	DSO-2090	DSO-2150	DSO-2250	DSO-5200 /5200A
Channels	2 Channels			
Impedence	1M 25pF			
Coupling	AC/DC/GND			
Vertical resolution	8 Bit			9 Bit
Gain Range	10mV-5V, 9 Steps			10mV-10V, 10Steps
DC Accuracy	+/- 3%			
Timebase Range	4ns - 1h 38 Steps			2ns-1h, 39 Steps
Vertical adjustable	Yes			
Input protection	Diode clamping			
X-Y	Yes			
Autoset	30Hz~40MHz	30Hz~60MHz	30Hz~100MHz	30Hz~200MHz
EXT. input	Yes			
Trigger Mode	Auto / Normal / Single			
Trigger Slope	+/-			
Trigger Level Adj.	Yes			
Trigger Type	Rising edge / Falling Edge			
Trigger Source	Ch1 / Ch2 / EXT			
Pre/Post trigger	0-100%			
Buffer size	10K-32K per ch			10K-512KB per ch
Shot Bandwidth	DC to 40MHz	DC to 60MHz	DC to 100MHz	100MHz
Max Sanple Rate	100MS/s	150MS/s	250MS/s	200MS/s / 250MS/s
Sampling Selection	Yes			
Waveform Display	port/line, waveform average, persistence, intensity			
Network	open / close			
Vertical Mode	Ch1, Ch2, Dual, Add			
CursorMeasurement	Yes			
Spectrum Analyzer				
Channels	2 Channels			
Math	FFT, addition, subtraction, multiplication, division.			
Bandwidth	40 MHz	60 MHz	100MHz	200 MHz
Cursor	Frequency, Voltage			
Data Samples	10K-32K/Ch			10K-1M/Ch

We carry a LARGE selection of Power Supplies, Soldering Equipment, Test Equipment, Oscilloscopes, Digital Multimeters, Electronic Components, Metal and Plastic Project Boxes, Electronic Chemicals, PC Based Digital I/O Cards, Panel Meters, Breadboards, Device Programmers, and many other interesting items. Check out our website at:
www.CircuitSpecialists.com

New Year, New Price
CSI-Station1A

Easily our best value in our selection of soldering stations. O.E.M. manufactured just for Circuit Specialists Inc., so we can offer the best price possible! The CSI-Station1A features a grounded tip & barrel for soldering static-sensitive devices and uses a ceramic heating element for fast heat up & stable temperature control.

The control knob is calibrated in Fahrenheit & Celsius (392° to 896°F and 200° to 480°C). One of the nicest features is the high quality comfort grip soldering iron. The iron connects to the station via an easy screw-on connector making iron replacement a snap. The 1 meter length iron cord provides plenty of length for users to set up the station in a convenient location. Another nice feature is the soldering iron holder. Made of rugged aluminum,it is a seperate piece from the main station & allows the user maximum convenience.....you don't have to reach all the way back to the station to store the iron. Yet another feature is the stackable design of the CSI-Station1. The main station is designed for an additional unit to be placed on top of it allowing for space saving placement of the CSI-Station1A. Also included at no additional charge is one user replaceable ceramic heating element so that you will be prepared! Large selection of soldering tips available too.

Item#
CSI-STATION1A **$29.95**

www.circuitspecialists.com/csi-station1a

Programmable DC Electronic Loads

Thease devices can be used with supplies up to 360VDC and 30A. It features a rotary selection switch and a numeric keypad used to input the maximum voltage, current and power settings. These electronic DC loads are perfect for use in laboratory environments and schools, or for testing DC power supplies or high-capacity batteries. It also features memory, and can also be connected to a PC, to implement remote control and supervision.

360V/150W (CSI3710A) $349.00
www.circuitspecialists.com/csi3710a
360V/300W (CSI3711A) $499.00
www.circuitspecialists.com/csi3711a

200MHz Hand Held Scopemeter with Oscilloscope & DMM Functions

NEW ITEM

Includes 1 Year USA Warranty

You get both a 200 MHz Oscilloscope and a multi function digital multimeter, all in one convenient lightweight rechargeable battery powered package. This power packed package comes complete with scopemeter, test leeds, two scope probes, charger, PC software, USB cable and a convenient nylon carrying case.

- 200MHz Handheld Digital Scopemeter with integrated Digital Multimeter Support
- 200MHz Bandwidth with 2 Channels
- 500MSa/s Real-Time Sampling Rate
- 50Gsa/s Equivalent-Time Sampling Rate
- 6,000-Count DMM resolution with AC/DC at 600V/800V, 10A
- Large 5.7 inch TFT Color LCD Display
- USB Host/Device 2.0 full-speed interface connectivity
- Multi Language Support
- Battery Power Operation (Installed)

Item # DSO1200 **$739.00**

www.circuitspecialists.com/DSO1200

60MHz Hand Held Scopemeter with Oscilloscope & DMM Functions

- 60MHz Handheld Digital Scopemeter with integrated Digital Multimeter Support
- 60MHz Bandwidth with 2 Channels
- 150MSa/s Real-Time Sampling Rate
- 50Gsa/s Equivalent-Time Sampling Rate
- 6,000-Count DMM resolution with AC/DC at 600V/800V, 10A
- Large 5.7 inch TFT Color LCD Display
- USB Host/Device 2.0 full-speed interface connectivity
- Multi Language Support
- Battery Power Operation (Installed)

Item # DSO1060 **$529.00**

www.circuitspecialists.com/DSO1200

60MHz Hand Held Scopemeter w/Oscilloscope, DMM Functions & 25 MHz Arbitrary Waveform Generator

- All the features of the DSO1060 plus a 25 MHz Arbitrary Waveform Generator.
- Waveforms can be saved in the following formats: jpg/bmp graphic file, MS excel/word file
- Can record and save 1000 waveforms
- DC to 25 MHz Arbitrary Waveform Generator

Item # DSO-8060 **$659.00**

www.circuitspecialists.com/DSO-8060

Programmable DC Power Supplies

- Up to 10 settings stored in memory
- Optional RS-232, USB, RS-485 adapters
- May be used in series or parallel modes with additional supplies.
- Low output ripple & noise
- LCD display with backlight
- High resolution at 1mV

Model	CSI3644A	CSI3645A	CSI3646A
DC Voltage	0-18V	0-36V	0-72V
DC Current	5A	3A	1.5A
Power (max)	90W	108W	108W
Price	$199.00	$199.00	$199.00

Ideal for Law Enforcement, Post Fire Inspection, Plumbing, Facilities Maintenence, Security Companies, & Many Other Uses!

Aardvark II

Dual Camera
Wireless Inspection Camera
With Color 3.5" LCD
Recordable Monitor
Your Extended Eyes & Hands!

See It!
Clearly in narrow spots, even in total darkness or underwater.
Find It!
Fast. No more struggling with a mirror & flash light.
Solve It!
Easily, speed up the solution with extended accessories.
Record It!
With the 3.5" LCD recordable monitor, you can capture pictures or record video for documentation.

Full specifications at www.CircuitSpecialists.com/aardvark

The Aardvark Wireless Inspection Camera is the only dual camera video borescope on the market today. With both a 17mm camera head that includes three atachable accessories and a 9mm camera head for tighter locations. Both cameras are mounted on 3ft flexible shafts. The flexible shaft makes the Aardvark great for inspecting hard to reach or confined areas like sink drains, AC Vents, engine compartments or anywhere space is limited. The Aardvark II comes with with a 3.5 inch color LCD monitor. The monitor is wireless and may be separated from the main unit for ease of operation. Still pictures or video can also be recorded and stored on a 2GB MicroSD card (included). The Aardvark's monitor also has connections for composite video output for a larger monitor/recorder and USB interface for computer connection. Also included is an AC adapter/charger, video cable and USB cable. Optional 3 ft flexible extensions are available to extend the Aardvark's reach (Up to 5 may be added for a total reach of 18 feet!).

Hot New Product!!
Low Cost
$99
Aardvark Jr
Inspection Camera

Item # AARDVARK II **$249.00**

Full details At:
www.CircuitSpecialists.com/AardvarkJr

0-30V / 0-5A . DC Power Supply

The CSI530S is a regulated DC power supply which you can adjust the current and the voltage continuously. An LED display is used to show the current and voltage values. The output terminals are safe 4mm banana jacks. This power supply can be used in electronic circuits such as operational amplifiers, digital logic circuits and so on. Users include researchers, technicians, teachers and electronics enthusiasts. A 3 ½ digit LED is used to display the voltage and current values.

ww.circuitspecialists.com/csi530s

Item # CSI530S **$84.95**