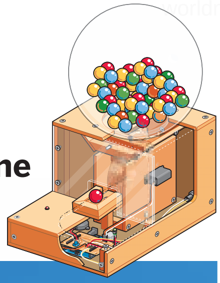
Make:

technology on your time

Build a Secret-Knock Gumball Machine
page 92 »



JOIN THE

ARDUINO REVOLUTION

MAKE STUFF SMART!

Automatic Yogurt Maker
Retro Skype Phone
Trippy Night Light

PLUS:
Sous Vide Cooker
DIY Beehive
Code 72: Real-Life MacGyvers

Make This

HELICOPTER ROCKET

3 Hours, Start to Launch!

O'REILLY®

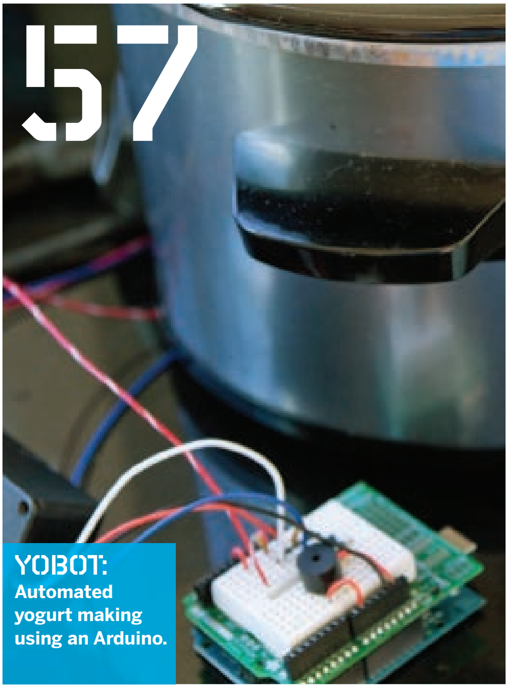
makezine.com

Volume 25

Make:

technology on your time

ARDUINO REVOLUTION



YOBOT: Automated yogurt making using an Arduino.

ON THE COVER: MAKE Production Designer Gerry Arrington and his son, Ethan, enjoy a day outdoors with our amazing $5 Heli-Rocket. Learn how to make one of your own on page 82. Photographed by Garry McLeod.

Columns

Vol. 25, Jan. 2011. MAKE (ISSN 1556-2336) is published quarterly by O'Reilly Media, Inc. in the months of January, April, July, and October. O'Reilly Media is located at 1005 Gravenstein Hwy. North, Sebastopol, CA 95472, (707) 827-7000. SUBSCRIPTIONS: Send all subscription requests to MAKE, P.O. Box 17046, North Hollywood, CA 91615-9588 or subscribe online at makezine.com/offer or via phone at (866) 289-8847 (U.S. and Canada); all other countries call (818) 487-2037. Subscriptions are available for $34.95 for 1 year (4 quarterly issues) in the United States; in Canada: $39.95 USD; all other countries: $49.95 USD. Periodicals Postage Paid at Sebastopol, CA, and at additional mailing offices. POSTMASTER: Send address changes to MAKE, P.O. Box 17046, North Hollywood, CA 91615-9588. Canada Post Publications Mail Agreement Number 41129568. CANADA POSTMASTER: Send address changes to: O'Reilly Media, PO Box 456, Niagara Falls, ON L2E 6V2

David Feind: ShopBotter

A veteran theatrical technician, David Feind builds sets for TV, film and industrial shows and has also worked as a cabinetmaker for 25 years. His career has taken him from *Die Hard 3* to building a cruise missile for the Jamie Lee Curtis movie *Virus* ("The cruise missile came out great," says Feind. "The movie…not so much.")

"My ShopBot is my favorite power tool," says Feind. "I remember what first attracted me was that ShopBot is an affordable and practical way to handle large sheets of material."

Sometimes Feind will work from a received design, other times he will design products from scratch. "I appreciate the flexibility of the ShopBot. Rather than needing two or three tools to do something, I can accomplish multiple tasks. Also, I can use my ShopBot as an input device. The ShopBot Probe can trace an existing object and bring that drawing into the software so I can recreate it perfectly – over and over."

"With a ShopBot, you're only limited by your own imagination."

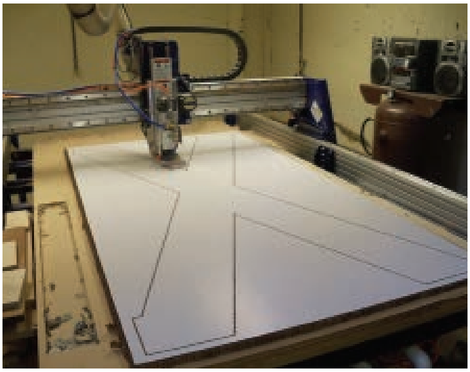


Photo by Island Creative Management

WHERE'S DAVID San Carlos, CA

BUSINESS Parts and Templates
partsandtemplates@att.net

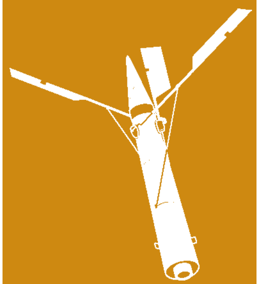
SHOPBOT TOOL PRS Alpha w/ Spindle

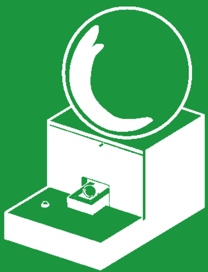
INSPIRATIONS His Dad (retired from the Marine Corps, he built furniture including a dining room table and sideboard from ammunition crates). Architects Charles and Henry Greene. Multiple childhood trips to the Smithsonian.

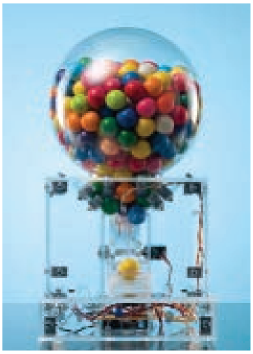
PROJECT Display for industrial show at Moscone Center

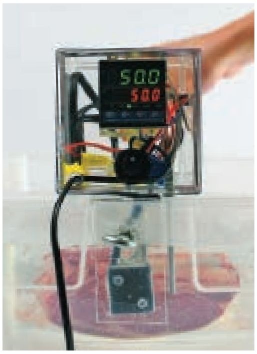
"This is a perfect example of 'why ShopBot.' We had to be able to achieve a clean cut on both sides of tricel material. There were close to 800 pages of drawings from Island Creative and their designers, and they needed it cut in 10 days. We translated the drawings into ShopBot software, and got the job done on time. This wouldn't have been possible with traditional tools."

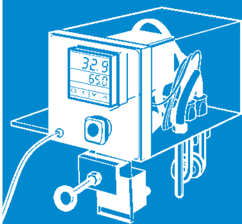
Make: Projects

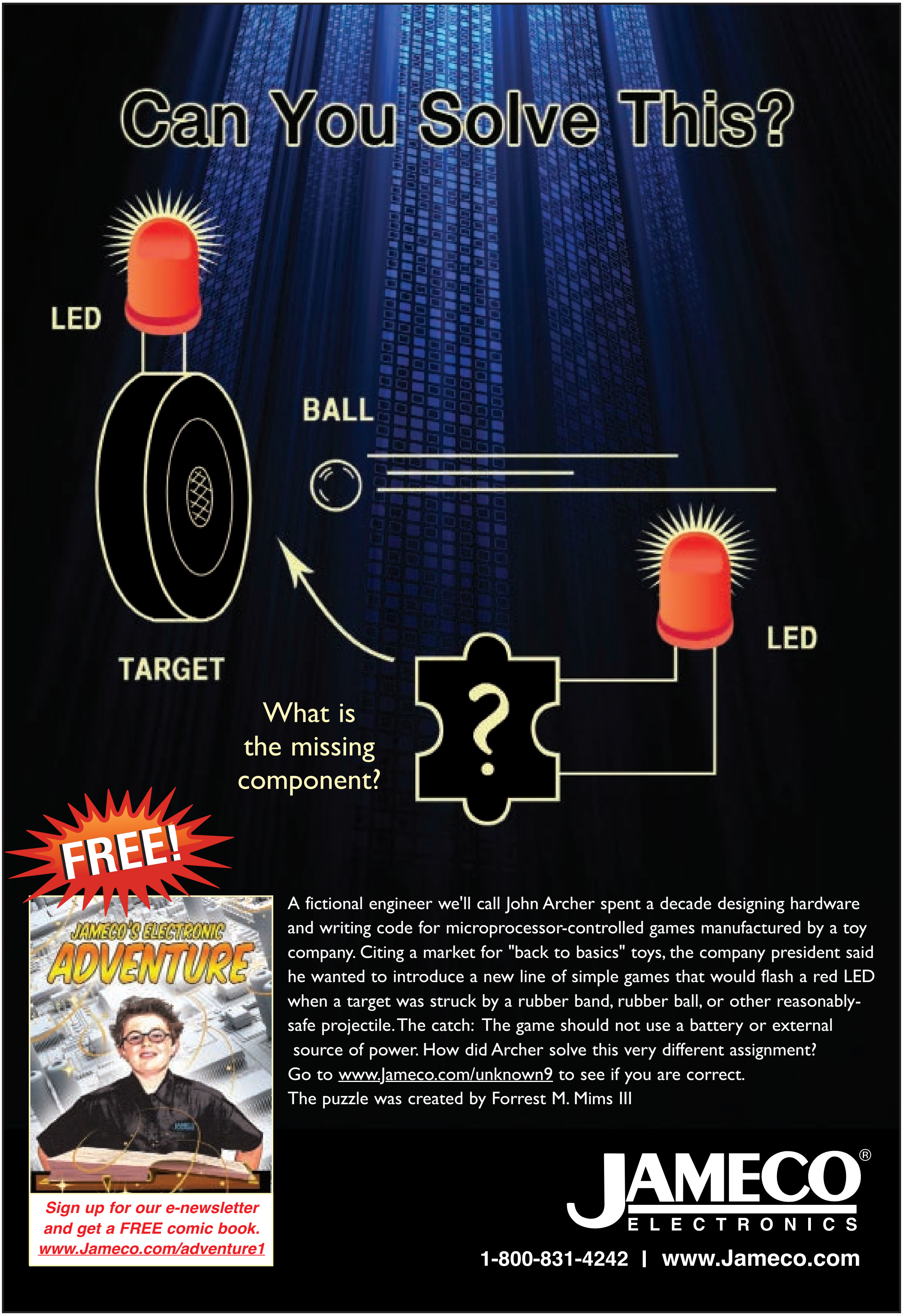












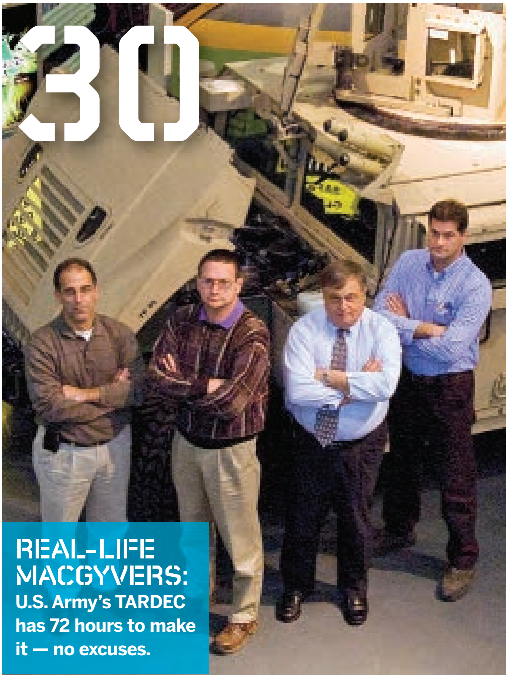
Can You Solve This?
LED
BALL
TARGET
LED
What is the missing component?
FREE!
JAMECO'S ELECTRONIC ADVENTURE
Sign up for our e-newsletter and get a FREE comic book.
www.Jameco.com/adventure1
A fictional engineer we'll call John Archer spent a decade designing hardware and writing code for microprocessor-controlled 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/unknown9 to see if you are correct.
The puzzle was created by Forrest M. Mims III
JAMECO® ELECTRONICS
1-800-831-4242 | www.Jameco.com

Volume 25

Make:

technology on your time

Maker



REAL-LIFE MACGYVERS: U.S. Army's TARDEC has 72 hours to make it — no excuses.

READ ME: Always check the URL associated with a project before you get started. There may be important updates or corrections.

1.23

You can make it!

Family fun project


KECON
Let your geek shine.
Meet Peter Madsen and Kristian von Bengtson, two of the brains behind the Copenhagen Suborbitals project. Peter and Kristian used SparkFun's Logomatic board to record vital data during the testing of their rocket. Ultimately, Kristian will man the spacecraft as it is launched into suborbital space.
Whether you're outfitting a shirt with LEDs, or sending a rocket into orbit, the tools are out there. Explore a new world and let your geek shine too.
sparkfun™ ELECTRONICS
Sharing Ingenuity
WWW.SPARKFUN.COM
© 2010 SparkFun Electronics, Inc. All rights reserved. All other trademarks contained herein are the property of their respective owners. Follow Peter and Kristian as they near their goal of launching a manned rocket into space at www.copenhagensuborbitals.com. Good luck and have a safe flight!

"Art without engineering is dreaming; engineering without art is calculating."
—Steven Roberts, the "high-tech nomad"

Make: technology on your time

FOUNDER, GM & PUBLISHER, MAKER MEDIA
Dale Dougherty
dale@oreilly.com

EDITORIAL

EDITOR-IN-CHIEF
Mark Frauenfelder
markf@oreilly.com

MANAGING EDITOR
Keith Hammond
khammond@oreilly.com

SENIOR EDITOR
Goli Mohammadi
goli@oreilly.com

PROJECTS EDITOR
Paul Spinrad
pspinrad@makezine.com

STAFF EDITOR
Arwen O'Reilly Griffith

EDITOR AT LARGE
David Pescovitz

CREATIVE DIRECTOR
Daniel Carter

DESIGNER
Katie Wilson

PRODUCTION DESIGNER
Gerry Arrington

PHOTO EDITORS
Sam Murphy
smurphy@oreilly.com
Kathryn Roach

COPY EDITOR
Gretchen Bay

EDITORIAL ASSISTANT
Laura Cochrane

ADMINISTRATIVE COORDINATOR
Ramona Minero

PUBLISHING

ASSOCIATE PUBLISHER & GM, E-COMMERCE
Dan Woods
dan@oreilly.com

DIRECTOR, RETAIL MARKETING & OPERATIONS
Heather Harmon Cochran

OPERATIONS MANAGER
Rob Bullington

DIRECTOR, PRODUCT DEVELOPMENT
Marc de Vinck

FULFILLMENT MANAGER
Sue Sidler

SINGLE COPY CONSULTANT
George Clark

DIRECTOR, MAKER FAIRE, SALES & EVENTS
Sherry Huss
707-827-7074
sherry@oreilly.com

SENIOR SALES MANAGER
Katie Dougherty Kunde
katie@oreilly.com

SALES ASSOCIATE PROJECT MANAGER
Sheena Stevens
sheena@oreilly.com

MAKER FAIRE SALES & MARKETING COORDINATOR
Brigitte Kunde
brigitte@oreilly.com

ONLINE

DIRECTOR OF DIGITAL MEDIA
Shawn Connally
shawn@oreilly.com

EDITOR-IN-CHIEF
Gareth Branwyn
gareth@makezine.com

SENIOR EDITOR
Phillip Torrone
pt@makezine.com

ASSOCIATE EDITOR
Becky Stern

DIRECTOR OF TECHNOLOGY
Stefan Antonowicz
stefan@oreilly.com

WEB DEVELOPER
Madelin Woods

COMMUNITY MANAGER
Matt Mets

PUBLISHED BY
O'REILLY MEDIA, INC.
Tim O'Reilly, CEO
Laura Baldwin, COO

Printed in the USA by Schumann Printers, Inc.

CUSTOMER SERVICE
cs@readerservices.makezine.com

Manage your account online, including change of address:
makezine.com/account
866-289-8847 toll-free in U.S. and Canada
818-487-2037, 5 a.m.–5 p.m., PST

Visit us online:
makezine.com

Comments may be sent to:
editor@makezine.com

MAKE TECHNICAL ADVISORY BOARD
Kipp Bradford, Evil Mad Scientist Laboratories, Limor Fried, Joe Grand, Saul Griffith, William Gurstelle, Bunnie Huang, Tom Igoe, Mister Jalopy, Steve Lodefink, Erica Sadun, Marc de Vinck

CONTRIBUTING EDITORS
William Gurstelle, Mister Jalopy, Brian Jepson, Charles Platt

CONTRIBUTING WRITERS
Tim Anderson, Ed Baafi, John Baichtal, Abe Connally, Larry Cotton, Frank DeFreitas, Douglas Desrochers, Cory Doctorow, Bob Goldstein, Saul Griffith, William Gurstelle, Mikal Hart, Scott Heimendinger, Steve Hoefer, Tom Igoe, Jon Kalish, Laura Kiniry, Bob Knetzger, Frank Landis, Andrew Lewis, Tim Lillis, Forrest M. Mims III, `Mike Outmesguine, Tom Parker, Kevin Quiggle, Chris Reilly, Dustyn Roberts, Zitta Schnitt, Terrie Schweitzer, Leland Sindt, Chris Singleton, L. Abraham Smith, Bruce Stewart, Jerry James Stone, Gever Tulley, Cy Tymony, Marc de Vinck, Dale Wheat, Thomas Wilson, Robert M. Zigmund, Lee David Zlotoff

CONTRIBUTING ARTISTS
Roy Doty, Brian Kelly, Bob Knetzger, Tim Lillis, Garry McLeod, Rob Nance, Charles Platt, Damien Scogin, Jen Siska

ONLINE CONTRIBUTORS
John Baichtal, Chris Connors, Collin Cunningham, Adam Flaherty, Kip Kedersha, Matt Mets, John Edgar Park, Sean Michael Ragan, Marc de Vinck

INTERNS
Eric Chu (engr.), Brian Melani (engr.), Tyler Moskowite (engr.), Ed Troxell (photo), Nick Raymond (engr.)

MAKER MEDIA CARES

PRINTED WITH SOY INK

MAKE is printed on recycled, process-chlorine-free, acid-free paper with 30% post-consumer waste, certified by the Forest Stewardship Council and the Sustainable Forest Initiative, with soy-based inks containing 22%–26% renewable raw materials.

PLEASE NOTE: Technology, the laws, and limitations imposed by manufacturers and content owners are constantly changing. Thus, some of the projects described may not work, may be inconsistent with current laws or user agreements, or may damage or adversely affect some equipment.

Your safety is your own responsibility, including proper use of equipment and safety gear, and determining whether you have adequate skill and experience. Power tools, electricity, and other resources used for these projects are dangerous, unless used properly and with adequate precautions, including safety gear. Some illustrative photos do not depict safety precautions or equipment, in order to show the project steps more clearly. These projects are not intended for use by children.

Use of the instructions and suggestions in MAKE is at your own risk. O'Reilly Media, Inc., disclaims all responsibility for any resulting damage, injury, or expense. It is your responsibility to make sure that your activities comply with applicable laws, including copyright.

Contributors

Tom Igoe (*Getting Started with Microcontrollers* and *Arduino Game Controllers*) likes playing with electronics, mechanics, and programming, making things that let people express themselves, and unusual clocks. He has written two books for makers: *Physical Computing* with Dan O'Sullivan, and *Making Things Talk*. Tom teaches at the Interactive Telecommunications Program at NYU, is a co-founder of Arduino, and believes that open fabrication can change the world. He's a fan of flat-track roller derby and lives in Brooklyn with a cat named Noodles. Currently living the dream of working with monkeys, he needs plenty of sunlight and wants to visit Svalbard someday.

Andrew Lewis (*Vintage VOiP*) is a self-proclaimed hopeless tinkerer. "I can't resist trying to bend technology to my will, which is probably why I'm not allowed to have anything nice." He's never afraid to control anything with an Arduino, so it all gets a makeover, from autoharps to sewing machines. Andrew lives in rural Staffordshire, England, with his Jack Russell terrier, Sophie. He does "technical things for money," trawls secondhand markets ("even rescuing and reusing old tins or bottles gives me a sense of satisfaction, because objects seem to have more value when there is history attached to them"), enjoys a good murder mystery, and is a splendid cook (he didn't say that, we did). upcraft.it

Brian Kelly (*Code 72* photography) was born to eat ice cream and grill outdoors, but also finds time to photograph people, places, and architecture for a wide variety of advertising, corporate, and editorial clients. He's based in Grand Rapids, Mich., and accepts interesting projects just about anywhere. Brian has been making short films and corporate videos since 2006. His documentary film *Karma Generator* screened at the prestigious Waterfront Film Festival in 2008. He currently employs HDDSLR technology and is very adept at using hybrid cameras.

Scott Heimendinger (*Sous Vide Cooker*) is a proud food geek. About four years ago, he started a blog to document his adventures in cooking. Back then, he was cooking normal food using normal equipment. But in 2009, a single egg at a local restaurant changed his life. It was the first time he'd experienced sous vide cooking: the yolk had the texture of pudding and the white was perfectly soft and delicate. It was a perfect egg, and he needed to know how to make it. Since then, he's devoted his blog to the relentless pursuit of food geekery. Next up on the DIY list: chamber vacuum sealer, rotary evaporator, rotor/stator homogenizer, and more lasers!

Steve Hoefer (*Smart Gumball Machine*) is a globally recognized inventor and creative problem solver. His creations have been featured on Asian TV, European art magazines, and American TV, radio, and print. He encourages people to engage positively with technology, and he works to remove the physical and imagined boundaries between the two. He is active in open source software and hardware, and is a promoter of community innovation and creativity. He's currently on a tour of hackerspaces, teaching a variety of workshops. You can see more at grathio.com.

Frank DeFreitas (*Holography*) is "one part retro-revolutionary, one part vintage-visionary." He saw his first hologram in 1968 and has been a full-time holographer since 1994. He still rides his bicycle around town, types on a manual typewriter, and collects historical holography memorabilia (20,000 pieces of it!). He may love history, but it goes cheek-by-jowl with the future; his current project involves "synthesizing simple three-dimensional holograms from the new Fuji FinePix 3D camera files." He lives with his wife of 35 years in downtown Allentown, Pa., in a house from the 1800s, and says simply, "My work is my fun."

BY DALE DOUGHERTY

The Little Engine That Could

Talk about coming from out of nowhere. Arduino could be the name of a high-flying Italian fashion designer or top chef. It just happened to be the name of a bar frequented by art students from the Interaction Design Institute Ivrea near Milan, Italy.

In 2005, an associate professor at the Institute, Massimo Banzi, offered the name of that bar for a simple new microcontroller aimed at artists and other non-technical users.

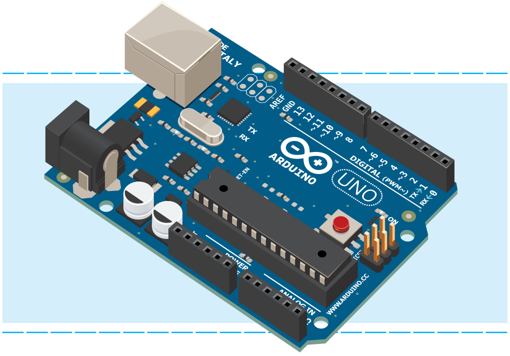
Banzi and colleagues, building on the work of others, wanted a cheap and uncomplicated way to control sensors, lights, sounds, motors, and other elements of what could be a museum exhibit, a performance, or an article of clothing.

From these humble origins, Arduino has surprisingly become a tinkering platform for all kinds of people, even engineers. At the Open Source Hardware Summit in Queens, N.Y., just before World Maker Faire last year, you could feel the enthusiasm from engineers thinking like artists, and artists thinking like engineers. They were working together and learning from each other.

While Arduino is familiar to many makers, it's unknown to most people. Arduino doesn't have a hype machine behind it and there are no ads on TV touting "Arduino Inside." As far as I can tell, no analyst is studying the Arduino market; as a commercial technology, it's practically off the radar. Perhaps those who don't know Arduino don't understand why you'd use one. It's not just another gadget.

Yet Arduino continues to pick up steam. A popular 20th-century children's book, titled *The Little Engine That Could*, told the story of a little blue engine that did what larger locomotives either refused to do or didn't believe was worth doing. The little engine, normally used to move cars in and out of the roundhouse, set off to pull a train of freight cars up a hill. The little engine climbed the hill, meeting the difficulty with determination, saying, "I think I can."

Arduino is the little engine that has proved to be more than capable for an amazingly wide range of projects. While largely ignored by the "big boys," Arduino users are the ones saying, "I think I can."



The Arduino community is growing fast, defined by cooperation and creativity more than competition. These tinkerers are discovering more imaginative and functional applications and sharing them, which only spurs others to think about what's possible.

It reminds me of the emergence of the World Wide Web, which also came from out of nowhere: Geneva, Switzerland. Those who were extremely technical would talk about all the limitations of the web, but miss how its simplicity and openness allowed so many people to do what they wanted with it.

Like the web, Arduino began with modest ambitions. It's not the most powerful microcontroller. Its virtues are being cheap and open and easy to use. Each of these virtures is important — you can freely share hardware designs and code, you can use it with any OS — but being cheap is first.

Cheap means you can try out Arduino with little investment. You don't have to know in advance whether it will do what you want — or even know exactly what you want. You can experiment and find out without risking a lot of money. An Arduino board is cheap enough that you wouldn't feel bad breaking it, burning it up, or leaving it behind embedded in a project.

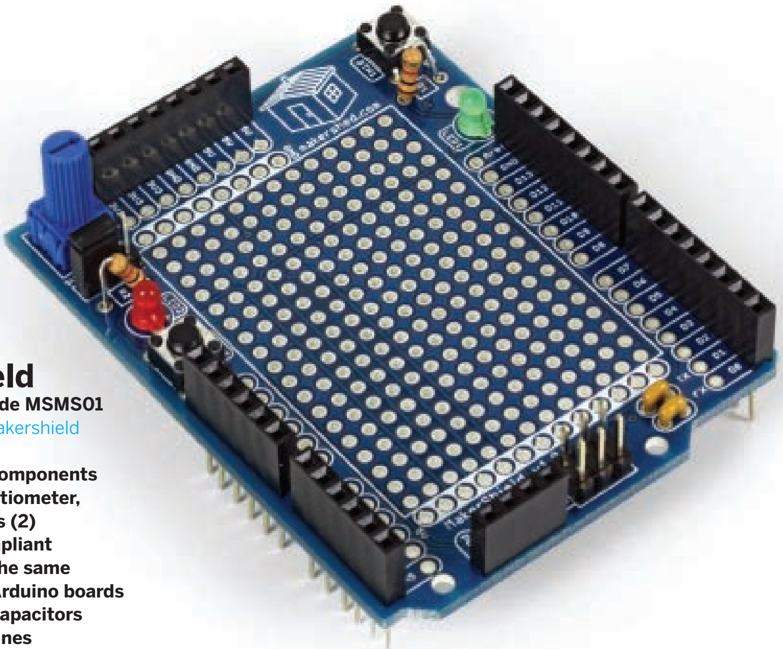
You wouldn't do that with an iPhone or a PC, but you can do it with Arduino, and for DIY, that's revolutionary. In this issue of MAKE, we devote a special section to Arduino as an engine and as an ecosystem. We want to see even more people join the Arduino revolution, saying "I knew I could."

Dale Dougherty is the founder, general manager, and publisher of Maker Media.

Illustration by James Provost

The MakerShield

The ultimate open source prototyping shield for Arduino and Netduino microcontrollers.



MakerShield
$16.95 **Product code MSMS01**
makershed.com/makershield

- **User-definable components**
- **Includes a potentiometer, button, and LEDs (2)**
- **3.3V and 5V compliant**
- **ICSP header in the same location as the Arduino boards**
- **Power-filtering capacitors on 5V and 3.3V lines**

If you aspire to do more than blink an LED with your new Arduino or Netduino, you need a prototyping shield that allows you to build a circuit. After listening to our readers and customers, we created a new kind of prototyping shield that supports the needs of noobs as well as advanced users. Introducing the MakerShield!

Unlike conventional prototyping shields, the MakerShield lets you create circuits the way *you* want, and easily change them without having to solder. All of the MakerShield's major components and pins are user-assignable, allowing you to jump from any component header pin to any pin on the microcontroller. Make all the changes you want. Just jump and go!

The MakerShield's potentiometer lets you switch between 5V or 3.3V signals, so the growing numbers of Netduino users can use MakerShield, too.

Being able to change the pins connected to the onboard LEDs, button, and potentiometer allows beginners to learn Arduino software with ease, while more advanced users will appreciate the convenience of the onboard components, the incredible flexibility, and the ability to stack another shield on top — the MakerShield uses stackable header pins and retains the original ICSP pin locations of the Arduino.

We invite you to check out the new MakerShield Kit in the Maker Shed store today!

Dan Woods is MAKE's associate publisher and general manager of e-commerce.

READER INPUT

Jailbreak catch-22, kids ♥ robots, ladder logic, and space zillionaires.

✉ In Volume 23, Cory Doctorow's "Make Free" column stated that it is illegal to jailbreak devices due to the Digital Millennium Copyright Act (DMCA). However, this act was updated in July 2010 and it is no longer illegal.

—Nicky Semenza, Santa Clara, Calif.

AUTHOR'S REPLY: Good point, Nicky, but not entirely true. Under the July rulemaking by the U.S. Copyright Office, it's not illegal, for the next three years, to jailbreak a mobile phone handset — but it's probably still illegal to jailbreak a tablet.

And it's still illegal to traffic in, create, or distribute all jailbreaking tools. So it's a catch-22: it's legal to jailbreak a phone, using illegal tools, if you can get them.

✉ This is what happens when you're retired: I believe that in Volume 23, "Primer: Programmable Logic Controllers," in Figure A, the + terminal should be connected to the base of the reset switch, not to one side. I understand how these work, but if someone is just starting out it could cause confusion, especially if they're starting to learn about ladder logic. Hope this helps.

I really like your magazine; it encourages younger people to make things, not just stare at screens.

—John Deming, Wilmington, Del.

EDITOR'S REPLY: John, you're right, we drew that switch wrong. Thanks for the sharp eyes.

✉ I wish to congratulate you on your fine magazine. There was a slight omission in your Volume 24 article "Zillionaires in Space." You forgot about Guy Laliberté, founder of Cirque du Soleil, who not only flew into space but spent a total of 12 days aboard Soyuz. Being Canadian we are proud of that fact.

—Ronald Santerre, Montreal, Québec

✉ Thanks for furthering the DIY movement. Three years ago, I didn't own a soldering iron, let alone know how to use one. Bre Pettis taught me how to solder via the MAKE Weekend Projects podcast (makezine.com/podcast). Since then, I've

built a TV-B-Gone and hacked several circuits, and now I'm the happy owner of a Sparkle Labs DIY Design Electronics Kit.

—Andy Johnson, Taylorsville, Utah

✉ I recently discovered MAKE (the magazine and website) and I really love it. It's inspired me, and I've been using it to inspire my three kids, too. The other day my 8-year-old asked me how to build a robot, and then promptly went and looked it up online at makezine.com.

—Kendra Hoffman, Chaska, Minn.

MAKE AMENDS

In Volume 24's "Lego Antikythera," we mistakenly ran a photo showing Andrew Carol's incredible Babbage difference engine made of Lego, instead of his incredible Antikythera mechanism made of Lego. The corrected article is available in the MAKE Digital Edition and also at makezine.com/24/made_legoantikythera.

In Volume 24's "Map the Chemical Composition of the Moon," holding the camera by hand works well, not because the telescope's mass reduces the camera's vibration, but because the camera doesn't touch the telescope.

In Volume 23's "Primer: Programmable Logic Controllers," Figure A, the + terminal should connect to the base of the reset switch, not to one side.

In Volume 23's "Fooling Around with Pulleys," the load in Figure A is supported by 5 rope segments, not 6.

Illustration by Roy Doty

BY SAUL GRIFFITH

The Art of Productive Procrastination

About four times a year I seem to be able to concentrate well for eight hours in a row — in fact, about 48 hours in a row — and I just let it happen. I don't know when it will occur or why, but it's typically when some manic inspiration strikes.

The rest of the time, I am rarely, if ever, doing what I should be doing. I've never been able to go straight down the to-do list, to the chagrin of friends and co-workers, yet still everything ends up getting done, and a little more. Since that seems to be the way I'm wired, I have found some simple ways to productively procrastinate. Here's what works for me; maybe it will work for you.

I gave up on trying to do exactly what I was meant to be doing in favor of always doing something. Frankly, I'm not sure we're designed to focus on only one thing for eight or ten hours in a row. I've always found that it's useful to have something else to be doing when you're too burnt out to face the next thing on your list. That way, flipping back and forth between the two projects prevents focus fatigue.

Now, the most important thing is to make sure your other project isn't "browsing on YouTube" or "catching up on Facebook." Make it a project that forces you to learn, because you want to.

Always have a learning project in mind. For me, it's typically learning some new tool, some new math, some new physics, or some new programming skill. It doesn't really matter what it is, just have something on the back burner. It helps if it's a skill you might need in your next project.

Then figure out something fun that requires that skill, like making a Sierpinski-triangle chopping board. Make sure you need your new skill to complete your weird and fun new project. The desire to finish the project will force you to learn the skill, and that skill will be available to you when you tackle a more serious project in the future.

I'll illustrate this with my most recent example. I have projects in the future where I know I'll need to do a lot of data visualization. I also have projects where I want to use more algorithm-based design. Christmas is coming up, and I want to give my friends who have children something cool and handmade, so I decided to make an alphabet book

Flip between two projects to prevent focus fatigue.

completely algorithmically: a computer program wrote, typeset, and produced the entire thing.

Why? I've always had a passion and fascination with fonts. And I need more day-to-day MATLAB skills. It's not something that needs to be done immediately — I can do it in the hours between other projects and work, and at the end of it I'll be a better programmer, understand fonts, colors, and visualization- and algorithm-based design better, and I'll have an awesome gift for my kid and others.

I've been at it for about three weeks now, spending maybe an hour a day (more when on airplanes, less when at home). It looks close to being finished. Magically, I've learned a whole bunch of skills that had always avoided me because I wasn't motivated to learn them — because I found a way to motivate myself. That's like a full-semester programming course finished in three weeks while skill-building and distracting me just enough from real work to make my real work more productive. (The next question is, what should I be doing while I avoid writing my MAKE column?)

I love Clay Shirky's concept of cognitive surplus, and the fact that there are more people with more time to contribute to more cool things than ever before, and that we can share all this learning and doing because we now have the web. I'll be able to share my book and my code with the world. Someone will improve the code, or change it, or find a creative link or nugget in it, and the world is improved. Everyone's a winner.

Harness your procrastinating self by fooling yourself into being motivated. Find recipes for your own ideal procrastination projects. Keep a list of them handy. You'll never find yourself zombie-eyed in front of a video game ever again. You'll be creating something new instead.

Saul Griffith is a new father and entrepreneur. otherlab.com

Jan	Feb	Mar
Apr	May	Jun
July	Aug	Sept
Oct	Nov	Dec

MAKER'S CALENDAR

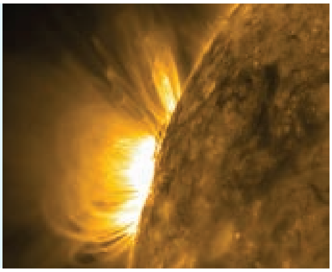
COMPILED BY WILLIAM GURSTELLE

Our favorite events from around the world.

Sun-Earth Day

March 20, worldwide

Established by NASA and the European Space Agency, Sun-Earth Day encourages people to explore the way the Sun influences life on Earth. This year includes a special webcast that highlights several sun sites, such as Hovenweep, Utah; Chichen Itza, Mexico; Cahokia Mounds, Ill.; and Sunspot, N.M. **sunearthday.nasa.gov/2011**



FEBRUARY

» Elemental Matters: Artists Imagine Chemistry

Opens Feb. 4, Philadelphia, Pa.

To celebrate the International Year of Chemistry (chemistry2011.org), the Hach Gallery at the Chemical Heritage Foundation is exhibiting the work of seven contemporary artists whose artwork centers on the elements, producing surprising and evocative works of art. **chemheritage.org**

» North American Handmade Bicycle Show

Feb. 25–27, Austin, Texas

Last year, 150 exhibitors and 7,200 visitors attended this event, which showcases the talents of bicycle builders from around the world. Bespoke frames and handmade components come together at NAHBS. **handmadebicycleshow.com**

» Final Flight of the Space Shuttle

Feb. 27, Kennedy Space Center, Fla.

When *Endeavor* leaves the launchpad to deliver spare parts to the International Space Station, it will likely be the final blastoff for the venerable Space Shuttle Program. After 30 years and 134 missions, those birds will fly no more. Well done, NASA! **nasa.gov/shuttle**

MARCH

» National Pyrotechnics Fair

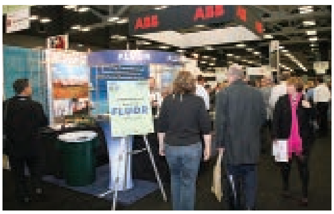
March 8, Tultepec, Mexico

The event includes some of the most spectacular and most dangerous homemade fireworks on the planet. This not-for-the-faint-of-heart happening culminates with the "Burning of the Bulls," a celebration of San Juan de Dios, patron saint of Tultepec's pyrotechnicians. **makezine.com/go/pyrofair**

Renewable Energy World Conference and Expo

March 8–10, Tampa, Fla.

For makers with a green streak and professionals in the renewable energy industry, this event's got presentations and technical sessions related to technology, business strategies, and policy covering the wind, solar, biomass, hydro, geothermal, ocean/tidal/wave, bio-power, biofuels, and hydrogen energy industries. **renewableenergyworld-events.com**



APRIL

» Printemps des Sciences

March 28–April 3, Belgium

The Printemps des Sciences is an annual springtime festival that seeks to raise interest in scientific careers among young people. The outreach event is organized by several large Belgian universities. **printempsdessciences.be**

» HAuNTcon

April 27–May 2, Louisville, Ky.

HAuNTcon delivers three days and nights of how-to education, haunted house tours, a trade show, a hearse show, and a costume ball. The event gives haunters (as they call themselves) a place to network, share tips and stories, get fresh ideas, and check out the latest in haunting props. **hauntcon.com**

✱ IMPORTANT: Times, dates, locations, and events are subject to change. Verify all information before making plans to attend.

MORE MAKER EVENTS:

*Visit **makezine.com/events** to find events near you, like classes, exhibitions, fairs, and more. Log in to add your own events, or email them to **events@makezine.com**. Attended one of these events? Talk about it at **forums.makezine.com.***

Photograph (Sun) courtesy NASA

BY CORY DOCTOROW

Untouched by Human Hands

I spent most of 2008 researching my novel *For the Win*, which is largely set in the factory cities of South China's Pearl River Delta. If you own something stamped MADE IN CHINA (and you do!), chances are it was made in one of these cities, where tens of millions of young women have migrated since the combination of Deng Xiao Ping's economic reforms and the World Trade Organization agreement set in motion the largest migration in human history.

It's difficult to characterize the products of these factories: everything from high-priced designer goods to the cheapest knockoff originates there (on average, one container per second leaves South China for America, every second of every hour of every day).

But there's one characteristic almost all these products share: they're produced on an assembly line, and they're supposed to look like it. Indeed, it's almost impossible to imagine a mainstream store that sold handmade goods for the purpose of daily use by average people.

The notion of "handmade" has undergone several revolutions in the past century, its meaning alternating between "precious and artisanal" and "cheap and inferior."

Artisanal fashions have likewise swung between the two poles of "rough and idiosyncratic" and "all seams hidden, every rough edge sanded away."

Today, the fit and finish that the most careful, conscientious artisan brings to her creations usually ends up making them look machine-finished, injection-molded, seamless as if they were untouched by human hands, not because they were lovingly handled until every blemish was gone.

What's more, the increasing awareness of the environmental and human cost of intensive manufacturing has started to give factory goods a whiff of blood and death.

Your new mobile phone was made by a suicidal Foxconn worker, from coltan mud extracted by slaves in a brutal dictatorship, shipped across the ocean in a planet-warming diesel freighter, and it's destined to spend a million years in a landfill, leaching poison into the water table.

Is there a boardroom somewhere trying to figure out how to make your next Happy Meal toy, laptop, or Ikea table look like it was handmade by a MAKE reader?

Which leads me to wonder: is there a boardroom somewhere where a marketing and product design group is trying to figure out how to make your next Happy Meal toy, laptop, or Ikea table look like it was handmade by a MAKE reader, recycled from scrap, and sold on Etsy?

Will we soon have Potemkin crafters whose fake, procedurally generated pictures, mottoes, and logos grace each item arriving from an anonymous overseas factory?

Will the 21st-century equivalent of an offshore call-center worker who insists he is "Bob from Des Moines" be the Guangzhou assembly-line worker who carefully "hand-wraps" a cellphone sleeve and inserts a homespun anti-corporate manifesto (produced by Markov chains fed on angry blog posts from online maker forums) into the envelope?

I wouldn't be surprised.

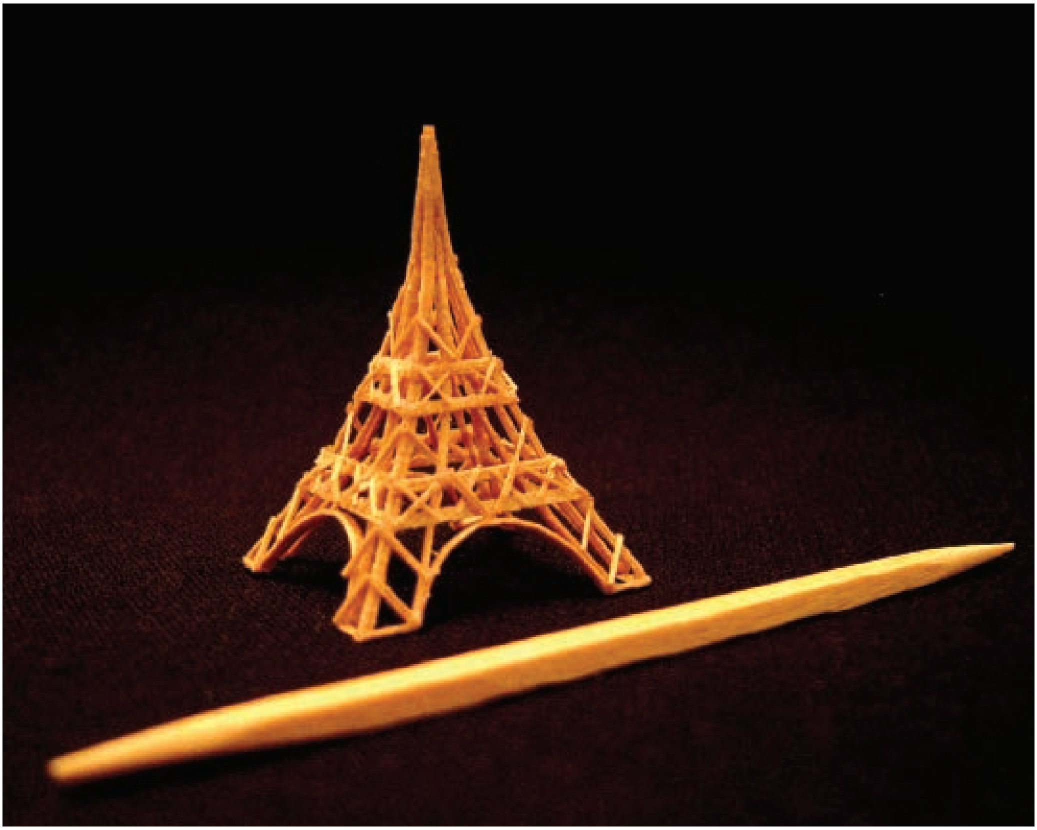
Our species' capacity to commodify everything — even the anti-commodification movement — has yet to meet its match.

I'm sure we'll adapt, though.

We could start a magazine for hobbyists who want to set up nostalgic mass-production assembly lines that use old-fashioned injection molders to stamp out stubbornly identical objects in reaction to the corporate machine's insistence on individualized, 3D-printed, fake artisanship.

Cory Doctorow's latest novel is *Makers* (Tor Books U.S., HarperVoyager U.K.). He lives in London and co-edits the website Boing Boing.

Photography by toothpickart.com



Toothpick Artistry

The online exhibit is called "The Essence of Patience," and it takes just one look at **Steven J. Backman**'s work to grasp the truth in that title.

Using just small hand tools, enormous patience, and lots of toothpicks, Backman creates amazingly intricate works of art. From detailed replicas of historic buildings, to a functional remote-control boat, his work pushes the boundaries of what we'd expect could be created out of tiny slivers of wood.

Backman grew up in San Francisco, and many of his sculptures reflect the region. Famous icons like the Golden Gate Bridge, the Fairmont Hotel, and the city's cable cars figure prominently in his art.

His largest sculpture is of the city's famous bridge, a 13-foot-long replica using 30,000 toothpicks that took more than two years to complete. But he also created a tiny version of the span that's smaller than a single toothpick.

Backman first started making things out of toothpicks in elementary school, where he made a DNA molecule out of toothpicks and beans. Nearly 40 years later he's still fascinated with the medium.

Unlike carving a solid piece of wood, toothpicks remind Backman of building blocks, but on a scale that allows him to produce striking results. "Once I create a sculpture, it looks like it was carved from a solid piece of wood," he proudly exclaims.

His work is created with the simplest of tools — razor blades, tweezers, needlenose pliers, and glue. He doesn't use loupes or magnification, even on the tiniest pieces. He prefers square center toothpicks with flat ends, and he gets his massive supply from Diamond.

Backman's work has been featured in the Empire State Building lobby, hotels in San Francisco, the Golden Gate Bridge's 50th anniversary celebration, and Ripley's Believe It Or Not museum. Toothpicks as an artistic medium might seem improbable, but Backman seems to have found his calling — even though, as he readily acknowledges, "It is truly a painstaking process, but it's a labor of love!"

—Bruce Stewart

One Toothpick at a Time: toothpickart.com

Quickup Camper

Over the years, **Jay Baldwin** has studied with Buckminster Fuller, served in the U.S. Army's ski troops in Alaska, and worked as an editor at the seminal *Whole Earth Catalog*. The maverick industrial designer, who still rides a mountain bike at the age of 77, has designed and built a prototype for what he calls the Quickup Camper, a collapsible camper top that transforms a standard-sized pickup truck into a fuel-efficient, highly maneuverable recreational vehicle.

Baldwin built the Quickup Camper in the barn of his Northern California home, with the help of students from California College of the Arts, where he teaches. His wife, **Liz Fial**, made the interior cabinets, counter, bunk bed, and seating out of Finnish plywood.

The Quickup Camper takes about 60 seconds to open into an 8'×8' living space with 6'4" of head-room. It has hot and cold running water and a solar-powered refrigerator. About the only problem he's had is finding a heater small enough not to overheat the interior, which is enclosed in a shell of carbon fiber-covered structural foam insulation.

Don't get him started on traditional RVs, which he refers to as "petro pigs" with the maneuverability of a "condo-on-a-skateboard." He gets 20mpg at a speed of 70mph with the Quickup Camper bolted to his Ford's bed — double the gas mileage of most RVs. In fact, the Ford gets better gas mileage with the 800-pound Quickup Camper attached than without it. That's because, folded, it fits within the contour of the truck's bed and improves aerodynamics.

"One of the beauties of this Quickup Camper is I can take it into downtown San Francisco and park it in a municipal garage. It fits just fine. On the street it's perfectly legal. It's no bigger than a pickup," he says.

The Quickup Camper made an appearance at the 2008 Maker Faire Bay Area. But until he finds a manufacturer for his camper top, Baldwin will be driving around with the only one on the planet. He figures it would cost between $10,000 and $12,000, in addition to the cost of the truck itself.

—Jon Kalish

A New Breed: quickupcamper.com

Photograph by Liz Fial

Photograph by Michel de Broin

Invisible Sculpture

In the spring of 2004, artist **Michel de Broin** approached a large rock in a forest in the Vosges Mountains, France. He carried with him glue, cement, and pieces of mirror.

Having received an invitation to "reflect upon the notion of transparency," de Broin responded by affixing pieces of mirror to the rock, maintaining its contours, until the rock's surface was entirely covered in mirrors. The result is *Superficial*.

In *Superficial*, the object's environment is enlisted as a kind of fifth material. Depending on the position of the viewer, the mirrors render the surroundings accurately while mutating the significance of what's being seen. The rock seems to disappear into its surroundings while simultaneously appearing highly conspicuous.

Associations traditionally made with mass and transparency are inverted. Contrasts between natural and artificial are amplified in the form of alien presence. By reminding the viewer of his/her presence, *Superficial* intervenes in what might otherwise be a continuous experience of nature.

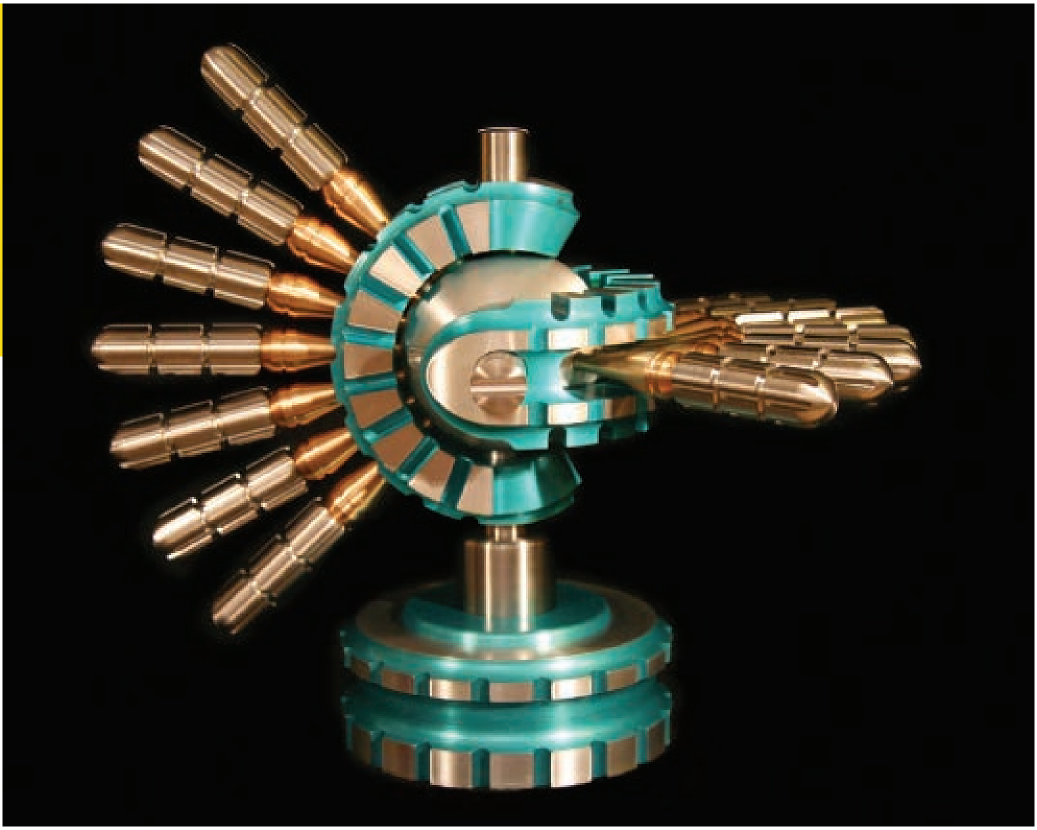
Canadian-born and now Paris-based, de Broin often uses common materials that have been displaced and reconfigured, such as old televisions, light sockets, and street signs. His award-winning work (in 2007 he won Canada's Sobey Art Award) renders the familiar strange, and in doing so it tends to exacerbate certain assumptions.

De Broin sees his objects as mischievous, intervening in the dynamic between individuals and "modern utopian aspirations." He reminds us that any reality, no matter how well accepted, is still an interpretation or representation of itself.

"The main idea for me regarding art is to try to open a small gap in the meaning/construction of reality so that viewers can construct a meaning for themselves," De Broin told the art blog View on Canadian Art. "The subject succeeds when people are questioned by it, [when] they have to participate in the creation of meaning."

—Thomas Wilson

Making Strange: micheldebroin.org



Tools of Artistry

For artist **Chris Bathgate**, the learning process is his creative process. His intricate sculptures showcase the perfection of metal and the nuts and bolts of machining. Each one takes countless hours of turning and milling bar stock into something that defies age and the touch of the human hand.

"The process of learning has been my biggest source of inspiration," Bathgate says. "So you could say a lot of my visual ideas are spawned by the act of machining itself."

But he wasn't always a machinist. After a brief stint in art school, he "got a job and just started buying tools ... once I bought my first small milling machine and realized the potential for making some very precise and interesting works, I was hooked." He polished his craft with manual tooling before teaching himself CAD and making the switch to CNC machines.

A true maker, he builds many of his tools from scratch. "In addition to the multiple machine tools I've built," says Bathgate, "I have built countless fixtures, jigs, hand tools, digitally operated kilns, set-up apparatuses, electroplating tanks, anodizing equipment, and many other thingamajigs in between."

And making the machines has profoundly affected his work. While he's not above mining existing parts for metal he admires — such as recycling copper cores from a linear accelerator's scrap bin for their luminous oxygen-free alloy — he never uses preexisting shapes ("they tend to have their own visual references and meanings"). He's also started making detailed CAD drawings of each piece, whether the parts were originally designed in CAD or just evolved on the mill or lathe.

But what are these unique hunks of metal? Futuristic visions? Alien creatures? "Viewers tend to take away from my work what they bring to it, [but] I strive to keep my work in that vast gray area where it never quite becomes anything specific," says Bathgate. "I feel it is far more interesting that way."

—Arwen O'Reilly Griffith

Mental Metal: chrisbathgate.com

Photograph by Chris Bathgate

Photograph by John Wells/The Field Lab

Biker Laundry

When life offers up an old washing machine and a bicycle, what do you do? The dump isn't a viable option, and recycling seems like a cop-out. So if you're **John Wells**, you build the Bike-O-Worsher.

Wells isn't one to back away from a challenge. In 2007, he switched from a life in New York (almost 20 years in New York City) and headed for the desolate desert near Terlingua, Texas, just a hop, skip, and a jump across the Rio Grande from Mexico.

Within the first few months, he'd built himself a modest house, a solar cooker, and an outdoor shower. In the ensuing years, he's added a wide array of projects, all designed to help him "learn about how to prosper comfortably living a back-to-basics life."

The Bike-O-Worsher is his latest build. Besides the old washing machine and bike, Wells used some pulleys and belts; after about four days of building and troubleshooting, he was in business.

"It takes as long as a normal washing machine," he says. "But I prefer the wash and soak, and wash and soak, since I'm out of shape. The mechanism is very easy to pedal. It's like riding a bike along a flat road."

The Bike-O-Worsher works just like any washing machine — you pedal forward for the agitation and pedal backward for the spin cycle. Wells fills the tub manually and hops on to wash the clothes. He drains the water into a graywater capture, then does a spin cycle. He follows with another water refill and agitation cycle for the rinse, then drains and spins again.

"Just a bit of exercise and your clothes are as clean as using a store-bought washer," he says. "Why buy exercise equipment when all that energy is wasted?"

Bike-O-Worsher 2.0 is already in the works and will be a front-loader that spins horizontally and uses much less water per load. And so far, Wells reports that the reactions have been positive. He quips, "My favorite comment is, 'Why don't you hook the bike up to an electric motor?' "

Living where he gets about 315 days of sunshine a year, Wells says he'll have no need for a Bike-O-Dryer. "That's what a clothesline and the sun are good for," he enthuses. —*Shawn Connally*

John Wells' Field Lab: thefieldlab.blogspot.com

Photograph by Bill McHugh

Backyard in Motion

Some animals have all the luck. Just ask retired mechanical contractor **Bill McHugh**, who's spent more than 40 years creating fun kinetic sculptures that run on air, sun, water, birds, and most typically, squirrels.

In his backyard in Narberth, Pa., they line up by the dozen to try out his peanut-loaded machines. They're especially fond of the *Whirl-a-Squirrel*, a whip-like contraption that runs on 200ppd (peanuts per day); and the *Twirl-a-Squirrel*, a cymbal-clanging, bell-ringing piece that plays Jane Fonda-type exercise tunes and operates on 2,000 peanuts weekly.

McHugh began his hobby by crafting wood pieces, but switched early on to more abstract sculptures using aluminum and stainless steel, materials unencumbered by snow and virtually maintenance free.

Each of McHugh's pieces requires an average of 200–400 hours to create, but he doesn't feel comfortable putting a price on his time and energy. Instead, he makes each piece available for long-term loan. Four pieces are currently on loan: one with a local art collector, one with a Philadelphia Zoo official, one with the nearby Lower Merion Conservancy, and the fourth with his daughter in Virginia.

While machines like his solar-powered *DNA Is for the Birds* — a corkscrew-shaped bird feeder — and the aptly named *Hitchcock* — a 10-foot-tall, bird-driven, rotating water fountain — are quite impressive, McHugh's favorites are his four *Pterodactyl* sculptures, one of which stands in his front yard.

With a 10-foot wingspan, it employs solar energy to flap, chomp, and rotate, and features an intercom that McHugh uses to greet visitors, most of whom he lets wander through his Backyard in Motion free of charge (he's cautious about small children because of possible injury). Those who McHugh thinks qualify also receive a tour of his workshop.

Despite decades working in relative obscurity, it was only a matter of time before Hollywood called. "Rosie O'Donnell recently requested the *Whirl-A-Squirrel*," McHugh says. "I told her it was promised to somebody else, but I'd put her on the list."

—Laura Kiniry

Kinetic Sculpture: backyardinmotion.com

Photograph by Paul O'Connor

Cinema with Sol

A tiny, solar-powered cinema is making its way across Europe, promising patrons an intimate movie-viewing experience and red-carpet treatment.

The Sol Cinema is a mobile, solar-powered movie house run by the U.K.-based charity **Undercurrents**. The tiny cinema, which seats eight adults or 12 kids comfortably, shows all kinds of films, especially those on environmental issues such as climate change. It's like Current TV but with ushers.

The micro movie house — originally a two-berth caravan from the 1960s that cost about $100 — took "a [whole] year of head-scratching to be able to fit everything," says Undercurrents co-founder **Paul O'Connor**. And it took an additional four months to actually convert it.

Artist **Jo Furlong** and volunteers **Ami and Beth Marsden** completed it in 2009. Renovation costs were minimized through very dedicated dumpster diving and many trips to the local junkyard.

The most expensive components of Sol Cinema are two 120W solar panels ($1,400 each). The panels juice four lithium-ion electric car batteries that were imported from Hong Kong due to their lack of availability in the U.K. And while it takes a full day of sunshine to top off the batteries, the movie house rarely uses a full charge.

During a typical event, they showcase a series of shorts from 11 a.m. to 7 p.m., and then finish it off with a full-length feature film. The feature is projected on an outdoor screen using a 1,500-lumen projector, which cost $1,600. O'Connor handled the projection and video equipment design and installation.

The mobile movie house is transported by a Volkswagen Eurovan powered by waste vegetable oil. Last year was Sol Cinema's first season, with bookings throughout Ireland and the U.K. For 2011, Undercurrents is hoping for a full European tour with its sights set on the Cannes Film Festival.

It wouldn't be the best venue for watching *Avatar*, but it's a surefire hit for the follow-up to Al Gore's *An Inconvenient Truth*. Don't forget the organic, fair-trade popcorn! *—Jerry James Stone*

Sol Cinema Specs: thesolcinema.org

BY FORREST M. MIMS III

Capturing and Studying Airborne Dust, Smoke, and Spores

Unless you're inside a laboratory clean room or wearing a dust mask, each breath of air you're now inhaling probably includes a fair number of dust particles and mold spores and, during spring and fall, a few pollen grains (Figure A). If you're in good health, your respiratory system will filter out most of these particles.

You can find out more about these particles with an inexpensive microscope and a DIY air sampler.

Passive Air Samplers

In this column we'll experiment with the simplest air samplers, those that rely on gravity or wind to deposit particles in the air onto adhesive tape or a bare microscope slide.

Let's begin with the microscope slide covered with ragweed pollen shown in Figure B. Behind the slide is the windshield of my pickup truck, also coated with pollen. The pollen grains are held in place by the molecular attraction known as van der Waals force, which is strong enough to secure the pollen on the windshield of a car moving along a highway at 70mph.

Figure D shows an air sampler I've used while driving the highways of Texas and New Mexico. It's simply a plastic vacuum cleaner attachment with a microscope slide taped across the wide end. The assembly is taped to the hood or a side window of a car.

This passive sampler has collected a wide variety of particles in the air, including the *Alternaria* spore in Figure C that was photographed by holding a digital camera at the microscope eyepiece.

The biggest drawback of these passive air sampling methods is that they provide only the particles, with no information about their concentration in the air. Yet even a simple passive air sampler can lead to a scientific discovery, as my daughter Sarah learned when she was in high school.

"Stuff in the Air"

For her first major high school particle project, Sarah tried various methods to collect African dust that often drifts over Texas during summer. These included loops of bare adhesive tape, a car vacuum cleaner equipped with a filter, and a sampler she made from an enclosed muffin fan and a coffee filter. All these methods collected many dust particles, but they were difficult to find when the filters and tape were examined with a microscope.

Ordinary microscope slides worked much better. Each afternoon during her study, Sarah placed a clean microscope slide on a 12-foot tower in our field. She stored the exposed slides in a dust-tight microscope slide box that kept them separated. And she used online NASA satellite images and Navy Research Lab dust and smoke forecasts (online at www.nrlmry.navy.mil/aerosol) to verify that the dust she collected fell from the sky on days when Sahara dust was at our place.

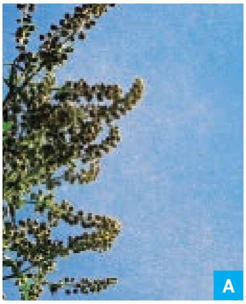
Sarah's Sahara dust project, which she called "Stuff in the Air," earned many science fair awards, including a grand prize from the Texas Junior Academy of Science.

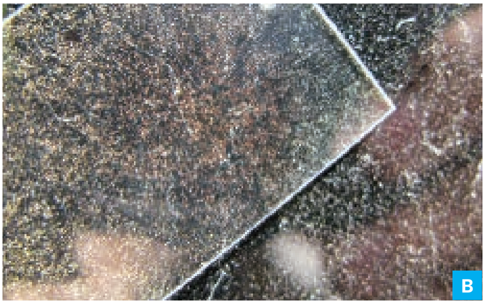
"Smoke Bugs"

Sarah expanded her air sampling research to look for mold spores that might be accompanying dust from distant places. Scientists had already discovered that mold spores were crossing the Atlantic along with dust from Africa. So she decided to look for spores in dust that often arrives in the United States from Asia each spring, by exposing microscope slides each day for two weeks when Asian dust was expected over our place.

Although the Asian dust clouds passed north of our place, Sarah's microscope slides were loaded with spores. Atmospheric scientist Dr. Tom Gill told Sarah that smoke from agricultural fires in Yucatan, Mexico, was passing over our site during the week or so that she was exposing her slides.

Sarah used a microscope to find and count 22 genera of fungal spores and many black carbon particles (smoke) that she observed during four scans across the length of each microscope slide. She entered the data in a spreadsheet and made





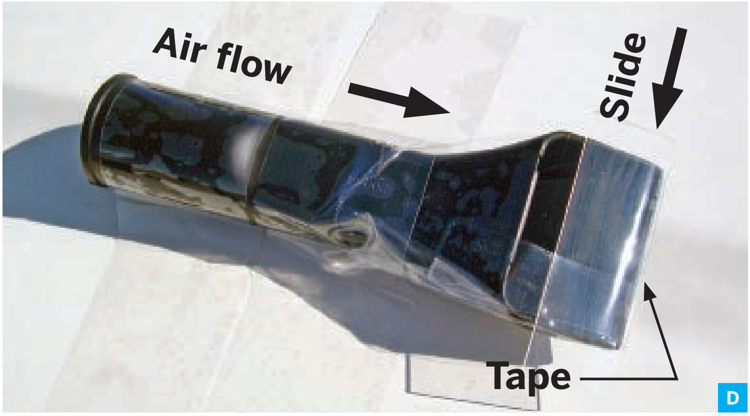


Fig. A: Pollen blowing from a giant ragweed plant, one of the worst causes of hay fever.
Fig. B: Pollen accumulates on windshields.
Fig. C: *Alternaria* spore, about 25μm, collected using the sampler. Ragweed pollen is about 20μm in diameter, and human hair 60μm–100μm.
Fig. D: Air sampler made from a vacuum cleaner inlet fixture and a microscope slide.

an x-y scatter graph of the number of spores versus the number of black carbon particles on each slide (Figure E, following page).

The correlation coefficient (r^2) is a statistical parameter that expresses how well two variables agree with one another, with 1.0 being perfect agreement. In this case, r^2 was 0.78, which strongly suggested that the fungal spores were associated with the smoke.

Sampling the Air from a Kite

Sarah wanted to be sure that most of the spores on her microscope slide had arrived with smoke from Yucatan. The year after her initial discovery, we made two trips to the coast, the first when considerable smoke was arriving from Yucatan and the second several months later when there was little smoke from Yucatan.

On both days the breeze was arriving from Yucatan, and on both days Sarah stood in or near the water and flew a simple DIY air sampler from a kite at an elevation of about 50–100ft to avoid contamination from Texas spores (Figures F and G).

After 30 minutes of exposure, she retrieved the kite, removed the sampler, and used her microscope to examine the slide (Figure H). The results of the kite flights clearly showed significantly more spores and black carbon particles on the smoky day than on the clean day.

Sarah's discovery that smoke plumes from distant fires can include many spores and bacteria led to more awards, including another grand prize in the Texas Junior Academy of Science and a NASA web page (earthobservatory.nasa.gov/Features/SmokeSecret).

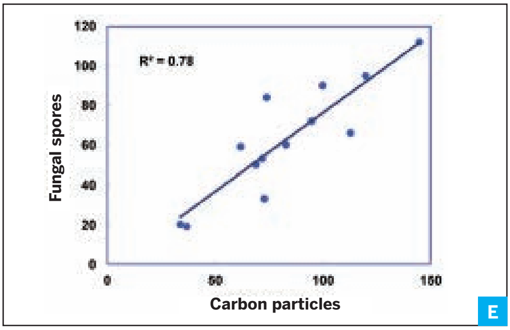
Since we could find no reports about the long-distance transport of microbes in smoke plumes, I helped Sarah convert her "Smoke Bugs" science report into a formal scientific paper, which was published on a priority basis by *Atmospheric Environment* ("Fungal spores are transported long distances in smoke from biomass fires," vol. 38, February 2004, pp. 651–655).

How to Make an Airborne Air Sampler

Smoke and dust can travel halfway around the Earth, and you can use the airborne sampler method to look for spores, carbon particles, pollen, and dust that arrive at your location.

Sarah made her airborne sampler from a 16oz plastic cup, 2 binder clips, a fishing tackle snap swivel, a microscope slide, and kite string. Follow

Photography by Forrest M. Mims III



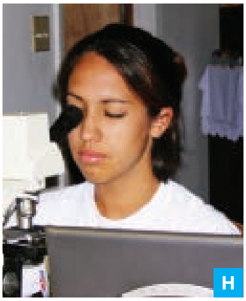


Fig. E: This scatter graph shows an association of mold spores and carbon particles (smoke) collected on microscope slides during 13 days when smoke from Yucatan drifted over South Texas. Fig. F: A DIY passive air sampler flies from a delta kite at Padre Island, Texas, to capture spores in smoke from agricultural fires in Yucatan. Fig. G: At the bottom of the air sampler, binder clips hold the microscope slide in place. Fig. H: Sarah Mims inspects microscope slides for dust, smoke, and spores.

these steps to make one in just a few minutes.

1. Use a ¼" paper hole punch to make 4 equally spaced holes below the rim of the cup (Figure G).
2. Cut four 20" lengths of sturdy kite string and tie a string through each hole.
3. Tie the opposite ends of the 4 strings to the eyelet end of a sturdy snap swivel. (The snap end will be attached to the kite.)
4. Use a hobby knife to cut a 1" long slit, parallel to and 1" from the bottom of the cup. Use caution when making the cut.
5. Cut a ¼" slit from each end of the 1" slit straight down toward the bottom of the cup. Fold the rectangular cutout section outward.
6. Repeat Steps 4–5 on the opposite side of the cup.
7. Insert a clean microscope slide through one of the cutout openings so that its ends rest on the 2 cutout flaps.
8. Secure each end of the microscope slide to the plastic flaps with binder clips.
9. Protect the microscope slide from contamination by inserting the completed air sampler inside a clean plastic bag. Keep the 4 tether lines outside the bag, and secure the open end of the bag around the lines with a wire tie.

Flying the Sampler

The air sampler is designed to be flown from a kite rather than a balloon. The kite method assures that a strong current of air will flow through the plastic cup and deposit some of the particles it contains onto the microscope slide.

Sarah flew her sampler from a store-bought delta kite with a 33" keel and a 66" base (fully extended trailing edge). The kite proved surprisingly stable with the air sampler attached. Plan ahead and select a site that's well away from trees, power lines, and buildings. Avoid flying the kite if the wind is gusty or if dust is blowing nearby.

Just prior to releasing the kite, the protective plastic bag should be removed from the sampler. A flight of at least 10 minutes will probably be necessary. Keep the bag handy so you can store the sampler inside it after the flight.

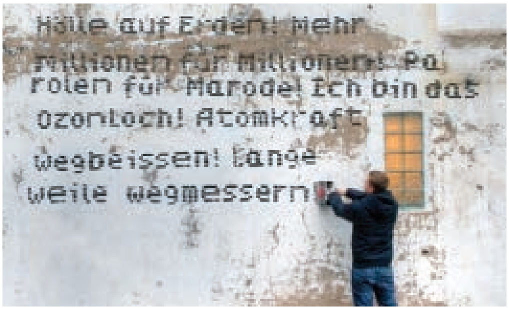
Handle exposed slides by their edges. Store them in a microscope slide box with spacers, or cover the exposed side of the slide with clear adhesive tape.

Forrest M. Mims III (forrestmims.org), an amateur scientist and Rolex Award winner, was named by *Discover* magazine as one of the "50 Best Brains in Science." He edits *The Citizen Scientist* (sas.org/tcs).

The Computer with 1,000 Faces

It has become fashionable in some circles to snicker over the Arduino's widespread popularity. But bagging on this microcontroller is like decrying the number of projects that have breadboards or batteries. Arduino is a part, one that's cheap, versatile, and user-friendly enough to have found its way into cool projects of staggering diversity. Here are a few of our favorites from Make: Online.

LilyPad Arduino Embroidery In this CRAFT video, associate editor Becky Stern shows how easy it is to use the LilyPad sewable Arduino microcontroller to create programmable electronic embroidery and other "soft circuitry." makezine.com/go/lilypad



Digital Paint Roller The txtBOMBER uses seven solenoid-actuated pens and an Arduino brain to create a handheld digital wall printer for making dot-matrix graffiti. makezine.com/go/txtbomber

Arduino Nerf Sentry Our very own John Park used an Arduino to turn a Nerf Vulcan blaster into a sentry gun that fires spongy projectiles of fun upon all intruders who trigger its proximity sensors. makezine.com/go/arduinonerf

Photograph by Michael Thad Carter (top right)

Garduino Featured in MAKE Volume 18 and on Make: Projects (makeprojects.com) the Garduino waters plants when they're thirsty, turns on grow lights, and alerts you if the temperature drops. makezine.com/go/garduino

Wall in Motion Students at the University of South Florida used Arduinos, proximity sensors, LEDs, and servomotors to create Flex, a wall that moves, and animates a light display, in response to people walking past. makezine.com/go/flexwall

Play Zork on a Typewriter Designed for text-based games and interactive storytelling, the Automatypewriter can type on its own and detect what you type. makezine.com/go/automatypewriter

NOTE: There are more great microcontroller projects online on Make: Projects. Have you posted your project yet? Get started here: makeprojects.com/info/start

Introducing the MAKE eBook Library
Did you know that digital back issues of MAKE are available to non-subscribers? All 25 volumes, plus our Halloween special and our 2011 Ultimate Workshop and Tool Guide, are now available in device-friendly PDF format.
In the Maker Shed: makezine.com/go/ebook

Code 72

UNDER THE GUN: Engineers (left to right) Mike Manceor, Bob Washburn, Thom Mathes, and John Schmitz lead the rapid-response team at the U.S. Army's tank and automotive engineering center in Warren, Mich.

Meet the real-life MacGyvers who engineer life-or-death solutions while the clock ticks.

By Lee Zlotoff

Imagine for a moment you're an engineer, and the top brass where you work call to say, "We've got a problem in the field — and you've got 72 hours to solve it." And by "solve it" they don't just mean *think* of an answer, they mean design it, build a prototype, test it, get user feedback, work out the bugs, retest it, and get it into production so it's on its way to the customer — *all in 72 hours.*

Wait, it gets better. Imagine now the company you work for is the U.S. military, where the top brass really are top brass, there's enough red tape to choke a Clydesdale, and your customers are soldiers in the field whose lives actually depend on your solution. Feeling a tad nervous now? Fingers a little sweaty around your calculator? Could that clock on the wall be ticking any louder?

Welcome to the world of TARDEC or, to us non-acronym types, the Tank Automotive Research, Development and Engineering Center, where this type of challenge is a regular occurrence — and is quietly referred to in the jargon of the crew as a "Code 72."

While we're all too familiar with stories of military hardware that takes years if not decades to produce and ends up billions of dollars over budget, the guys at TARDEC rarely enjoy such indulgence. Based in Warren, Mich., TARDEC came about in the late 1950s to marry the design and manufacturing expertise of Detroit's car companies with the military's expanding need for ground vehicles, running the gamut from tanks and trucks to the armored troop transports, Humvees, and remote-controlled robots in use today. And given that our military has the largest fleet of ground vehicles on the planet — around 250,000, give or take a few — that covers a lot of territory in every sense of the word.

So TARDEC routinely has its hands full, its budgets tight, and its projects sharply focused. Especially when it comes to a Code 72. Case in point: the Gunner Restraint System.

Back in September of 2008, then-Secretary of the Army Pete Geren realized that in Iraq and Afghanistan more than five soldiers a month were getting killed or seriously injured by being thrown from open gun turrets when their heavily armored, mine-resistant, ambush-protected vehicles (MRAPs) rolled over on accident or in combat. And this was simply an unacceptable loss.

So the call went down to TARDEC's executive director for product development, Thom Mathes, to find a solution and get it out to the field of battle *within three days*. 72 hours. No extensions, no excuses, no room for failure. Everyone knew this was a Code 72 — or as Thom likes to say, with his Cheshire Cat smile, "They offered us another opportunity to excel."

Having spent 10 years at Chrysler before moving to TARDEC almost 30 years ago, Thom tries to minimize the bureaucracy into what he calls their *Monster Garage* approach: forget titles or departments — just put your heads against the problem and find a way to get it done.

In the case of finding a way to secure those gunners, Thom turned to three of his top engineers: team leader John Schmitz, whose father worked at TARDEC before him and who is never without his Swiss Army knife because he thinks *MacGyver* rocks; Mike Manceor, a mechanical engineer who coincidentally started working at TARDEC the same day as John and has stayed there for more than 23 years because, to him, nothing beats the rush of creating things for a living; and "Magic" Bob Washburn, an electronics engineer who earned his nickname by frequently being called upon to stroke TARDEC's high-end but fidgety computer systems into cooperating, as in: "Why won't this program do what I tell it? Beats me. Call Magic Bob — he'll figure it out."

Project start: John, Mike, and Magic Bob drop what they're doing, tell their wives not to bother waiting up, and huddle in a room with Thom to see what they're up against. The fleet of MRAPs is made by five different manufacturers and has 12 variant models — and although TARDEC has several physical MRAPs to work with, there are no reference drawings or engineering specs.

Whatever restraint system they devise has to work for all models. What's more, it has to be simple enough for any soldier to install. There are only so many mechanics in the war zones, and if vehicles have to be pulled out of service for an installation, it'll create an operational nightmare and most likely never happen.

It also has to be made from parts that already exist in the military stock catalog, because to design and manufacture something from scratch will be way too costly and time-consuming to even think about. And, perhaps most important of all, it has to be both comfortable and convincing enough that the gunners in those MRAPs will actually want to use it. No point in coming up with something your customer won't buy.

So they break it down into the basic components: a plate to attach the system to the floor of the MRAP, a seatbelt retractor they can attach to the plate, and a harness for the gunner that can attach to the retractor. Mike takes on the plate and heads off to start measuring bolt holes in the MRAPs they have in the shop. Magic Bob starts the hunt for the best retractor, and John starts on the harness. In the meantime, Thom will reach out to the wounded warriors who survived these rollovers, at Walter Reed Army Medical Center in D.C., for any wisdom they can offer, and will also talk to crash experts who work at some of the 56 automotive R&D firms within the greater Detroit area. It's *Monster Garage* time, people, and the clock is ticking.

It's been like this at TARDEC since at least October of 2003, when the shop started running seven days a week. It still does, and 80-hour workweeks are not uncommon. When John, Mike, and Magic Bob first came to TARDEC, the designers and prototype makers were like two separate camps — the engineers weren't allowed to even pick up a wrench. You gave the shop guys your drawing and hoped for the best. Not so anymore. Thom and the others knew a good idea could come from even the most novice machinist, and as deadlines became more urgent and machines more sophisticated, there was no way they could be effective with that barrier. So the shop techs are now part of the team and are consulted by the engineers before anything gets made. Even the electronics-focused

Everyone knew this was a Code 72 — or as Thom likes to say, with his Cheshire Cat smile, "They offered us another opportunity to excel."

MONSTER GARAGE: Maj. Preston Hayward (top left) demonstrates the Gunner Restraint System that prevents gunners from being thrown from open turrets in MRAP armored vehicles during combat. TARDEC engineers conceived, designed, built, and tested it, configured mounting plates for 12 variant MRAPs, retested, and sent it on to the assembly line — all in less than 72 hours. Kits arrived on the battlefield within days.

Photography by Brian Kelly (top right and left, bottom right)

Magic Bob can run every machine in the shop, and it's one of the first places the other guys know to look for him.

Suffice it to say, within the first 24 hours of getting the call, the guys and their six shop techs have a restraint system mounted in an MRAP and ready for testing. For that they turn to a "user jury" of combat vets stationed at the same Detroit Arsenal where TARDEC is based. There's no substitute for experience, and even the most imaginative designer can't know what's going on in gunners' minds when they're keyed up and out on a mission.

To accurately assess what happens when an MRAP rolls over, they've got just the thing — a massive rollover cage called an "egress assist trainer," which they had originally designed for testing the Humvee. Despite their wide and squat frames, when Humvees were beefed up with additional armor plates, they were notorious for rolling over. And their doors weigh close to 800lbs — not an easy thing to flip open when the vehicle is lying on its side. (Some MRAP doors weigh upward of a ton — and yeah, TARDEC found a way to fix that problem too, with an electronically assisted door opener.)

To train soldiers how to escape a Humvee in the event of a rollover, TARDEC was tasked with building something that would flip the massive vehicles over, so troops would know exactly how to get out in a hurry when under attack. Initially they were charged with building 70 egress trainers, but these proved so effective and valuable that they eventually made close to 300, almost all of which are now in use overseas.

Adapting the trainer for an MRAP let the team put the restraint-system prototypes through their paces. Needless to say, after eight hours of rapid-rollover testing, there are issues. The seat belt retractor torque is too weak, the harness connection isn't right, and there's no way to get one plate to fit in all 12 variations of the vehicles.

Fine. So they'll make at least two kits with different plates, they'll go back to work on the harness and retractor, and Thom will start on cutting the red tape with the suppliers, so once the bugs are ironed out and it's ready to go, the plates, retractors, and harnesses can be fabricated and assembled as quickly and efficiently as possible. Slam back another cup of coffee — we're not close to being finished and almost half the time is gone.

While all this is happening, the other 200-plus engineers and shop techs at TARDEC are busy with a laundry list of ongoing projects. Like increasing the fuel efficiency of all the ground vehicles. In 2009, 431 million gallons of fuel were consumed in Iraq and Afghanistan. Even a 1% improvement in fuel efficiency would cut 6,400 tanker truck convoy missions, which are prime targets for what Mathes calls "precision guided martyrs." Reduce the need for fuel, reduce the danger — not to mention the considerable savings in cost. So TARDEC has a two-pronged approach to try to *double* the fuel economy for the Humvee alone: one using a hybrid electric design, the other using high-efficiency diesel.

It's now hour 63 in the race, and the guys are ready for the next eight-hour round of testing. This time everything works: the plates all fit into the existing bolt holes, and the pumped-up retractor system now withstands an additional 500lbs of force beyond the 2,500lbs initially required. This is maker hacking at its best.

With less than an hour to go, Thom picks up the phone and calls upstairs — "We're ready." If the funding is approved, they can send designs and prototype kits to the manufacturers within the hour. Half an hour later, the word comes back: the freshly minted Gunner Restraint System, now called the MRAP GRS, is a "Go!"

Within five days, the GRS kits start arriving in the battle zone. Within three months, more than 8,500 kits are deployed and installed in MRAPs.

Maybe when they have some downtime and aren't so wiped out, the guys can tell you about some of their other fast-break wonders. Like the BPMTU (Battery Powered Motorized Traversing Unit) or the OWM (Overhead Wire Mitigation) Kits — meanwhile, check those out on the web.

For now, Thom, John, Mike, and Magic Bob merely exchange a satisfied nod and head home to get some rest. Just another day at the office, right? That is, at least until the next Code 72 comes knocking at the door.

Lee D. Zlotoff is a writer/producer/director among whose numerous credits is creator of *MacGyver*. He is also president of Custom Image Concepts (customimageconcepts.com).

SHAKE IT, DON'T BREAK IT: (Above) The "egress assist trainer" rollover cage was originally designed for training personnel to escape from flipped Humvees. (Below) Overhead Wire Mitigation (OWM) kits let armored vehicles lift wires that are strung across roads out of the way, rather than breaking them.

Things of Science

Miniature monthly science surprises! By Bob Knetzger

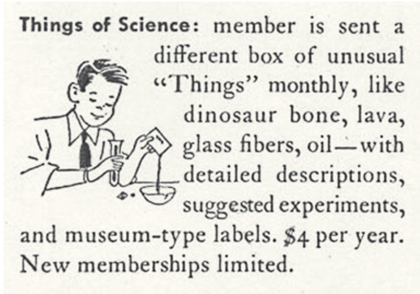
Among my childhood memories of toy gyroscopes, souvenir radiometers, and other scientific kid stuff is a unique series called "Things of Science." I nostalgically recall its clever combination of simple yet effective elements: the anticipation of a magazine subscription combined with the mystery of a "Free Inside" cereal prize, crossed with the fascinating variety of a science fair together with the sense of accomplishment from completing a merit badge — all mailed to my house once a month in a little blue box.

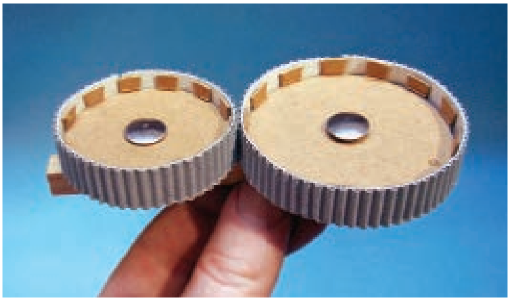
If you were one of the lucky subscribers to Things of Science ("memberships limited"!) you know just what I mean, and if you weren't, here's what you missed.

In 1940, the nonprofit group Science Service created Things of Science (ToS) as an educational outreach program to spread news about science to the public. The idea was to put actual samples of real things into the hands of newspaper editors, teachers, and students. Each month, Science Service produced a kit on a different scientific topic, such as optics or electricity, and included small samples, such as lenses or wires, along with booklets containing information and simple experiments to do at home. These materials were collated into small blue boxes and mailed to subscribers. There were hundreds of different ToS kits produced over nearly 50 years.

The list of scientific topics reads like an encyclopedia: enzymes, seashells, modern electronics, carbon black, slide rules, papermaking, fossils, probability, horology, polarized light, licorice, air pollution, the mollusk, simple machines, and on and on. You never knew what scientific surprise next month's little blue box would bring!

Photography by Bob Knetzger





AN EDUCATION IN A BOX: Things of Science subscribers received a monthly box of "unusual things," such as fossils or high-tech materials (like the "glass fibers" in this 1948 ad). One kit had strips of corrugated cardboard to make gears. Imagine what today's postal inspectors would make of the enzyme kit!

The one common element of all Things of Science kits was the "thing." By placing an actual thing in your hand, otherwise abstract or distant topics became real and touchable. Instead of just reading a book about sound, the ToS Sound kit, Unit No. 243, provided you with a complete (if tiny) sound lab featuring various sound makers: a nylon string, a harmonica, a reed, a siren, a whistle, a plastic membrane, and a booklet with experiments. You'd build a drum, explore the relationship between tension and pitch with a vibrating string, and experience how air pressure affects volume in wind instruments. With each ToS, you'd hear, see, feel, smell, and taste science.

Yes, taste! I remember the Taste kit, Unit No. 265, from November 1962. It came with a packet of table salt, a piece of bubble gum, some strips of impregnated paper, and several mysterious, unmarked pouches of white powder (imagine sending this through the mail today!).

Besides the tried-and-true grade school science-fair experiment of mapping the locations of salt, sour, and bitter taste bud areas on your tongue, this kit came with samples of phenylthiocarbamide (PTC) for testing "taste blindness." I learned that some people cannot taste the bitter PTC and that this inability is hereditary; children of two nontaster parents are also nontasters. According to the accompanying booklet: "It has been found that among American whites, three out of ten are 'blind' to PTC's bitter taste. The proportion is somewhat different for other races. The Chinese are reported to be 94% tasters and so also are the American Indians. This fact is used as evidence that the Indians originally came from Asia." From tasting salt to anthropological migration — that's just a "taste" of the cleverness of ToS kits.

Can you smell the science? In December 1961, ToS subscribers found a special star-covered "Christmas Unit" edition of ToS in their mailboxes: Incense, Unit No. 254. It included different sample cones of incense, including pine, sandalwood, rose, wisteria, "oriental," violet, jasmine, piñon, gum arabic, and, appropriately for the season, gum olibanum, better known as frankincense.

The experiments had kids mixing and sampling gum arabic (used in foods as well as incense) and burning and smelling the cones. This multisensory approach to learning is particularly effective. The sense of smell's olfactory bulb is part of the limbic

A PIECE OF THE SPACE AGE: The Space Materials kit contained samples of the same space-age stuff used to make *Echo I*, NASA's first communications satellite in 1960. When I received this kit, the very same passive radio wave-reflecting satellite balloon was orbiting the Earth, visible from my backyard.

system and the hippocampus, directly associated with memory and learning (and emotions). To the old saying "I hear and I forget, I see and I remember, but I do and I understand," perhaps we can add "I smell and I *feel*" because scents are very emotionally memorable.

One month the "thing" would be a common substance (like table salt), but presented in new ways; the next month, the "thing" would be an "out of this world" material. December 1963's ToS arrived as a thin cardboard folder instead of the usual blue box. Huh? Inside was Space Materials, Unit No. 278, with five shimmery, silver swatches of "Satelloon" skins: high-tech (for the time) laminates of foils and plastics.

These were samples of the same space-age material used to make *Echo I*, the passive radio wave-reflecting satellite balloon that was then orbiting the Earth and was visible from my backyard. I could see the brilliant reflective balloon — 800 miles up, streaking through the night sky at 16,000mph — and hold the very same vacuum-metalized aluminum-coated mylar right in my hand. I was touching the space age!

Of course, every ToS kit wasn't so exciting. Sometimes I'd rip open that month's latest mystery offering and find a clunker, like Cotton, Unit No. 270, from April 1963. It came with some puffs of fibers, a bit of thread, and few swatches of woven cotton cloth. Snore. Even as a third-grader I was already too jaded to do the boring-sounding experiments and learn about carding, spinning, and weaving cotton. No matter, there was always next month. It was that *intermittent* positive reinforcement that kept me coming back.

As the product of a nonprofit organization, the ToS kits had to be low-cost. The modest subscription fee ($5 a year in the 1960s) covered only the postage and printing. Samples and materials were donated by companies. The "science" part of ToS was written by the staff of Science Service, now known as Society for Science & the Public, publishers of *Science News*.

One fun fact: for a while in the 1980s, the ToS kits were produced by a company owned by Andrew Svenson Jr., whose father, under the pen name Jerry West, wrote *Hardy Boys*, *Tom Swift Jr.*, and *Happy Hollisters* books. (The endpaper illustration from the 1958 edition of *Tom Swift Jr.* shows Tom in his orbiting space lair, filled with

You never knew what scientific surprise next month's little blue box would bring!

FIVE DECADES OF THINGS: Things of Science ran from 1940 to 1989, delivering without fail a monthly box of materials with an experiment booklet. One of the companies that produced the kits was owned by a man whose father wrote *Hardy Boys* and *Tom Swift Jr.* books (the latter shown here in his orbiting science lab).

every child's dream of scientific things: rockets, subs, beakers and flasks, a microscope, blueprints, and electronic control panels.)

My family wasn't quite so dedicated to science. My mom threw out all my old ToS kits long ago (along with the comic books and baseball cards), so I turned to eBay to find them again. Sure enough, I got some old friends back and some other vintage ToS boxes that were new to me. I'm having a blast going through them.

Looking at them now, I'm impressed by how simply and cleverly the kits were designed. The writing is direct and conversational. The entire ToS series is so earnest and straightforward — no "edutainment" here.

Could a Things of Science-style program work today? Unlike years ago, many cities now have excellent children's science museums. The kids' section of any bookstore is filled with slick, four-color "book and thing" titles and plastic snap-together kits giving even the least-patient kid fail-proof, fast results. Cable TV shows nature and science programming in high definition.

The competition for kids' limited time and attention with all these well-heeled for-profit entities would be daunting. U.S. manufacturers struggle against cost-cutting pressures with little room in their budgets for sending out free stuff. Society is more litigious now, and in a time of heightened peanut allergy awareness, sending out raw industrial materials with sharp edges, chemical compounds, and pouches of mystery powders could keep an army of liability lawyers busy.

And yet — there's a certain undeniable coolness about holding a real thing in your hand, and feeling, smelling, or tasting it. That's some "thing" that even the internet can't provide.

Photograph (mailing label) by George Moody

Resources

» Society for Science: societyforscience.org
» George Moody's excellent cataloging of ToS Units: ecg.mit.edu/george/tos
» More on the Svenson family, real and fictional: happy-hollisters.com/memoriesofandrew.html

+ Tell us whether you think Things of Science could work today. Go to makezine.com/tos.

Bob Knetzger is an inventor/designer with 30 years' experience making all kinds of toys and other fun stuff.

You can make it!

1+2+3 Urban Survival Button

By Cy Tymony

Emergency survival kits are designed for the wilderness and enclosed in pouches or mint containers. But what about survival situations in the city?

Prepare yourself by making a Sneaky Button Survival Kit, which contains practical items you can use to protect yourself in urban situations.

YOU WILL NEED

- Garment button, hollow
- Magnet, neodymium, small
- Jacket
- Key fob, retractable
- Velcro strip
- Mirror, small, round
- Scissors

Optional:
- Matchbook
- LED, super-bright
- Battery, 3V button cell, CR1025 type or 1.5V watch batteries (2)
- Tape
- Staple

1. Insert the magnet.

Get a hollow garment button from an old coat or a sewing supply store. It's got 2 parts: a button and a separate cap or cover. Open the cover, put the small magnet inside, and press the button back together.

2. Attach to your jacket.

With the scissors, remove a button from your jacket and punch a small hole where it was attached.

Remove the key ring (if supplied) from the key fob. Push the fob's extendable cord through the jacket's small hole and loop it through your button's shank.

3. Attach the mirror.

Cut a small, round piece of velcro and stick it to one side of the key fob. Peel the other side of the velcro backing and press the back of the small mirror to it.

Use It

Pull out the key fob and use the mirror to see behind you while you're walking, or when using an ATM.

To test for counterfeit currency, bring the button near the edge of a folded bill. If the bill is legitimate, it'll move toward the magnet, due to the iron particles in the currency's ink.

Going Further

Store more emergency items inside the button:

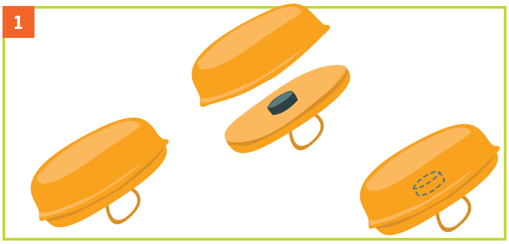
» **Fire:** Tear off some match heads and a small piece of the matchbook striker material. (Wrap the striker in tape so the matches don't strike accidentally.)

» **Compass:** A small, straightened staple can make a handy compass in a pinch. Simply stroke it multiple times with the magnet.

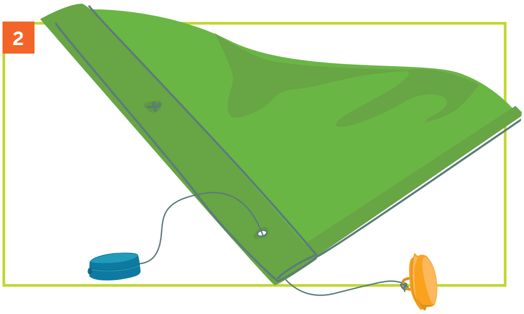
» **Light:** Store a super-bright LED and a small 3V battery (or two 1.5V batteries). Insulate the batteries with tape so they won't drain their charge.

Cy Tymony is the author of the *Sneaky Uses for Everyday Things* book series. He lives in Los Angeles and his website is sneakyuses.com.

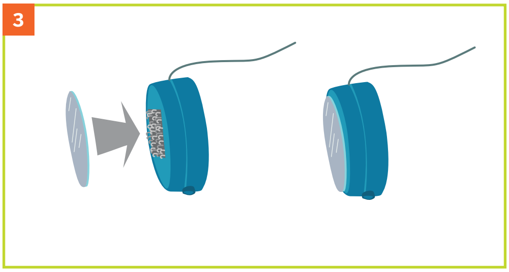
1



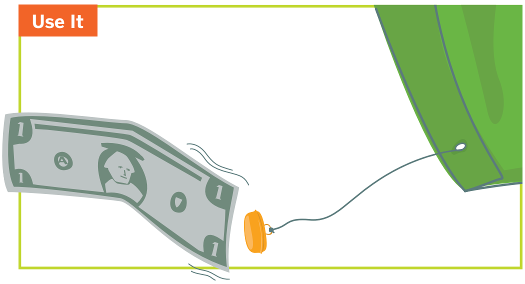
2



3



Use It

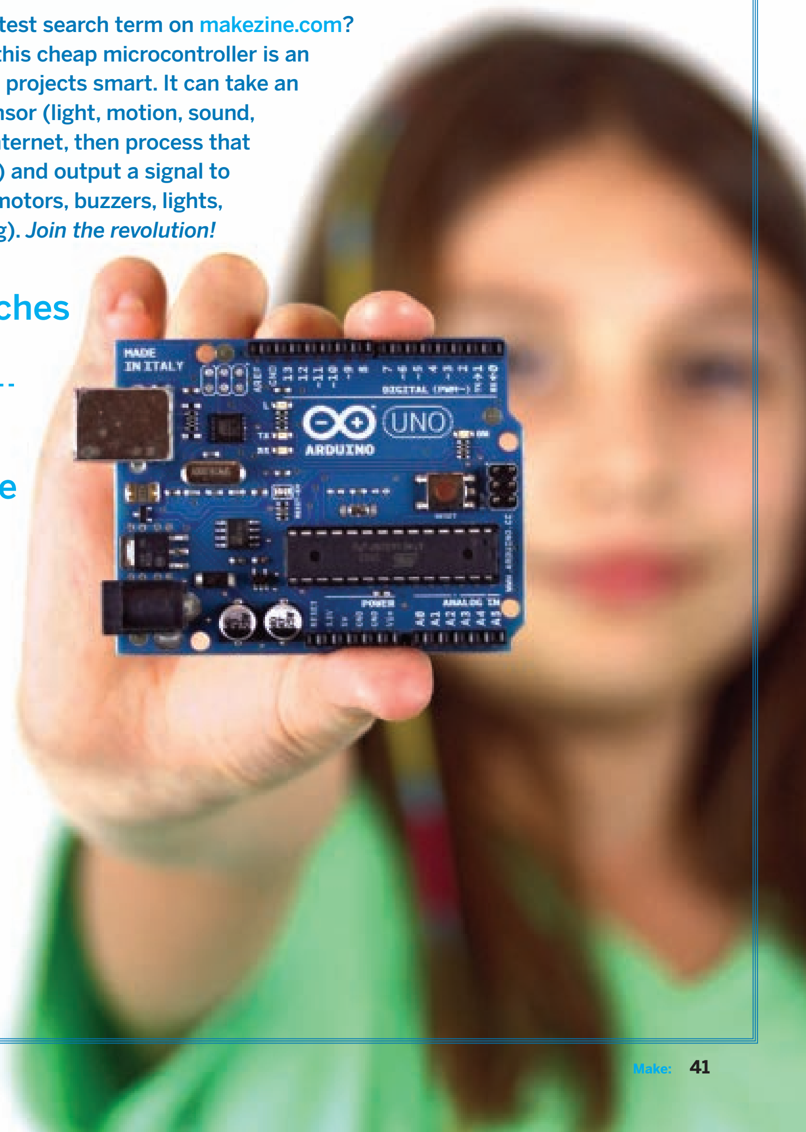


Illustrations by Tim Lillis

JOIN THE ARDUINO REVOLUTION

Everything you need to know to get started with Arduino.

Why is "Arduino" the hottest search term on makezine.com? The answer is simple: this cheap microcontroller is an easy way to make your projects smart. It can take an input signal from any kind of sensor (light, motion, sound, temperature, etc.) or from the internet, then process that data (using a program you write) and output a signal to control electrical components (motors, buzzers, lights, solenoids, computers ... anything). *Join the revolution!*



Control Switches and Motors

Talk to Chips and Hardware

Make Sounds

Light LEDs

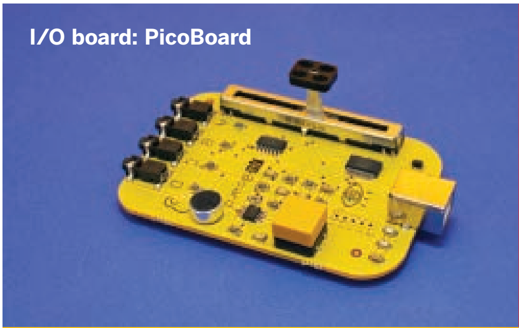
Talk to the Internet

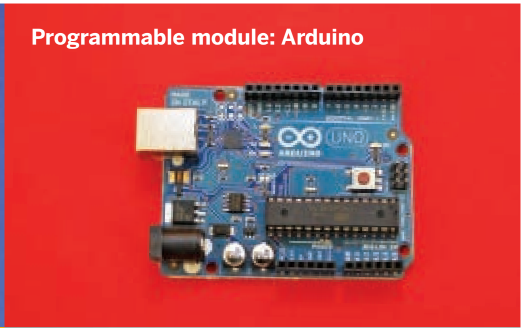
See the anatomy of an Arduino at makezine.com/arduino.

Photograph by Marc de Vinck

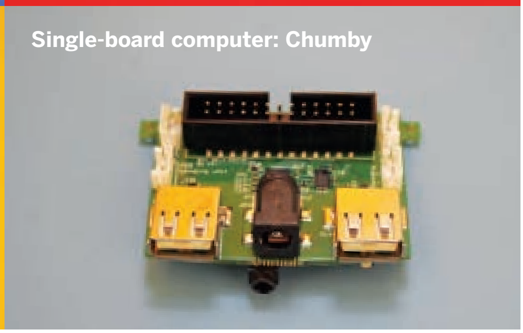
Getting Started with Microcontrollers

Choose the right controller for your project and your skill level. BY TOM IGOE


I/O board: PicoBoard


Programmable module: Arduino


Bare processor: PICAXE


Single-board computer: Chumby

Picking your first microcontroller can be bewildering, because there are so many on the market. Here's what you need to know to get started.

TYPES OF CONTROLLER SYSTEMS

I/O boards (examples: PicoBoard, Phidgets)
You don't program these devices directly, but you can control their input and output (I/O) from a desktop programming environment like Scratch, Processing, or Flash. They don't work when detached from the computer. If you're familiar with desktop programming, this is a good place to start.

Programmable modules (examples: Arduino, BASIC Stamp) These devices can operate independently of your computer once programmed. Like I/O boards, they have a lot of the electronics onboard: communications, power management, and connectors. They require more learning than I/O boards, but let you build smart things that don't have to be attached to your computer. More expensive than bare processors, they make up for it in convenience. For most people, this is the best level at which to start.

Bare processors (examples: Propeller, PICAXE)
These are the guts of the other microcontroller categories. You have to build all the supporting circuitry and connectors yourself, but they offer the lowest price. These generally require an additional piece of hardware to program them as well. They're better for more experienced hobbyists.

Single-board computers Experienced programmers get excited about boards like the Beagle Board or the Chumby. I don't recommend starting

Photography by Ed Troxell

here. They're usually more difficult to interface hardware to than simpler controllers. For experienced programmers, it's too tempting to stay in software rather than experimenting with circuits attached to their board. For software beginners, the challenge of getting an operating system started from the bootloader can be an insurmountable hurdle.

PICKING YOUR FIRST CONTROLLER

Focus on ease of use. You're learning to program and learning to build a computer. Starting simple will keep your enthusiasm high.

Don't be seduced by features. Apollo spacecraft made it to the moon and back with less processing power than most microcontrollers have. Don't be tempted by the fastest, or the one with the most memory or I/O, at the expense of simplicity.

What's the "Getting Started" guide like? The time you're most likely to give up on microcontrollers is in the first hour. The Getting Started guide should take you from zero to blinking an LED or reading a switch in a short time. Read it before you buy. If you don't understand any of it, or it doesn't exist, be wary.

How complicated is the programming environment? The program in which you write your code, called the integrated development environment (IDE), should be easy to understand. Download the IDE and check it out before you buy hardware. If you don't feel you can understand it quickly, keep looking around.

How expandable is the programming environment? I/O boards and very simple languages make getting started easy, but you'll reach their limits quickly. If you're experienced in programming or electronics, you want to feel liberated by a platform's simplicity, not limited by it.

Is the programming environment free? If not, don't bother. There are too many good free environments for a beginner to bother paying for the software alone.

What's the community knowledge base like? You're not just getting hardware, you're getting a community. Every controller has websites and email lists dedicated to its use; check them out, look at the code samples and application notes, read a few discussion threads. Do a few web searches for the microcontroller you're considering. Is there a lot of collected knowledge available in a form you understand? If nobody besides you is using your controller, you'll find it much harder to learn, no matter how great its features are.

How easy is it to add extra components? If there's a particular component you want to work with, check to see if someone's written an example for how to use it with the controller you're considering. Most controllers offer 16 or so I/O connections, which is plenty enough to get started. Tools for expanding your I/O, such as shift registers and multiplexers, are compatible with most controllers.

Is your operating system supported? Most microcontroller manufacturers focus on the Windows operating system. Some have third-party support for Mac OS X and Linux. Learning from friends is common, and being able to have the same user experience on different operating systems is helpful for that.

GET A STARTER KIT

Here are some components you'll need regardless of the platform you choose:

- » Breadboard
- » LEDs
- » Switches
- » 10kΩ resistors
- » 220Ω resistors
- » Photocells
- » Potentiometers
- » Speaker
- » Servomotor
- » Pliers
- » Wire strippers
- » Diagonal cutters
- » Jumper wire

WHAT SHOULD YOU CHOOSE?

For beginners, I recommend starting at the middle level, with programmable modules. You can get your first few projects going for under $100 including the controller and components. You'll save yourself time not having to learn some of the messy details of the support circuitry, hardware programmers, or setting up a development environment.

The first microcontroller you learn will most likely not be the last. It's addictive. So start simple, and grow into the advanced tools as you go.

There are lots of microcontroller environments out there! I looked at 25 for this article, and only scratched the surface. The following table covers 8 that I think are most interesting, ranging from I/O boards to bare microcontrollers.

I/O boards

PicoBoard

$50 picocricket.com/picoboard.html

The PicoBoard is a USB-based hardware accessory to the Scratch programming language. If your kids know Scratch already, it's a great way to introduce them to taking input from the physical world; you just read the inputs via the Scratch board watcher. It's more limited than its cousin, the PicoCricket kit ($250), but a whole lot cheaper.

Runs on: Windows, Mac OS X, Linux*
Requires: USB A-to-B cable
Inputs/outputs: Built-in button, slider, light sensor, sound sensor, 4 stereo jacks with alligator clips
Languages: Scratch*
Getting Started guides: Short, clear, and simple with good examples.
Knowledge base:
- Good examples on site
- Small knowledge base, but large collection of Scratch examples on the web that could be expanded through the use of the PicoBoard

Pros:
- Very simple, fun way to introduce kids to physical interfaces
- Alligator clips — explore the electrical properties of anything!
- Inexpensive for an I/O board

Cons:
- Not programmable
- Limited I/O
- Must be attached to a computer

*** The I/O boards are not programmable; APIs to control them are available in these languages.**

**** Full disclosure: I write a lot of the Arduino tutorial material, so I am biased.**

Phidgets

$78 phidgets.com

Phidgets are specialized hardware blocks that give your personal computer added physical functionality. Besides general sensor I/O, there are RFID modules, motor control modules, and display modules. If you already know how to code, there's an API that'll let you talk to a Phidget as easily as talking to a mouse or keyboard.

Runs on: Windows, Mac OS X, Linux*
Requires: Sensor modules, $7–$100 each
Inputs/outputs: 8 analog in, 1 digital in, 8 digital out (model 1018 interface board)
Languages: Director, C++, C#, Cocoa, Flash, Flex, Java, Max/MSP, Visual Basic, etc.*
Getting Started guides: Guides and code samples for each language are clear. Not many sample projects, but plenty of user-contributed examples on the web.
Knowledge base:
- Lots of code samples in many languages on site
- Plentiful examples on the web at large
- Lots of sensor and special function modules
- No electronics work required

Pros:
- Simple hardware interface for experienced software developers
- Lots of modules that work with no electronics knowledge

Cons:
- Very expensive for what you get. Compared to programmable modules, the average project cost can be 2x–3x more
- Not programmable
- Must be attached to a computer

Programmable modules

Lego Mindstorms

$280 mindstorms.lego.com

Lego Mindstorms kits are Lego's entry into the programming market, done as only they (and MIT) could do. The language is slick and fun, the parts are way cool, and just putting things together gets the geek in you excited. You're somewhat locked into the (expensive) Lego world if you choose this as your main platform, but it's a fun world in which to be locked.

Runs on: Windows, Mac OS X
Requires: Lego Mindstorms NXT 2.0 kit 8547
Inputs/outputs: 3 motor ports, 4 sensor ports, USB port, speaker, buttons, display
Languages: Mindstorms visual language, based on LabVIEW
Getting Started guides: Guides are clear and step-by-step. Build instructions are excellent.
Knowledge base:
- Good tutorials on site and bundled with kit
- Plentiful examples on the web from users

Pros:
- Graphical programming language is great for people who've never programmed and are intimidated by text-based languages
- Kit comes with Lego parts so you have building materials
- No electrical debugging: parts all work and are sealed
- Parts are well-designed and machined; they stay together
- Looks way cool

Cons:
- The leap from here to a text-based language is difficult for some beginners
- Price is high if you just want to do basic microcontroller projects
- You don't really learn electronics; you learn construction and interaction logic

BASIC Stamp 2

$49 parallax.com

The granddaddy of the hobbyist microcontroller market sets the bar for a simple programming language, good starter kits, and excellent documentation. Several models offer more speed, memory, or both. Parallax also sells sensor modules, actuator modules, project kits, robot boards, and good development boards for beginners, along with lesson plans and teaching guides.

Runs on: Windows (third-party applications are available for Mac OS X and Linux)
Requires: USB-to-serial converter, $15
Inputs/outputs: 16 general I/O plus 2 for dedicated serial. No analog input, but you can fake it with RCTime.
Languages: BASIC
Getting Started guides: Manual is clear and thorough, but dated: still uses 9-pin serial cable; doesn't take advantage of what a micro can do.
Knowledge base:
- Many excellent examples on Parallax site
- Countless examples on the web
- Parallax has a ton of application notes and curriculum plans for classrooms

Pros:
- Very large knowledge base
- Simple language
- Multiple models
- Lots of project kits and accessories available

Cons:
- Language is limited: no block IF statements, no passing of parameters to functions
- No analog inputs
- Best getting-started guide, *What's a Microcontroller?* costs $25

Note: Consider using the similar but more powerful BasicX BX-24 ($50, basicx.com), which is pin-compatible with the BASIC Stamp 2.

Bare controllers

Arduino

$30 arduino.cc
An open source platform designed originally for artists, designers, and other technical beginners. Descended from the Processing programming language, it's got an easy-to-use IDE that runs on all 3 major platforms. There are several models for specific applications, and dozens of third-party derivatives. It's expandable through breadboard circuits as with other controllers, or through special-purpose "shield" boards that stack on top of it to add functionality such as motor drivers, Ethernet connections, and many others.

Runs on: Windows, Mac OS X, Linux
Requires: USB A-to-B cable
Inputs/outputs: 20 including 6 PWM, 6 analog in
Languages: Wiring/Arduino, C/C++
Getting Started guides: Clear step-by-step instructions, from download to blinking LED.**
Knowledge base:
- Many simple examples included with download**
- Good reference guide to the commands
- Large knowledge base on Arduino site and elsewhere

Pros:
- Can be run as I/O board, using Firmata firmware
- Very large knowledge base
- Simple language, but expandable using C/C++
- Multiple models, for shields, breadboards, wearables, extra I/O pins
- Many shield modules
- Large number of open source derivative boards

Cons:
- C language constructs (semicolons, brackets, case sensitivity) are confusing for some beginners

Netduino

$35 netduino.com
Newcomer Netduino is the first ARM-based microcontroller module under $50. It follows the Arduino form factor, so it should be able to take Arduino shields. The IDE and language are intimidating for coding beginners, but Secret Labs promises expanded support for other languages and platforms in the near future.

Runs on: Windows (support for Mac OS X and Linux in Mono is promised soon)
Requires: USB A-to-micro-B cable
Inputs/outputs: 20 general I/O, including 6 analog, 4 PWM
Languages: C#
Getting Started guides: Clear and well-written. It took only a few minutes to get an LED blinking.
Knowledge base:
- Small knowledge base so far
- Several user-contributed tutorials
- Videos on the website for blinking an LED and reading a switch are clear and succinct, but tutorials don't explain all the code that shows up when you make a project
- There's an advanced tutorial on event handlers, but no basic tutorial on analog input, PWM, or serial

Pros:
- Compatible with Arduino shield modules
- ARM processor allows for multitasking and event handlers

Cons:
- Installation took me 60 minutes, during which I was able to complete installation for *all* the other tools mentioned in this article on another machine
- Microsoft Visual C# Express 2010 IDE is complex, not good for beginners

PICAXE

$3–$11 rev-ed.co.uk/picaxe
The PICAXE is a BASIC compiler for the PIC microcontroller. Revolution Education makes 3 IDEs: the main Programming Editor (Windows only), Logicator (Windows only), and AXEPad (Windows, Mac OS X, Linux). The first two allow BASIC, or programming by laying out a flowchart; the latter is text-based only, and lacks the debugging features of the others.

Runs on: Windows (IDE is available for Mac OS X and Linux, with fewer features)
Requires: Programming cable, $4, and breadboard adapter, $26
Inputs/outputs: Various: 5 for PICAXE-08, 32 for PICAXE-40X2
Languages: BASIC
Getting Started guides: Well-written, with lots of tutorials. Once you've assembled the microcontroller circuit on a board, getting an LED blinking is very quick.
Knowledge base:
- Many examples on Revolution Education site
- Plentiful examples on the web
- PICAXE BASIC is similar to Parallax BASIC, so code is easy to convert

Pros:
- Inexpensive to get started
- Simple language
- Graphical programming option (Windows only)
- Helpful code wizards (Windows only)
- Multiple hardware versions
- Plenty of examples

Cons:
- Requires more circuit-making ability than programmable modules
- Language is limited: no block IF statements, no passing of parameters to functions

Propeller

$8 parallax.com
The Propeller is Parallax' multi-processor controller. It's very powerful computationally, able to do multiple tasks at once by running 8 processors simultaneously. It's available as modules or bare processors and it can do a lot: video, music playback, and more. It's not great for beginners, but for the experienced programmer it's impressive.

Runs on: Windows (third-party application available for Mac OS X and Linux)
Requires: Prop Plug, $15, plus clock crystal, assorted support components
Inputs/outputs: 32
Languages: Spin
Getting Started guides: There is no Getting Started guide. Reference manual is very thorough, but it's not for the beginner; assumes familiarity with computer architecture and assembly language syntax.
Knowledge base:
- Plentiful examples on Parallax site, and some user-generated code on the web — but not many basic examples

Pros:
- Inexpensive to get started
- Multiple-core processor allows multitasking and other complex functions

Cons:
- Requires more circuit-making ability than programmable modules
- Language is too complicated for beginners — more like assembly language than any of the others

---

Tom Igoe is an associate arts professor at the Interactive Telecommunications Program (ITP) at NYU (itp.nyu.edu), a co-founder of the Arduino microcontroller platform, and the author of *Making Things Talk*.

Vintage VoIP

Convert a classic 1930s telephone into a Skype phone. BY ANDREW LEWIS

Photograph by Andrew Lewis

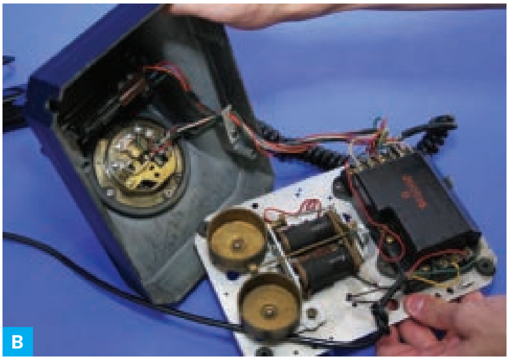
I use Skype, but I've never liked the design of Skype phones. Their cheap, lightweight, plastic cases and shrill alarms can't hold a candle to the sleek lines, comfortable heft, and melodic peal of older telephones. So I converted a 1930s Bakelite desk telephone to receive Skype calls. The new old phone detects when its receiver is off the hook, and its metal bells ring, just like they did in the phone's previous life.

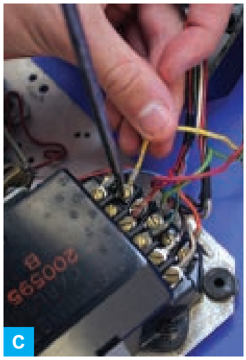
This project has 2 parts: converting the old telephone handset to work with a USB phone (aka Skype phone), and enabling the USB phone to drive the old phone's ringer. These modifications operate independently, so you can perform either one on its own — for example, if you just want to make an old telephone ring for a stage play.

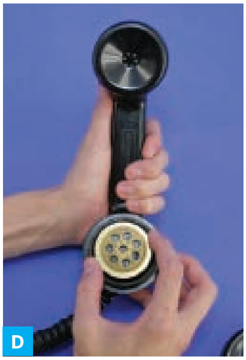
For the handset conversion, my basic approach was to disconnect the microphone and speaker from a USB phone and substitute connections to the vintage phone's own mic and speaker. To bypass the original wiring, I ran twisted-pair wires from a CAT5 cable through the old phone's original handset cord sleeving. I found that the USB phone didn't generate enough audio power to drive the vintage phone's speaker at adequate volume, so I disassembled a pair of USB speakers to extract a small amplifier board to boost the sound. The USB phone and amp fit inside the desk phone's base.

For the ringer conversion, I connected an Arduino between the computer and the base of the vintage phone so that I could interface with its hook and


A


B


C


D


E


F


G

ringer. The Python code uses the Skype4Py library to monitor Skype for incoming or active calls and chat messages. If a call comes in, the Python application tells the Arduino to start the ringer, and then turn it off if the handset is picked up or the call is missed. If you replace the handset while a call is in progress, the call is ended. The Arduino output pins that control the ringer can't supply enough power to drive the ringer coils directly, so they switch 2 transistors that route power from a 12V power supply plugged into the Arduino.

I made a particular effort to alter the phone as little as possible, and all the changes described here can be reversed to return the phone to its original condition. For Materials, see the following page.

1. MODIFY THE OLD HANDSET

The first step is to remove the screws from the bottom of the telephone to expose the electronics (Figures A and B).

Find the connector block at the rear of the telephone, near where the wires enter the base. Follow the 3 wires from the handset to this connector block, and disconnect them by removing the screws that hold them in place (Figure C). This should free the handset from the base of the telephone.

Unscrew the speaking cup (the bit you talk into) from the handset (Figure D). Some old telephones have a catch that prevents the speaking cup from coming loose. Look for a small hole in the side of the cup, and poke a fine screwdriver or paper clip into it. This should release the internal catch and allow the cup to unscrew.

Once the cup is removed, remove the windscreen, if present, and pull out the microphone underneath (Figure E). In some old phones, the microphone connects to the handset wires via a spring contact and metal pin fitted into the body of the handset. In other telephones, the microphone connects via wires screwed onto its back (Figure F). Functionally, the microphone, speaker, and cord connect the same way in both styles of phone.

Old handset cords also vary, but they generally contain 3 wires; the microphone and speaker share a contact, which is the vintage-phone way of letting you hear yourself talk through the earpiece. We'll connect the microphone and speaker separately with their own pairs of wires; circuitry inside the VoIP phone will take care of echoing the mic to the

Photography by Ed Troxell; Andrew Lewis (G)

MATERIALS AND TOOLS

- » **1930s-era desk telephone** Available through eBay, thrift stores, antique stores, and the like. These phones have big components and everything screws together, making them very easy to work with. Look for one with fabric-covered cords.
- » **USB VoIP telephone, aka Skype phone** can be found for $25 or less online. I used a CM-01 VoIP Handset from ebay.co.uk seller shop4usb.
- » **12V power supply** aka power adapter, aka wall wart
- » **Electrical tape**
- » **Cable ties, small**
- » **Screwdriver, flathead**
- » **Pliers**
- » **Wire cutters and strippers**
- » **Multimeter**
- » **Soldering iron with a fine tip**
- » **Solder**
- » **Desoldering braid** or solder sucker
- » **Glue gun and hot glue**

FOR THE HANDSET CONVERSION

- » **USB-powered speakers** with a headphone jack; not USB speakers that also receive signal via USB. $25 or less at drug and discount stores.
- » **Network cable, CAT5e** #278-1764 from RadioShack (radioshack.com), $10
- » **Voltage regulator, 7805 or 78M05** #276-1770 from RadioShack, $2
- » **Small, short screw** like a PC motherboard mounting screw, M3 or 6-32
- » **Machine screw and nut** for mounting voltage regulator. The screw should fit through a 3.5mm hole.
- » **Woven mesh tubing (optional)** to replace original handset cord if it's difficult to extract wires

FOR THE RINGER CONVERSION

- » **Arduino** I used a Duemilanove, but the newer Uno would also work, item #MKSP4 from Maker Shed (makershed.com), $35. You could also use a Nano, #MKGR1 from Maker Shed, $35, but see Final Notes.
- » **Network cable, CAT5e** another one, 2 total needed for both handset and ringer conversions
- » **Transistors, TIP120, TIP110, NTE261 or similar (2)** #276-2068 from RadioShack, $2
- » **Resistors, ¼W or ½W: 1kΩ (2), 10kΩ (1)** #271-1321, #271-1335 from RadioShack, $1 per 5-pack
- » **Project box, small** An Arduino-sized box like #PRT-10088 from SparkFun Electronics (sparkfun.com), $12, would look nattiest, but I just used a black ABS enclosure I had handy.
- » **Diodes, 1N4002 or similar (2)** #276-1653 from RadioShack, $3 per 25-pack
- » **Heat-shrink tubing** #278-1610 from RadioShack, $4

speaker. To obtain these wires, carefully cut away the outer sheath from a CAT5 networking cable and remove 2 twisted wire pairs.

Fabric-covered handset cords are the easiest to modify without altering the phone's appearance. Some have 4 strands braided together, but only 3 contain wires; the fourth is cotton. With these cords, disconnect them from the handset and unbraid them all the way to the base. Pull the original wires out of 2 cords, and snake the twisted pairs through, leaving extra wire at each end. Braid the strands back together (4-strand round-braid instructions are at makezine.com/go/braid) and run them back into the handset (Figure G, previous page).

With one-piece cords, try to pull the wires out of the outer sheath. If that isn't possible, you may need to compromise authenticity by substituting a different wire cover like Tech-Weave. As with the fabric cord, leave the twisted pair long at each end and run the cord back to the handset (Figure H).

Inside the handset, connect one wire pair to the microphone and the other pair to the speaker. To determine which pair of contacts connects up to the speaker, you may be able to follow the wires visually. If it isn't obvious (or if you want to confirm), you can use a multimeter. Look for a resistance of about 36Ω across 2 of the contacts.

With the phone in Figure I, I used pliers to remove a metal pin behind the microphone, severing the connection between mic and speaker. With the mic wire pair, I attached one wire to the back of the microphone and the other to the screw mount for the metal spring that the mic pushes against. One speaker wire screwed into the newly vacant pin-hole behind the mic using a PC motherboard screw, and the other to a nearby screw that I determined ran up to the speaker.

With a hollow handset, you can simply run the twisted pair for the speaker up through the handset and connect it directly (Figure J).

With these 4 connections in place, reassemble the handset.

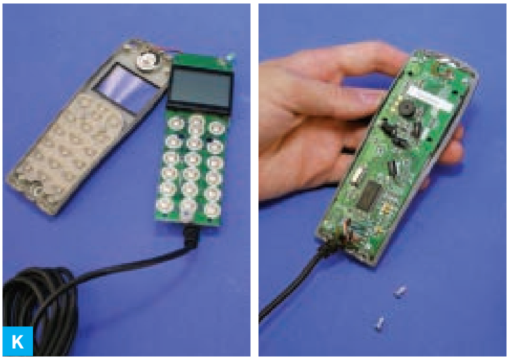
2. CONNECT THE USB PHONE

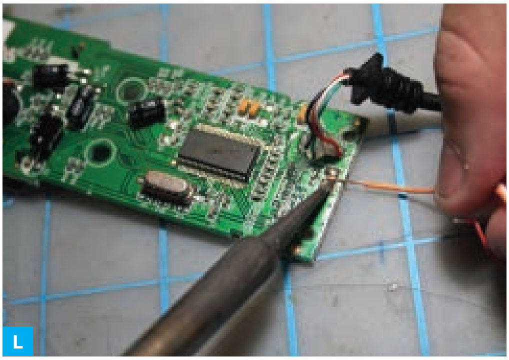
Break open your USB telephone's case to expose its main circuit board, which is basically a sound card (Figure K). Locate and desolder the points on the card where the phone's microphone and speaker connect. Strip the free ends of the wire pair from the vintage handset's microphone, and solder them to the sound card's mic contacts (Figure L).

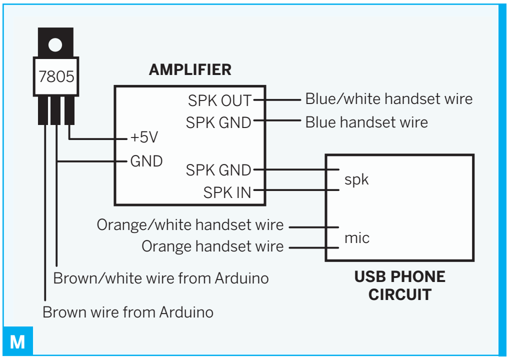

H


I


J


K


L



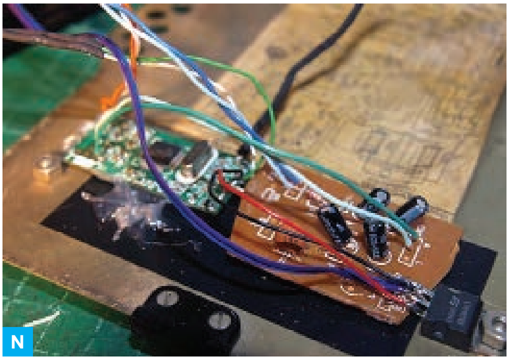
M

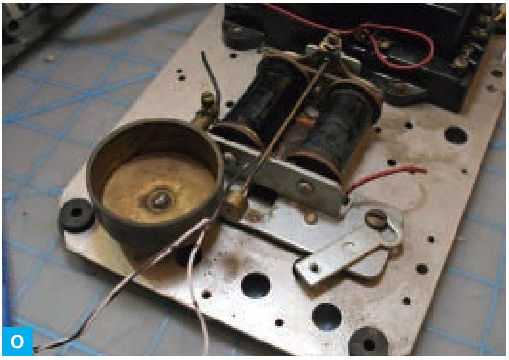
Photography by Andrew Lewis (Figure I)

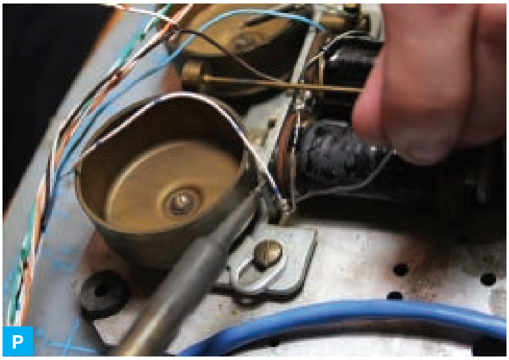
To extract the amplifier board from the USB-powered speakers, I found the screw holes in the case of the right speaker (which has the USB cable), then unscrewed it and broke the glue that held it together. USB speakers contain only one circuit board, which is the amplifier, so it's pretty straightforward. Four wires run to the speakers, 2 go to the USB port, and 3 go to the headphone input cable. All of them should be color-coded, so just note which wires go where and snip them.

Figure M shows how to wire up the amplifier, voltage regulator, and handset speaker wires. The amp's +5V and GND (ground) go to the third and second pins of the voltage regulator, respectively. Amplifier SPK GND (speaker ground) connects to both the speaker ground from the USB handset and one of the vintage handset speaker contacts; SPK IN from the amp's headphone cable (the plug tip for the right channel, possibly also marked Line In or Input) connects to the other USB handset speaker output; and amplifier SPK OUT to the right speaker connects to the other vintage handset speaker wire.

To install the voltage regulator, sound card, and amp, inside the body of the old phone, I hot-glued


N


O


P


Q

the boards and bolted the voltage regulator to the inside of its metal bottom plate, with electrical tape underneath for insulation (Figure N). The plate acts as a heat sink for the 7805 regulator.

Plug the USB phone cable into your computer, and Skype should recognize it as a USB handset within a few seconds. If you have any problems with the installation, double-check your solder connections, remembering that some circuit boards are double-sided, so wires must go completely through the board to make proper contact.

3. CONVERT THE RINGER

In the circuitry in the vintage phone's base, you should see 2 coils that move the clapper to strike the bells. Three wires connect to these coils, one running between the coils, and the other 2 disappearing into the circuitry. To isolate the phone's ringer, disconnect the 2 outer wires, but leave the wire that joins the coils together (Figure O).

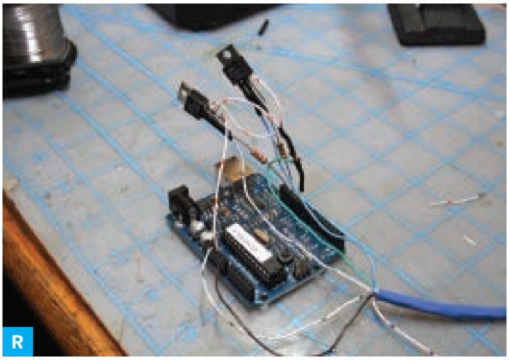
Strip the end of a CAT5 cable to expose about 5" of the twisted pair cable from each end. This cable will connect the Arduino to the internals of the vintage phone. CAT5 cable wires have all different colors, so I'll give wire-by-color instructions.

Solder the brown CAT5 cable wire to either end of the wire connecting the coils. Solder the blue and blue/white wires to the coil's 2 outer tabs, which you just disconnected from the original circuitry.

Solder a 1N4002 diode across the tabs of each coil, with the diode's stripe facing the wire connecting the coils together (Figure P). This diode will absorb any back-current when the coils deactivate.

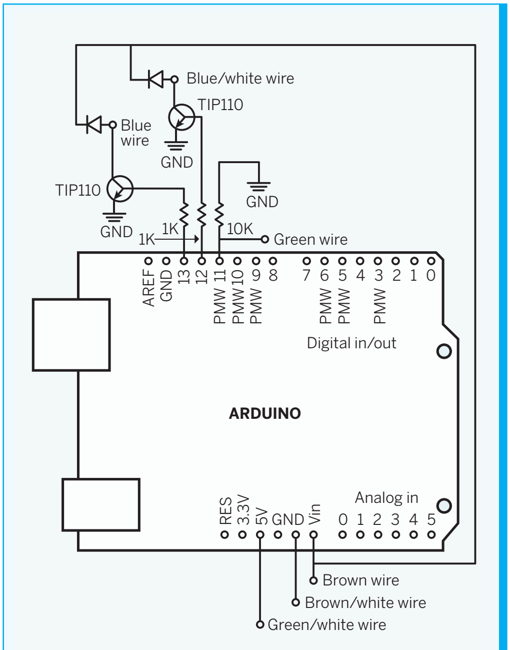
Now you need to find 2 terminals in the phone's circuitry that the hook switches disconnect when they're down (with the handset in the cradle) and connect when they're up. You may be able to figure this out by following wires around or looking at the yellowed diagram pasted on the inside of some old phones. If not, you can search and confirm by probing for continuity with a multimeter while pushing and releasing the buttons. On my phone, one terminal was hidden underneath a bell, which I had to unscrew, and the other was on the terminal block at the back of the phone (Figure Q).

Connect the green and green/white wires from the CAT5 cable to these 2 terminals. They should now work to indicate when the phone is on or off the hook. Neaten the new wires' routing with cable



R

S



Connections marked "GND" go to one of the Arduino GND pins. Wire colors refer to CAT5 cable, going into telephone. Resistors are 1K on pins 12 and 13, and 10K on pin 11.

T

ties so they don't get in the way of the clapper, and reassemble the phone.

At the opposite end of the CAT5 cable, wire the cable and components to the Arduino, following the schematic diagram in Figure T. Solder TIP110 or TIP120 transistors via short wires to the Arduino's ground (brown/white), Vin (brown), and digital pins 12 (blue/white) and 13 (blue) through 1kΩ resistors. Connect Arduino digital pin 11 to the green CAT5 wire and to the ground through a 10kΩ resistor (this prevents the on/off hook status from flipping randomly when the switch is open).

Rather than use any circuit board, I just free-form soldered the transistors to hang off short wires, and enclosed the resistors in heat-shrink tubing (Figure R). Finally, I enclosed the Arduino and components in a small project box, with a USB cable connecting to my computer and the CAT5 cable connecting to the vintage phone (Figure S).

On your computer, install Python (python.org) and the Skype4Py (sourceforge.net/projects/skype4py) and pySerial (pyserial.sourceforge.net) libraries. Download *phone_ring__serial.pde and skype2.py* from makezine.com/25/skypephone, then upload the first to the Arduino and run the second on your computer. Plug a 12V adapter into the Arduino, and start using Skype!

FINAL NOTES

You can use the Arduino software's Serial Monitor for debugging. When the telephone is first powered up, it should send "TELEPHONE READY." It should also send "OH" when the receiver is picked up, and "OK" when it's replaced.

You can also use Serial Monitor to connect to the vintage phone directly without using Skype. Send the command **r** to make the phone start ringing, **s** to stop the ringing, and **n** to trigger a single ring.

For the microcontroller, I used an Arduino Duemilanove I had on hand. But if I were remaking this project in the future, I'd consider using an Arduino Nano, wiring power directly to the board (the Nano lacks a power jack), and fitting all the circuitry inside the telephone.

Andrew Lewis is a keen artificer and computer scientist with interests in 3D scanning, computational theory, algorithmics, and electronics. He is a relentless tinkerer, whose love of science and technology is second only to his love of all things steampunk.

Photography by Andrew Lewis (Figures N, Q, and S)

You can make it!

Drag-n-Drop Arduino Programming

Modkit makes microcontrollers easy for kids and anyone else.

BY ED BAAFI

Photograph (author working on laptop) by Matt Cottam

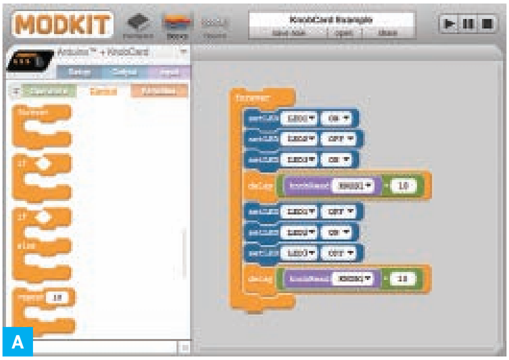
I'm excited to introduce you to Modkit, a drag-and-drop Arduino programming platform. I started Modkit because I wanted to develop physical computing and fabrication tools powerful enough to use in industry and university education, yet intuitive enough to be used by kids, artists, and other non-technical adults, without losing any of the power.

In this article, I'll show you how to build a Modkit "Crimp Card" to program your Arduino and add sensors and actuators (motors, lights, etc.), with no breadboards or soldering required.

THE ORIGINS OF MODKIT

This quest began years ago, around the time MAKE visited MIT's Fab Lab for an article in the premiere edition of MAKE magazine. At the Boston Fab Lab, I learned to program microcontrollers in assembly language, which isn't far from the 1s and 0s of the machine code that computers and microcontrollers can understand. But as powerful as writing in assembly can be, my colleagues and I quickly realized it wasn't necessarily the best tool for everyone, such as kids, artists, and well, almost everybody else.

We run a technology program, in conjunction with the Fab Lab, called Learn 2 Teach, Teach 2 Learn. Every year we hire high school students in Boston to learn, teach, and build projects that solve a need



A

We thought Scratch could be adapted to leverage the tools, community, and momentum around the popular Arduino project, while still retaining its simplicity.

in their lives and communities. Because these students in turn teach what they've learned to elementary and middle school students, we wanted to introduce them to technologies that not only could be used in serious projects but also could be easily transferred to their younger peers.

So when our collaborators from the Lifelong Kindergarten group at the MIT Media Lab released the Scratch programming environment (*see MAKE, Volume 15, page 174, and Volume 21, page 133*), it quickly became our choice for introducing programming, for both projects and teaching.

We thought Scratch could be adapted to leverage the tools, community, and momentum around the popular Arduino project, while still retaining its simplicity.

So we developed Modkit, a free, web-based editor that uses Scratch-inspired programming blocks (Figure A) and allows users to program Arduino (and Arduino-compatible) boards.

What makes Modkit unique is that hardware can be configured visually, and there's a text code view that allows users to view and edit code after it's created in the block view. This allows people to start programming the Arduino with graphical code and grow into writing traditional text code.

MODKIT CRIMP CARDS

We also felt users could benefit from a new way to develop their own circuits. Traditionally, this is done using a prototyping breadboard (see "Primer: Arduino" on page 62). However, for many beginners, a breadboard can be intimidating, as it's not always clear how things are wired internally.

Another alternative is an Arduino "shield," a separate board that connects to the Arduino to provide additional functionality. One limitation of shields is that, once soldered, they fix your components, making it hard to move components around to create new interfaces.

Modkit Crimp Cards aim to solve these issues, behaving like something between a shield and a breadboard. Like a breadboard, there's no soldering required, only simple crimping, so components can be swapped in and out of a project. Like a shield, the components are held firmly in place, which helps take the guesswork out of debugging your wiring.

We manufacture Crimp Cards using die or laser cutters, and sell them as a kit with all the necessary components (visit modk.it/crimpcards to order). But with a few common components and household tools, you can build your own Crimp Card with the instructions that follow.

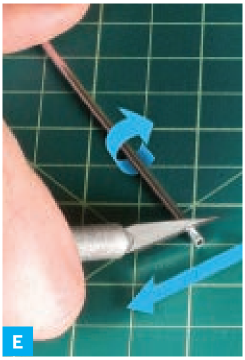
MATERIALS AND TOOLS

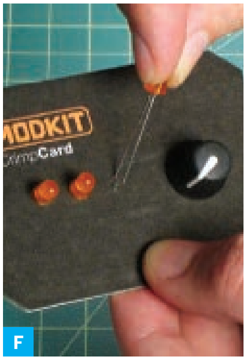
- » **Thin cardboard from a shoe box or cereal box**
- » **Glue for paper**
- » **LEDs (3)**
- » **Potentiometer, 10kΩ**
- » **$\frac{3}{32}$" and $\frac{1}{16}$" aluminum tubes, 1' each**
- » **Wire, solid core, 22 gauge, about 3' Optionally, use colors of red, black, blue, and purple.**
- » **Resistors, 150Ω (3)**
- » **Electrical tape**
- » **Arduino microcontroller board such as the Arduino Duemilanove or the Arduino Uno, part #MKSP4 from the Maker Shed (makershed.com), $35**
- » **USB cable**
- » **Computer running Windows, Linux, or Mac OS**
- » **Printer and paper**
- » **Modkit desktop software The basic version is free at modk.it.**
- » **Thumbtack and pen or similar punch tools**
- » **X-Acto knife**
- » **Scissors**
- » **Sandpaper (optional)**
- » **Wire strippers**

B


C

D


E



F

BUILD A MODKIT CRIMP CARD

1. Prepare the stencil.

Download and print a stencil sheet from makezine.com/25/modkit, and cut it out along the outside dotted line (Figure B). Glue it onto the cardboard, away from the box edges.

2. Punch and cut the card.

Once the glue is dry, push the thumbtack through the LED holes (Figure C) and through the small potentiometer hole, for the tab that prevents the potentiometer from slipping. Push the pen through the large potentiometer hole. Then use the scissors to cut out the card.

3. Clean up the card.

Snip any extra material that may have been pushed through from the stencil side to the front of the card. You can use sandpaper to clean any remaining rough holes on the front.

Cut and glue the front graphics to the front of the card, making sure to line up the potentiometer hole with the outline on the graphics (Figure D).

The front graphics are a bit larger than the card, so use scissors or an X-Acto knife to trim the excess paper to the edge of the cardboard.

4. Cut the crimp tubes.

You'll need six $3/8$" lengths of the $1/16$" aluminum tubing and four $1/4$" lengths of the $3/32$" tubing.

To cut the tubing, place it on a stable cutting surface and press the X-Acto knife down perpendicular to the tube, while you roll the tube back and forth with your free hand (Figure E).

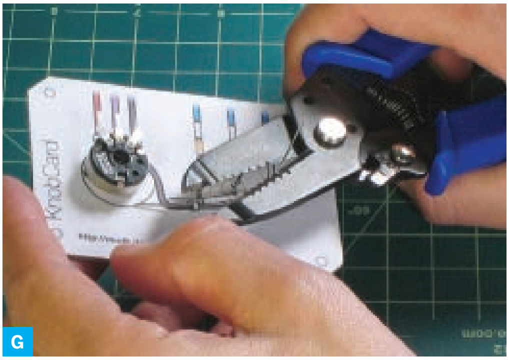
It should only take 4 or 5 rolls to cut the tube, but if it hasn't cut through completely, gently bend it at the score mark to break off the new crimp tube.

5. Place the components.

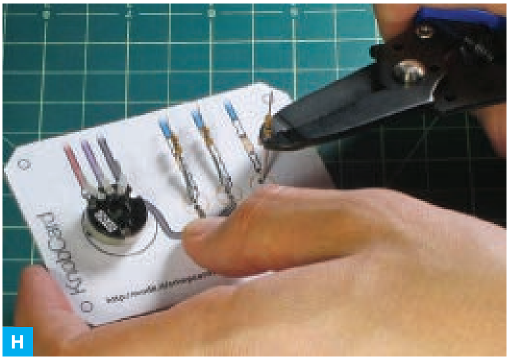
5a. Place the potentiometer through the back of the card (stencil side), add the washer and nut on the front, and screw down the nut until the potentiometer is secure. If you have a knob, place it onto the potentiometer.

Push each LED through its holes, with its negative lead in the bottom hole (Figure F). As a rule, an LED's negative lead is the shorter one, or the one on the side where the LED's plastic dome is flat.

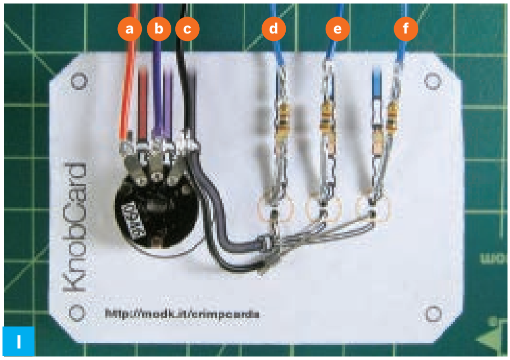
Photography by Collin Reisdorf



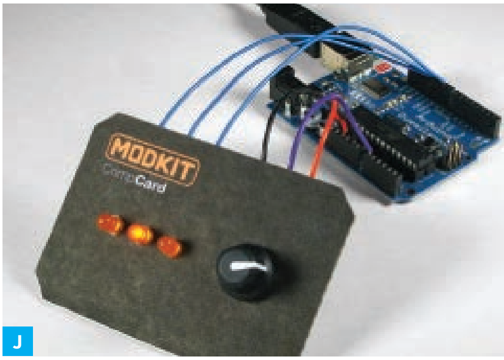
G



H



I



J

5b. Flip over the card, being careful to hold the LEDs so they don't fall out. Bend the LED leads to match those on the stencil. Cut the top (positive) leads to leave about ¼" exposed (Figure G), then trim the bottom (negative) leads to the length shown on the stencil.

5c. Cut the resistor leads, leaving about ¼" on each side.

Cut three 6" wires (shown in blue on the stencil), three 4" wires (black, red, and purple), and one 1½" wire (black), stripping ¼" of insulation off each end. (We color coded the wires on the stencil, but you can just use a single color.)

5d. Add a resistor to each LED by sliding a 1⁄16" crimp tube over the top (positive) LED lead, and then sliding a resistor lead into the top of the tube. Crimp each tube by squeezing it with the grips located at the nose of the wire stripper until it's tight (Figure H).

Now use 1⁄16" crimp tubes to connect the 3 blue 6" wires to the resistors.

Then use the 3⁄32" crimp tubes to connect the remaining wires as shown in Figure I:

» The 3 lower (negative) LED leads to the 1½" black wire
» The 1½" wire and the 4" black wire to the right lead of the potentiometer
» The 4" red and purple wires to the potentiometer leads.

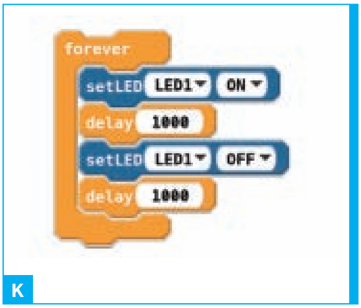
5e. For strain relief, apply tape over all the wires coming from the Crimp Card.

6. Plug the card into the Arduino.

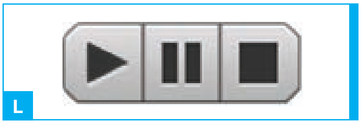
Plug the wires into their corresponding Arduino pins as follows (Figures I and J):

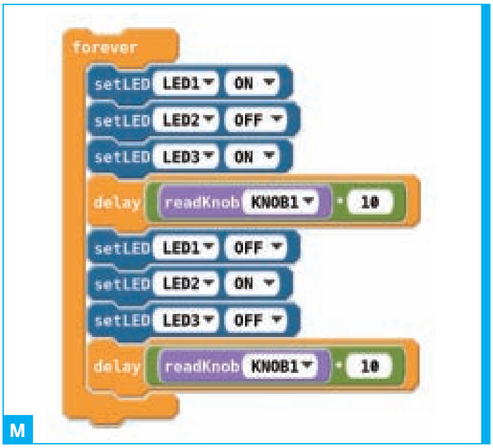
a: 5V
b: Analog In A0
c: GND
d: Digital 9
e: Digital 10
f: Digital 11

That's it! Now you're ready to write your first Modkit program. For this, we'll show you how to make some LEDs blink.



K


L



M

YOUR FIRST MODKIT PROGRAM: MAKE THE LEDS BLINK

1. Install the software.

Modkit is a web-based application and can't directly access your development hardware, so you'll need Modkit's desktop software to program your board. The basic version is available for free. Visit modk.it/download for system requirements and downloads.

2. Load the blink code.

We'll start with a simple blink example. First, plug your Arduino board into your computer with the USB cable. Then go to modk.it/examples/knobcard/blink and click the Load In Editor button.

This simple code example contains code blocks that tell the Arduino to turn on the first LED, wait for a bit, and then turn it off and wait for a bit more (Figure K). The fact that this code is wrapped in a `forever` block means that it will run continuously, blinking on and off until you stop it.

3. Send the code to your Arduino.

To compile and send the code to your Arduino board, press the Play button (Figure L). If this is your first time using the Modkit Editor, you may be prompted to choose your board type and serial port. Once these are set, press Play again, and in a few seconds, the left-hand LED on your Crimp Card should start blinking.

The `delay` block is set in milliseconds, so a delay of `1000` means the LED should be blinking in 1-second intervals. Try changing each delay to 200 milliseconds by typing `200` instead of `1000`, and press Play again. Can you guess how fast the LED will blink?

4. Add code to sense the knob.

You can add code to your program by dragging new code blocks and clicking them into your stack of blocks. Click the Output tab on the left side of the editor. and then drag a `setLED` block below the first `setLED` block in your stack and select `LED2` from the drop-down list.

When dragging blocks, the editor will highlight the area where the blocks will fit. If you make a mistake, you can drag a block to disconnect it. To delete a block or stack, simply drag and drop it back on the left pane, and it will disappear.

To create the program shown in Figure M, drag 3 additional `setLED` blocks, picking the LED numbers from the drop-down list, and then drag multiplication (*) blocks from the Operators tab and drop one into each `delay` block. Finally, drag a `readKnob` block from the Input tab into the left side of each multiplication block, and type `10` into the right side.

This program will blink all 3 LEDs, and the delay will be based on the position of the knob. Since the `readKnob` block returns values from `0` to `100`, we multiply it by 10 so that the delay will vary between 0 and 1,000 milliseconds, or 0 and 1 second. Press Play and try out your new program, twisting the knob to change the blink speed.

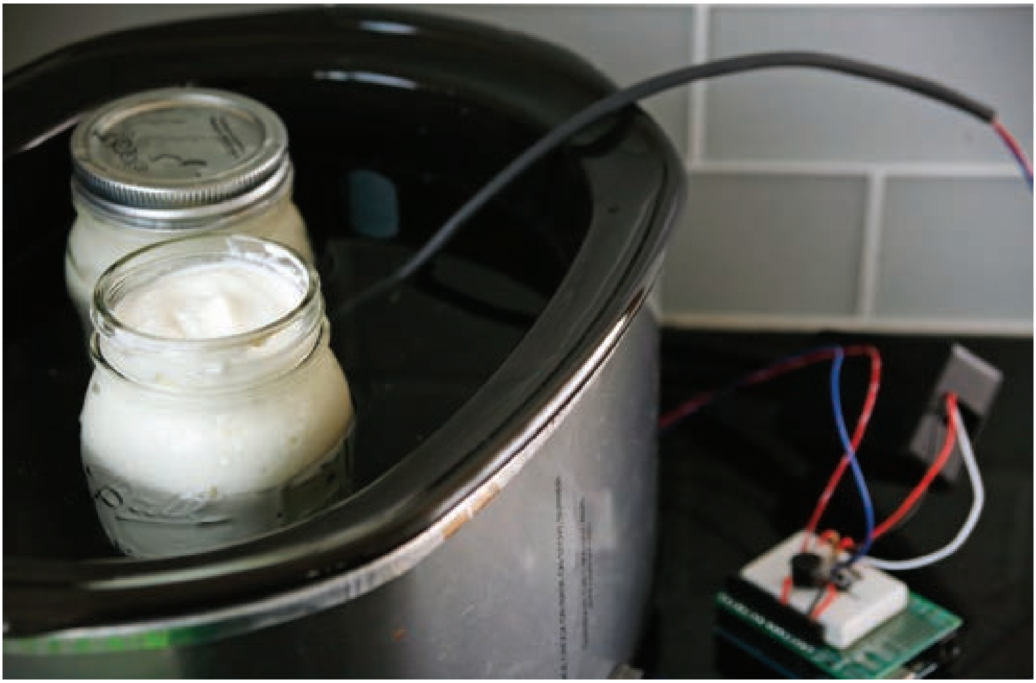
+ There's much more to do with Modkit! To learn more, visit modk.it/examples.

Ed Baafi founded Modkit to build tools that help people design, build, and program interactive projects without needing a background in engineering or computer science. He is a researcher in the High-Low Tech research group at the MIT Media Lab.

Yobot: Precision Fermentation

Build your own smart yogurt maker.

BY CHRIS REILLY



Photograph by Sam Murphy

I'm lazy. My impatience often leads me to botch important steps when I make yogurt. So to get better control over the fermentation process, I made a crockpot thermostat attachment to precisely control the temperature.

You can buy electric yogurt makers, but most of them only incubate; the heating/sterilization step still has to be done on the stovetop. I wanted to experiment with Arduino microcontroller programming and electronic circuit design in Fritzing (an open source circuit layout tool that lets users document and share designs), so why not combine them into something I enjoy doing?

With my old-school yogurt recipe (adapted from wikihow.com/Make-Yogurt), I'd use a stovetop and a candy thermometer to heat the milk to 185°F and cool it to 110°F, then use a warm oven or radiator to ferment it at 100°F. That takes a lot of attention, and more containers than I care to wash later. Even with a commercial yogurt maker, I'd probably have to heat the milk myself, and that's the step I'm most likely to botch.

Don't get me wrong — it's a great recipe as long as you're diligent. But the combination of boring, time-consuming, temperature-sensitive steps puts my diligence to the test; that's why the automation of an Arduino-controlled crockpot yogurt maker makes perfect sense to me.

HOW THE TEMPERATURE CONTROLLER WORKS

I plug my crockpot into a relay, which in turn is connected to the wall socket. I leave the crockpot switched on, so that the relay controls when power is supplied to the crockpot.

The relay is toggled on and off by an Arduino microcontroller, based on readings from a waterproof temperature-sensitive resistor — called a thermistor — placed inside the crockpot. I put my yogurt containers in a water bath inside the crockpot to ensure even heating, then submerge the temperature sensor in the water.

A voltage divider circuit is used to indirectly measure the resistance of the thermistor. In the code that runs on the Arduino, I use the Steinhart-Hart thermistor equation to translate the thermistor's resistance into temperature. This gives a pretty good idea of the temperature inside the crockpot.

In addition to the thermistor's resistance at a given time, the equation needs to be fed 3 coefficients, which can be calculated from predetermined thermistor resistances at different temperatures, shown on the manufacturer's data sheet.

You can use a simple online calculator (makezine.com/go/thermistor) to get your coefficients for a given temperature range, or do the calculations yourself (makezine.com/go/diy_calc). Since we'll be measuring a range between 100°F (38°C) and 185°F (85°C), I used resistance values measured at 86°F (30°C), 140°F (60°C), and 194°F (90°C) to calculate my coefficients.

The code that decides whether the crockpot should be on or off is very simple; it checks the temperature once per second and turns the relay on or off if the temperature is under or over the target temperature, respectively.

If the temperature control were variable, like a dimmer switch, then it might make sense to use a more complex control called a proportional-integral-derivative (PID) algorithm. PID controllers use a sophisticated set of rules to control things like heat or pressure and keep them from overshooting their target values (see "Sous Vide Cooker," page 104). For our purposes, though, a simple approach works fine.

1. PREPARE THE THERMISTOR

Since the temperature readings will be taken in the water bath, we need a way to keep the thermistor from getting wet. I found a great reference online for constructing a waterproof sensor (makezine.com/go/sensor1).

I waterproofed my thermistor by first soldering the thermistor onto 2 long (~3') hookup wires, then wrapping the exposed wires with electrical tape.

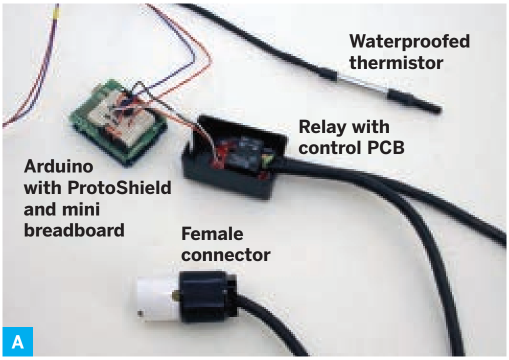
MATERIALS AND TOOLS

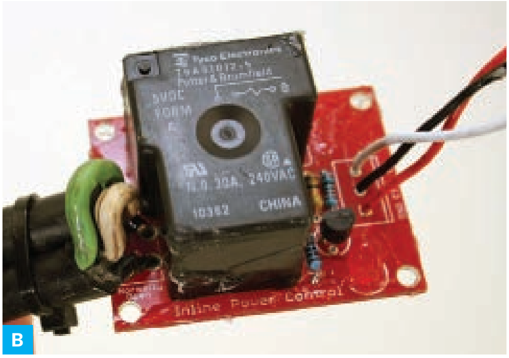
- » **Arduino Duemilanove USB-programmable microcontroller** This project may also work with the new Arduino Uno, but hasn't been tested.
- » **Arduino ProtoShield kit** part #MKAD6 from the Maker Shed (makershed.com), $15, includes:
  - ProtoShield bare PCB
  - Stackable headers, 6-pin (2)
  - Stackable headers, 8-pin (2)
  - LEDs, 5mm, yellow (2)
  - Resistors: 330Ω (2), 10kΩ (1)
  - Momentary push buttons (2)
  - Capacitors, 0.1µF ceramic (2)
- » **Mini breadboard** Maker Shed #MKKN1, goes directly on the ProtoShield. A separate breadboard can also be used.
- » **Relay control PCB** SparkFun #COM-09096
- » **Relay, SPST-NO, sealed, 30A** SparkFun #COM-00101
- » **Buzzer or small speaker** SparkFun #COM-07950
- » **Resistors, ¼W: 1kΩ (2), 10kΩ (3)** #690865 and #691104 from Jameco Electronics (jameco.com)
- » **Switching diode, 100V, 0.15A** Jameco #36038
- » **NPN transistor** Jameco #178597
- » **LED** Jameco #333973
- » **Thermistor, 10kΩ** Jameco #207037
- » **Moisture-seal heat-shrink end cap** #72855K24 from McMaster-Carr (mcmaster.com), $7 for 5
- » **Moisture-seal heat-shrink tubing, 6"–12"** McMaster #74965K64, $7
- » **Aluminum tube, ¼" inside diameter, 3"–6"** McMaster #7237K29, $7; or at hardware stores
- » **120V AC female connector** McMaster #7196K41, $11
- » **Extension cord or power cable, heavy gauge, 3-prong** one you don't mind cutting up
- » **Crockpot** Any slow cooker with an analog switch (manual on/off) should work.
- » **Glass canning jars with lids (2–3)** uniform size is ideal
- » **Plastic container or project enclosure** for relay
- » **Hookup wire, about 6'**
- » **Electrical tape**
- » **Heat gun or hair dryer**
- » **Soldering iron and solder**
- » **Multimeter or connectivity tester**
- » **Epoxy (optional)**

I slid a short length of aluminum tubing over the sensor, then used moisture-resistant heat-shrink tubing and a heat-shrink end cap (on the open end of the tube) to seal both ends (Figure A, top). You can use a heat gun or hair dryer to seal the tubing; you could also use epoxy to make the seals.

2. CONSTRUCT THE RELAY

The Relay Control PCB from SparkFun comes disassembled, so you'll solder the components to the





board: relay, 1K resistors (2), 10K resistor, diode, transistor, and LED. If you follow the component labels silkscreened onto the board, you shouldn't have much trouble.

A few tips:

» The relay will only fit in the correct orientation.

» The line on the diode must match the line on the board.

» The flat side of the transistor lines up with the flat side of the outline labeled 2N3904.

» The unlabeled circular outline is for the LED. Face the flat side of your LED's base toward the flat part of the outline. If your LED lacks a flat side, its shorter lead should face the flat outline.

SparkFun has a great tutorial for constructing a 120V relay outlet specifically for this relay board (sparkfun.com/tutorials/119). If you're uncomfortable working with high-voltage control or just don't feel like soldering as much, you can also check out SparkFun's PowerSwitch Tail, a premade version of this relay: sparkfun.com/products/9842.

CAUTION: Be extremely careful when working with line voltage, as it can actually kill you. These instructions assume you have knowledge of basic electrical safety practices — if that's not the case, seek expert guidance.

Figure A shows the yogurt maker's relay assembly. I deviated slightly from SparkFun's instructions by using a female connector instead of a GFCI outlet.

Use the extension cord's male end with 12"–14" of cord attached. Expose the cord's 3 wires in the middle of its length, then cut the black wire and solder the ends to the relay board's Load 1 and Load 2 connections (Figure B). This is where the line voltage toggles on and off.

Attach the 3 wires at the end of the cord to the female connector: the green/blue wire (ground) to the green screw terminal, and the black and white wires to the other 2 terminals. Use a connectivity tester to make sure the larger slot receptacle is connected to the larger prong on the plug.

The solder points on the relay are basically live exposed wires, so use a project enclosure to keep the relay from accidentally being touched. I used a discarded plastic petroleum jelly container, but you can also use a nice store-bought enclosure like the one shown in Figure C.

Photography by Ed Troxell

3. BUILD THE CIRCUIT

Assemble the circuit, following Figures E and F. The former was drawn in Fritzing (fritzing.org), an open source tool for creating interactive electronics. It offers a visual mode that lets you document circuits the way they look in real life — a great feature for those of us who aren't electrical engineers or are just visual thinkers.

The best part is that the visual mode is linked to a schematic diagram drawn with traditional electronics symbols, which can really help newbies to see the translation between the visual layout (Figure E) and the schematic layout (Figure F).

4. LOAD THE CODE

Download the code from makezine.com/25/yogurt and load it onto your Arduino. The program controls 3 stages of temperature for making yogurt:

Stage 1 heats milk to 185°F for 10 minutes, to sterilize it and denature enzymes in it. (It'll reach 185°F faster if you cover the pot with a towel or blanket.) When Stage 1 ends, the buzzer sounds for 1 minute.

Stage 2 cools the milk to 110°F. During this stage it's useful to remove the cover and any insulation to allow for faster cooling. When the temperature reaches 110°F, the buzzer will signal. The temp will hold at 110°F for 10 minutes, then Stage 2 ends. That's when you'll add yogurt or starter culture and seal the containers.

Stage 3 incubates the yogurt at 100°F for 8 hours then shuts off the heating element. This period can be increased to taste; the longer the fermentation, the more sour the yogurt. At the end of Stage 3, the alarm will sound for 10 minutes, at which point the yogurt containers should be refrigerated.

5. MAKE AUTOMATIC YOGURT

Wash everything! Before starting, gather up all the containers, lids, and stirring utensils to be used. Thoroughly wash them with very hot, soapy water. then rinse them all well with more hot water.

Then follow the recipe at right and let your Arduino do the cooking.

AUTOMATIC YOGURT

INGREDIENTS
» ½ gallon of milk
» 1 package dry milk
» Yogurt or starter culture

Let the starter yogurt come to room temperature before using it; make sure that it contains live cultures and isn't expired.

Step 1: Add the dry milk to the wet milk. Stir thoroughly.

Step 2: Fill the containers and place in a water bath in the crockpot.

Step 3: Loosely cover the jars to prevent condensation from dripping in, and insert the thermistor in the water bath.

Step 4: Plug the crockpot into the relay, and the relay into a power source.

Step 5: You can insulate the crockpot by covering it with towels or blankets to help the heating stage go faster. The milk will heat to 185°F.

Step 6: Plug the Arduino into a computer. The serial monitor will give feedback and instructions.

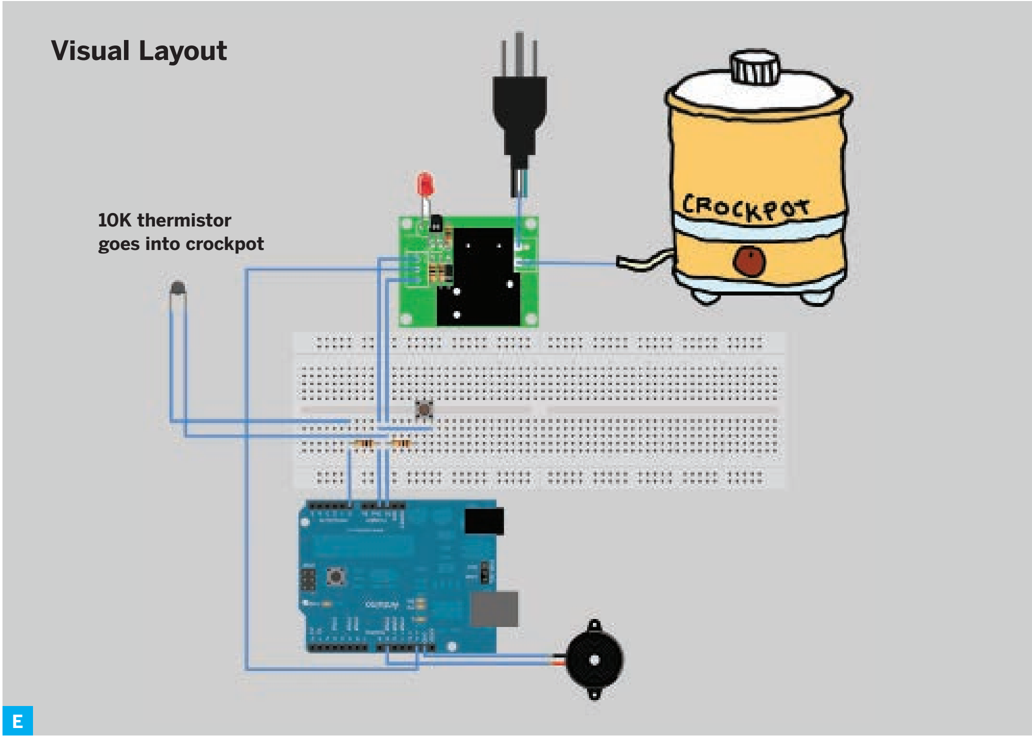
Step 7: After the milk has cooled to 110°F, add about 1 tablespoon of starter or yogurt per container.

Step 8: Seal the lids tightly and incubate for 7 hours (or more) at 100°F.

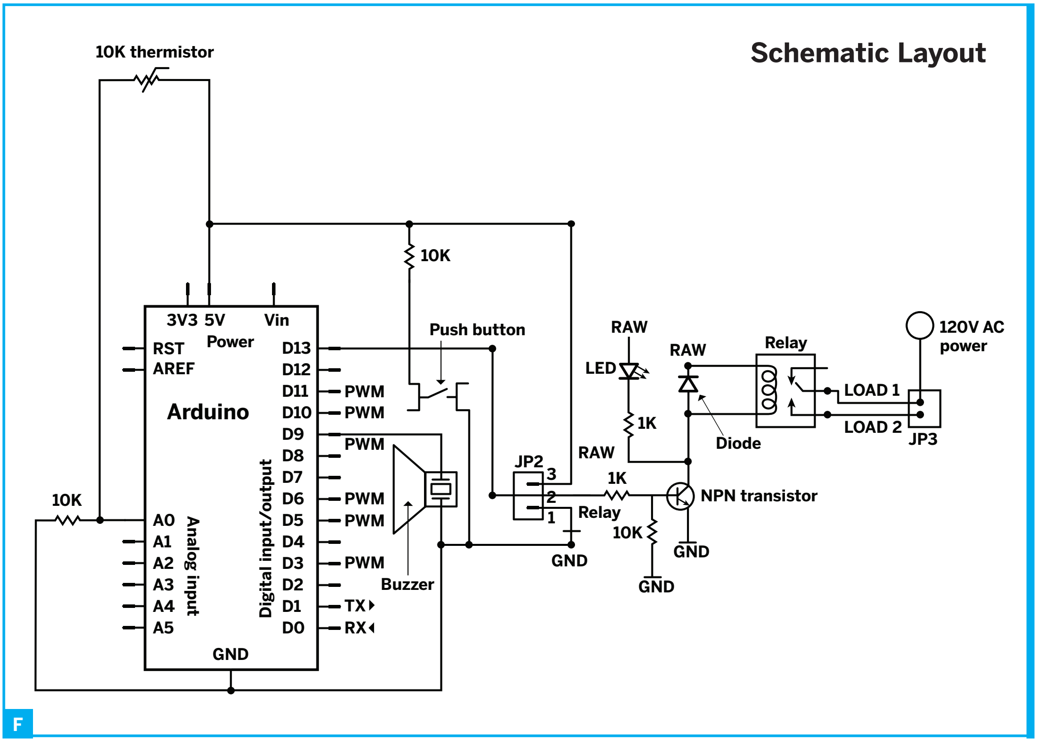
It's yogurt! Refrigerate after opening.

Photograph and illustration by Chris Reilly

Chris Reilly (rainbowlazer.com) is an artist, writer, and teacher who enjoys modded video games, virtual/augmented reality, scripting/programming, and kinetic sculpture. He researches and teaches CNC technology (diylilcnc.org), open source hardware development, and video game production at the School of the Art Institute of Chicago.


Visual Layout
10K thermistor
goes into crockpot
CROCKPOT

E

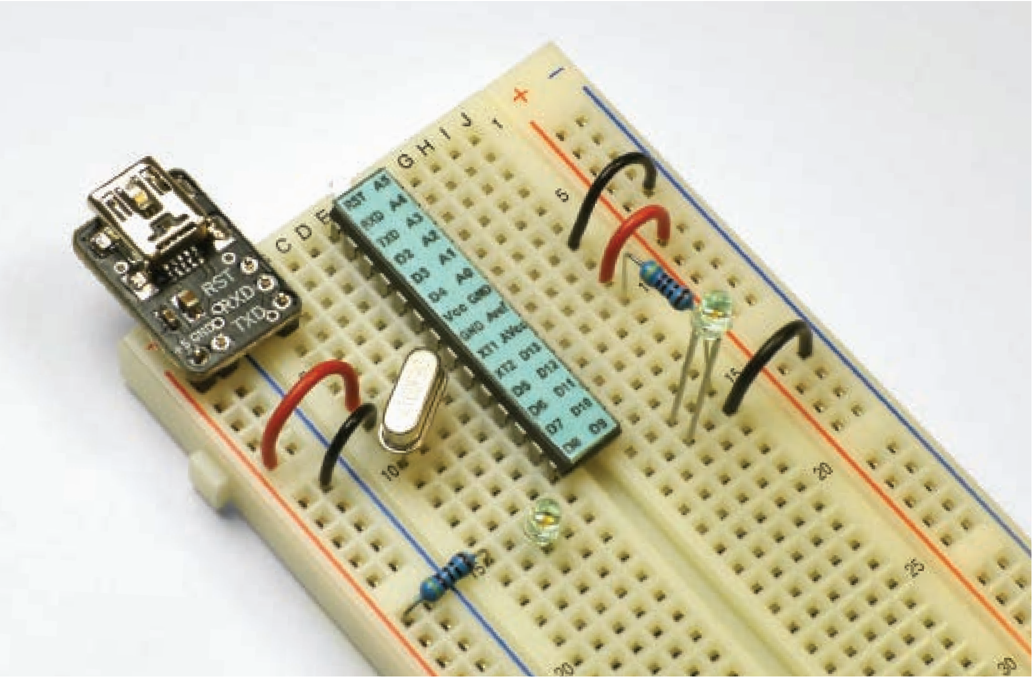

Schematic Layout
10K thermistor
10K
Push button
3V3 5V
Vin
Power
RST
AREF
Arduino
D13
D12
D11
D10
D9
D8
D7
D6
D5
D4
D3
D2
D1
D0
PWM
PWM
PWM
PWM
PWM
PWM
TX
RX
Digital input/output
A0
A1
A2
A3
A4
A5
Analog input
GND
10K
Buzzer
JP2
3
2
1
Relay
GND
RAW
LED
1K
RAW
1K
10K
GND
NPN transistor
GND
RAW
Diode
Relay
LOAD 1
LOAD 2
JP3
120V AC
power

F

Primer: Make and Use an Arduino

Build and program a microcontroller clone with no soldering.

BY DALE WHEAT



Last year, Brett Stephens and Chris Holloway of the Houston hackerspace Tx/Rx Labs (txrxlabs.org) invited me to help with their first workshop: how to build an Arduino from the ground up. It was a huge success — all 20 participants built their own Arduino clone on a solderless breadboard and walked away knowing how to program it.

I told Brett I would steal his idea and take it back to share with the Dallas Personal Robotics Group (dprg.org), of which I'm a member. Brett not only offered to give me his detailed 67-page instruction manual, he even volunteered to drive up to Dallas to help teach the class!

Exactly one month after the class in Houston, I held my first "Breadboard Arduino" class at a DPRG meeting. This article grew out of those experiences.

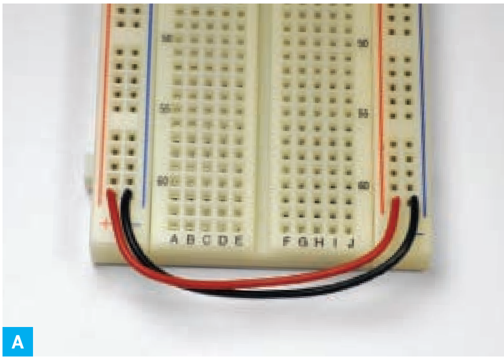
Without even picking up a soldering iron, you'll learn to make your own Arduino and program it to control lights and sounds, and listen to sensors and other hardware.

Photography by Dale Wheat

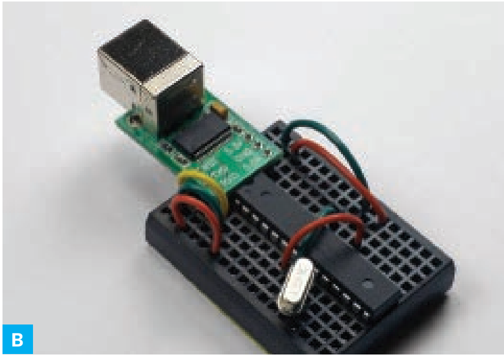
MATERIALS AND TOOLS

» Computer with at least one spare USB port
» USB cable
You can buy all the following materials as the Breadboard Arduino Kit, for $40 at dalewheat.com/store. If you prefer to source your components separately, check out makezine.com/25/arduinoprimer for tips on breadboarding with a different USB-TTL adapter than the one in the kit.
» Atmel ATmega328 microcontroller with Arduino-compatible bootloader installed
» Solderless breadboard
» Jumper wires
» USB-TTL adapter that plugs into a breadboard **Do *not* use a USB to RS-232 converter; it won't work and may damage your components.**
» LEDs, any color (2+)
» Resistors, ¼W, between 150Ω and 1kΩ (2+)
» Quartz crystal, 16MHz

ADDITIONAL COMPONENTS FOR EXPERIMENTS
» Piezoelectric transducer #273-073 from RadioShack (radioshack.com), $2
» Push-button switch, mini SPST (plunger type) #172700 from Jameco Electronics (jameco.com), $1
» Resistor, ¼W, 1MΩ RadioShack #271-1356, $1



A


B

» PART 1: HARDWARE

Link the Power Rails

Solderless breadboards let you electrically connect small wires by simply plugging them into a grid of holes. A small, spring-loaded clip inside each hole secures the wire in place. (If your jumpers fall out easily, try using slightly heavier-gauge wire.)

Rows are usually numbered from the top down, and the columns on either side of the central gutter are often labeled ABCDE and FGHIJ. I'll refer to specific holes, or *tie-points*, with names like A-1 and B-2; but don't confuse the tie-points with Arduino pin numbers, which have no hyphen.

For each row, tie-points A–E connect together electrically, as do F–J. For example, anything plugged into A-1 would connect to points B-1, C-1, D-1, and E-1.

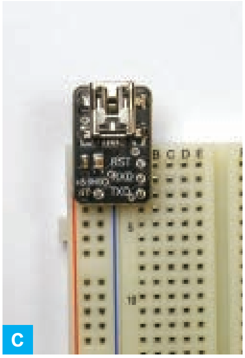
In addition, most breadboards have 2 columns running down each side, with each column connected electrically. These columns serve as the power rails, which put positive and negative power close to all the tie-points on the main grid (they may even be marked with "+" and "-"). The paired rails aren't connected across to the other side, however, so before anything else, wire each left-right rail pair together (Figure A). I use 3½" jumpers, which are long enough to bend over the end of the breadboard so I can still read the column labels.

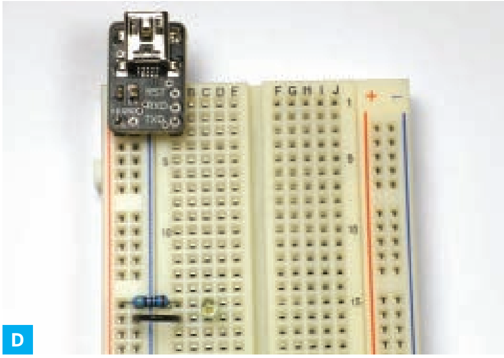
Some smaller breadboards lack these rails, in which case you can use point-to-point wiring to bring power and ground where needed (Figure B).

Install the USB Adapter

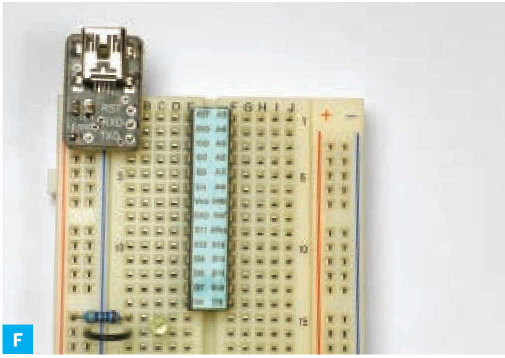
The USB-TTL (transistor-to-transistor logic) adapter connects your computer to your microcontroller to upload programs and enable the two to communicate. The USB cable also brings power to the microcontroller, although the microcontroller will operate standalone with a regulated 5V power supply. The adapter has pins for 5V, ground (GND), transmit data (TXD), receive data (RXD), and reset (RST).

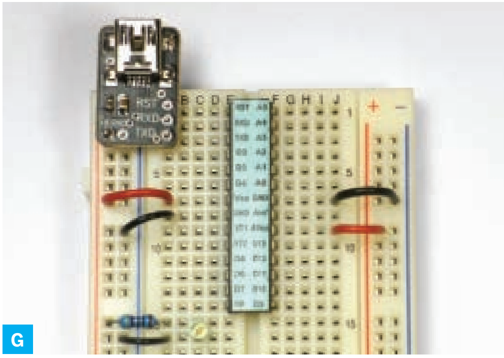
With my kit's adapter, plug it into the upper left corner with the connector end facing out and power and ground pins plugged directly into the breadboard's power rails (Figure C, following page). Other USB–TTL adapter modules will probably work best straddling the central gutter, with power and ground pins jumpered over to the rails.


C


D


E


F


G

TERMINOLOGY

This project only uses 2 power signals: positive and negative. The positive power supply is 5 volts DC, which is also called 5V, +5V, or Vcc (for voltage of the common collector), or simply power.

The negative power is also known as common, ground, or earth, even though it's not necessarily connected to the ground.

An electrical signal that connects to multiple points is called a bus, rail, or common (despite the ambiguity with "common" also meaning negative power).

On the software side, Arduino-land has some unique expressions. Verify means compile (automatically translate from a high-level language to machine code), and programs are called sketches, particularly if they're short.

To confirm that power is flowing, hook up an LED with its positive lead (aka anode, with the longer leg) in tie-point C-15 and its negative lead (aka cathode, marked by a flat spot on the edge of the plastic dome) in tie-point C-16.

Connect a 150Ω–1kΩ resistor between A-15 and the left side power rail, and install a short jumper wire between A-16 and the left ground rail (Figure D).

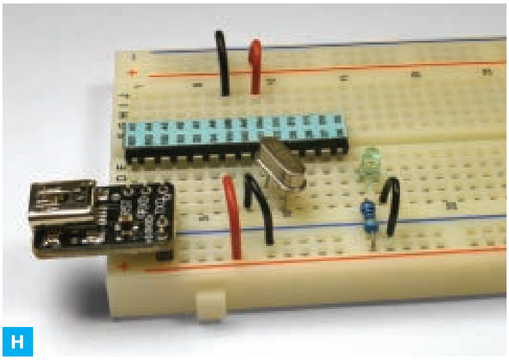
TIP: You can leave the component leads at their original length, but clipping them shorter makes them less likely to snag on things and pull out of the circuit. If you clip them, they should still go all the way down into the hole, to give the spring clip inside something to hold.

Plug a USB cable from your computer into the adapter. The LED should light, and your computer should recognize the installation of a new USB device (confirm if prompted by your operating system). From now on, whenever you change any components on the breadboard, be sure to remove the USB cable first so the LED is dark.

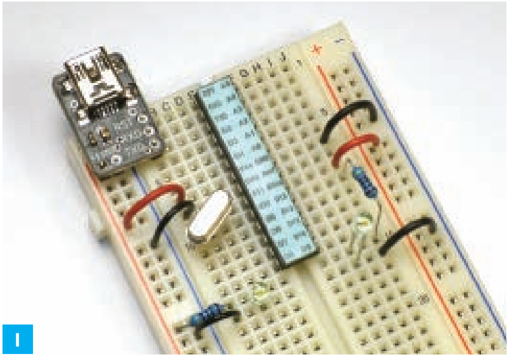
Install the Brain

It's time to plug in the brains of the Arduino: the Atmel ATmega328 microcontroller (Figure E). This chip may or may not have a pin-labeling sticker on it, but it's essential to plug it in oriented correctly.

As with other ICs (integrated circuits, aka chips)



H



I

packaged for soldering to PCBs, the "top" of the package is marked with a dot or notch, and the pins are numbered counterclockwise, starting with pin 1 on the upper left. Pin 1 is often marked with a small dot as well. Note also that the chip's physical pin numbering is unrelated to the Arduino's functional pin numbering; for example, the chip's physical pin 28, at upper right, functions as Arduino analog input pin 5 (A5).

Plug the ATmega into the breadboard, straddling the gutter vertically, with its top pins (1 and 28) plugged into row 1 (Figure F). Fit it in carefully, so all the pins seat correctly in their holes rather than bending.

To connect the chip to power and ground, jumper A-7 to the left-side power rail, A-8 to left-side ground, J-7 to right power, and J-9 to right ground (Figure G).

If you're using my kit's USB-TTL, your microcontroller chip is already connected via rows 1–3 to the correct pins for RST, RXD, and TXD.

Connect the Quartz Crystal

To install the 2-pin quartz crystal, plug it in either direction between C-9 and A-10, to connect the chip's XT1 (pin 9) to XT2 (pin 10) (Figure H). This 16MHz crystal provides the "heartbeat" for the chip, at up to 16 million instructions per second.

Install the Programmable LED

Store-bought Arduinos now come with a built-in LED attached to digital pin 13 (D13), which makes it easy to program test routines that blink the LED. To mimic this feature, install another resistor between J-10 and J-15 and another LED between H-15 (anode) and H-16 (cathode). To complete the circuit, jumper J-16 to the right ground rail (Figure I). This humble, monochrome, monopixel display can convey a lot of useful information.

That's it. You've just built an Arduino from scratch with your bare hands. This proves that you rock.

Plug the USB cable from your computer back into the USB-TTL adapter. Depending on what software was preloaded on your chip, the programmable LED should start blinking. If not, check your wiring and confirm that your chip's preloaded software is supposed to blink pin 13 on startup.

Now it's time to move from the realm of hardware into the murky depths (I mean the soaring heights) of software.

PART 2: SOFTWARE

The Arduino IDE

Download and install the free Arduino IDE (integrated development environment), available for Windows, Mac OS X, and Linux, from arduino.cc/en/Main/Software. This collection of software tools lets you write programs for the ATmega chip, convert them into the machine code that the chip understands, and upload the machine code to run whenever the chip powers up.

Launch the Arduino IDE. At the top of the main window (Figure J, page 67), the *menus* cascade into submenus, with the most frequently used commands also available as buttons on the *toolbar* just underneath. The Verify (or compile) button at far left, which looks like a Play button on a music player, tells the IDE to translate the program that's currently in the *editing area* below the toolbar into machine code.

The Upload button, with a right-pointing arrow and dots, saves the compiled machine code to the microprocessor chip's nonvolatile memory, where it remains even when power is removed from the chip.

Both Verify and Upload commands can take several seconds to complete, and the *status bar* at the very bottom tells you what's happening. The status bar also indicates the line number that the cursor is currently on in the editing area, which comes in handy when referring to lines of code or debugging.

Before you use the IDE, you need to configure it for your specific hardware. Your Arduino clone is functionally identical to the Arduino Duemilanove model, so click the menu item Tools → Board → Arduino Duemilanove or Nano w/ATmega328.

Then you need to tell the IDE which serial port to use when communicating with your Arduino hardware. Click on Tools → Serial Port ...

If there's only one selection available, use that, and if there are more than one, use the highest numbered COM port listed. When you first connected the USB-TTL adapter to your computer, it should have given the USB-TTL its own virtual serial port.

Software Experiments

Ready to dive into some Arduino programming? What follows is a series of experiments that starts out simple and progresses toward the moderately complex. Sadly, we'll only be able to scratch the surface of Arduino programming, but I hope this short introduction will inspire you to continue.

1. BLINK THE LED

No, this is not a story about an LED named Blink. We are going to make the LED blink. This sketch comes ready-packaged with the IDE as an example sketch. Open it up by selecting the File → Examples → 1.Basics → Blink menu item.

Every text color in the editing area means something different (Figure J). Program comments appear in gray — they help explain the code to humans without affecting its execution. You can add a short comment at the end of any line of code after // or include block comments of any length sandwiched between /* and */.

Orange represents a *keyword* or a word with a special meaning to the compiler. The first block of text that starts with an orange `void` defines the `setup` function. Like in math, *functions* in the Arduino programming language can take some or no *arguments*, do some magic, and possibly return a *value*. Function arguments are enclosed in parentheses, as in algebra, and the word `void` here declares that the function returns no value.

Unlike with mainstream C or C++ programming, Arduino's `setup` function executes once at startup to perform initialization, then the `loop` function repeats forever, ad infinitum. There is no `main` function.

In this sketch the `setup` function calls the function `pinMode`, which sets digital pins on the ATmega328 chip to function as either input or output pins. Here it sets the pin connected to our LED, pin 13, as an output (the pin modes all default to input).

The `loop` function calls `digitalWrite` to set pin 13 `HIGH`, which turns the LED on. (The blue words in ALL CAPS are generally *constants*, values that are defined elsewhere and not allowed to change.) Then `delay` waits about a second (its argument `1000` specifies milliseconds), `digitalWrite` sets pin 13 `LOW` (off), and delay waits another second. The `loop` function repeats, so the LED blinks on and off until the Arduino is powered off.

Let's give it a try. Compile the sketch by clicking the Verify button, wait until the status reads "Done compiling" and gives the program size, then click Upload to blast all those bytes into the chip. After you see the status "Done uploading," you should see your LED blinking. If not, check your board and serial port settings. Otherwise, congratulations! Unplug the USB cable, wait a while, and plug it back in; the blinking will start again, demonstrating that the program is stored on the chip in nonvolatile memory.

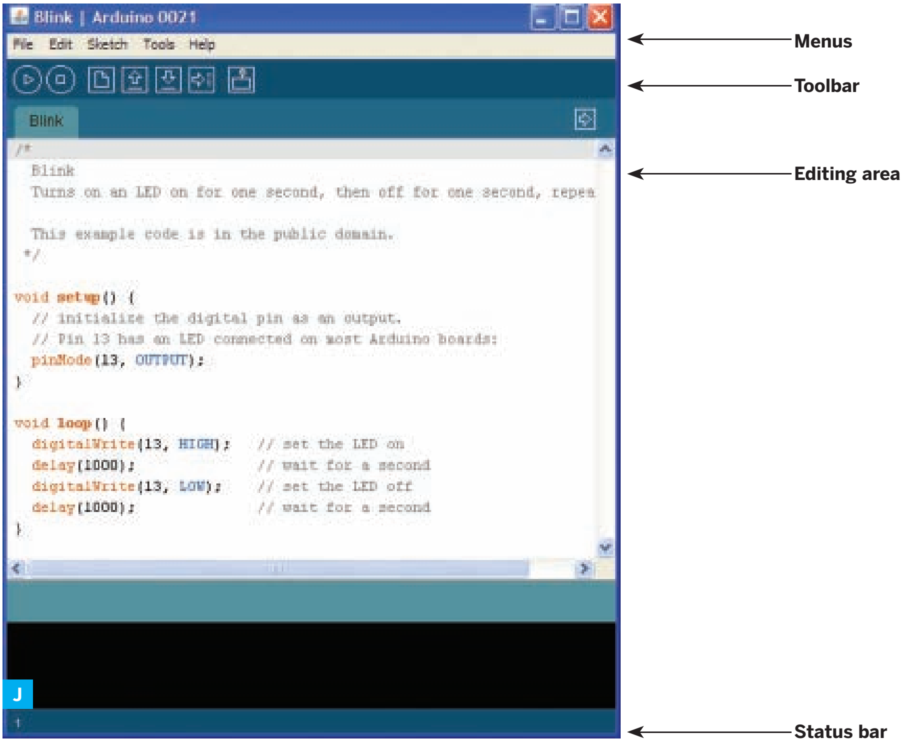
2. BLINK AT A DIFFERENT RATE

Let's expand on the *Blink* sketch — but to keep the original example intact, we'll save our changed version under a new name. With the *Blink* sketch in the editing area, move the blinking cursor down to the `loop` function and change the arguments to both `delay` functions from `1000` to `250`. Once you've made the 2 changes, do the Arduino Two-Step: click Verify and Upload. You should see an LED blinking at a noticeably different rate. Is it faster or slower?

Try again, changing the values passed to `delay` to any number between 1 and 4,294,967,295 (the biggest number that can be expressed with 32 bits, which will program a delay of almost 50 days).

Save your work by selecting the menu item File → Save As and enter a new sketch name over the suggested "Blink." The Arduino IDE will automatically change any spaces in the name to underscores and create a new folder to hold your sketch and its related files. These automatic actions maintain the IDE's multiplatform compatibility.

Notice that compiling and uploading the modified sketch didn't automatically save your work. This is something you need to do for yourself, and often.



3. FADE

Open the example sketch under File → Examples → 1.Basics → Fade, then compile and upload. This sketch uses pin D9 instead of D13, so you can get it to work by moving one leg of the program LED resistor from row 10 down to row 14 (Figure K). But to be safe, unplug the USB cable before moving the resistor or making any other hardware changes, then plug it back in. And instead of borrowing the D13 LED/resistor pair (and having to unborrow it later), you can give D9 its own.

The LED's brightness will appear to fade up and down, but it's actually cycling on and off at 490Hz, faster than the eye can see, using *pulse-width modulation* (PWM) to simulate an analog output. This sketch uses D9 rather than D13 because D9 (along with D3, D5, D6, D10, and D11) is one of the pins that can support PWM. Plus, in the code, the `analogWrite` function can pass it a value between 0 and 255, where 0 is completely off and 255 is completely on.

Line 11 of the *Fade* sketch (`int fadeAmount = 5;`) declares an integer variable `fadeAmount` as equal to 5. Mess with `fadeAmount` by changing that `5` to a `1`, then compile and upload. This changes the rate of the fading effect. Is it slower or faster? As before, save your work under a new name to avoid over-writing the example sketch.

Interestingly, the only values that will work properly for `fadeAmount` are 1, 3, 5, 15, 17, 51, and 85, which are the multiples of the prime factors of 255. You can either trust me on this or try it yourself. Go ahead — try 2 or 4 and see what happens. I dare you.

4. OLD SCHOOL "HELLO, WORLD"

The Arduino IDE includes a *Serial Monitor* window that lets you communicate with your Arduino via its virtual serial port. You open it with the rightmost toolbar icon, which depicts a screen hanging off a line.

Older versions of the Arduino IDE included the sample program *HelloWorld* (no spaces) that would output "Hello, world" to the serial monitor. But that code fell through the cracks, so we have to write our own.

Before we do that, click the Serial Monitor icon, and change the setting in the lower right-hand corner, which most likely says "9600 baud," to "300 baud." Then close the window and open up the *BareMinimum* example sketch, under File → Examples → 1.Basics → BareMinimum, which contains handy empty shells for both the **setup** and **loop** functions.

In any Arduino function definition, the arguments are listed up top between regular parentheses, and the body of the function, where all the work gets done, is enclosed by curly braces, { and }.

For our **setup** function, we want to add the following 2 lines after the opening brace (line 1) and before the closing brace (line 4):

```
Serial.begin(300);
Serial.println("Hello, world!");
```

All punctuation and capitalization must be exactly as shown here or the compiler may become displeased. The first line opens up the serial port connection at 300 baud, and the second line calls the **println** method to output the string **Hello, world!** over the connection.

Compile (ahem, I mean Verify) and upload the sketch, then open the Serial Monitor from the toolbar and see if it doesn't contain the classic greeting sent from the microcontroller. Save your "Hello, world" sketch under a new name (don't overwrite *BareMinimum*).

If you see garbage characters instead, double-check that the baud rate selection matches the argument passed to **Serial.begin**. Both should be 300.

This sketch has nothing in its **loop** function, so after it runs **setup** to say hello, it doesn't do anything. But if you close the Serial Monitor and open it again, our message prints out again.

This happens because the Arduino IDE sends a **reset** pulse to the chip whenever it opens or closes the Serial Monitor, which restarts the sketch stored on the chip. This makes the Serial Monitor a handy way to reset your Arduino program remotely.

5. COUNTING

Open the "Hello, world" sketch. Now we'll add some code to its **loop** function to print out a number, increment by 1, and repeat. The variable **x** will hold this number, but we don't need to worry about **x** in **setup**. Instead, we can declare it after **setup** but before **loop**, which makes its scope global (visible to other functions).

Add this line immediately below **setup**, to *initialize* the value of **x** as **0**:

```
int x = 0; // our counting variable
```

Then add the following 2 lines to **loop**:

```
Serial.println(x);
x++;
```

Serial.println here outputs the value of **x** just as it output **Hello, world**, and the **x++;** is shorthand for **x = x + 1;**

Save under a new name, compile, upload, and open the Serial Monitor. At the relatively low baud rate of 300 bits per second (about 30 characters per second), you should be able to keep up with your Arduino's output.

See what happens when you increase the baud rate, changing it in the Serial Monitor before you run a new sketch. What's the fastest you can get it to go?

6. SOUND

We've controlled LEDs that blink and fade light, now let's add a small piezoelectric transducer (speaker) to output sounds. Plug the piezo element into the breadboard below the power LED, connect a jumper between pin D8 and one speaker pin, and another between the other speaker pin and ground (Figure L).

Load the sketch File → Examples → 2.Digital → toneMelody into the editor. To fix a bug, scroll down to line number 35, as indicated in the status bar, and add a line above it so that they both read as follows:

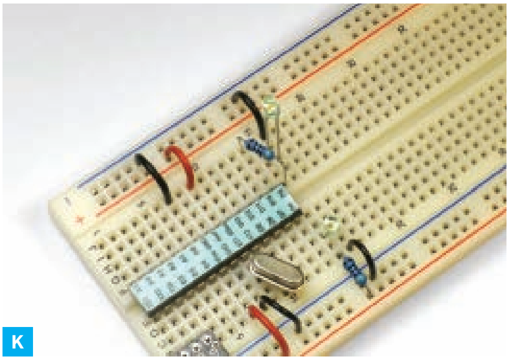
```
noTone(8); // add this to fix the bug
tone(8, melody[thisNote],noteDuration);
```

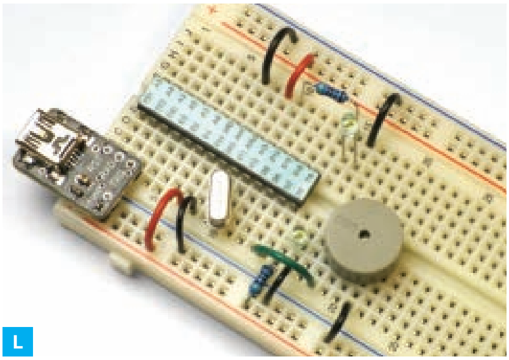
Compile and upload this sketch to your Arduino, and it should treat you to a short but familiar musical phrase. To hear the little ditty again, you can open and close the Serial Monitor.

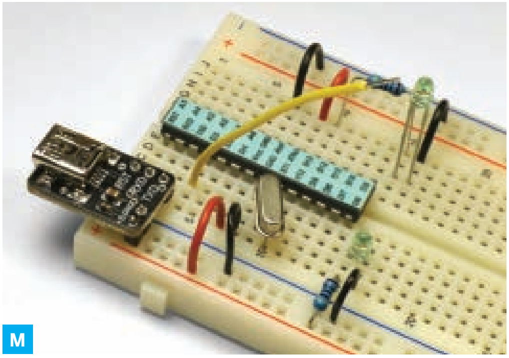
7. INTERACTIVE FADING

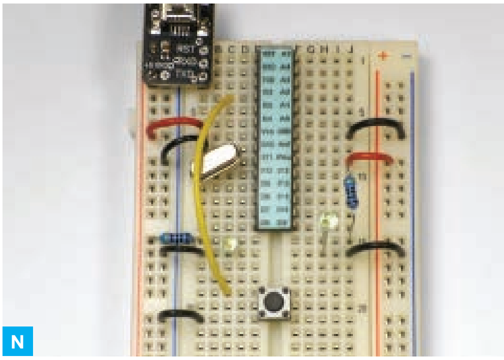
Now let's experiment with some *input devices*. For starters, the Serial Monitor is a built-in input device that can send commands or data from your computer to the Arduino.

Open, compile, and upload File → Examples → 4.Communication → PhysicalPixel. Open the Serial Monitor and set the baud rate to 9600. Make sure


K


L


M


N

you have at least one LED still hooked up to D13.

In the Serial Monitor window, type in a capital `H` and click the Send button or press Enter on your keyboard. The LED should come on and stay on. Try a capital `L` and the LED should go off. `H` and `L` correspond to `HIGH` and `LOW`, respectively, and lowercase letters won't work.

In the code, scroll down to program line 42 and look for the line that reads:

```
if (incomingByte == 'H') {
```

This is a *conditional statement* that checks for the `H` character. In the Arduino programming language, as in its ancestors C and C++, == is an *equality operator* that evaluates to either true or false, whereas = is an *assignment operator* that sets values.

Also note that single quotes delineate the `H` character in the code; double quotes are reserved for *strings of characters*, such as `Hello, world!`

You can change the `H` and the `L` (line 46) to just about any other single characters. Try changing them to `0` and `1` and see what happens.

8. BUTTON

Now let's add an input device: a jumper wire hooked up to act like a button. Unplug the USB cable, install one end of a 2"–3" wire in hole C-4, then replug the USB (Figure M).

Compile and upload File → Examples → 1.Basics → DigitalReadSerial. Open the Serial Monitor and set the baud rate to 9600.

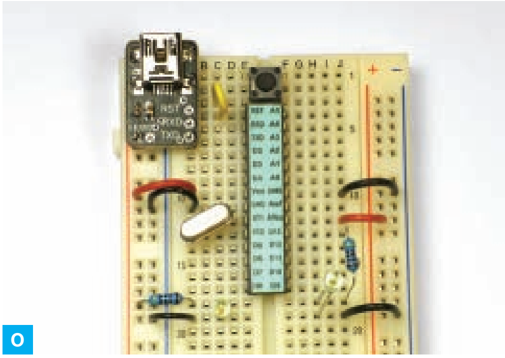
You should see a rapidly scrolling list of ones or zeros or both. Plug the free end of the jumper into the ground rail and you'll see only zeros scrolling past. Move the jumper end to the power rail and you'll see only ones.

When the end of the jumper wire is left unconnected, the voltage present on the input pin of the microcontroller is subject to change by stray ambient electromagnetic radiation. Connecting it to a stable voltage level puts it in a steady, known state.

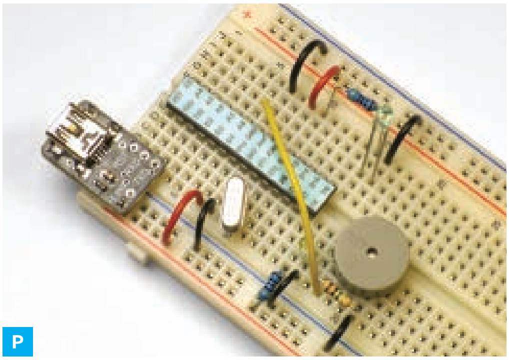
Now, add the following program statement anywhere within the `setup` function:

```
digitalWrite(2, HIGH); // enable pull-up resistor
```

This statement enables a built-in *pull-up resistor* that connects digital pin 2 (D2) to positive power



O



P

whether it's configured as an input or an output. This puts the pin in a steady `HIGH` voltage state until its voltage is *pulled down* by being connected to ground.

Compile and upload your modified sketch, then open the Serial Monitor again and repeat the experiment with the jumper wire. Notice anything different? What happens when the jumper wire is not connected to either power rail?

Now let's use a real button. Install a small SPST push button on the breadboard. Connect the loose end of the jumper to one button terminal and use a short jumper wire to connect one of the other button leads to ground (Figure N, previous page).

If you remembered to unplug the USB cable as you should have before modifying the hardware, you might have noticed that the Arduino IDE does not take kindly to having its connection dropped. Lots of reddish, ochre error messages spew across the status bar. Closing and reopening the Serial Monitor will sometimes solve this. If not, try shutting down and relaunching the IDE.

After installing the push button, try out your new invention. What happens when you press the button? What happens when you hold it down for a long time? What happens when you release the button?

You can add a dedicated reset button to your circuit, to easily interrupt and restart your program. I recommend uprooting everything and moving it all down to make room for the reset button at the top end of the breadboard, where it will always be easy to reach. If your reset button straddles the center gap over rows 1 and 3, install a short jumper under the adapter between A-1 and the ground rail, and another between C-3 and C-4 (Figure O). This connects the button such that pressing it pulls the reset pin down, resetting the chip.

9. KNOCK, KNOCK

Now let's use the piezo speaker as an input device, specifically, a knock detector. If you've left the speaker installed from the previous experiment, move its jumper wire going to digital pin 8 (D8) over to analog input pin 0 (A0). Also install a 1MΩ resistor across the speaker leads or from A0 to ground (Figure P).

Open the example sketch File → Examples → 6.Sensors → Knock. On line 28, change the threshold value from `100` to `3`. This will make our knock sensor more sensitive. Compile the sketch and upload it to your Arduino.

Open the Serial Monitor window and knock on the speaker. When the Arduino detects your knocking, it will print the message "Knock!"

The piezo transducer can work as either a speaker or a microphone, transforming electrical energy into mechanical vibrations, or vice versa. When you knock, it transforms the mechanical energy into a small electrical signal that the Arduino can detect. Pretty amazing, no? (To go further, make the Secret-Knock Gumball Machine on page 92.)

The World's Your Arduino

We've only begun to explore what you can do with your new, handmade Arduino. You can find more fascinating and enlightening experiments to perform at makezine.com/25/arduinoprimer. Some require additional hardware, but many only require a little imagination.

I hope you've enjoyed this introduction. Please let me know what you think. You can reach me at dale@dalewheat.com.

Dale Wheat (dalewheat.com) is a full-time freelance writer and part-time student. He likes things that blink or beep.

Get in the Game

Make all kinds of game controllers using an accelerometer or other sensors. BY TOM IGOE

Making stuff is more fun when you do it with other people. So is gaming. So why not combine the two? Here's a game you can play against your friends using game controllers you build yourselves.

The game is simple. A ball drops from the top of the screen. Your goal is to keep it from hitting the ground. Each player has a paddle on-screen (Figure A, following page). You can move your paddle left, right, up, and down. You get a point for each new paddle the ball hits (bouncing it off your own paddle multiple times doesn't help). The more players you have, the easier it is to keep the ball in the air. You get 5 balls per game.

To move your paddle up, send the command `u`; to move down, send `d`; to move right, send `r`; and to move left, send `l`. That's it. The fun part is figuring out what kind of game controller to build, and how to program a microcontroller to read your actions to send those commands.

» MAKE A BALANCE-BOARD GAME CONTROLLER

You can make a balance-board controller using an accelerometer (Figure B). This board is made from 2 discs of stiff 3-ply cardboard, on a rocker made of 2 semicircles notched to fit perpendicular to each other. An Arduino and accelerometer are mounted on the center of the balance board. Balancing and controlling the paddle is challenging but fun!

1. Attach each axis of the accelerometer to a different analog input of your microcontroller (Figure C).

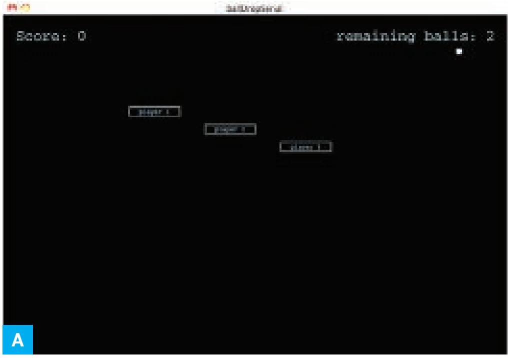
2. Now you need to find the range of values that the accelerometer's 2 sensors return (you'll need to do this no matter what sensors you use). Download the *SimpleReader.pde* code at makezine.com/25/gamecontroller, upload it to your Arduino, and open the Serial Monitor.

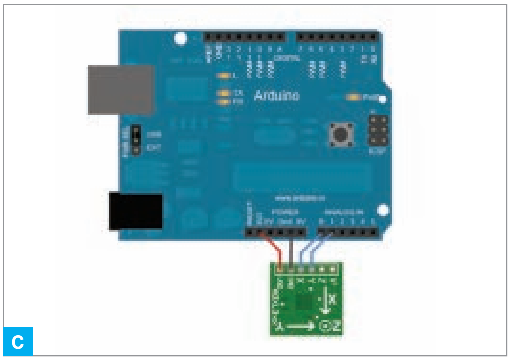
Photography and illustration by Tom Igoe

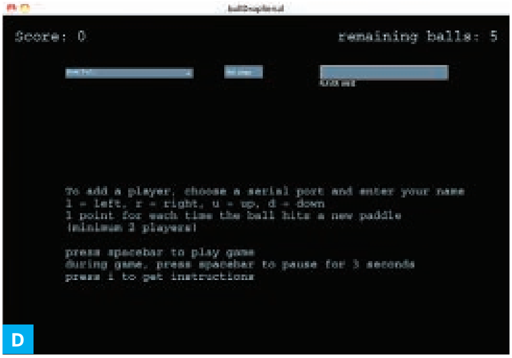
MATERIALS AND TOOLS

You can build your game controller using any microcontroller that can communicate through a serial connection to your computer. The code and hardware examples shown here are done on an Arduino, but any microcontroller will work, as long as it can communicate at 9,600 bits per second.

- **» Arduino Uno microcontroller Maker Shed #MKSP4 (makershed.com), $35**
- **» 3-axis accelerometer, type ADXL335 Adafruit Industries #163 (adafruit.com)**







3. Tilt the accelerometer from side to side on one axis, so that only one value changes. Write down the upper and lower extremes of the changing sensor value, and the center value when the accelerometer is level.

Set it down, level. How many points does it drift across without you touching it? This is your threshold. For example, if it drifts by 10 points either way around a central number, then you could safely pick a threshold of 12 (a cushion of a few extra points is a good idea). Then do the same on the other axis.

4. Enter these values into the *BasicController.pde* code that you can download at makezine.com/25/gamecontroller. Here's an example of what to enter:

```
int left = 400; // minimum for sensor 1
int right = 610; // maximum for sensor 1
int up = 400; // minimum for sensor 2
int down = 610; // maximum for sensor 2
int center = 500; // center value for both
int threshold = 12; // drift across center for both
```

In the main loop of your program, you'll use these values along with the accelerometer's readings to determine when to send `r`, `l`, `u`, or `d`. Here's a snippet that reads the sensor for the horizontal axis:

```
// read the first sensor:
 int horizontalSensor = analogRead(A0);
 // if the reading is between minimum and
    maximum...
 if ((horizontalSensor >= left) && (horizontalSensor
    <= right)) {
  //... and the reading's above center by the
    threshold amount:
  if (horizontalSensor > center + threshold) {
   Serial.print("l");
  }
  //... or it's below center by the threshold:
  else if (horizontalSensor < center — threshold) {
   Serial.print("r");
  }
 }
```

5. Make sure there's a position at the center where your controller sends no values. You want to be able to keep your paddle still onscreen. If there's no rest position, increase your threshold. If the rest position is too large, decrease the threshold.

Then apply the same logic to the reading for the vertical axis, changing the variable names as needed, and substituting `u` and `d` for `l` and `r`.

E

F

G

H

6. Fire up the game, choose your name and serial port (Figure D), and start playing.

7. If you find your paddle is too sensitive, or too sluggish, it's likely sending its messages more frequently than the game is reading them. Try adding a delay of 50 milliseconds at the end of your loop, like so: `delay(50);`

That's it! You can use this same method for lots of different sensors by adjusting the minima, maxima, thresholds, and center value. I used it with an accelerometer, with a joystick (which contains 2 potentiometers), and with 2 infrared ranging sensors, by changing only those values.

MORE CONTROLLER IDEAS

Figures E–H show a few other controllers. In Eric Beug and Lesley Flanagan's seesaw controller (Figure E), an accelerometer moves with the seesaw to move the paddle left and right.

In Daniel Liss' eyeglasses controller (Figure F), a photocell over each eye detects whether the eye is open or closed, and the paddle moves toward whichever eye is open.

Scott Varland's helmet controller (Figure G) is another accelerometer project; tilting your head moves the paddle where you want to go.

And in Raphael Zollinger's handshake controller (Figure H), shaking the left or right hand moves the controller in the right direction. Force-sensing resistors in the hand determine how hard you're holding.

Try your own combinations. All you need are 2 analog sensors, a microcontroller, some building materials, and a little imagination.

+ The game code, controller source code, and simple sensor reader are available for download at makezine.com/25/gamecontroller. The game source code is written in Processing (processing.org). If you want to modify the controller code or sensor reader, you'll need the ControlP5 library available from sojamo.de/libraries/controlP5.

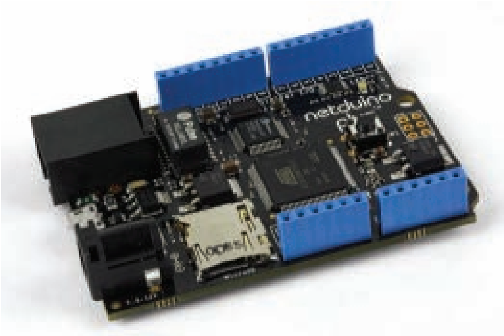
Tom Igoe is an associate arts professor at the Interactive Telecommunications Program (ITP) at NYU (itp.nyu.edu), and a co-founder of the Arduino microcontroller platform. He is the author of *Making Things Talk*.

Photograph by Estee Wah (E)

Micromania

Everything Arduino: Resources, kits and parts, cool projects and people.

The Secret's Out with Secret Labs



The "secret" part of Secret Labs can now be told: it's an open source electronics platform based on Microsoft's .NET Framework called the Netduino ($35, netduino.com).

This amazing piece of hardware allows the millions of .NET software programmers to enter the world of physical computing and hardware development, using a language they already know and love.

The Netduino features an Atmel 32-bit microcontroller that runs at 48MHz. When you couple that with the 20 general-purpose pins, a whole world of possibilities opens up. With just a few lines of code, you can control a variety of different motors, LEDs, servos, and sensors with ease. It also has 128K of memory to store your program, a fair amount of space in the world of microcontrollers.

The Netduino board wasn't the only secret that Chris Walker (who runs Secret Labs) had up his sleeve. When I met with him recently, he showed me two boards: the Netduino+ and the Netduino Mini.

The Netduino+ features onboard Ethernet, allowing it to connect online with ease, and the Mini is so small (only about ½"×1"), it seems impossible to pack all that power into such a small footprint.

Secret Labs is a company you should keep an eye on. Walker has a lot more projects and products under development, and they're all just as interesting as the Netduino. I'd love to tell you all about them, but they're secret.

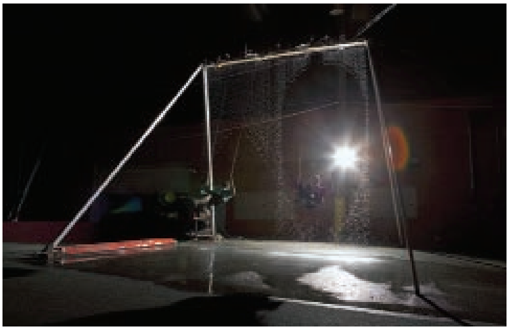
—Marc de Vinck

Swinging in the Rain

Maker Faire 2010 attendees got a chance to swing through an artificial rain shower and emerge perfectly dry. The swing, called *Deus Ex Machina*, was created by Andrew Ratcliff, Andrew Witte, Ian Charmas, and Michael O'Toole of Dash 7 Design.

The swing consists of an 18' frame with seats suspended from aircraft cables and has 30 gallons of water sprinkling down from the crossbar every minute, controlled by 273 independently actuated solenoids.

When the installation's Kübler incremental encoders detect the seats passing under the rain shower, it shuts down the relevant solenoids, and the swingers pass through unscathed. ***—John Baichtal***



Deus Ex Machina photograph by Paul Sobota

makerfaire.com/pub/e/3425

makezine.com/go/swing

iPhone LED Suit

"A disco party in my pants."

After attending Dragon*Con 2009 empty-handed, Miami maker Marc DeVidts knew he had to make a killer costume for 2010. What he came up with has been described as "a disco party in my pants" — a brilliant, full-color, full-body LED suit that's controlled by his iPhone.

As awesome as the suit is in action, DeVidts' saga of inventing it is equally impressive. He threw down for 200 Bliptronics RGB LED modules and one tiny SparkFun Arduino Pro Mini to control them all. He taught himself to design circuit boards in SolidWorks, then CNC-milled a custom controller board and sewed the whole kit into a white shirt and pants. An electrical engineer by trade, he even taught himself to write iPhone apps.

"Being a true iPhone user, my first thought was 'Oh man, it would be sweet to have an app for that,' " DeVidts writes. "However, there was one small problem. I didn't know how to write apps, and I didn't even own a Mac!"

One trip to the Apple Store fixed that. His app sends commands to the suit via a Lantronix wi-fi to serial module, to create customizable, rippling waves of light and color, flashing effects, even an audio-triggered display that turns him into a human equalizer.

—Keith Hammond

LED suit build log: makezine.com/go/ledsuitlog

LED suit in action: makezine.com/go/ledsuitvid1, makezine.com/go/ledsuitvid2

The Uberhoop

Uberhoop (sites.google.com/site/uberhooper) dazzled Faire-goers at World Maker Faire in September; the spinning hoops and blinking lights disguised custom hardware and sophisticated Arduino programming.

Maker Christian Miller, a doctoral student in computer science in Austin, Texas, admits that because the Arduino project supplies a free programming environment, "the software development part of the process was a breeze." The hardware side, on the other hand, required 400 solder joints! Thanks to a soldering party with over a dozen volunteers at the ATX Hackerspace, six hoops were there in New York, spinning their way into people's imaginations. *—Arwen O'Reilly Griffith*

Foiled Again!

The folks at Nerdkits (makezine.com/go/nerdkits) have an easy tutorial explaining how to use their ATmega168-based microcontroller kit to make a candy jar that lights up when you reach inside.

You line the container with two sheets of aluminum foil to turn it into a big capacitor, then use the microcontroller to cyclically charge and discharge it while timing the discharges.

When someone inserts a hand, the capacitance is increased, which lengthens discharge time. Detecting this, the controller lights some LEDs — the "Hello, world" of the hardware realm — but with its digital output pins you could almost as easily sound a klaxon or trigger a mini-guillotine.

—Paul Spinrad

Want some Arduino project ideas?

40 Top Projects

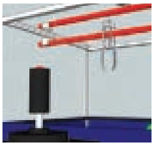
For Arduino inspiration, check out Hack N Mod's list of Top 40 Arduino Projects of the Web, a well-rounded compilation.

Earning the well-deserved Number 1 spot is the laser harp (pictured above), which is both visually and conceptually impressive. Number 11 goes to Ganzbot, an endearing Arduino robot head housed in a cardboard box. He reads tweets out loud and expresses himself with moving eyebrows and LED eyes and a mouth.

Other gems: a door lock with a colored LED light sequence as its combination, a sensor that tweets when your houseplant needs water, a breathalyzer microphone, and an elegant and practical turn-signal biking jacket. Get inspired!

—Laura Cochrane

» makezine.com/go/hacknmod



MEGA CLAW GRABS A PRIZE

An open source version of the classic child-robbing claw game.

Mega Claw (megaclaw.com) was created by seven-person hacker collective SplitReaction to make a big splash at the World Maker Faire in New York City.

To move the claw around the 10' game area, the makers repurposed cheap cordless drills from Harbor Freight run by an Arduino Pro Mini. They designed the claw in Google SketchUp and cut out the parts on a band saw. (Their design files are available at thingiverse.com/thing:4238.)

Mega Claw was a hit at the Faire. Players manipulated the game's giant joystick and grabbed balloons to earn prizes. The project earned a prize of its own, taking home an Editor's Choice Blue Ribbon award. *—John Baichtal*

Flying Wi-Fi Sniffer

First there was wardriving — driving through your neighborhood to discover wi-fi networks. Then a pilot in a Cessna airplane brought us warflying. Now, an Arduino-based plane called the WASP (Wi-Fi Aerial Surveillance Platform) makes warflying relatively cheap, easy, and fun.

The robotic progeny of Mike Tassey (RedQueen) and Rich Perkins (WhiteQueen) of Rabbit-Hole (rabbit-hole.org), the WASP is an unmanned aerial vehicle (UAV) with a few optional extras installed. Namely, a wi-fi detection system and autonomous flight control.

The 6'-long surplus Army drone is made to look like a Russian fighter. It's larger than most R/C planes, making it suitable to carry aloft the various radios, controllers, GPS, and sensors used for automated flight as well as its RF "payload."

The payload can be chosen from a number of off-the-shelf modular radios the team has put together. The main component is a Pico-ITX computer running the Backtrack wireless testing suite. Attached to that are a 3G modem, wi-fi, ZigBee, and Bluetooth modules, and a video downlink device.

They're also planning to intro-

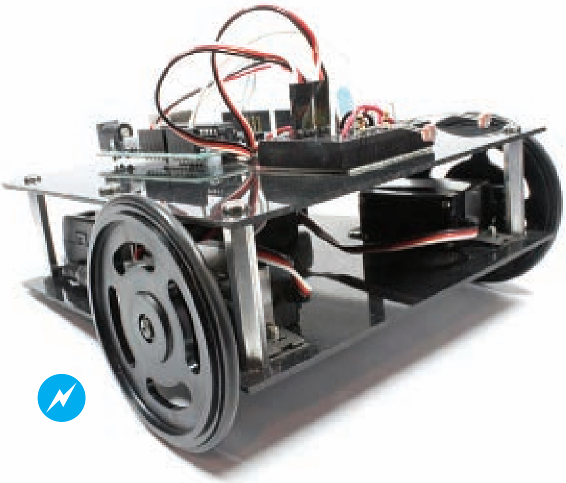
duce a GSM cellular "tower" that will fit into the airframe to create a cellphone tap in the sky.

The WASP is manually controlled for takeoff and landing but flies itself once airborne using GPS waypoints and ArduPilot. If the WASP finds an interesting radio source, it can fly circles over the target and collect (or deliver) RF signals.

A portable base station on the ground runs Windows XP and collects telemetry from the flight via ZigBee or 3G. The base station can connect several users to the airborne WASP.

—Mike Outmesguine

Photograph by Stephen Hobley (laser harp)



Blu Robotics Kit

photonrobotics.com

Photon Robotics founder Bill Hunt realized early on that "a lot of people are excited about the idea of robotics but don't quite know where to begin." Existing kits are often either too complicated for beginners or not relevant to real-world experience. Solving this problem is a noble goal, and Hunt didn't waste any time, despite the fact that he was 14 years old. A year later, Photon Robotics is hard at work fine-tuning the Blu robotics kit.

Hunt is an engineer at heart, so his favorite part of the technical design process "has definitely been experimenting with the wonderful Arduino platform." He picked up the new Uno controller module at World Maker Faire in September 2010 and has already integrated it into Blu's design.

His love affair with Arduino can only benefit future roboticists: "I'm hoping that the experience I've had with the Arduino will be passed on to Blu's users," says Hunt. Now that's a good place to start. *—Arwen O'Reilly Griffith*

A SHIELD FOR THAT

One of the most fun things about Arduino is the lively ecosystem of add-on boards called shields.

Shields ride on top of the Arduino board and plug directly into the headers, so it's easy to swap them in and out to add functionality.

Do you want Ethernet connection, servomotor control, audio playback, or GPS? There's a shield for that. How about SD memory, XBee wireless radio, or any sensor you can name? That too.

To get an overview of the expanding universe of shields, check out the comprehensive, searchable Arduino Shield List (shieldlist.org) maintained by Jonathan Oxer. The list boasts descriptions and pin usage details "for 196 shields from 83 makers, and counting!"

—Keith Hammond

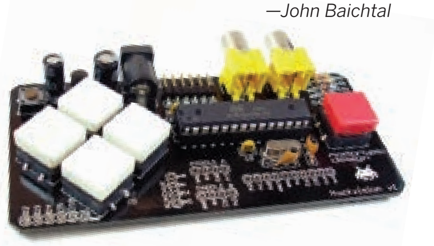
Retro Gaming Future

nootropicdesign.com/hackvision

Hackvision ($48; kit $38), an Arduino-based video game system designed by Michael Krumpus, connects to a television via standard RCA plugs, and enables users to play retro games like Pong and Space Invaders.

Big directional buttons and a red "fire" button are mounted directly on the board next to resistors and the ATmega328 microcontroller, or users can solder in headers for a Wii nunchuck or game paddle. The platform is eminently hackable, with an open source design and downloadable device libraries.

—John Baichtal



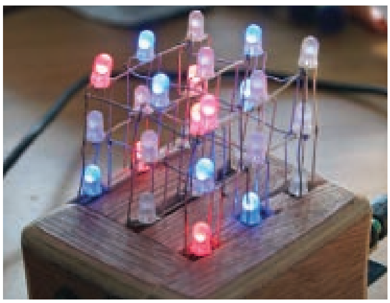
TickTack³

makezine.com/25/ticktack3

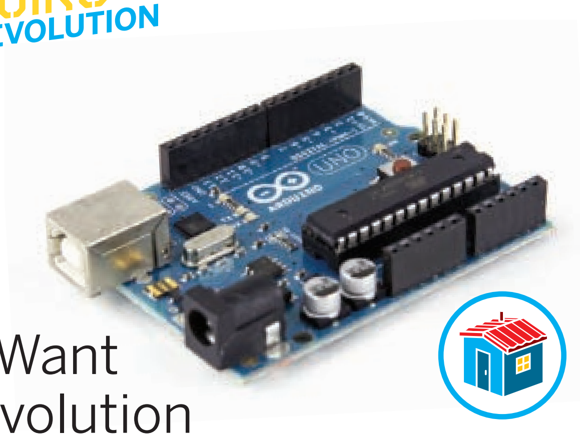
When SparkFun Electronics had its infamous Free Day last year, Matthew Katzenstein tried to get an LED Cube 3×3×3 kit, a programmable 3D cube of RGB LEDs. But the online giveaway was halted after 90 minutes of overwhelming participation, and Katzenstein's gift request, like many others, didn't go through. So he decided to make his own, TickTack³.

He improved on the original's fragile design and made it interactive by adding a knob and programming it to play 2- and 3-player versions of color-coded 3D tick-tack-toe. The original costs $70, but you can make TickTack³ for just $20. Download a materials list, schematic diagram, and EagleCAD file for the circuit board at makezine.com/25/ticktack3.

—Paul Spinrad



KITS AND PARTS



Say You Want a Revolution

makershed.com

The Maker Shed store carries hundreds of DIY kits, tools, components, and books for electronics and robotics, but the hottest category is definitely Arduino-related kits. A simple search for "Arduino" results in more than 60 different products ranging from the ever-popular Getting Started with Arduino kit, to the vast array of shields that allow you to connect sensors, log data, blink more LEDs than you ever thought possible, and even connect to the internet. Each product is hand selected and tested by the staff of MAKE. If it's maker-made and plays nice with the Arduino, chances are we carry it in the Maker Shed.

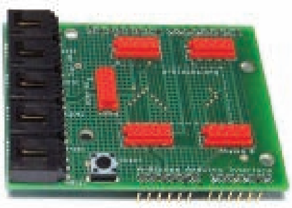
—Marc de Vinck

Getting Started with Arduino Kit v2.0

$74.95 **Product Code MSGSA**

Projects Pack for Arduino v2.0

$99.95 **Product Code MSAPK**



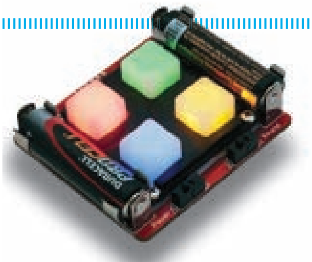
A-Blocks

a-blocks.org

Dutch company A-Blocks produces ready-to-use, easily combined Arduino building blocks ($5–$62). The A-Blocks system consists of an interface board on top of an Arduino, to which you can connect both A-Block modules and Phidget sensors. The coding is minimal: a bit to "glue" the blocks together and, sometimes, to add some specific behaviors. It's a timesaver.

For example, to build a home alarm that has on/off RFID tags, a door open/closed detector, and wireless transmission to the internet, it might take days on a regular Arduino, but with A-Blocks, it takes a few hours. It's ideal for teaching.

—Laura Cochrane



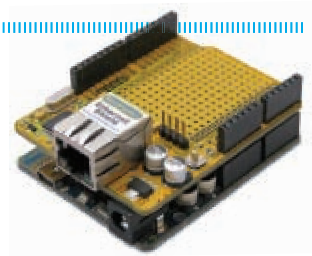
Simon Says Solder

sparkfun.com/products/9883

SparkFun brings the iconic 80s into the 21st century with the through-hole soldering version of the Simon Game kit. Great for beginners and beyond, the kit comes neatly packed with the preprogrammed ATmega328 microcontroller, all electronic components and hardware, inspirationally clear instructions, and even batteries (gasp!) — all for $25.

(Seriously, check out the downloadable instructions on the product page: clear beyond belief. You couldn't mess up if you wanted to.) Spend less than an hour building it, and countless hours trying to beat the machine.

—Goli Mohammadi



Ethernet Power

makezine.com/go/poe

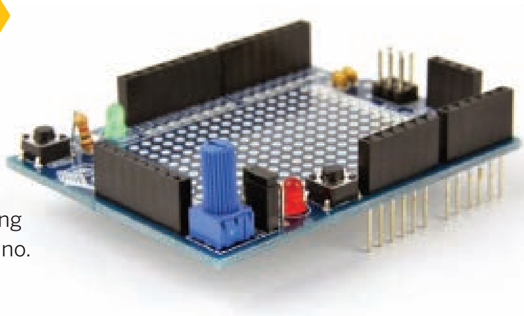
Do you like making Ethernet-enabled Arduino sensors but hate running wires to power them? Well, rather than building a complicated energy-harvesting system or relying on batteries, you might want to take a look at powering them using the same Ethernet cables used to transmit data to them.

This can be a bit messy to implement, but luckily the folks at Freetronics have cooked up the Ethernet Shield With PoE ($43) to do the dirty work for you. Just plug it in, hook up a single power supply at your router, and you're good to go!

—Matt Mets

MAKE's Essential Arduino Site

makezine.com/arduino



Microcontrollers have been a hot topic in the DIY scene for the past few years. One of the fastest-growing and most popular categories on Make: Online is Arduino. When I last checked, we had more than 1,100 articles about this amazing little microcontroller.

If you take into account that most articles on makezine.com have multiple tags, you quickly realize that "Arduino" is in fact a Top 3 category.

You may be asking why it's so popular, and I have a theory. It wasn't that long ago that the early Arduino serial boards were changing the way most people thought about programming microcontrollers. It made the whole process of programming the physical world accessible to most individuals.

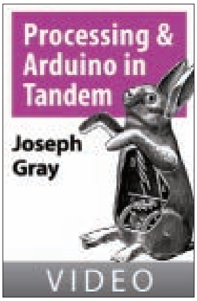
Jump forward a few years, and with the recent release of the Arduino Uno, almost anyone, including both my kids at around age 6 (with a little bit of help), can be programming interactivity within a few minutes of plugging the board into their computer.

This ease of programming, coupled with the Arduino's open source licensing, has built a community that's passionate about creating and sharing. It's why you can find so many new projects posted every day on the internet that use the Arduino in one way or another.

With the incredible amount of Arduino projects and information available online, it's great to have such a large collection of curated projects in one place.

Be sure to check out makezine.com/arduino and start exploring all that we have to offer. You'll find everything from my own "How-to Tuesday: Arduino 101" videos to interactive art, robotics, and more.

—Marc de Vinck



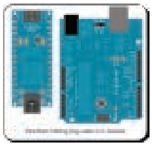
CRASH COURSE

makezine.com/go/tandem

If you're an artist wanting to take your work to the next dynamic level, check out the O'Reilly *Processing and Arduino in Tandem* video course, an 8-hour, in-depth introduction to the open source Processing programming language, Arduino microcontrollers, and how to make the two work together.

The video is taught by Joseph Gray, a Seattle-based artist and designer well versed in creating interactive art shows and dynamic sculptures. Though the pace is fast, the instruction is engaging and easy to follow, and you don't need any previous programming or electronics experience. When you get the materials and follow along, by the end, you'll have built the Arduino-powered Project Box, written code to control it, and be itching to delve further.

—Goli Mohammadi



The Cheat Sheet

Enjoy hacking on the Arduino but have a hard time remembering which pin is which and what function to use to get it to sleep? I certainly do, which is why I keep a copy of Gavin Smith's "Arduino Cheat Sheet" handy while hacking circuits.

From programming syntax, to common library functions, to specifications and pinouts for the different Arduino chips, the cheat sheet is full of useful information that can be hard to remember. Download it at makezine.com/go/cheatsheet.

—Matt Mets

The 1-Minute Microcontroller

Super Awesome Sylvia's how-to videos (makezine.com/go/sylvia) always leave my smile muscles aching, and the "Super Simple Arduino" episode of her *Mini Maker Show* is the best yet. In it, the 9-year-old host demonstrates two Arduino projects that literally take seconds to make: an adjustable LED strobe, and a beeping "randomly influenced finger flute." Her explanations are charming and clear. Watching Sylvia, you'll realize that the Arduino is a great toy for any child who's old enough to push a wire lead into a header.

—Paul Spinrad

1+2+3 PET Bottle Purse

By Zitta Schnitt

You can make it!

Follow these instructions to build a cool purse out of the bottoms of 2 PET bottles and a zipper. You can use the container as a coin purse or a storage box.

YOU WILL NEED

- 16oz PET bottles (2) such as soda bottles
- 9" zipper
- Tape
- Nylon thread
- Vise (optional)
- Hacksaw
- Small pair of scissors
- Thin needle
- Thick, sharp needle
- Pliers or seam ripper

1. Cut the bottles.

Cut off the bottoms of both bottles using the hacksaw, while holding the bottle in place with a vise. Then trim the burrs and any excess material with the small scissors.

2. Punch the holes.

Download, print, and cut out the hole-punching template found at makezine.com/25/123_bottlepurse. Tape the template, dots facing up, to the outside of the bottle, the top edge of the template flush with the cut bottle edge.

Holding the thick needle with a pair of combination pliers or vise-grip pliers, punch holes in the bottle, using the dots on the template as a guide. (You can also use a seam ripper to punch the holes.)

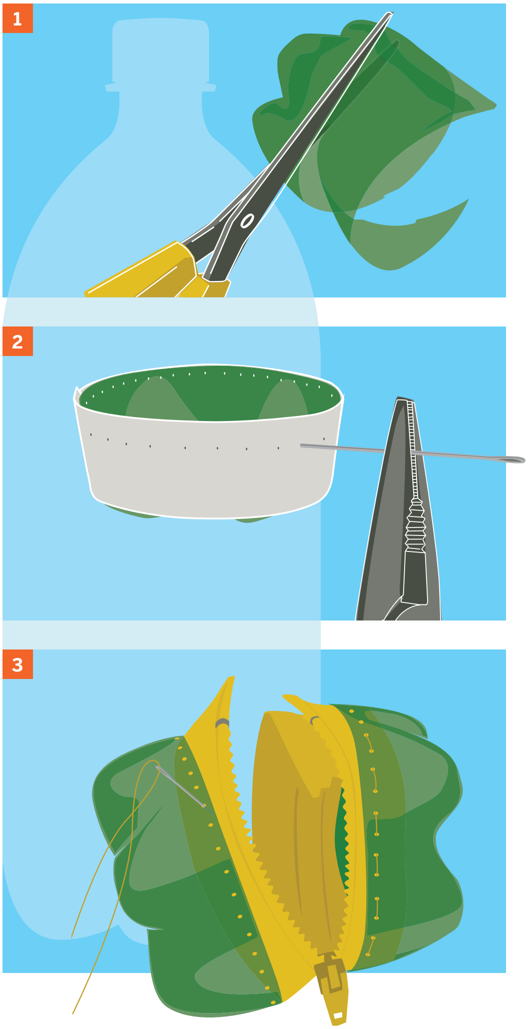
When you're done, move the template to the other bottle bottom and repeat the hole-punching process.

3. Attach the zipper.

Unzip the zipper before sewing it into both bottle bottoms. With the thin needle and thread, begin stitching 1¼" below the zipper's top end, leaving it loose for now (later you'll overlap it with the zipper's bottom, and stitch them both down).

Stitch the zipper around one bottle bottom, and then around the other (inverted) bottle bottom.

To finish, overlap the zipper's bottom end with its top, and sew both ends to the bottle bottoms through the remaining holes.



Illustrations by Tim Lillis

Zitta Schnitt, born in Vienna, spent her childhood in Austria and Hungary. In art school she focused on textile design. Now she studies industrial design at the University of Applied Arts in Vienna, with a special focus on tableware and lighting design. zittaschnitt.com

Make: Projects

Our "Big 3" projects will make you the hit of the neighborhood. Wow your friends with a model rocket whose spring-loaded helicopter rotors pop out for a gentle return to Earth. Then, keep them guessing with a gumball machine that knows your secret knock. Finally, treat them to a unique, slow-cooked gourmet meal, sous vide style.

Five-Dollar Helicopter Rocket

82

Secret-Knock Gumball Machine

92

Sous Vide Immersion Cooker

104

Photograph by Garry McLeod

$5 HELI-ROCKET

By Doug Desrochers

THE GREAT DESCENT

Use toilet paper tubes, coat hangers, and rubber bands to build a high-flying model rocket that transforms into a helicopter midflight, then spins dramatically back down to Earth.

For years, I've been in charge of my local Cub Scout pack's model-rocket derby. I've also flown some unique rockets of my own, including a Redneck Rocket made from paper towel tubes and Walmart-bag parachutes; and a Scissor-Wing Rocket that converts into a glider after the boost phase. The HeliRocket is a twisted hybrid of these two concepts.

The HeliRocket is propelled by a standard model-rocket engine, but instead of the engine's ejection charge deploying a parachute, it releases the rocket's 3 tail fins. The fins then swing up into a helicopter blade configuration, which slows the rocket's descent by converting much of its kinetic energy into drag and angular momentum (spin).

Some commercial kits enable a similar "spin recovery," but they use separate helicopter blades. Having the tail fins perform double duty makes for a cool "transformer" effect and reduces rotational drag for faster spinning. Using household items, you can build this rocket for nothing more than the cost of the engine. (To fire it, you'll also need a launchpad, with either a fuse or a launch controller and igniter.)

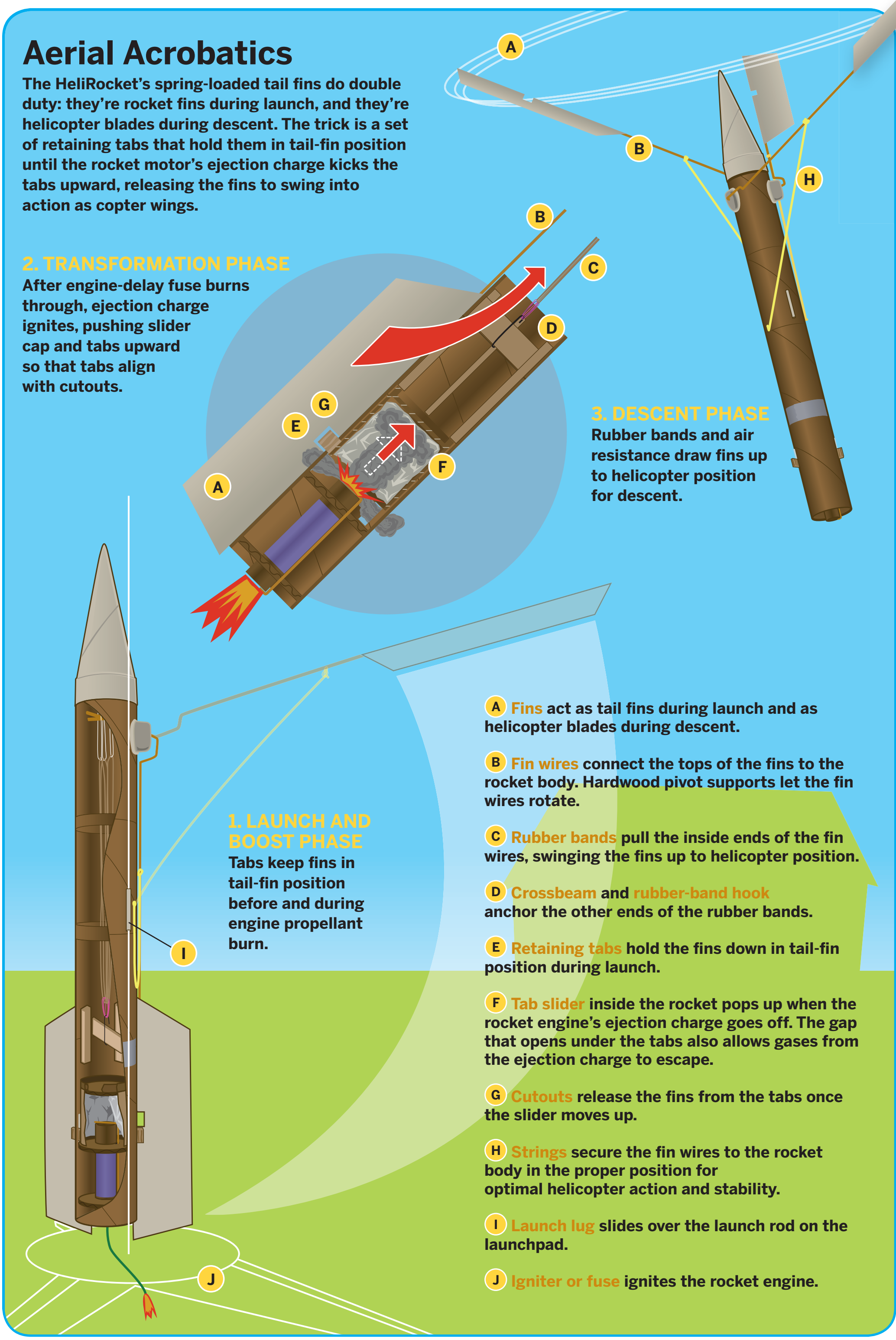
Set up: p.85 **Make it: p.86** **Use it: p.91**

Doug "Beads" Desrochers (beads27@cox.net) has a B.S. and an M.S. in aerospace engineering, served in the U.S. Navy as a test pilot and test pilot instructor, and is a local model-rocket and fireworks menace. This project is his third article for MAKE magazine, of which he is a charter subscriber. He's currently a systems engineer and test pilot supporting the Department of Defense near Washington, D.C.

Photography by Garry McLeod

Aerial Acrobatics

The HeliRocket's spring-loaded tail fins do double duty: they're rocket fins during launch, and they're helicopter blades during descent. The trick is a set of retaining tabs that hold them in tail-fin position until the rocket motor's ejection charge kicks the tabs upward, releasing the fins to swing into action as copter wings.



1. LAUNCH AND BOOST PHASE

Tabs keep fins in tail-fin position before and during engine propellant burn.

2. TRANSFORMATION PHASE

After engine-delay fuse burns through, ejection charge ignites, pushing slider cap and tabs upward so that tabs align with cutouts.

3. DESCENT PHASE

Rubber bands and air resistance draw fins up to helicopter position for descent.

A Fins act as tail fins during launch and as helicopter blades during descent.

B Fin wires connect the tops of the fins to the rocket body. Hardwood pivot supports let the fin wires rotate.

C Rubber bands pull the inside ends of the fin wires, swinging the fins up to helicopter position.

D Crossbeam and **rubber-band hook** anchor the other ends of the rubber bands.

E Retaining tabs hold the fins down in tail-fin position during launch.

F Tab slider inside the rocket pops up when the rocket engine's ejection charge goes off. The gap that opens under the tabs also allows gases from the ejection charge to escape.

G Cutouts release the fins from the tabs once the slider moves up.

H Strings secure the fin wires to the rocket body in the proper position for optimal helicopter action and stability.

I Launch lug slides over the launch rod on the launchpad.

J Igniter or fuse ignites the rocket engine.

Illustration by Tim Lillis

SET UP.

Photograph by Ed Troxell

MATERIALS

[A] Thin cardboard, flat plastic, or other light, thin, stiff material You can use a cereal or 12-pack soda box or plastic cut from a liquid laundry detergent container.

[B] Corrugated cardboard, 8"×4" by about 1⁄16" like from a gift box rather than a thick moving box, with corrugations running the long way

[C] Scrap hardwood, 2"×1"×½" is plenty I used some scrap cherry trim.

[D] Scrap balsa wood A broken balsa rubber-band plane will supply enough material; also available at most hobby stores.

[E] Packing tape or duct tape

[F] Metal repair tape or aluminum foil

[G] String, about 5'

[H] Gorilla Glue or other polyurethane glue

[I] Wood glue

[J] Super glue

[K] Fast-setting epoxy

[L] Toilet paper tubes (5)

[M] Paper towel tube

[N] Wire coat hangers (2)

[O] Plastic drinking straw

[P] Rubber bands (3)

[Q] Paper clips (2)

[R] Model rocket engine, C6-3 (one-time use) C6 is about the right size and power, and a 3-second delay between end of thrust and ejection charge is about right. Available from most hobby stores in 3-packs.

[T] Model rocket launch-pad (includes launch rod) You can make your own — for more casual rocketeers, a 3' length of ⅛" wire rod (about $3 from hardware stores) stuck in the dirt does the trick. You can also buy a launchpad and controller together as a "launch system" for about $23 from Apogee (makezine.com/go/apogee) or Quest Aerospace (makezine.com/go/quest).

[S] Fuse with lighter, or igniter with launch controller I use a pocket lighter and 3" of American Visco waterproof cannon fuse, $8 for 50', from CannonFuse.com (makezine.com/go/fuse). You can also use an igniter (shown here) and a launch controller (1 igniter per launch). Igniters cost about 50 cents each in packs of 6, and controllers cost $20–40; both are available wherever model rocket engines are sold.

TOOLS

[NOT SHOWN]

Dremel or other rotary tool with grinding drum

Saw for cutting small wood shapes

Vise

Drill and #45 (.082") drill bit or other size to match coat hanger wire

Wire cutters

Pliers (2)

Hobby knife

Scissors

Ruler

Marker

Small clamps, clips, or spring clothespins (at least 4)

MAKE IT.

BUILD YOUR YOUR HELI-ROCKET

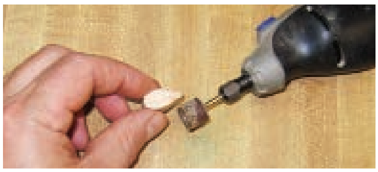
START **Time: 3–4 Hours Complexity: Easy to Moderate**

1. MAKE THE TOP TUBE AND FINS

1a. Split a toilet paper (TP) tube lengthwise, then cut off a 1"-wide ring. Trim a bit off the ring circumference so that it fits inside another TP tube without overlap. This is the first of several "doubler" reinforcements that are needed to strengthen parts of the rocket body.

1b. Completely cover the outside of the ring with wood glue, and slide it into place in one end of another TP tube so that it's flush with the end. This will be the top end of the top tube.

1c. Take a scrap piece of hardwood and cut 3 flat teardrop-shaped pivot supports about ½" wide and thick. Use a rotary tool to get a rough aerodynamic shape, but let's be honest here: the overall rocket is not very streamlined.



1d. Drill a hole centered in the wide part of each pivot support that allows your hanger wire to fit through snugly but still spin. A #45 bit usually works.

Using plenty of Gorilla Glue, evenly space and attach the pivot supports to the top tube on the end with the doubler, trailing edges pointed down. After the glue is dry, redrill the holes to clear out excess glue, and gently drill through the cardboard tube.

Photography by Douglas Desrochers

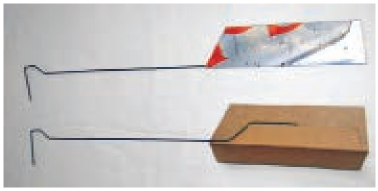
1e. Cut two more 1" doubler rings from the tube cut in Step 1a, coat them with glue, and use them as joints to connect 2 more TP rolls in series behind the top tube, to make a tube about 12" long. Inside the bottom end, use Gorilla Glue to mount a sturdy crossbeam (about ½" deep and ⅛" thick) for the rubber-band hook. (With thinner balsa or plastic, glue together a double layer to make the beam strong enough.)

1f. Cut pieces for 3 tail fins out of balsa or thin cardboard. Each fin should be about 2" wide, 8" along the long edge, and 7¼" along the shorter edge. See makezine.com/25/helirocket to download a template, or cut something similar. With cardboard, cut butterfly pieces that fold together into the tail fin shape. With balsa, cut 6 pieces (3 pairs) with the grain running the long way.

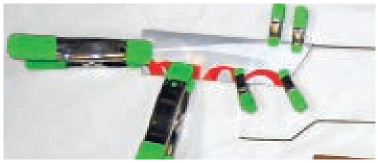
1g. Straighten the hangers with pliers and cut 3 lengths of 18" (you can get 2 per hanger). Put in a 90° bend 2" from the straightest end of the wires, a 45° bend in the same direction 1" farther down, and a second 45° bend back in the opposite direction, spaced to accommodate the wood pivot support thickness. Mark the wires 11" from the 2" bend, for positioning the fin later, then put 2 more 45° bends, approximately 1" apart, starting at 12" to form an internal support for the fin.

1h. Use pliers to slightly angle these last 2 bends 8°–10° away from the plane of the bends at the pivot end, offsetting the fin supports in the same direction on each wire.

1i. Cut three 6"×1" strips of thin corrugated cardboard in the same general shape of the fins, corrugations running the long way. Assemble the fins using Gorilla Glue, sandwiching the corrugated cardboard strip and wire between the folded soda-box cardboard (or 2 balsa pieces); the extra cardboard inside helps form a roughly symmetrical airfoil cross section. The 11" mark on the wire should be at the top edge of the fin.



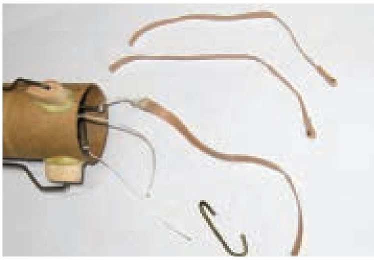
1j. Clamp the fins with small clamps, clips, or clothespins, and let dry.



1k. Insert one fin wire through a pivot support, all the way up to its first bend. Grip the pivot support with pliers, and while keeping the fin parallel to the tube, bend the internal 1½" part of the wire over about 45° using the second pliers. Repeat for the other 2 wires. Test-swing each of the fin wires up perpendicular to the tube in both directions, and if there's any internal rubbing against the tube, use wire cutters to trim the excess wire back.

1l. Tie a 3" piece of string near the end of each wire, and put plenty of super glue over the square knot.

Cut 3 rubber bands to equal lengths, and use super glue to make a loop at one end of each. Tie the string from each wire to the loop of a rubber band.



1m. Tie the free ends of the 3 rubber bands together with string, and connect them at equal length using a short hook, cut and bent from some extra hanger wire. Carefully push the bands and strings down the tube.

2. MAKE THE ENGINE MOUNT

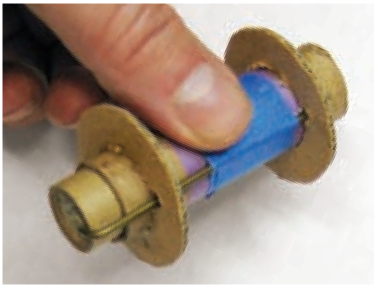
2a. Cut TP tube cardboard to wrap around a rocket engine. Gorilla-glue the cardboard over the engine and let dry (I used rubber bands and some tape to hold it in place during drying).

2b. For the engine holder, cut a piece of coat hanger long enough to clip around the length of the engine with one end bent to extend fully across the engine's diameter (approximately 1¼") and the other end bent to stop before the engine's center (approximately ¼") to leave room for the fuse or igniter.

2c. Cut 2 centering rings out of corrugated cardboard, with inside diameters fitting around the engine, and outside diameters fitting inside a TP tube. Cut a small notch inside each to fit around the engine-holder wire.

2d. Fit an engine into the mount by sliding it through the tube until only ½" sticks out the bottom. Line up the clip wire with its short end at the bottom of the engine. At the top, carefully use the tip of a hobby knife or a drill bit in hand to punch a hole through the holder tube for the long end of the clip wire to run through. Tape the long section of the clip securely to the tube; the clip should flex just enough for you to remove and replace the engine.

Finally, Gorilla-glue a centering ring around at each end.



3. MAKE THE TAB SLIDER

The tab slider needs to be tough since it catches the “backfire” from the rocket engine’s ejection charge, which normally deploys a parachute.

3a. Cut a 2½" length of TP tube and reduce its diameter so that it slides easily inside the paper towel tube. Glue a 1"-wide doubler (as in Step 1a) inside the bottom of this tab slider tube, using wood glue.

3b. Cut a corrugated cardboard disk to fit in the top end of the tab slider, and secure it inside with plenty of Gorilla Glue, making a tight seal.

3c. Once the glue is dry, line the inside of the slider with metal repair tape or layers of aluminum foil.

4. MAKE THE BOTTOM TUBE

4a. Cut a 6½" length of paper towel tube. Then glue the 2 centering rings of the engine mount inside the 6½" tube, so that the engine’s bottom is roughly flush with the end of the tube. Once the glue is dry, measure the distance from the top of the 6½" tube to the top of the upper engine-mount centering ring, and mark where that ring is on the outside. Measure up 3" and mark the tube again. Put the tab slider inside the 6½" tube, with the sealed end up (away from the engine). It should easily slide up and down, with minimal gaps around the perimeter.

4b. Cut 4 strips of thick (or double-layer glued) scrap balsa, 2½" long and approximately ¼"–⅜" wide, and glue them one at a time evenly spaced around the top of the 6½" tube, with the lower part of the strip at the same depth as the 3" mark made earlier. They should extend about 1½" from the top rim. The lower parts of these strips provide the stop for the tab slider, which should now move freely up and down about ½".

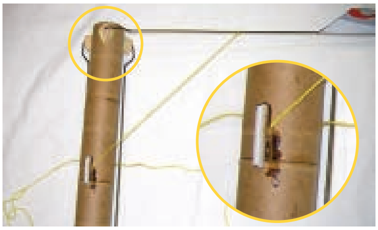
4c. Following the same even spacing as the pivot supports, cut three 1"×½" rectangles out of the 6½" tube, starting just above the mark you made at the upper engine-mount ring and cutting upward 1", being careful not to cut into the slider.

4d. Use Gorilla Glue to attach two ½"-square balsa tabs through each hole to the now-visible tab slider. Space the tabs apart just enough to accommodate a fin, and take care not to glue the slider to the 6½" tube. With the tab slider pushed up, a gap should appear below the tabs, where the hot gas will exhaust.

5. FINAL ASSEMBLY

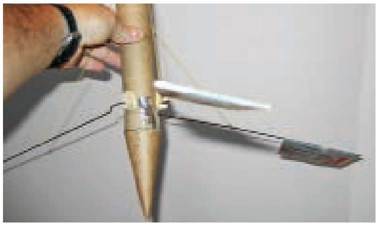
5a. Cut 3 lengths of string about 15", and epoxy each to the top tube, directly down from a pivot point and 9" away.

5b. Make a hook tool from scrap hanger wire, pull the rubber-band hook down inside the top tube, and latch it into the balsa crossbeam. The fins should pivot up, but don't worry if they don't go all the way out — aerodynamic forces will take care of that during flight.



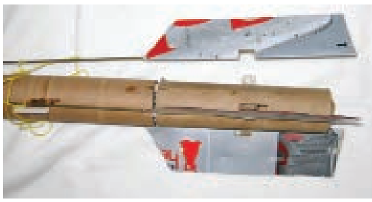
5c. Tie and epoxy the other end of each string to the midpoint of each fin wire, calibrating the length so that each wire angles up about 10° when its string is taut. This angle is called a *dihedral*, and it increases stability. Put drops of super glue on the knots, and trim excess string.

5d. Make a nose cone using cardboard and duct tape, foil, or whatever — it should be as close to symmetrical as possible. Secure it centered to the top, and make sure it doesn't interfere with the pivoting motion of the fin wires.

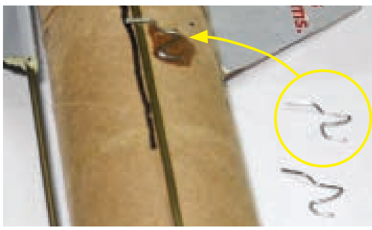


5e. With the rubber-band hook in place, slide the 6½" tube into the upper section (NASA calls this "assembling the stack") and tape the tubes together. You'll need a launch lug to guide the rocket up the launch rod. Cut a 2" length of plastic straw and slice it lengthwise. Wrap the plastic around the launch rod so that it fits closely but can still slide, and tape it together. Glue this 2" lug onto a small spacer of the same length, approximately ¼" wide by ⅛" thick (balsa, plastic, scrap wood, or even cardboard will work fine). Glue the lug and spacer vertically on the bottom tube, placed so it doesn't interfere with the pivoting fin wires.

5f. Make cutouts on each fin, sized and positioned so that when the tabs slide up, the fins are released. The completed fins should look like ceiling fan blades when fully extended but will radiate out perpendicular to the tube when secured in the tabs.



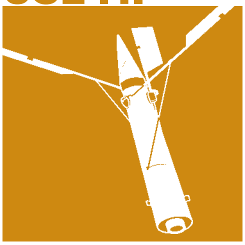
5g. Finally, the TP tubes sometimes flex during blastoff, which may cause a fin to pop out from behind the tab. Adding a small paper clip retaining wire above each fin prevents early fin deployment. Bend a paper clip segment so that about ¼" of it angles away, and epoxy it with the fins in the tabs, so that the fins can only slide in the direction of the rubber band pull.



FINISH X

NOW GO USE IT »

USE IT.



CARDBOARD ROCKET COPTER, BLASTOFF!

LAUNCH TIPS

The completed HeliRocket should weigh about 130 grams (4.6oz) without its engine, so a C-series rocket engine will take it up about 300 feet. In my experience, a C6-3 engine works best; the "3" indicates a 3-second time delay on the backfire, which lets the rocket coast down a bit before the fins are released.

As with other model rockets, you should launch the HeliRocket under calm wind conditions. With fuse or igniter in place in the engine, just slide the rocket's launch lug over the launchpad's rod, step away, count down, and fire.

RECOVERY AND REUSE

The HeliRocket is very quickly reusable, as long as you don't fire it at a crazy angle. Just unclip the spent engine, discard it and clip a new engine into the engine mount.

The descending rocket spins so fast that a pivot support can pull loose if a fin hits the ground. To prevent this, catch the rocket in your hand before it touches down, which is a fun thing to try anyway.

+ See a video of the HeliRocket in action at makezine.com/25/helirocket

Photography by Garry McLeod (top), Douglas Desrochers

SECRET-KNOCK GUMBALL MACHINE

By Steve Hoefer

CANDY RAPPER

Make a cute candy vending machine that only dispenses treats when you knock the secret rhythm on its front panel.

One of the best things about exhibiting at Maker Faire is giving attendees a challenge. For the 2010 Maker Faire Bay Area, I decided to combine a past project of mine, a door lock that opens only when you give a secret knock, with a standard crowd pleaser: candy.

The result was this Secret-Knock Gumball Machine, which tempted and tested the crowds at Maker Faire to guess the right rhythm and receive a treat. Since the knock was not terribly secret (I happily handed out hints), it distributed hundreds of gumballs over the event's two days.

The "secret" knock defaults to the famous "Shave and a Haircut" rhythm, but you can program custom knocks by simply pressing a button and knocking a new pattern. The machine only listens for the rhythm, not the tempo, so the correct knock will dispense a treat whether you perform it fast or slow.

Inside the machine, a piezo sensor picks up sounds from the front knock panel, while an Arduino microcontroller recognizes the target pattern and controls a servo-driven gumball-dispensing wheel. You can build the Secret-Knock Gumball Machine with its inner workings visible or hidden, depending on whether you want to show off the mechanism or keep it a mystery.

Set up: p.95 **Make it: p.96** **Use it: p.103**

Steve Hoefer is a technological problem solver in San Francisco. He spends much of his time trying to help people and technology understand each other better.

Photograph by Garry McLeod

Open Sez Me

A Sweet, satisfying gumballs tempt passersby.

B A piezoelectric sensor detects knocks on the front panel.

C An Arduino microcontroller listens for the knocks and tracks the relative times between them.

D If the Arduino determines that the input knock rhythm matches the target pattern, it switches on the servomotor.

E The servo rotates the dispensing wheel through the gumballs, which causes one of them to fall into its delivery hole.

F As the wheel completes its rotation, it dumps the gumball from its delivery hole into the delivery tray.

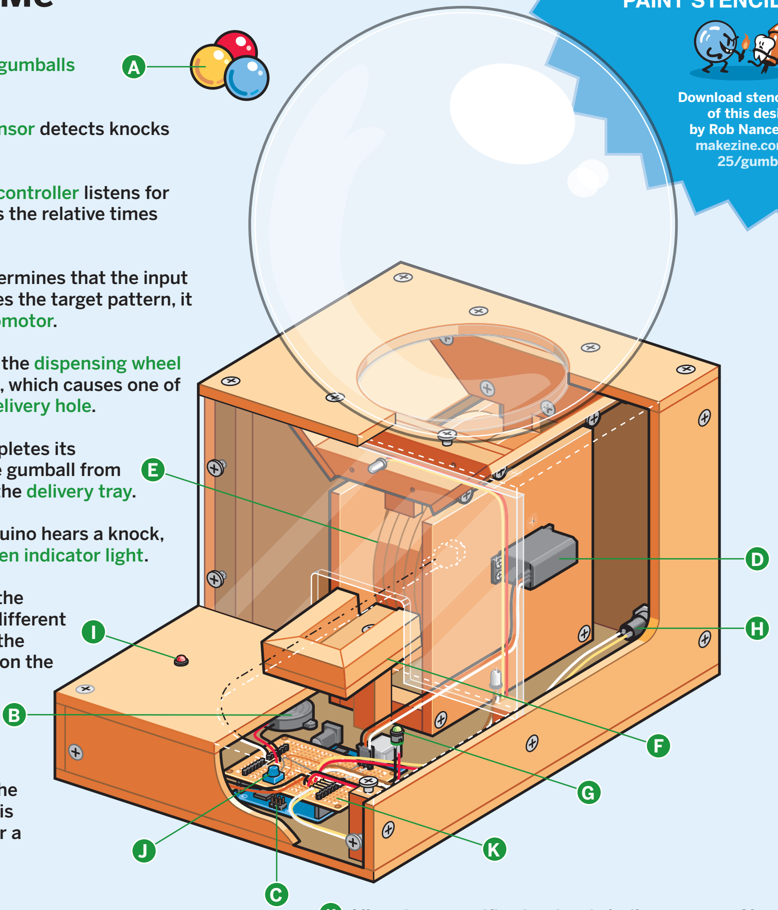
G Whenever the Arduino hears a knock, it also flashes the green indicator light.

H You can program the Arduino to listen for different patterns by pressing the programming button on the back and knocking a new sequence.

I The Arduino shines the red indicator light while the programming button is pressed and whenever a knock pattern fails to match the target.

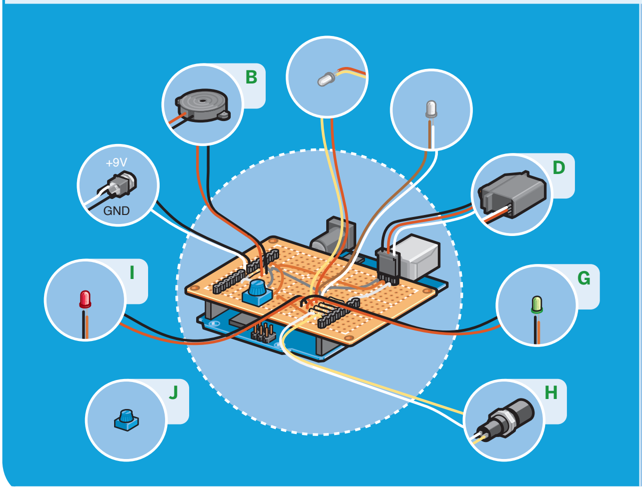
J The sensitivity potentiometer sets the level of loudness the circuit will recognize as a knock.

K All project-specific circuitry is built onto a perf board shield that plugs on top of the Arduino. This lets you unplug the microcontroller for use in other projects.



BONUS SPRAY PAINT STENCILS

Download stencils of this design by Rob Nance at makezine.com/25/gumball.



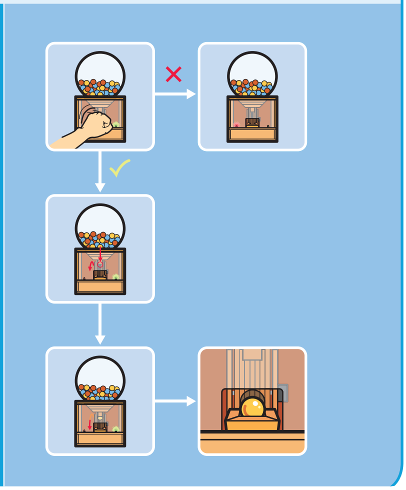
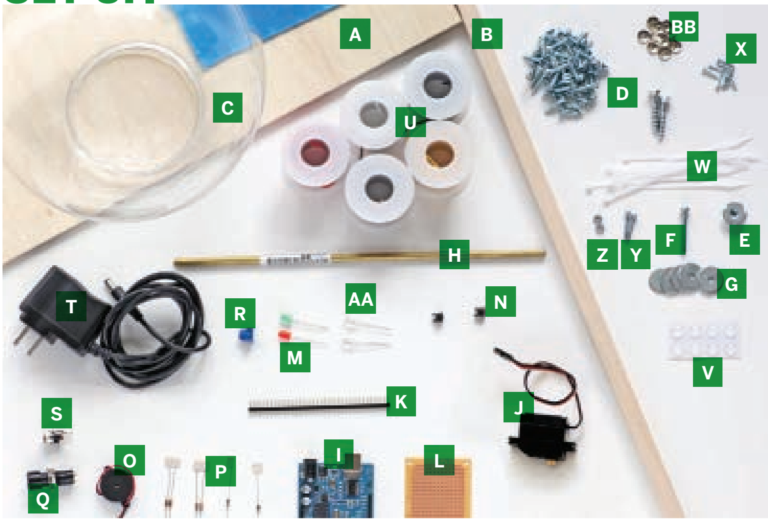


Illustration by Rob Nance

SET UP.



MATERIALS

[A] Plywood, ¼" thick, or ³⁄₁₆" clear acrylic, 2'×4' Use plywood to hide the internal workings or acrylic for maximum visibility. Plywood is much easier to work with than acrylic, which tends to crack, chip, and scuff. To give a view inside with minimal hassle, I cut all my pieces out of plywood except for the 7¾"×5¹⁄₁₆" acrylic access panel in front.

[B] Wood stock, ½" square, 64" long

[C] Clear plastic light globe, 8" diameter with 4" opening item #3202-08020-002 from 1000bulbs.com, $8

[D] Wood screws: #8, ¾" long (65); and #12, 1¼" long (2)

[E] T-nut, ⅜"

[F] Bolt, ⅜", 1¼" long

[G] Washers, ¾" OD, ³⁄₁₆" ID (5)

[H] Brass tube, ⁷⁄₃₂" diameter, at least 2" long

[I] Arduino Uno microcontroller item #MKSP4 from Maker Shed (makershed.com), $35. You can also use an older Arduino Duemilanove or equivalent, but in one test build an Uno worked while the Duemilanove mysteriously did not.

[J] Servomotor, small, high-torque Turnigy S3317M, #TGY-S3317M from hobbyking.com, $10

[K] Male breakaway headers, 40×1 pin #PRT-00116 from SparkFun Electronics (sparkfun.com), $3

[L] Perf board, 0.10" pitch, at least 15×20 holes #276-149 from RadioShack (radioshack.com), $2 (or cut from a larger piece)

[M] LEDs, 5.5mm: red (1), green (1)

[N] LED holders, panel mount (2) #276-0079 from RadioShack, $1 each

[O] Piezo buzzer, 1" diameter #273-059 from RadioShack, $3

[P] Resistors: 100Ω (2), 150Ω (2), 10kΩ (1), 470kΩ (1)

[Q] SPST momentary pushbutton, panel mount #275-609 from RadioShack, $3

[R] Potentiometer, 10kΩ linear trim #COM-09806 from SparkFun, $1

[S] Power jack, 5mm×2.1mm coaxial (size M), panel mount #274-1582 from RadioShack, $3

[T] 9V DC power supply with 5mm×2.1mm center-positive plug #MKSF3 from Maker Shed, $7

[U] Insulated solid-core wire, 22 gauge, multiple colors, around 8' total length plus another 5' if making the transparent case

[V] Stick-on rubber feet (8)

[W] Small cable ties (8)

[X] Wood screws, round head, #2, ½" long (6)

[Y] Machine screws, round head, #4-40, ½" long (2)

[Z] Nuts, #4-40 (2)

[NOT SHOWN]

Gumballs, 1" diameter a standard box of 850

[OPTIONAL FOR VISIBLE INTERIOR]

[AA] LEDs, high brightness white (2)

[BB] Thumbtacks (2)

TOOLS

[NOT SHOWN]

Drill or drill press and assorted drill bits

4" hole saw

1" Forstner (flat bottom) drill bit

Wood saw(s) for cutting straight lines and curves; a table saw and jigsaw, or a handsaw and miter box

Vise

Screwdrivers

Hammer

L-square ruler

Pencil

Masking tape

Pipe cutter or hacksaw

Sandpaper, 80- and 120-grit

Wire cutters and strippers

Needlenose pliers

Thick cardstock or cardboard, 6"–12" square

Soldering iron and solder

Computer with printer and internet connection

USB cable

Photography by Steve Hoefer

MAKE IT.

BUILD YOUR SECRET-KNOCK GUMBALL MACHINE

START **Time:** A Weekend **Complexity:** Moderate

1. BUILD THE CASE

1a. Download the templates from makezine.com/25/gumball and print them at full size. Transfer to plywood or acrylic, and cut all pieces to size. Mark all screw and rail locations for later drilling and placement. To avoid confusing or flipping parts, temporarily mark the outside of each part with masking tape. Sand all edges smooth.

Use a 4" hole saw to cut the hole in the top plate as well as the 7 dispensing wheel disks (2 stir plates and 5 center disks). You can use the remnant of the top plate hole as one of the dispensing wheel disks. Drill the ¼" axle holes (which are off-center on the stir plates) as shown on the template.

TIP: To minimize chipping when working with acrylic, support the work well, tape both sides of all cuts, and use fine-bladed saws at low speed — or have your local plastics retailer cut the pieces.

1b. Attach 3 rails to each side panel where indicated on the template. The front and back rails are 3/16" from the edge. The 2 side pieces should mirror each other, with rails on the inside of the case.

NOTE: Unless otherwise specified, all attachments in this project are made with #8 ¾" wood screws, and holes should be pilot-drilled to prevent splitting.

1c. Attach the bottom panel and front panel to the side panels.

1d. Drill holes for the power plug and programming switch in the back panel, then attach it to the side panels.

1e. Test the knock plate, top panel, and access panel for fit, and pre-drill the holes, but don't attach them yet.

2. BUILD THE GUMBALL FUNNEL

2a. Attach the top and bottom rails to each funnel end piece.

2b. Attach the funnel side pieces to the funnel ends. Be sure to put the rails on the outside of the funnel.

2c. Attach the completed funnel to the bottom of the top plate with 4 screws.

3. BUILD THE DISPENSING WHEEL

3a. Make a sandwich of round plates: the 5 dispensing wheel plates stacked in the middle and one stir plate on each end, with the stir plates rotated 30° from each other. Push the brass tube through the axle holes and use an L-square to check for true.

Clamp, pre-drill, and secure the whole stack with a #12 1¼" wood screw on each side.

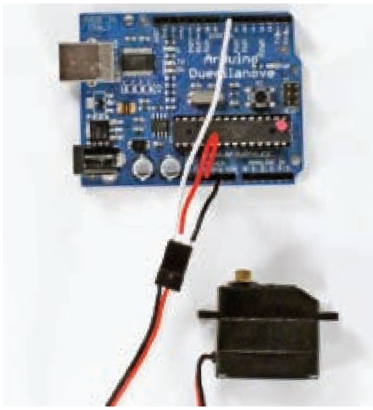
3b. Clamp the dispensing wheel (in a drill press if available) and drill a 1" deep hole using a 1" Forstner bit. The hole should point directly down toward the axle, centered between the stir plates and equidistant between their peaks. To help gauge the right hole depth, place a masking tape flag 1" from the bottom of the drill bit.

Toss a 1" gumball in the hole to check for fit. It should fall in and out easily, and the top of the gumball should sit flush with the wheel.

3c. Attach the servo horn (the X-shaped plastic piece that connects to the servo shaft) to the center of the wheel using four ½" #2 screws. Space the screws so that they don't intrude into the 1" hole drilled in the previous step. To center the horn, sight its drive-shaft mounting hole through the axle hole of the dispensing wheel.

4. MOUNT THE DISPENSING WHEEL

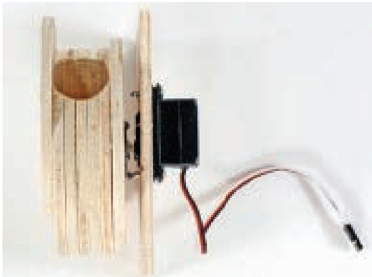
4a. Before attaching the servo, we need to zero its rotational position using the Arduino. Download and install the latest version of the Arduino application from arduino.cc/en/Main/Software, and attach the Arduino to your computer with the USB cable. Use 3 solid-core wire leads to temporarily connect the servo's red wire to Arduino +5V, the servo's black (or brown) wire to Arduino ground (GND), and the servo's white (or orange) wire to Arduino Digital Out 7.



4b. Download *servo_reset.pde* from makezine.com/25/gumball, open it in the Arduino application, and upload it to the microcontroller. The servo should rotate fully one way, then the other, then stop.

4c. Remove the wires connecting the servo to the Arduino. Use the servo's supplied mounting screws to attach it to the dispensing wheel right side plate, with the servo shaft centered to line up with the 7/32" center hole in the left side plate. Before screwing, fit the included rubber bushings and metal grommets into the servo mounting holes to provide strain relief, with the curved edges of the grommets facing the wood. It's OK if the screw points run out the other side of the plate.

4d. Orient the dispensing wheel next to the right side plate so that the gumball hole tilts back 45° toward what will be the back of the machine. Slide the servo horn over the servo shaft and secure it with the included servo horn screw that runs through the center of the horn.

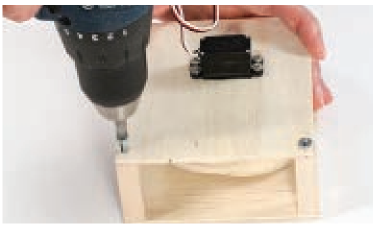


4e. Use a hacksaw or pipe cutter to cut a 1 7/16" length of brass tube for the axle. Insert the tube into the dispensing wheel axle hole. It should be flush with the stir plates.

4f. Hammer the T-nut firmly into the hole on the dispensing wheel left side plate, on the outside (not the wheel side).

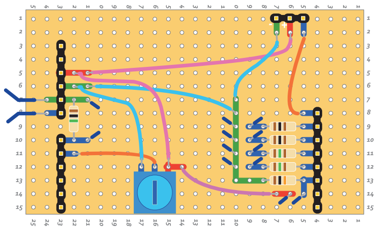
4g. Place a washer on the 3/16"×1¼" bolt, thread it entirely through the T-nut, and tighten the bolt firmly. Place three 3/16"×¾" washers on the inside of the bolt.

4h. Insert the end of the bolt into the dispensing wheel axle. Attach the left and right side plates to each other with the 2" dispensing wheel rails. If there's a lot of play between the T-nut and the dispensing wheel, add an additional washer before screwing the plates together.



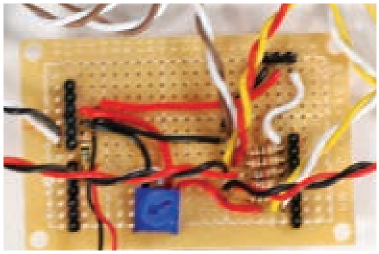
5. ASSEMBLE AND TEST THE ELECTRONICS

For all connections, refer to the schematic and circuit board layout at makezine.com/25/gumball.



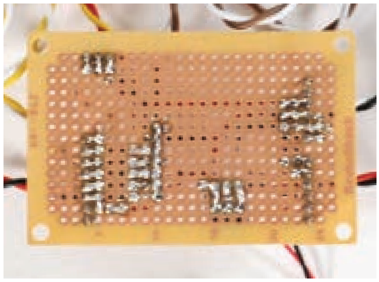
5a. Cut and strip leads for all off-board components. You'll need four 5" leads for the indicator LEDs and four 14" leads for the power connector and programming button. If you're making a transparent case, you'll also need two 12" leads and two 16" leads for the white LEDs.

5b. Cut the breakaway header pins into rows of 8, 6, 6, and 3. On all but the 3-pin header, use needlenose pliers to push the pins flush with the plastic spacer.

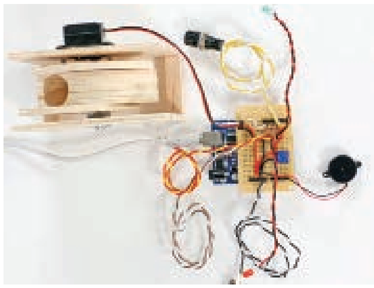


5c. Solder the headers in place on the perf board, following the layout diagram *SNGM_layout.pdf*. The 3-pin header's long pins should point up, for connecting the servo, while the other header pins should run down through the bottom of the board for plugging into the Arduino.

5d. Solder the rest of the components to the perf board following the layout or schematic diagram *SNGM_schematic.pdf*. Pay special attention to the polarity of the LEDs (longer leg is positive) and power connector (center pin is positive). Omit the 2 white LEDs and 100Ω resistors if you're not making a transparent case.



5e. Plug the completed shield into the top of the Arduino, aligning the 8-pin header with digital pins 0–7. Plug the servo into the 3-pin header on the shield, taking care to observe the polarity (because plugging the servo in backward can damage it).



5f. Connect the Arduino to your computer with a USB cable. Download the code file *secret_knock_gumball_machine.pde* from makezine.com/25/gumball and upload it to the Arduino.

After you upload the sketch (or apply power for the first time) all the LEDs should light, and the servo should rotate the dispensing wheel to its start location. When the lights go out, it's ready to listen for a knock.

5g. Turn the sensitivity potentiometer down until the green LED stops blinking. If the light stays on no matter how far you turn it, check the piezo sensor wiring for shorts or bad connections.

5h. For testing, adjust the pot so that the green light stops blinking without any input, but gently tapping the knock sensor makes it blink.

Tap the "Shave and a Haircut" rhythm. The green LED should blink and the servo should rotate. If you included the white LEDs, #1 should blink and #2 should stay lit for a few seconds. If an incorrect rhythm is entered, the red LED should blink.

Press the programming button, and both red and green LEDs should light. Tap a new rhythm, release the button, and wait a second. The red and green lights should echo the rhythm visually, and tapping this new rhythm should rotate the servo. Resetting or unplugging the Arduino will revert it to the default "Shave and a Haircut."

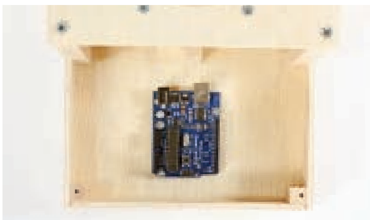
6. WIRE AND COMPLETE THE CASE

6a. Secure the top panel to the rest of the case with 4 screws. Align the panel so the offset 4" hole is toward the back of the device.

6b. Secure the dispensing wheel assembly to the bottom panel with 2 screws. Rotate the dispensing wheel to check for clearance against the back panel and the funnel, and remove material wherever it hits the wheel.

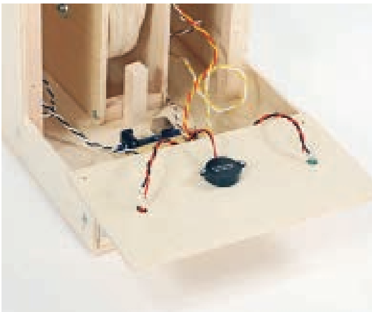
6c. Attach the gumball tray support directly in front of the dispensing wheel, with the support's angled end on top, and inclining down toward the front of the case.

6d. Unplug the shield from the Arduino. Attach 4 rubber feet to its underside and use two 4-40 bolts and nuts to attach it to the bottom panel. Insert the screws up from the bottom through the pre-drilled holes, and thread the nuts on top. Do not overtighten the nuts.



6e. Attach the knock sensor to the center back of the knock plate using two ½" #2 screws. The piezo buzzer should fully contact the back of the knock plate, and the screws shouldn't penetrate through to the other side of the wood. I attached the sensor "upside down" with its mounting tabs away from the wood to keep the screws from running through, but the sensor will function facing either way.

6f. Insert the LED holders into the front of the knock plate and slide the red and green LEDs into the holders from the back.



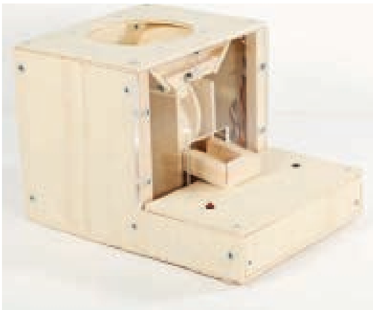
6g. Attach the power plug and programming button to the back of the case with their mounting hardware.

6h. If you're making a transparent case, attach the white LEDs to illuminate the action. Tighten a small cable tie around the base of each white LED and secure it in place with a thumbtack. Place LED #1 behind the right side pillar pointing inward, and LED #2 on the bottom front of the dispensing funnel, pointing to the gumball tray. Use the remaining cable ties to secure and manage the cables inside the case.

6i. Screw the knock panel into place.

6j. Screw the gumball tray sides to the gumball tray, and screw the tray down onto the tray support post. Before attaching it permanently, check for gumball clearance against the dispensing wheel and the access panel.

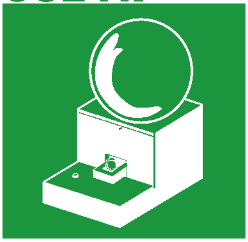
6k. Stick a rubber foot on each bottom corner to prevent the machine from sliding around as people knock. Finally, attach the access panel. You're done!



FINISH X

NOW GO USE IT »

USE IT.



KNOCK YOURSELF OUT!

MACHINE SETUP

1. Plug the power adapter into a handy outlet and plug the other end into the back of the machine.

2. Fill the plastic globe with your favorite 1" gumballs (or other candy balls) and put it on top. To prevent spilling the balls all over the floor, put a piece of thick cardstock or cardboard over the hole in the globe, invert it in place, then slide the card out.

3. Program your new secret knock or leave it with the standard "Shave and a Haircut" knock.

4. Knock and enjoy!

TIP: To make the machine more responsive, loosen the screws that hold down the knock plate and gumball tray, so that they move and clatter when you knock.

TREAT SELECTION

The 1" ball is a vending industry standard for not only gum, but also jawbreakers and other candies, bouncy balls, toy capsules, and empty capsules that you can fill with your own small objects.

An 8" globe holds about 200 gumballs, which are usually sold in quantities of 850 — but you can also buy 1" gumballs by the pound at bulk candy stores, and there are sellers on eBay that offer them in smaller quantities. I also found a box of 500 gumballs at Smart & Final.

+ Visit makezine.com/25/gumball for project templates, Arduino code, a schematic diagram, and a layout/wiring diagram.

Photograph by Liz Smith (bottom)

SOUS VIDE IMMERSION COOKER

By Scott Heimendinger

COLLABORATE ON
Make: Projects
makeprojects.com

PRECISION GOURMET

I'm fascinated by sous vide cooking, in which foods vacuum-sealed in plastic are immersed in a precisely temperature-controlled hot water bath to achieve optimal doneness.

But most sous vide (soo-veed) cooking machines are commercial models that cost north of $2,000, and the first "home" version, the countertop SousVide Supreme, is priced in the neighborhood of $450 (not including vacuum sealer), which is still a steep investment for something that essentially keeps water warm.

I decided to build a better device on the cheap. Behold, the $75 DIY Sous Vide Heating Immersion Circulator! By scrapping together parts from eBay and Amazon, I created a portable device that heats and circulates water while maintaining a temperature accurate within 0.1°C. And unlike the SousVide Supreme, it mounts easily onto larger containers, up to about 15 gallons, for greater cooking capacity.

The water is heated by three small immersion heaters and circulated by an aquarium pump to keep the temperature uniform. An industrial process temperature module controls the heaters, and an eye bolt lets you clamp the entire apparatus to the rim of a plastic tub or other container.

To cook sous vide, you also need a vacuum sealer, which this project does not include. I bought a good one new for about $112.

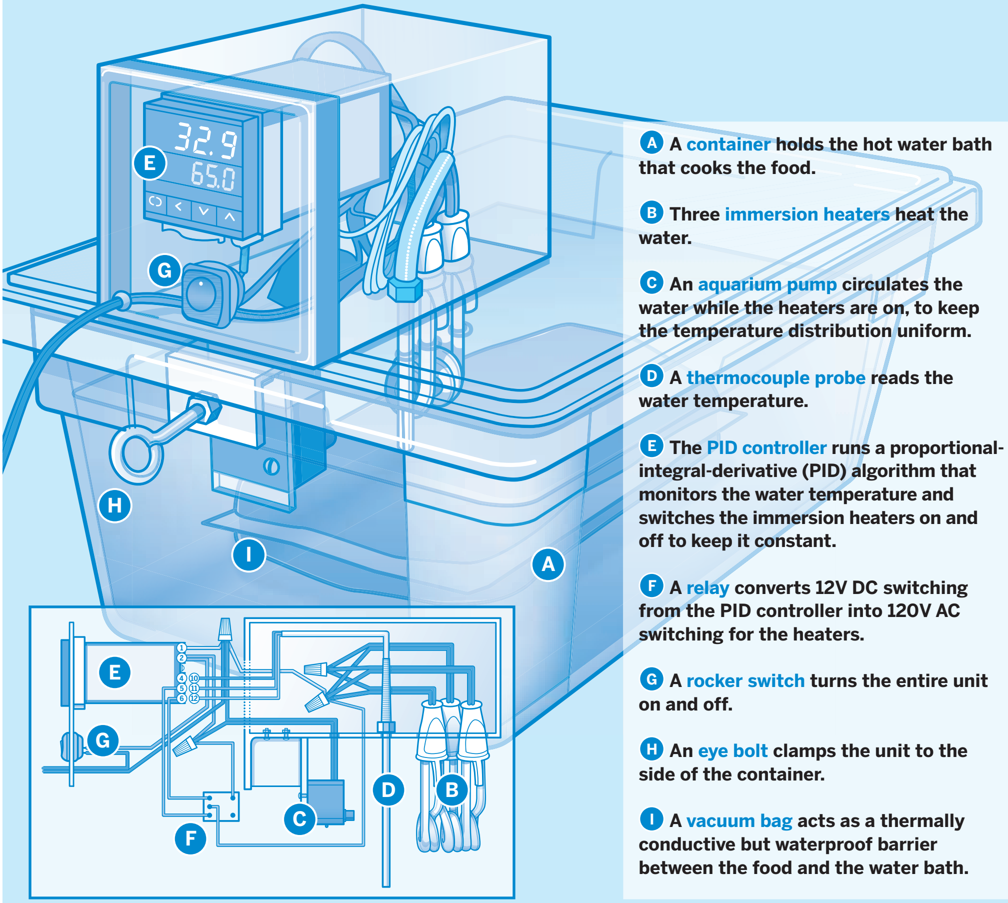
Set up: p.107 **Make it: p.108** **Use it: p.115**

Scott Heimendinger runs the blog SeattleFoodGeek.com where he offers tips for geeky recipes and projects.

Photography by Scott Heimendinger

Slow Food

Cooking sous vide ("under vacuum" in French) means vacuum-sealing ingredients in plastic and immersing them in a relatively low-temperature bath (typically around 140°F/60°C) for long periods of time. The process retains cell structure and juices that foods lose through other cooking methods.



A A **container** holds the hot water bath that cooks the food.

B Three **immersion heaters** heat the water.

C An **aquarium pump** circulates the water while the heaters are on, to keep the temperature distribution uniform.

D A **thermocouple probe** reads the water temperature.

E The **PID controller** runs a proportional-integral-derivative (PID) algorithm that monitors the water temperature and switches the immersion heaters on and off to keep it constant.

F A **relay** converts 12V DC switching from the PID controller into 120V AC switching for the heaters.

G A **rocker switch** turns the entire unit on and off.

H An **eye bolt** clamps the unit to the side of the container.

I A **vacuum bag** acts as a thermally conductive but waterproof barrier between the food and the water bath.

PID Control

A PID temperature controller is like an advanced thermostat. Regular thermostats click on and off when the measured temperature passes fixed thresholds, and the resulting temperature oscillates around the thresholds. This creates an opportunity for temperature "carry-over" in the food, like the way meat's internal temperature can rise after you take it out of the oven.

In contrast, PID controllers use the PID algorithm, common in industrial control applications, to track temperature changes and calculate how much ON time will raise the measured temperature (the process variable) to the target (the set point) asymptotically, rather than overshooting it.

By predicting, mathematically, the impact that turning on the heaters for 1 second will have on overall water temperature, the PID controller enables extremely precise temperature control and stability over long periods of time.

NOTE: Be aware of a minor technical nuance — the model CD101 PID controller used in this project is accurate when used in Celsius mode. For unknown reasons, people have experienced inaccurate readings and fluctuating temperatures when the controller is set to Fahrenheit.

Illustration by Damien Scogin

SET UP.

MATERIALS

[A] Digital PID temperature controller, model CD101 with PT100 support and voltage pulse output Some model CD101 units will not work; you need to make sure that the 5th character of its code (on a sticker on the side) is V, for voltage pulse output. I bought mine (code FK02-VM*AN-NN) for $39 from eBay, or see Sure Electronics #RDC-TE11113 (sureelectronics.com), $33.

This controller has 12 screw terminals. Posts 1 and 2 connect to AC power, 5 and 6 control a relay for the heater, and 10–12 connect to a thermocouple. If you use a different controller, refer to its datasheet for the functionally equivalent connection locations.

[B] Clear acrylic storage container, 7"×4"×4" item #B000NE80GO from Amazon (amazon.com), $5

[C] Immersion heaters, Norpro 559 (3) Amazon #B000I8VE68, $6 each

[D] SPST rocker switch, heavy duty #275-693 from RadioShack (radioshack.com), $3

[E] SPDT mini relay, 7V–9V DC, 12A RadioShack #275-005, $4. You may want to upgrade to a solid-state relay for a more robust, reliable build. The wiring is the same for an SSR as it is for the relay included in the build instructions. A good option is Lightobject #ESSR-25DAC, $14.

[F] Aquarium pump with suction cup "feet" Catalina Aquarium #A801 (catalinaaquarium.com), $10

[G] Clear acrylic sheet (plexiglass), 7"×2"×1/4" thick From hardware stores or online vendors; you may have to cut it to size with a band saw, table saw, fine-toothed hacksaw, or jigsaw.

[H] Eye bolt, 1/4"×2", with nut

[I] Machine screws, #4-40, 1/2" long, stainless steel, with matching nuts (2)

[J] Sheet metal screws, #6-40, 3/8" long, stainless steel (2)

[K] Super glue

[L] Waterproof silicone caulk aka tub and tile caulk

[M] Wire, 22 gauge, stranded, 1'

[N] Wire, 14 gauge, solid core, 4'

[O] Wire nuts, large (4)

[P] Electrical tape

[Q] Thermocouple temperature sensor probe with 3 leads #PT100 from Virtual Village (virtualvillage.com), or from eBay, $6

[NOT SHOWN]

Large, straight-sided container to hold the water. Because the bath doesn't get very hot or touch the food, plastic is OK; I use a 17qt plastic storage bin I bought for $6 at Bed Bath & Beyond (bedbathandbeyond.com).

TOOLS

[S] Phillips screwdriver

[T] Wire cutters and strippers

[U] Multimeter

[V] Soldering iron and solder

[W] Hot glue gun

[X] Rotary cutting tool with router bit such as a Dremel

[Y] Hobby knife

[Z] Drill and drill bits: 1/8", 5/32"

[NOT SHOWN]

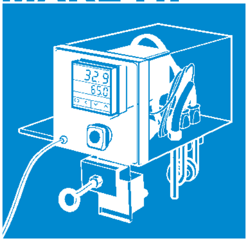
CNC laser cutter (optional)

Marker

Stove and oven mitt

Bowl of cold water

MAKE IT.



BUILD YOUR SOUS VIDE IMMERSION COOKER

START **Time:** 6 Hours Over 2 Days **Complexity:** Easy

1. CUT THE ACRYLIC ENCLOSURE

This is the most difficult part of the project. For your cooker to be sturdy, water-resistant, and decent looking, the mounting holes must be cut precisely. I used a CNC laser cutter I have access to at work, but with a steady hand you can achieve the same results using a rotary tool like a Dremel.

1a. Download and print the cutting template in *Cutouts.pdf* from makezine.com/25/sousvide. This template matches the heaters, controller, and switches I used, so you'll need to adjust the shapes and sizes if you use different parts.

1b. Following the template, cut out the 3 holes for the immersion heaters on one side of the acrylic container, near its base.

1c. Mark and cut the small oval-shaped hole for the water pump power cord and a circular hole for the thermocouple.

1d. Follow page 2 of the template to mark and cut openings in the lid of the container for the controller, switch, and power cord.

2. MOUNT THE IMMERSION HEATERS

2a. Cut the power cord off each heater, leaving an 8" tail of wires from each coil end.

2b. Using your hobby knife, scrape the flat sides of the heater handles to remove the lettering and flatten the circular rim where it meets the cord.

2c. Fit all 3 heaters into their holes in the container, such that the coils stick out and line up. Your finger should be able to fit through all 3 coils at once. Trim the holes if needed to make the heater handles fit snugly.

2d. Caulk a fully waterproof seal around the heater handles on the outside of the enclosure and let dry overnight.

For added strength, add hot glue over the caulk after it's dried. The hot glue only needs 5 minutes to dry before continuing.

3. ATTACH THE MOUNTING BRACKET

3a. Cut a 7"×2" rectangle out of ¼" acrylic. The cut sides don't have to be perfect as long as the rectangle dimensions are approximate. Use a rotary cutter to rout a shallow recess that's the same shape and size as the nut for your eye bolt, centered at one end, as shown on page 3 of the template. Drill a ¼" hole through the center of the recess for the eye bolt to pass through.

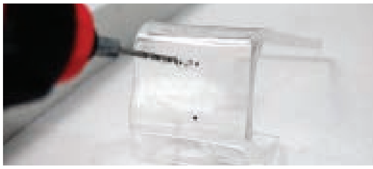
3b. It's time to bend the acrylic sheet. Clear a countertop near your stove, put a bowl of cold water within reach, and mark the bend point lines from the template onto the acrylic.

NOTE: If your countertop is too thick or thin for this process, you'll need to bend the sheet around a hard, stable object, approximately 2" thick, such as a 2×4.

Turn one of the front burners on high, hold the undrilled end of the acrylic with an oven mitt, and place the drilled end with bend lines a few inches above the burner, moving the acrylic around and turning it to heat both sides. If it begins to form small bubbles, move it away from the heat a little. When the acrylic starts to curl away from the heat, it's ready to bend.

3c. With the recess for the nut facing outward, bend the acrylic around the edge of the counter (or other object) approximately along the marked lines to form a "J." While in place, press the middle segment of the J against the edge of the countertop to make it flat. Immediately drop the acrylic into cool water so that it holds its form.

3d. Position the middle segment of the J bracket against the heater side of the enclosure, with the bracket's nut side aligned along the enclosure's rim. This is how you'll bolt these 2 pieces together. Mark and drill two ⅛" holes through the bracket and 2 matching holes in the enclosure. Check that the 4-40 machine screws fit through both pairs of holes.



3e. Liberally apply super glue to the underside of the bracket, and bolt it to the enclosure with the machine screws and nuts.

3f. Liberally apply more super glue to the recessed area for the eye bolt nut, and set the nut in place. Reinforce the bond with hot glue, but don't obstruct the hole in the nut.

For more support, you can cut a 1½" acrylic round with a hole fitting the nut, and glue it to the J-clamp overlapping the opening. This will provide extra gluing surface and take stress off the nut.

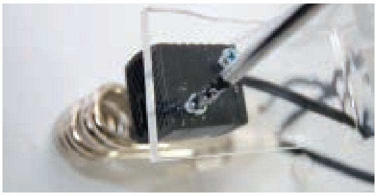
4. ATTACH THE PUMP

4a. Cut the water pump's power cord about 8" from the pump. Reserve the severed cord, which will become the cooker's power cord.

4b. Position the pump on the long side of the bracket, with its cord pointing toward the enclosure and its top aligned with the top of the heater coils. The pump should be placed so its outlet is more or less pointing through the coils. Mark the locations of the suction cup feet on the bracket.

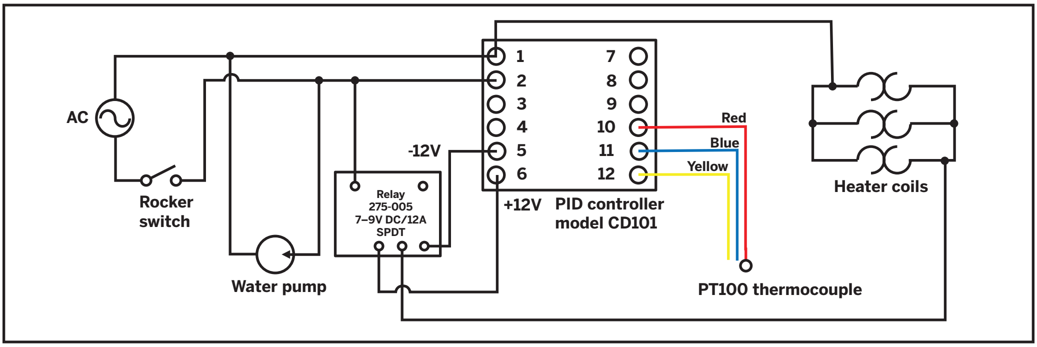
4c. Drill two 5⁄32" holes through the bracket at the exact centers of the suction cup locations. Check that the holes can accommodate the sheet metal screws, and enlarge them if needed by wiggling the drill bit.

Remove the suction cups from the pump, and attach the pump to the bracket with the sheet metal screws in the suction cup holes.



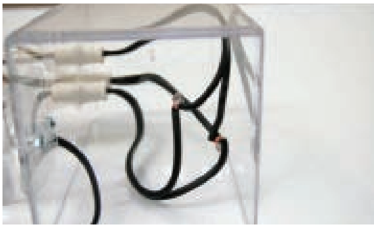
5. COMPLETE THE WIRING AND ASSEMBLY

For the big picture, refer to this wiring diagram.

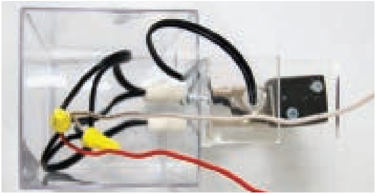


CAUTION: Don't ever power on the heater coils unless they are submerged in water! (They'll burn out.) And if you're unfamiliar working with AC power, seek assistance from someone who's knowledgable.

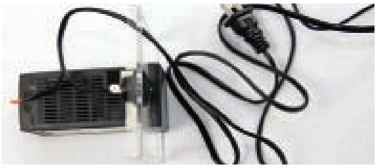
5a. Split the heater cords apart and strip about 2" of shielding away from each end. Twist and solder together 1 lead from each of the heaters, then repeat for the other set of leads to wire the 3 coils in parallel.



5b. Cut two 18" 14-gauge wires and strip ⅜" of shielding from each end. Use wire nuts to bundle one end of each wire with a set of heater leads. One wire will connect to AC power, and the other will connect to the relay.

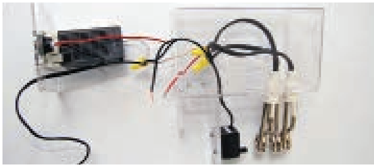


5c. Remove the mounting ring from the PID controller, fit its front panel out the opening you cut for it in the acrylic lid, and secure the controller from the back by replacing the mounting ring.



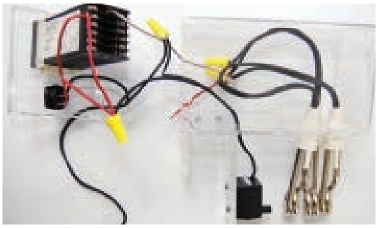
5d. Fit the power switch into its hole in the acrylic lid and secure it with its included mounting nut. Run the power cord you cut from the water pump through its hole nearby. Separate its leads, strip ¼" of shielding, and solder 1 lead to 1 leg of the power switch.

5e. Cut and strip 3 short (about 4") lengths of 14-gauge wire. Twist-bundle 1 wire with the free power cord lead, 1 lead from the water pump, and 1 of the 14-gauge leads from the heaters. Secure the bundle with a wire nut, and screw the short wire to terminal #1 on the controller.

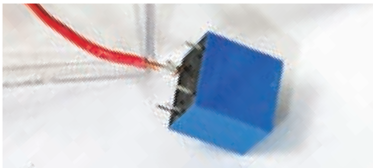


5f. Solder the other 4" piece of 14-gauge wire to the unconnected leg of the power switch, and then screw the other end to the #2 terminal on the controller.

5g. Make a new bundle connecting the free lead from the water pump, the wire from the #2 terminal, and the third short length of 14-gauge wire.

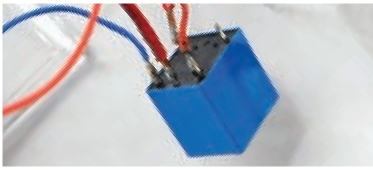


5h. Orient the relay so that its underside (with the pins) faces you, and the row with 3 pins is facing down. Solder the free lead from the previous step to the upper left pin of the relay.



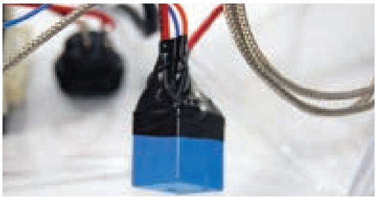
5i. Solder the free lead from the heater bundle to the middle bottom pin of the relay.

5j. Cut and strip two 6" lengths of 22-gauge wire. Solder one to the bottom right pin of the relay and connect it to the #5 terminal on the controller. Solder the other piece to the relay's bottom left pin and connect it to the #6 terminal.

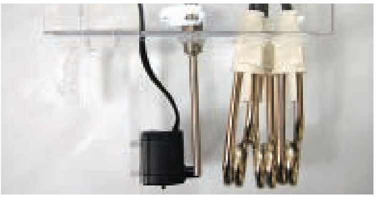


5k. Insulate and strengthen the relay connections with a liberal application of hot glue. Wrap the relay and wires with electrical tape.

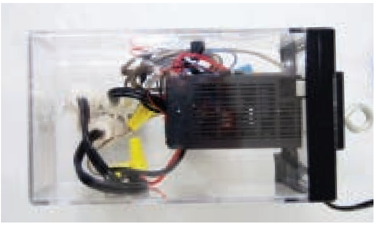
5l. Apply hot glue around the holes for the power cord and water pump cord, to prevent steam or water from leaking into the enclosure.



5m. Insert the PT100 thermocouple into its hole in the enclosure and secure with hot glue. Connect its red lead to the controller's #10 pin, its blue lead to the #11 pin, and the yellow lead to the #12 pin.



5n. Put the acrylic lid on the open end of the container, and apply a neat strip of electrical tape around the edges to hold it on. (Once you've verified that the cooker works, you can then glue the seam.) Insert the eye bolt through the nut.



5o. Everything is wired up and assembled. Now you probably want to see if it works. BUT WAIT! Don't turn the machine on (ever!) unless the coils are submerged in water or else they'll burn out in about 5 seconds. I learned this the hard way.

5p. Fill a container with enough water to cover the top of the circular part of the heater coils by ¼"–½". Mount the cooker inside with its bracket hanging over the lip, and tighten the eye bolt to secure.

Plug in the cord and flip on the power switch. If the PID controller turns on and the pump starts pumping, that's a good sign. Note that the heaters may not warm up yet, depending on what the controller's default target temperature is.

CAUTION: 120V AC current is dangerous and misuse can injure or kill. Seek the assistance of a qualified electrician if needed. Take care to keep the control box and wires out of the water or other liquids. And once you've tested your cooker, seal the lid on the control box.

6. PROGRAM THE CONTROLLER AND TEST

For the model CD101 PID controller manual and a full sequence of photos showing how to program it for this project, visit makezine.com/25/sousvide.

6a. The PID controller defaults to a different thermocouple than the PT100 used here, so you need to reconfigure it for a PT100 sensor. Press and hold the SET button until `AL1` appears, then keep pressing SET until you see `LCK` (lock) with `1000` underneath. To unlock the settings, we need to change this value.

6b. Use the left-arrow key to move the cursor and the up-arrow key to change `LCK` to `1100`. Then press SET again to exit the menu.

6c. Push and hold the SET and left-arrow keys together until you see `COD` (code). Use the arrow keys as before to set this value to `0000`, then press SET again to see `SL1`, the sensor type symbol.

6d. Change `SL1` to `1100`, for the PT100 sensor. Press and hold the SET and left-arrow keys again to exit. Then re-lock the settings by setting the value of `LCK` back to `1000`, as in Steps 6a–6b.

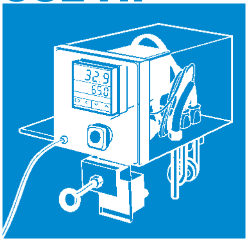
6e. It's time to heat things up. The top line of the display (green) shows the current temperature, and the second line (orange) shows the target. Set a target temp by tapping the SET button, using the up and down arrows to specify the target, and pressing SET again to confirm. 50°C is good for testing.

If all's well, the OUT1 light will turn on and you'll hear a soft clicking as the relay turns on the heaters. The temperature reading will increase, and you'll hear more frequent clicking as it approaches the target.

FINISH X

NOW GO USE IT »

USE IT.



MAKE THE ULTIMATE STEAK 'N' EGGS

VACUUM SEALING

A good vacuum seal is essential to sous vide. Without air pockets, the heat transfers into the food evenly and efficiently. A good seal also dramatically decreases the risk of bacterial contamination from long incubation at low temperatures.

I use a FoodSaver V3835, which I bought for just over $100 with a 20% coupon at Bed Bath & Beyond. It uses rolls of plastic, so you can make the bags any size you need. I recommend against using cheaper handheld sealers. I've tried a few and had problems creating a reliable seal.

With any sealer, always wipe clean and dry the end you're going to seal, or the plastic layers might not melt together properly.

Using a zip-lock bag instead of a vacuum sealer is actually OK for cooking times up to 4 hours, but also riskier for leaking and contamination.

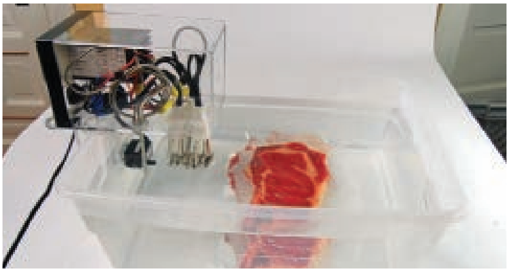
KITCHEN TESTS

To reveal the power of sous vide, cook an egg in the shell (no plastic needed) at 64.5°C for 1 hour. This yields an amazing transformation: perfectly soft whites, not runny or rubbery, and a yolk with the consistency of a rich pudding. It's impossible to achieve this through any other cooking method, and it's spectacular the first time you experience it.

I expanded on this amazing transformation by breading the yolks and quickly deep-frying them to add a crunchy shell. See my recipe at makezine.com/25/sousvide.

Next, try cooking a good, thick steak. Unlike conventional cooking methods, sous vide gives you a perfect medium-rare steak all the way through. To finish the steak, pat all sides dry and sear like crazy with a propane torch (the kind plumbers use).

Some things that don't work well sous vide, in my experience, are broccoli, kiwi, and strong aromatics. Note also that if you sous vide with too much garlic, your whole house will smell like feet.



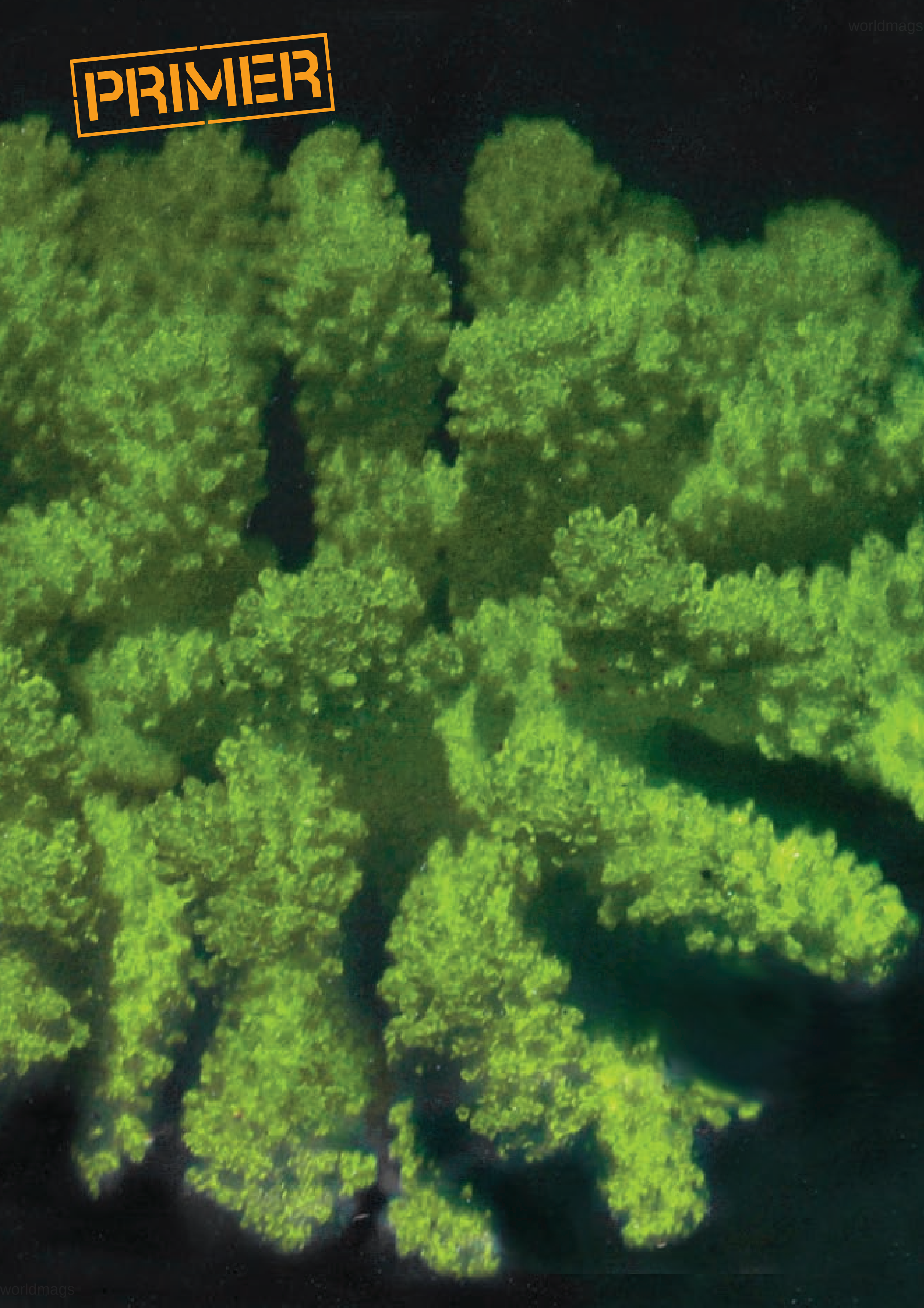
GENERAL-PURPOSE TEMP CONTROLLER MOD

Several people have modified this project so that the relay switches a power outlet that the heaters and pump plug into, and the probe plugs into an added jack. Then, if a heater burns out, you plug in another one rather than having to mess with glue and solder. Modularizing the control and power components this way also lets the machine double as a precise temperature controller for a smoker box, such as Alton Brown's DIY flower-pot smoker (see makezine.com/25/sousvide). You just plug in the hotplate filled with wood chips.

Pick up a few steaks and break out your blow torch — it's time to start cooking like a geek!

+ For hole templates, the CD101 PID controller programming instructions and manual, Scott's Deep-Fried Sous Vide Egg Yolks recipe, and other resources, visit makezine.com/25/sousvide.

PRIMER



HOLO-GRAPHY

Capture magical 3D images without a camera.

By Frank DeFreitas

Holograms have fascinated both scientists and artists since the 1960s with their ability to re-create three dimensions. Back then, the equipment needed to produce holograms was cumbersome and expensive, but today you can create them using $5 laser pointers and other inexpensive and readily available materials.

Most light sources emit a mishmash of wavelengths at different phases, but the coherent light from a laser is at a single wavelength and all aligned in the same phase. When laser light strikes a motionless object, its reflections are just as unchanging as its source — and the interference between the source and reflected waves produces a regular interference pattern, just as regular wave patterns form around objects in a wave tank.

This pattern is the key to making holograms, and it's easy to create.

Photograph by Sam Murphy

When you position a film plate near the object, it captures these wave patterns as they exist in a thin slice of space. Where the reference beam (from the laser) and the object beam (aka signal beam, reflected from the object) combine additively, the developed film image reflects or transmits more light, and where the 2 waves cancel, the image reflects or transmits less.

Take the object away, and the film does the same thing to light that the object originally did, when you illuminate it at the same wavelength from the same angle. This film plate is a hologram, and they're surprisingly easy to make.

In this tutorial you'll learn to do *inline holography*, shining the laser directly through the film plate onto the object. In addition to some basic equipment, you'll need darkness and stillness. Holography's long exposure times allow ambient light to wash out the film image, and even the slightest vibrations during exposure — movements measuring a quarter of the laser light's wavelength (around 140 nanometers) — will jostle the interference pattern enough to blur or erase the film image.

THE BASIC SETUP

Our setup has 3 stations: the *laser* supplies the light to record the hologram; the *spreading lens* spreads the laser beam so it covers both the hologram plate and the object to be recorded; and the *stage* holds both the plate and the object.

The laser and lens are secured to heavy steel pipes to reduce vibration, and the stage consists of a concrete paver sitting atop 3 Sorbothane tripod feet, to reduce vibration even more (Figure A).

1. FIND AND PREPARE YOUR WORKSPACE.

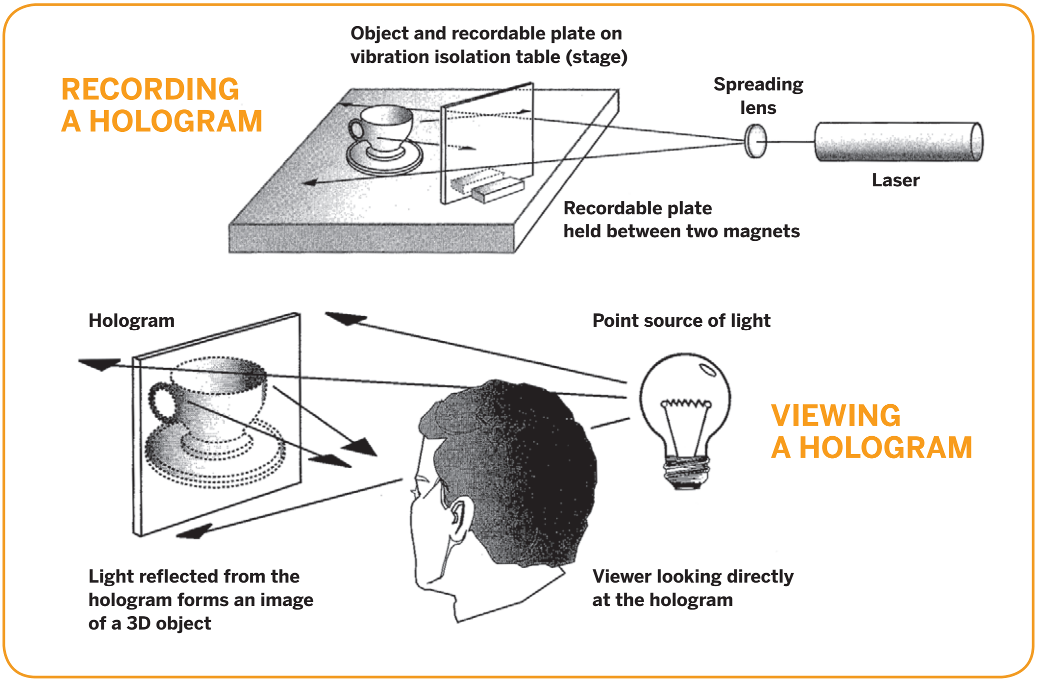
Ideally, you can set up on a concrete floor in a basement or garage. If you must use an upper floor, set up next to a wall instead of in the middle of the room where the floor is less stable. If you use a table, make sure it doesn't wobble. Avoid air currents from heating or air conditioning ducts.

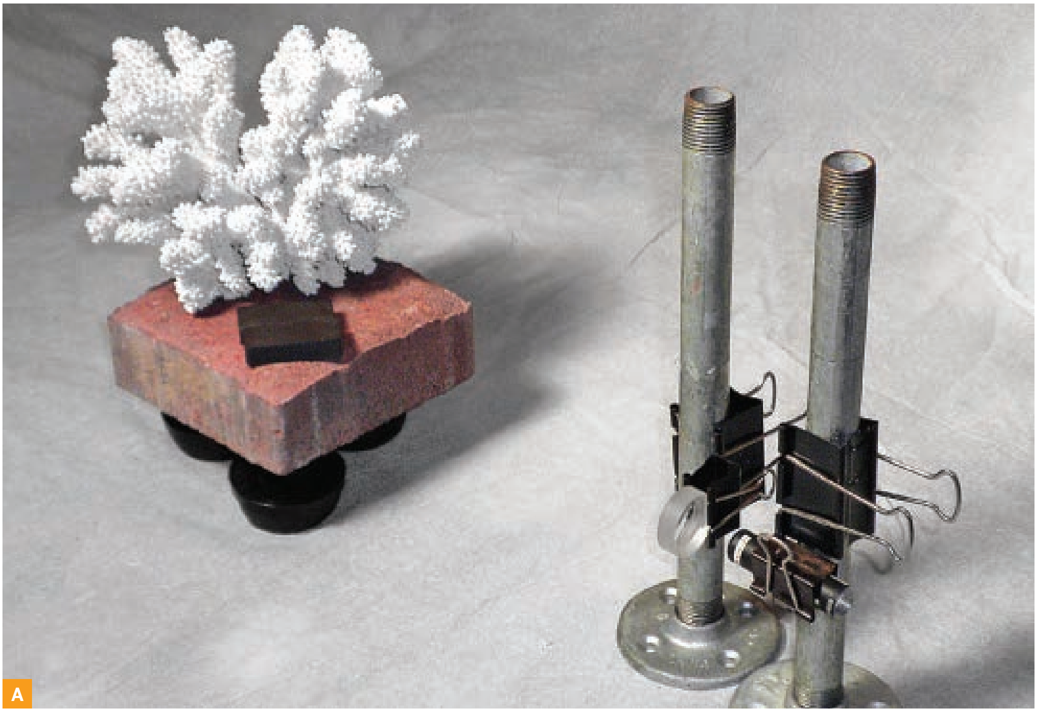
Sound is vibration, and vibration ruins holograms, so your area must be quiet. No radios, no television, no one jumping around the house during the exposures. Exposure times are less than 30 seconds, so try to have everyone cooperate for this time.

MATERIALS AND TOOLS

Object to capture Use something light-colored, smaller than the holographic plates you use, and solid enough not to move with air currents. Coral, porcelain figurines, ceramics, pewter, and metal car models are classics, and 3D printing offers a limitless new source. Items such as feathers, flowers, or fabric won't work (unless you use a high-powered pulse laser).
Holographic plates, red-sensitive, PFG-03M For beginners, I recommend 2½"-square plates such as item #S3P-06306 from Integraf (holokits.com), $27 for 6
Processing chemistry kit: JD-4 hologram developer Integraf #JD4, $17
Distilled water
Laser pointer, red Any red pointer should work.
Batteries for laser pointer. These should be fresh. Internal batteries last about 2 hours total, so for longer life and more reliable power, connect 2 AA batteries in a AA holder, listed as optional below.
Double-concave (DCV) lens item #AX27490 from Anchor Optics (anchoroptics.com), $10. Keep the lens clean and free of dust, fingerprints, etc.
Sorbothane hemisphere foot mounts (3) Anchor Optics #AX35264, $17 each
Card or cardboard rectangle, black, about 3"×5"
Magnets, flat, rectangular (2)
Concrete paver or stepping stone, 8"×8"×2" or similar, from a home center
Plumbing pipe nipples, galvanized steel, ½"×8" (2)
Floor flanges for ½" plumbing pipe (2)
Leviton Glow Guide nightlight, green from hardware stores or online; sold in 2-packs for around $8
Binder clips, 1" (2) and 2" (2)
Plastic containers, small must accommodate the processing fluids and the size of your holographic plate
Masking tape
Construction paper, dark
Measuring cup
Tongs or spring clothespin
Rubber gloves
Goggles
Apron
Paper towels
Spray paint, flat black (optional)

OPTIONAL (to extend laser "on" time):
AA batteries (2)
Battery holder, 2x AA #270-408 from RadioShack (radioshack.com)
Alligator clip leads (2) RadioShack #278-1156





A

Photography by Renae Novarese and Frank DeFreitas; illustrations by Stephen Michael

Don't bring your object or materials in from outdoors or from one room to another immediately before you begin to make a hologram. Everything should be in the room for several hours so it can adjust to any temperature changes.

To make sure your workspace is dark enough, work at night, cover windows with dark construction paper, and use masking tape around door edges. If you can see around the room after being in it for ten minutes, there's still too much light.

2. SET UP THE EQUIPMENT.

Place the 3 Sorbothane feet (3 feet are more stable than 4) on your work surface. Then rest the concrete paving stone flat on top. The block's mass provides a very solid foundation.

Place your object on the concrete block, leaving room adjacent for the holographic plate. The object should fit behind the plate; for instance, don't try to record an 8" object with a 2.5" plate.

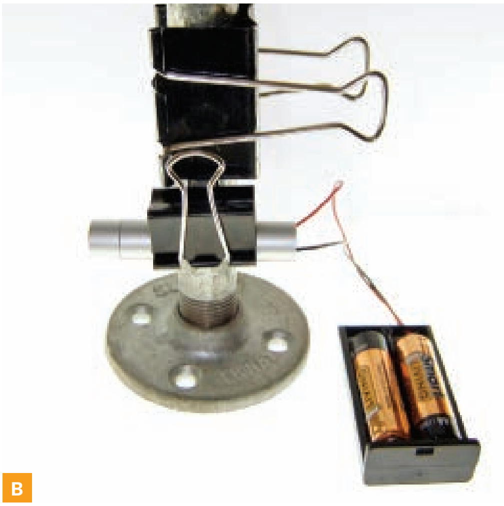
To assemble adjustable laser and lens holders, screw each 8" steel pipe nipple into a floor flange as a base. Clip the laser pointer and lens (make sure the lens is spotless) each in a 1" binder clip, and clip a handle from each to a pipe using a 2" binder clip. The binder clip that holds the laser will also press its "on" button to keep it on.

If you're using an external battery pack to power the laser (Figure B), use alligator clips to connect one battery wire to the pointer's inside spring and the other wire to the other contact inside, along the edge.

Set up your 4 processing trays near your setup so they're ready to go right after the exposure. Follow the instructions that come with your processing kit to prepare the chemical baths, using only distilled water. Familiarize yourself with the processing sequence because the light won't be good for reading when it comes time to perform it. Arrange the plastic trays in sequence: developer, first rinse, bleach, and final rinse. Put a pair of tongs or a spring clothespin handy nearby.

3. CAPTURE THE IMAGE.

With the laser button clipped "on," position the laser's unspread beam, the red dot, in the middle of the object. Maneuver the lens into the beam's path and position it to spread the laser light (Figure C) just enough to cover the object. This will take some



B

adjusting. Finally, double-check to make sure the object is evenly lit with laser light.

Block the laser light by gently propping the black card against the front of the lens. This card will act as the "shutter" during exposure.

Plug in the green nightlight as a safelight and turn off all other lights. Take a holographic plate out of its box and close the box back up. Gently press a corner of the plate between your lips and note which side sticks: that's the emulsion side.

Position the plate on the concrete stage with its emulsion side facing the object, glass side facing the laser. You can gently rest the plate against the object, or else fix the plate upright between 2 flat rectangular magnets (Figure D; shown in full light).

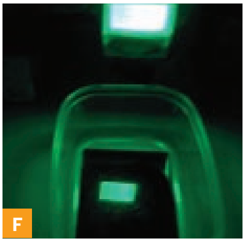
Now is the time to become very quiet. Sit on the floor next to your black shutter card. Allow a few minutes for the setup to settle down. This "settling time" is one of the most important steps in holography.

To make the exposure, gently lift the card and allow the laser light to shine through the plate onto the object (Figure E). Typical exposure times are 15–30 seconds for a 2½" square plate and a standard laser pointer, but this will depend on the power of your pointer and the freshness of your batteries. Do not move during this time. When your exposure is complete, place the shutter card back down to block the light. Your hologram is ready to be processed.

C

D

E



F

G

WARNING: All photographic chemistry needs to be handled with care, so wear gloves, goggles, and an apron when working around it. Keep your work area clean. Keep all chemicals, mixed and unmixed, out of reach of small children.

4. PROCESS THE PLATE.

Pick up the plate with the tongs or clothespin and process it under the safelight according to the instructions. Throughout the process, be sure to wear goggles, an apron, and gloves; only handle the plate with tongs or a clothespin; and always put the plate in its bath emulsion-side up.

Holographic film and processing chemistry varies and changes over the years, but processing usually runs as follows: Immerse the exposed plate in the developer bath for several seconds to several minutes (depending on materials and exposure) until it becomes visibly dark (Figure F). Remove the plate to the first rinse water and agitate for a minute.

Immerse the rinsed plate in the bleach bath until it is no longer dark. Then, once the plate has cleared, rinse it in the final bath water and let it air dry.

Air drying may take up to an hour if the weather is humid. Don't be tempted to hurry it along with a hair dryer because the heat can collapse the emulsion, leaving very little, if any, holographic information.

5. VIEW THE IMAGE (HOPEFULLY).

If everything has worked properly, you'll now have a seemingly magical holographic image to show friends and family. To view it, use an unfrosted light bulb, halogen spotlight, or sunlight on a sunny day. Frosted light bulbs, fluorescent lights, and cloudy days just give a blurred image.

Hold the hologram so that its emulsion side is facing away from you. Start out with the light coming in at the same angle that you used to create the hologram. Look for the image in the light that reflects off the hologram (Figure G shows an 8"×10" version). Have patience; sometimes it's hard to find the image. You can spray-paint the emulsion side flat black to make the image stand out brighter.

Your hologram was made with red light, but due to exposure and processing variations, the final hologram image might be red, orange, or even green. If it's red, you can use the original laser pointer in dark surroundings to reconstruct the image more vividly than with full-spectrum light.

Most importantly, if you attempt to make a hologram and it doesn't turn out, that's OK. This happens sometimes, even to experienced holographers like me. Try again later — or better yet, come back to it the next day. Knowing when to take a break is important to success.

However, your adventure doesn't need to end with a few holograms. The very same system can be used for a number of other laser art projects that are just as interesting. Have fun, and visit holoworld.com for additional resources and project ideas for lasers and holograms.

Frank DeFreitas (holoworld.com) designs and facilitates world-class laser and holography educational programs. He has his full-time studio in Allentown, Pa.

YOUR OWN HONEY COW

Make a simple beehive and enjoy the sweet fruits of your labor. By Abe Connally

Beekeeping is an ancient DIY art, practiced by amateurs and makers for centuries. Anyone can produce natural honey at home by making their own hive. Here's how to build a cheap and simple beehive called the Honey Cow.

The Honey Cow is designed to mimic nature. Unlike commercial hives, it doesn't have frames, foundations, or excluders. Instead, it just has top bars, allowing the bees to do what they'd do in a fallen log: build beautiful, natural combs. Because it's less intrusive to the bees, it's easier to make and to manage, so it's a perfect hive for beginners.

Once you have a hive, you'll want to gather a few extra bits of equipment, like a veil, a smoker, and a bee feeder. You can procure bees by capturing a swarm or buying a "package" or nucleus from a fellow beekeeper. After one full summer, you'll reap the reward: wonderful, homegrown honey.

Photography by Josie Moores

1. Make the barrel.

Choose a food-grade container to avoid potentially dangerous chemicals. Saw it in half lengthwise, making sure there's a bunghole in each half for the bees to enter (Figure A, following page). Now you've got 2 barrels; you'll use one per hive.

Clean your barrel well. You never know what was in it. On one end of the barrel (the end that used to be the top) there's a rim of plastic that protrudes. Cut it away (Figures B–C).

Rub the interior with beeswax. This will remove any foreign smell and make it more attractive to a hive. A drop or two of lemongrass oil is good, too.

2. Make the frame.

Measure your barrel rim and cut 1×2 lumber to make a frame that fits around it. For example, if your barrel is 36"×24", cut 2 lengths of 37" and

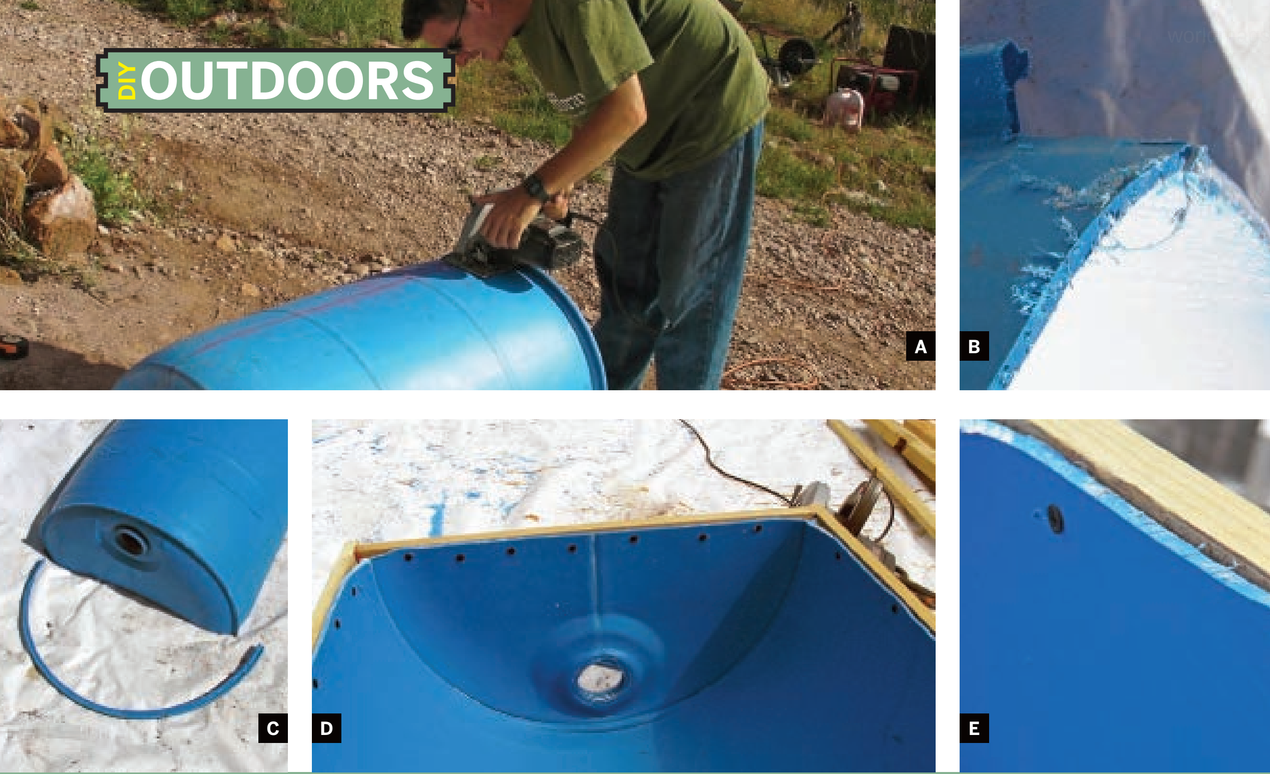


Fig. A: Saw the 55gal barrel in half lengthwise, making sure that the half you use for your hive has a bung-hole. Bees will use it as an entrance to the hive. Figs. B–C: Trim away the protruding rim of what used to be the top of the barrel. Figs. D–E: Build a frame of 1×2 lumber to fit exactly around the rim of your barrel, then screw the barrel to the frame.

MATERIALS AND TOOLS

55gal plastic barrel preferably food grade (makes 2 hives)
1×2 lumber (nominal), 22' Standard 1×2 lumber is really ¾"×1½".
2×4 lumber (nominal), 8' lengths (2) It's 1½"×3½".
1½"×1" lumber, 46' This lumber should have true dimensions of 1½"×1".
Tin sheet, 3'×4'
1½" wood screws (20) for the 1×2 frames
2" wood screws (10) for the 2×4 legs
½" screws (70) for the barrel and tin roof
Bungee cord or tie wire
Thin wood molding, about 45', or natural fiber string and beeswax
Beeswax
Lemongrass oil (optional)
Circular saw or jigsaw
Drill
Tinsnips
Tape measure
Marker

2 lengths of 25" (the extra inch allows you to screw one piece into the next).

Glue and screw the frame together. Then screw the barrel's rim into the frame (Figure D).

3. Build the legs.

Lay a 2×4 flat. Mark one edge 40" from either end (points A and C) and mark the opposite edge 36" from either end (points B and D), as shown in Figure F. Cut the board into 3 pieces, along the lines between A and B and between C and D.

Butt the 2 long pieces together at their pointed ends (A and C) to form an inverted V. Lay the short piece across them to form an A, then screw it down.

Repeat to make the other leg.

Screw a leg to each end of the barrel's frame, and put several ½" screws through the barrel into the leg for a good, sturdy fix (Figure G).

4. Make the top bars.

Cut twenty-three 24" lengths of 1½"×1" lumber. These are the bars to which the bees will attach their honeycombs (Figure H). On each bar, you need to provide a guide so that the bees will make straight combs. There are several ways to do this:

» Screw a thin piece of molding, 20" long, centered on each top bar so that you leave 2" free at each end of the bar. This molding will face downward, into the barrel, when the bar rests on the frame. Rub beeswax on the molding (Figure I).

» Or attach twine coated in beeswax, centered on

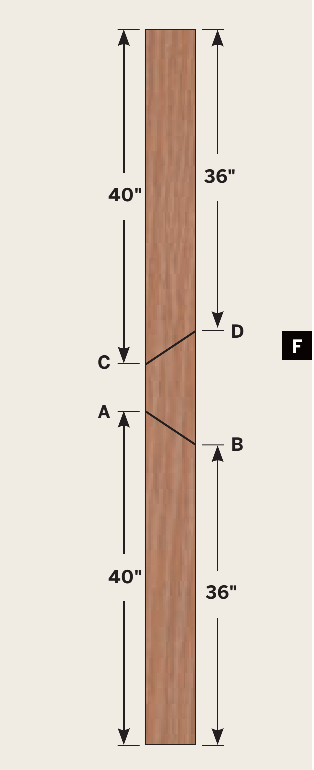


Fig. F: To make each leg, cut an 8' length of 2×4 lumber as shown in this cut diagram. Fig. G: Screw the legs to the barrel frame, then screw the barrel to the leg as well. Fig. H: The top bars lie across the barrel; the bees will attach their honeycombs to these bars. Fig. I: Screw in a thin piece of 20"-long molding centered on each top bar. Fig. J: A simple roof of sheet tin will keep the weather out.

each bar, leaving about 1"–2" free at each end.

» Or carve a narrow groove, about ¼" wide, into each bar and fill it with molten beeswax. Again, leave 1"–2" intact at either end of the top bar.

5. Make the roof.

Make a frame of 1×2 lumber to fit around the barrel frame with a ¼" gap on all sides. For example, if you cut 25" and 37" lengths for the barrel frame, now cut 27½" and 39½" lengths for the roof frame.

Screw the sheet of tin centered onto the roof frame. Bend the excess tin down and screw these edges to the sides of the frame. Using the tinsnips, trim any excess hanging below the frame.

Secure the roof on top of the barrel frame with a bungee cord (Figure J) or with wire.

6. Get some bees.

You can buy a "package" of queen and bees, but it's much more satisfying to capture a swarm.

When dealing with bees, you can't think of them as individuals. It's the hive, as a whole, that is the animal. And each year, if conditions are right, the hive will reproduce. If they've filled the space they inhabit and food is abundant, the bees will create another queen and the hive will split, creating a swarm that will leave in search of a new home.

The swarm is laden with honey, and preoccupied, and consequently very docile. If you come across a swarm on a branch, you can shake the bees off, into a box. Take the box to your hive and empty it into your Honey Cow. They'll do the rest.

In future articles, we'll cover accessory and harvesting equipment for natural, simple, low-cost DIY beekeeping.

CAUTION: Wear protection (as shown on page 123) when handling swarms, because bees can always sting, even when they're docile.

Resources

» Tools, accessories, and DIY kits for top bar hives: goldstarhoneybees.com

» Natural beekeeping forum: biobees.com

» Author's page: velacreations.com/bees.html

» Join a local bee club; they love to help beginners.

Abe Connally and Josie Moores are a young, adventurous couple living in an off-grid hideaway with their 2-year-old. Their experiments with energy, architecture, and sustainable systems are documented at velacreations.com.

MAKING A CAGE TRAP

Catch pesky animals humanely with a homemade slammer. By William Gurstelle

Consider the natural relationship between making things and urban homesteading. Even the smallest garden requires a bit of technology to be successful. Seedlings must be watered, weeds pulled, and garden pests kept at bay. All these tasks require paraphernalia of some kind.

Our modest-sized garden is in a residential neighborhood in Minneapolis. Despite the urban locale, there are hundreds of wild animals in the area. I've observed raccoons, squirrels, rabbits, voles, chipmunks, and, once, an opossum. By far the most plentiful wild mammals are the rabbits.

Any evening, summer or winter, a walk up and down a single block all but guarantees a half-dozen rabbit sightings. Typically they're sitting quietly on front lawns, eating. These rabbits eat a lot: grass and clover certainly, but also perennial plants, young trees, annual flowers, and most irksome to gardeners like me, vegetables. There are so many rabbits and they are so hungry that every garden must be well fenced.

Luckily, fencing is cheap and effective; so far no adult rabbit has breached the integrity of our garden's chicken-wire fence, and our harvest hasn't materially suffered from the leporine onslaught.

But not all is rosy in the urban homestead. Outside the fence, rabbits continued to feast on ornamental plants and shrubs. And smaller creatures slipped through the chicken wire: chipmunks took the tomatoes and battered the basil without remorse. For several days in a row, as our tomatoes neared ripeness, my wife and I entered the garden with happy anticipation only to discover an animal pest had beaten us to the fruit.

As a maker, I decided to make something to take care of the problem.

Photograph by William Gurstelle

MATERIALS

Pine board, 1×6, about 48" long A nominal 1×6 board is really ¾" thick by 5½" wide. You'll cut it into many smaller boards (see Step 1).
Screw eye, medium
Screw eyes, small (2)
Music wire, .056" OD, 14" length
Tee hinges, 2", with mounting screws
Hardware cloth, ¼" mesh: 36"×24" (1), 8"×8" (1)
Thin wood or cardboard, 2"×6" strip for the trigger plate
String, 3'
Nails, 4D, 1½" long (1 box)

TOOLS

Table saw or buy wood pre-cut to size
Hammer
Wood glue
Screwdriver
Needlenose pliers (2)
Wire cutters
Staple gun
File

Removing small nuisance animals can be accomplished in a variety of ways. The most straightforward is the simple noose snare. It's merely a loop of picture-hanging wire placed over a known animal runway and secured to a stake in the ground. The animal enters the loop headfirst and becomes ensnared when the loop pulls tight.

I didn't have great luck with the snare. The rabbits, perhaps sensing danger with native bunny intuition, would hop around it or over it. The one time I caught a rabbit, the animal was able to back off and loose itself from the trap. After a while, the animals avoided the snared runway altogether.

I considered building a deadfall trap, the staple of many a Roadrunner cartoon. When the animal takes the bait, a trigger lever causes a weight to fall and crush the unsuspecting prey. Such a trap can be made to any size; indeed, Wile E. Coyote often used a large safe for the deadfall. But in reality, a deadfall trap is easier to envision than to make.

An adult rabbit is pretty large, so the deadfall needed to crush Thumper has to be big. Devising a trigger that could support a rabbit-crushing rock, yet reliably trip when the rabbit was in position, was a difficult proposition. Plus, I didn't want to accidentally hurt my dog, or have to dispose of a mangled but still breathing animal. So I set aside the deadfall idea.

The Cage Trap

The cage trap is the king of animal traps: reliable, humane, and fun to build. I've enjoyed great success with these devices. A cage trap consists of a wire mesh box with one hinged door. The door is held open by a movable latch, which retracts, shutting the door, when an animal steps on a trigger plate located deep inside the box. The garden pest is held securely and humanely inside the box until the trapper disposes of the animal.

I built this trap in a few hours for about $5 in material costs. Once built, it trapped its first chipmunk in less than half an hour. Chipmunks and rabbits, not being especially bright animals, are quite easy to trap with the appropriate bait.

1. Cut the boards.

From your 1×6 board (which is really ¾"×5½"), first cut an 8" length. This is the trap's door.

Cut a 24" length and from it, rip 4 strips each ¾"×24" and one ¾"×22½" (long frames). From the remaining 24" scrap, cut 2 strips ¾"×8" for the trigger support board and small screw eye support board.

From the remaining board, cut 8 strips ¾"×7" (end frames), and 2 boards 1½"×7" for the medium screw eye support board and hinge support board.

2. Build the frame.

Build the cage trap frame from the 5 long frame and 8 end frame pieces as shown in the assembly diagram (Figure A, following page), using nails and glue. This size of trap works well for squirrels, rabbits, and chipmunks; you can scale it up for use with larger animals.

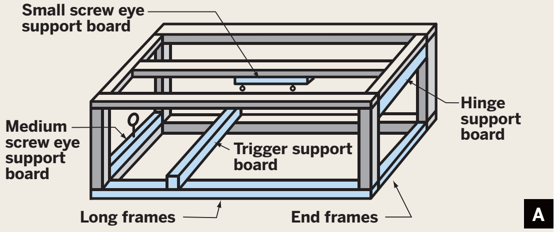
3. Attach the support boards.

Following the assembly diagram, attach the trigger support board, hinge support board, medium screw eye support board, and small screw eye support board. The small-eye board must be set back far enough to clear the cage door when it's fully opened.

4. Attach the hardware.

Screw in the hinges and screw eyes as shown in the hardware diagram (Figure B).

DIY OUTDOORS



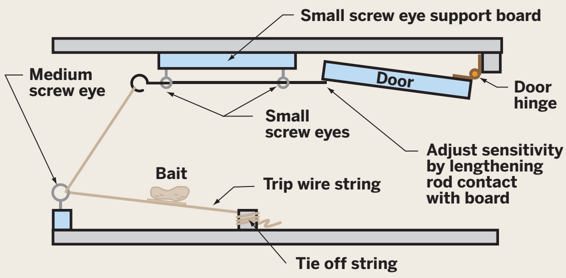


Fig. A: Frame assembly diagram. Fig. B: Hardware diagram. The latch releases the trap door when the animal takes the bait and trips the trigger string. Fig. C: To make the door lock, bend a 6" length of music wire. Fig. D: The latch is just a length of music wire with a loop in one end. Fig. E: Staple the hardware cloth to the cage frame. Fig. F: Run the string from the latch through the medium screw eye, then tie it off.

5. Make the door lock.

Cut a 6" length of music wire and use 2 pairs of needlenose pliers to bend it into the door lock, as shown in Figure C. Attach the lock to the door with the staple gun.

6. Attach the door.

Attach the door to the hinges. Test the door to make certain it closes easily and the door lock operates correctly. Adjust as necessary.

7. Bend a loop in the latch.

Cut an 8" length of music wire and use the 2 pliers to bend one end into a tight loop (Figure D).

8. Attach hardware cloth.

Attach the large piece of hardware cloth to the trap frame using the staple gun (Figure E). Bend the hardware cloth so it fits tightly against the frame, cut off any excess, and file the cut ends smooth.

Attach the small piece of hardware cloth to the open end using the staple gun.

9. String the latch and trigger.

Tie the string to the loop on the music wire latch. Insert the latch through the small screw eyes until ¼" of the latch end supports the door. Run the string through the medium screw eye and tie it off on the trigger support board (Figures B and F), following the hardware diagram.

10. Set the trap.

Choose the location of the trap carefully, away from people and pets, but in the area where the pest animals have been most active. Conceal the trap by placing it under vegetation.

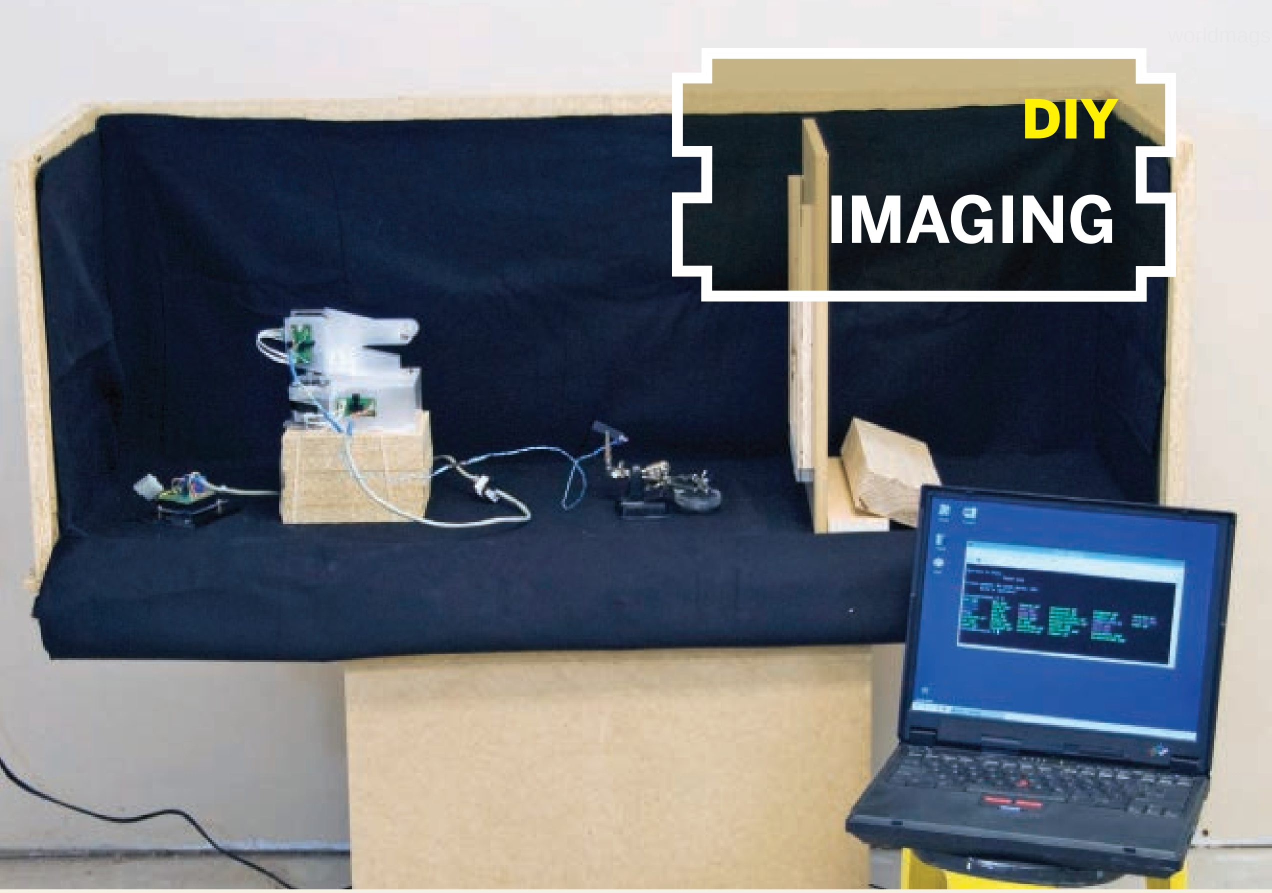
Place your trigger plate over the string and bait the trap by placing the animal's favorite food on the plate. When the animal enters the cage and takes the food, it will cause the latch to release and the door to shut.

Chipmunks and squirrels seem to love peanuts. Rabbits are fond of Romaine lettuce. Carnivorous animals are attracted by raw liver.

Once the animal is trapped, you have various options for disposing of it, ranging from releasing it at an appropriate location to preparing a delicious rabbit fricassee. Irrespective of your disposal method, use caution to avoid being bitten or unnecessarily stressing the animal.

William Gurstelle is a contributing editor of MAKE.

Photography by Ed Troxell (C, E), William Gurstelle (D), Sam Murphy (F)



LASER POINTER PRINTER

A scanning laser "paints" on photo paper.

By Leland Sindt and James Underwood

Photography by Leland Sindt and James Underwood

Several years ago, we came up with an idea for a new project: mount a paintball gun to a computer-controlled turret and turn it into a gigantic "inkjet" poster printer. Searches revealed that this had been done already by Benjamin Gaulon (aka Recyclism), who called it PrintBall.

We talked about scaling up the design with multiple paintball guns, but back-of-the envelope calculations determined that a 640×480-pixel image would require a 53'×40' canvas (at 1" per pixel) and over $6,000 in paintballs. Without a government grant and a drive-in movie screen, it wouldn't work.

We needed cheaper ordnance — and what could be cheaper than photons? A high-powered infrared CO_2 laser could burn an image into many materials, but we didn't like the cost and danger involved. So we settled on an ordinary red laser pointer tracing onto light-sensitive black-and-white photo paper.

This project was particularly enjoyable because it brought together 3 completely different areas of development: the interaction of laser light and photo paper, the mechanical/electrical design of the mirror gimbal that aims the laser beam, and the software to pull it all together. The images the system produces have a wonderful mixed digital/analog quality that results from this unique combination.

Lego Technic Prototype v.0

Early darkroom testing revealed that the biggest limitation to our approach was per-pixel exposure time. It took 5 seconds for the laser to burn one completely black pixel onto the white photo paper. For a 640×480 image, this translated to an exposure time of up to 17 days, depending on the image darkness, so we decided to keep the images small and make a light-tight box to enclose the apparatus.

MATERIALS AND TOOLS
Stepper Motor Version v.2

Laser card module item #COM-00594 from SparkFun Electronics (sparkfun.com), $8
Arduino microcontroller We used a Duemilanove, but you can also use the newer Uno, #MKSP4 from Maker Shed (makershed.com), $35.
Stepper motors, with cable (2) #ROB-09238 from SparkFun, $15 each
Stepper motor drivers, EasyDriver v4 (2) #ROB-09402 from SparkFun, $15 each. We used the older EasyDriver v3, which is no longer available.
Arduino ProtoShield Kit SparkFun #DEV-07914, $17
Relay, 12V #275-241 from RadioShack (radioshack.com), $5
Transistor, NPN, TIP120 RadioShack #276-2068, $2
Diode, 1N4004 RadioShack #276-1103, $1
Resistor, 1kΩ RadioShack #271-1321, $1 for 5
Cat5e cable with RJ45 connectors RadioShack #55010844, $7
RJ45 jack to connect to Cat5e cable, easy to scrounge, or #195630 from Jameco Electronics (jameco.com), $2
Wire, 22 gauge, insulated we pulled this from an old Cat5e cable
Front surface mirror, less than 1" square #FS-01 from Images Scientific Instruments (imagesco.com), $4
Plexiglass, ⅛" thick, about 2' square
Machine screws, #4-40, ⅜" (8)
Washers, #6 (8)
Steel rod, ⅛" dia., 3" long for pivot points
Setscrew collars, ⅛" (7) for pivot points
Setscrew collars, ¼" (2)
MDF (medium density fiberboard), ¾" thick, 4'×8' sheet to make lightproof box
"Third hand" adjustable parts holder
Fabric, black, 4'×2' to line box
Duct tape, black
Photo paper, black-and-white
Laser cutter
Soldering equipment
Drill
Allen wrench and screwdriver
Wire cutters and stripper
Red darkroom light
Darkroom or other dark environment
Sandpaper
Table saw

We prototyped our first versions of the laser-aiming gimbal using Lego Technic blocks, but we couldn't get them to aim accurately; the gears backlashed a bit with movement. Also, the gearing ratios we needed for precise, pixel-by-pixel control required a long gear chain, which compounded the issue — especially when the laser reversed direction to paint a new line.

Servomotor Version v.1

We shifted to using R/C hobby servos to power the gimbal. Outside the box, a serial servo controller provided a simple interface to the project's control software, which ran on a PC.

The controller interface offered well over 1,500 points of rotational resolution in each plane, but the actual hardware could only accurately manage about ⅛th of this. We went with images 175 pixels square, to allow for reasonable exposure time and let us test and tune our software/hardware design.

The Lego gimbal aimed the laser, but for more precision we decided to keep the laser still and bounce its beam off a small mirror mounted to a gear-driven metal pin axle. This halved the range of travel required by the gimbal because the angle of incidence equals the angle of reflection (thank you, 7th grade science class).

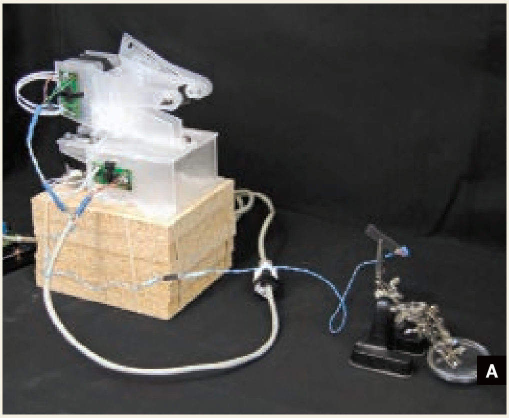
To design the new gimbal, we collaborated using Google SketchUp Pro (sketchup.google.com) and exported the designs to a laser cutter for rendering in plexiglass. We measured and modeled our servo and servo controller in SketchUp and designed the gimbal mechanism around them.

The motors mount to the gimbal with machine screws, and we glued the servo horns to the plexiglass gears. To make a gearing mechanism with a 1:4 ratio, we used the free Ruby script Involute Gears (ohyeahcad.com/download), which generates gear drawings in SketchUp.

For image processing, we expected to write a daunting amount of code. Luckily, the Python Image Library (pythonware.com/products/pil) offered functions for opening, converting, and reading each pixel as a data point, and the Python Serial Library (pyserial.sourceforge.net) facilitated communications with the servo controller via serial port.

This all gave Leland, who did the coding, a great excuse to learn Python, and in the end, the control code turned out to be fairly simple. It basically scales any image down to a maximum dimension of 175 pixels, then converts it to grayscale and assigns each pixel a time value from 0 to 5 seconds. Download our code at makezine.com/25/laserpointer.

In terms of physical control, the code directed the servos to move the laser beam across each line horizontally, pausing for the appropriate amount of time for the laser to expose each pixel. Then the laser would line-feed downward one pixel and start back across the paper in the opposite direction —



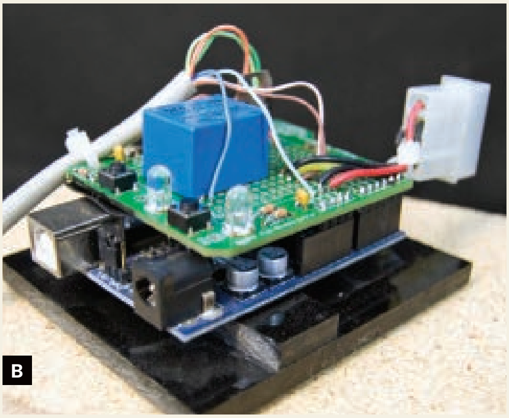


Fig. A: Both the mirror gimbal, raised on blocks, and the laser pointer, held in the third hand, are slideable into different configurations inside the light-tight box. Fig. B: We used an Arduino and our own Arduino shield for the stepper motor printer.

like a kid eating corn on the cob. The code had to reverse the data for every other row, but it made for a more elegant and efficient solution.

We also built a simple light-tight box out of MDF to house the gimbal, laser, and photo paper, with enough room inside to let you adjust the position of each element to vary the image's composition. We ran Python on the PC, which communicated via serial cable with the servo controller outside the box. The controller communicated to the servos inside the box through long servo cables we ran through a small hole, lightproofed with black duct tape.

Inside the box, we clamped the laser pointer in a "third hand" holder near the gimbal. To hold the photo paper, we built a small vertical easel out of MDF. It helped to raise the gimbal on a pedestal of stacked MDF so that the beam from the mirror hit the photo paper more squarely (Figure A).

For testing, we set up in a windowless basement bathroom, which we equipped with a red darkroom light, and we processed the photos in batches in Leland's father's darkroom.

Stepper Motor Version v.2

The servo-based images taught us many lessons. The largest of these is that when you ask a servo to hold a single position for 7+ hours, it burns out.

We discovered that our cheapo Brand X laser pointer didn't emit a concise dot, but instead shined an oblong shape that resulted in pixel overlap and overly dark images. So we substituted a mini-laser card module that clipped neatly into the third hand.

The mirror that we used, scavenged from an old CD drive, was also suboptimal. Like all back-surface mirrors, its front surface reflected some light, which added noise to the image. And our gimbal design, which relied on assembling the plexiglass pieces with glue, required steady hands and much patience.

Life moved on for both of us, and we put the project on hold. But we were inspired to revive it after it was featured on Make: Online and generated interested feedback from the community there.

We decided to completely redo our gimbal design and control system. We built an Arduino shield (Figure B) to interface between the computer and both the relay that switches the laser on and off and a pair of pulse-driven controllers for small stepper motors.

Unlike servos, stepper motors aren't capable of absolute positioning, which added complexity to the code. Design ideas included end-stop switches and self-calibrating software routines, but we decided to reset and recalibrate manually between exposures. We developed Arduino code to accept commands from Python and modified our servo Python code to work with the new Arduino stepper interface.

Like before, we used SketchUp to design our new gimbal (Figure C, following page). But this time we made the pieces interlocking so they could be snapped together and glued to add strength. And we used a better-performing front-surface mirror.

We connected the Arduino and shield to the PC outside the lightproof box via serial cable, and we wired it to the gimbal inside using Cat5e cable with male/female RJ45 ends for easy connection and disconnection. Arduino digital pin D6 connects up to the transistor relay circuit on the shield to switch the laser (see makezine.com/25/laserpointer for a diagram). We soldered the shield connections using 22-gauge wire. Pins D2 and D4 determine the direction the stepper motors travel, and pins D3 and D5 move the motors one step with each output pulse.

Lining the inside of the box with black fabric took care of light scatter. Images came out dark, but we fixed this by tweaking the exposure down to sub-second intervals. Still, pixel overlap and limitations in relay reaction time give the developed images a slight haze. Call it the medium's unique texture.

Moving Forward — What's Next

This has been a truly satisfying project, synthesizing aspects of mechanical design, electronic design, and programming, with each phase governed by both artistic goals and physical constraints. And as with any challenging project, the most valuable product wasn't the artifact we created, but the knowledge we gained and the tools we can use on our next project.

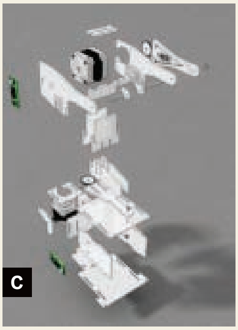
The warping caused by the laser's radial movements translating onto a flat plane (Figure E) adds to the light-painted images' appeal, and you can compose infinite variations of this effect by simply scooting the laser, gimbal, and easel around to change their relative positions inside the box.

The next step might be to increase the power of the laser and use wood or paper canvases. As higher-power lasers become less expensive, this becomes increasingly attainable.

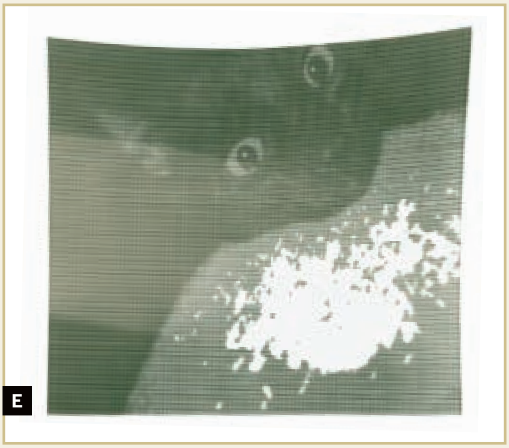
We weren't able to make sense of some of our pixel overlap and light-scatter issues, but we're glad to share our source material. If you take this project to the next level, let us know how it goes!

+ For project templates, source code, schematic diagrams, laser pointer images, and other resources, visit makezine.com/25/laserpointer.

Leland Sindt is a business continuity consultant for Hitachi Data Systems. He enjoys beer brewing, woodworking, programming, and problem-solving. James Underwood is a graduate of the University of Kansas School of Architecture and Urban Planning. He enjoys beer brewing, bread baking, and talking his friends into collaborating on ridiculous projects.


C


D


E

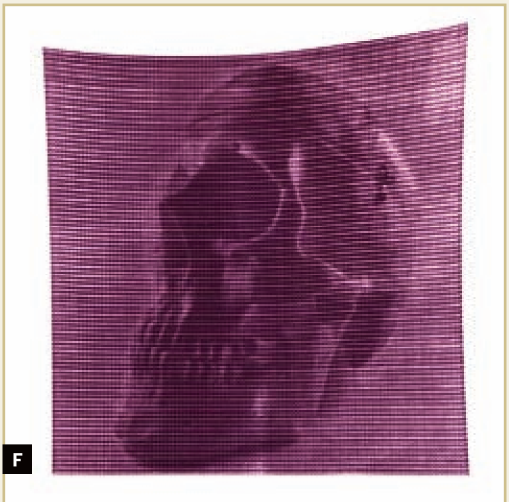

F

Fig. C: Exploded view of the stepper version gimbal, with motors and controller boards. Fig. D: A test print from the earlier servomotor version of the printer. Figs. E–F: Sample prints from the stepper motor version of the printer.

LED PAPER CUTTER ASSIST

Modify an el-cheapo paper cutter to make it way more accurate. By Larry Cotton

Take one el-cheapo paper cutter. Drill some holes, add a linear scale and some weight. Presto: you now have one very good paper cutter. What happened?

X-Acto makes nice guillotine paper cutters, but the plastic base model I bought isn't known for robustness or accuracy. It skitters around your desk as you accidentally whack your photo of Aunt Martha down the middle. Adding weight stablizes it.

So why drill holes in it? Because you'll add several 5mm-diameter bright white LEDs below the cutting blades, pointing up. These cast a shadow of the fixed blade onto the underside of the paper you're cutting.

Note: All bets are off for cardboard or more than a few sheets of paper. Remember, this thing really isn't up to those tasks anyway.

The blade's sharp shadow, which is readily visible through the paper from the top, tells you exactly where you'll cut. It's perfect for cutting, say, an 8½"×11" sheet of photos into smaller ones, as it makes penciled guidelines virtually unnecessary.

1. Drill the holes.

Each LED requires two 1/16" holes for its leads. Use 3 or 4 LEDs to disperse the light along the blade's length. I used 4, which required 8 holes.

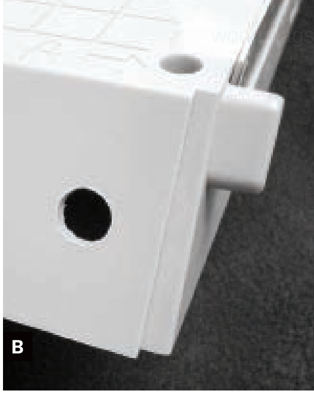
The holes must be near the bottom edge, in order for the LEDs to clear the moveable blade when it's all the way down (Figure A).

I had a rotary SPST switch lying around, which was easy to mount in a 5/16" hole on the front surface (Figure B). Your modifications may vary depending on the switch you use.

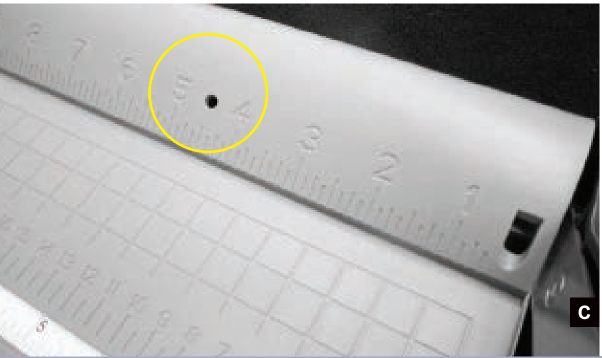
The last hole simply clears a screw for attaching the LED power supply to the bottom (Figure C). If you use a 9V battery, you may not have to drill this hole — or build a power supply either.

Photography by Larry Cotton

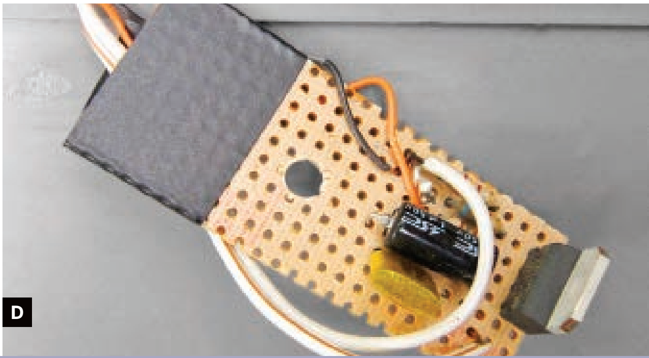
A



B



C



D

Fig. A: Drill 2 holes for each LED, near the bottom edge. Fig. B: I mounted a rotary SPST switch in a 5/16" hole on the front surface. Fig. C: The last hole clears a screw for attaching the LED power supply to the bottom.

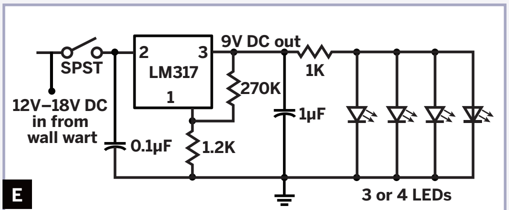
Fig. D: The power supply wired on a perf board. Fig. E: The wiring circuit diagram (below).

MATERIALS

Paper cutter, X-Acto model 23262 or 02623X
  The newer 23262 has a 1/16" scale; the older 02623X doesn't.
12"×½" scale, self-adhesive (optional) if you have the 02623X
Switch, SPST
LEDs, bright white (3–4) such as Jameco #320531, jameco.com
Perf board or prototyping breadboard
LM317 voltage regulator IC
Capacitors: 0.1μF (1), 1μF (1)
Resistors: 270Ω (1), 1kΩ (1), 1.2kΩ (1)
Wall-wart transformer, 115V AC to 12V–18V DC
  Others will work, but you might have to change other components to compensate.
Wire, 22 gauge such as telephone wire
#8-32 machine screw and nut
Scrap steel or iron to add weight
Masking or electrical tape

TOOLS

Drill and drill bits
Hot glue gun and glue
Soldering iron, small
Screwdriver
Pliers, regular and wire cutting
X-Acto knife to strip insulation
Scissors



E

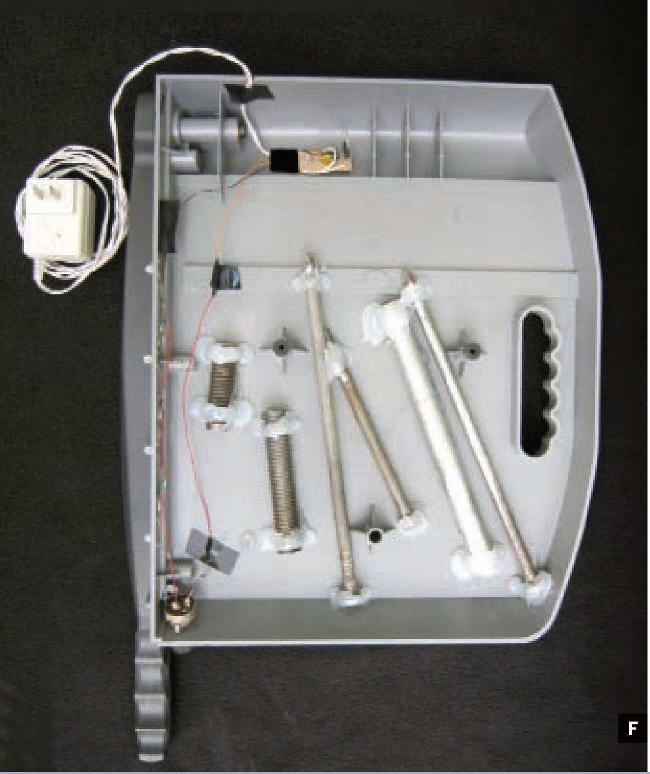
2. Wire the power and LEDs.

Wire the power supply on either a perf board (Figure D) or a small prototyping circuit board, according to the circuit diagram (Figure E, above). Mount it to the underside of the paper cutter, near the rear, with an 8-32 machine screw and nut.

Wire the LEDs next. Trim their leads to about 1", enough to bend at 90° against the outside of the fixed blade, pointing straight up. The short, stiff wires are strong enough to hold the LEDs in position.

Noting their polarity, connect the LEDs in parallel with the power supply. Use one current-limiting 1kΩ resistor to serve all the LEDs. Wire the switch in series at the input of the wall wart.

I hot-glued most of the wiring to the inside walls of the paper cutter. It's quick and easy, albeit a bit messy.



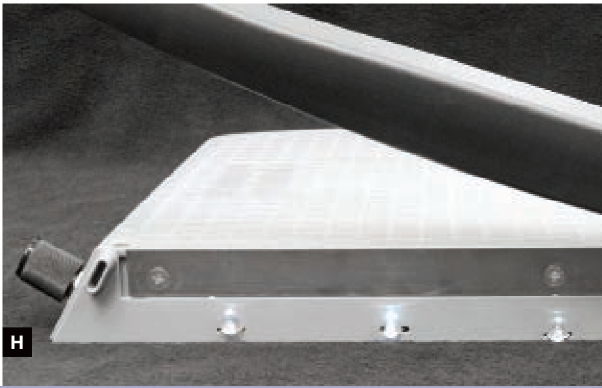


Fig. F: Add copious amounts of hot glue over both ends of all rods and heavy bolts to strap them in.
Fig. G: You can add a ½"-wide scale in the recess in the top of the 02623X if you like.

Fig. H: Turn on the LEDs and adjust your paper using the fixed-blade shadow. The shadow shows you exactly where the moving blade will cut.

3. Add weight.

Any heavy steel or iron scraps, rods, or large bolts will do for weight. There's plenty of room underneath the cutter for them. You can either box them in with a piece of corrugated boxboard, or just hot-glue them to the underside of the paper cutter's base. Lightly sand the smooth underside of the base to achieve better adhesion.

Squirt copious amounts of hot glue over both ends of any rods or heavy bolts to "strap" them in place (Figure F).

4. Add the scale (optional).

X-Acto model 26232 already has a scale (ruler) accurate to 1⁄16". The older model 02623X has only a ⅛" scale molded in the plastic, but there's a recess in the top for you to add a ½"-wide scale that's graduated in finer increments (Figure G). Get a self-adhesive scale, or use your own adhesive, such as double-sided tape.

Use It

Turn on the LEDs and adjust your paper using the fixed-blade shadow (Figure H). Keep your paper's edges parallel to the grooves molded into the paper cutter's top surface. You can use the scale for even more precise cuts.

The LEDs will make photo borders pop out at you with no guesswork about where you'll cut. Aunt Martha now stands a fighting chance of survival.

Larry Cotton is a retired engineer and part-time math teacher who lives in New Bern, N.C., and likes to listen to, write, and play anything musical.

Make: TIPS !

Sharpen Sandpaper with an Old Tire

When my drum sander gets clogged I sand a chunk of old tire with it, and then it cuts well again. The rubber works like a big eraser to scrub the sawdust out from between the grains of abrasive on the sandpaper. You can buy products that are meant to do this, but an old tire works plenty good, and you'll never have trouble finding one.

—Tim Anderson

Find more tools-n-tips at makezine.com/tnt.

GIANT SPIN ART

Spin canvases at high speed and drop paint on them. By Bob and Pete Goldstein

Spin art is a children's activity, often found at school fairs. Kids drop paint onto a spinning square of paper, making beautiful, colorful patterns. As adults, we imagined it would be fun to scale this up, and up, and up. Our friends envisioned injuries, or worse — an elaborate, spinning contraption flying high into the sky and disappearing.

So we tried it, and after burning out the motor from a box fan, we realized that a corded power drill would work better. Corded power drills are cheaper than battery-operated drills, and they typically have greater torque. Many even have a speed knob on the trigger, offering more control.

Building the Frame

We mounted our drill pointing straight up, by sandwiching it between two 2' planks and then screwing 2 more 2' planks onto the ends, to make a stable

MATERIALS AND TOOLS

1×6 wood boards, 24" long (4) for the frame
1×4 wood board, 30" long
Small wood block about the width of your drill
$\frac{5}{16}$" spike T-nut
$\frac{5}{16}$" bolt, $1\frac{1}{2}$" long
Loctite threadlocker glue
$1\frac{1}{2}$" wood screws (14)
Canvases Pre-stretched, 20" round canvases can be found online at misterart.com for about $10 each, shipped in packs of 6.
Paint We used water-based acrylic paints in squirt bottles.
Cardboard
Electric drill, corded with a lock to keep the trigger held in, and a way to control speed. Ours has a variable speed wheel on the trigger.
Remote switch extension cord
Dremel high-speed rotary tool to shape the bolt

Photography by Patrick Harrell

Fig. A: Corded drill mounted on the wooden base. Fig. B: To make an adapter that holds the canvases, hammer a T-nut into the center of a 1×4, screw a bolt through the T-nut, and chuck the bolt into the drill.

Figs. C and D: Two wood screws secure a canvas to the adapter. Fig. E: A round canvas mounted and ready for painting. Fig. F: Spin art in progress, with cardboard around it to catch any flying paint.

H-shaped frame. To keep the drill from sliding, we also screwed a small wood block against its handle at the bottom. The drill sits snug in the frame, and it can be lifted out easily (Figure A).

We locked the drill's trigger in the On position, and then plugged the drill into a switched extension cord so we could turn it on and off remotely.

Attaching the Canvas

To make an adapter for attaching the drill chuck to the canvas, we drilled a pilot hole through the center of a 30" plank and hammered in a 5/16" spike T-nut. We ran a 5/16" bolt through the nut and added a drop of Loctite to make it hold. To strengthen the drill's grip on the bolt, we used a Dremel to shape it like a hexagonal drill bit (Figure B).

Then we drilled a wood screw through each end of the plank, to point up when the bolt points down. We screw the screws farther to attach them to a canvas frame (Figures C and D), and then flip the assembly over and clamp the bolt in the drill chuck (Figure E).

Spinning the Canvas

We've found that any canvas spins pretty smoothly as long as it's centered on the adapter. For canvases more than 2'–3' across, just make sure not to spin for more than a few minutes at a time, to avoid burning out the drill's motor.

We put a ring of cardboard around the setup and some paper on the ground so that high-speed paint wouldn't cover the neighborhood (Figure F).

WARNINGS:

- **Watch your knees! And don't lean over the device so far that you fall onto it.**
- **Stop the device immediately if it ever tips over. We stake ours to the ground.**
- **Keep electrical plugs out of the area where the paint is; don't mix electricity and liquids.**
- **To minimize risk of electrocution, plug the device into a ground-fault circuit interrupter outlet. These are often labeled GFI or GFCI, with test and reset buttons on them.**

The Results

Giant spin art has been fun to do with friends. We usually spin 20" round canvases, but sometimes try larger ones. Our record size so far is 4'×3'. Surprisingly, just about every canvas comes out great.

Bob and Pete Goldstein are brothers who rarely build anything based on their half-baked ideas.

SOLAR TV REMOTE

Juice your flipper with sunlight.

By Sparkle Labs

Powering a TV remote with the sun means you won't have to buy batteries every six months to surf from the sofa. Low-voltage devices that use power only occasionally, like remotes, are good candidates for a solar-powered trickle-charge. Here's how we modded our remote to live off the sun, using rechargeable batteries connected in parallel with a small solar cell and a diode to keep any power from trickling back out (Figure C).

1. Determine what you need.

First, open your remote to see what kind of batteries you'll need and how many. Our remote used 2 NiMH AAA batteries. Remote batteries connect in series, so add up the battery voltages to determine what size solar panel you'll need. Disposable AA and AAA batteries put out 1.5V, but the equivalent NiMH rechargeables produce 1.2V.

Look for a solar cell that generates a higher voltage than the batteries (a total of 2.4V in our case). Make sure it's a flexible cell that fits on the back of your remote so you can attach it easily.

Use a multimeter at its lowest DC voltage setting to test the polarity of the cell. The needle or number will read positive with the meter's black probe (–) on the ground terminal and red probe (+) on power.

Switch the meter to test for continuity and probe to find the pair of contacts, (–) in one battery compartment and (+) in the other, that aren't electrically connected. These 2 contacts are where we'll patch in the solar panel.

2. Put it together.

Use a sharp knife to scrape the plastic off the 2 solar cell contacts, then dab a bit of hot solder on these points. Position the solar cell against the back of

Photography by Amy Parness and Ariel Churi

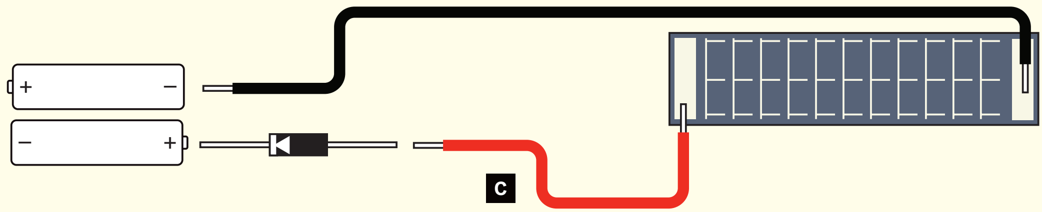


Fig. A: Cut a notch in the battery compartment door to route the wires through. Stick the solar cell to the back of the remote with double-sided tape, and cover the solar cell's contacts and the diode with electrical tape.

Fig. B: To smooth the wire connections in the battery compartment, solder copper tape over the ends of the wires. Fig. C: Wiring diagram. Solder the diode directly to the cell, or farther away, depending on your remote.

MATERIALS AND TOOLS

Remote control
Rechargeable batteries, NiMH size and number to replace the remote's batteries
Solar cell, flexible film Match voltage and size to your remote (see Step 1). We used a 3V 25mA cell measuring 4½"×1", item #227993 from Jameco (jameco.com), $4. You can also find these on other electronics websites and eBay.
Wire, 24–28 gauge, insulated, about 2'
Diode, 1N4004 or similar item #276-1103 from RadioShack (radioshack.com), $1
Copper tape, ½"-square pieces (2)
Double-sided tape
Electrical or foam tape matching the color of your remote
Multimeter
Hobby knife
Wire cutters and strippers
Soldering iron and solder
Rotary tool or small file

the remote, and determine the wire lengths needed for neatly connecting the 2 unconnected battery contacts to the solar cell contacts.

Cut 2 wires to length, strip the ends, and solder one wire between the negative (–) contacts of the battery and solar cell. Solder the diode's cathode side (without the stripe) to the positive (+) contact of the solar cell, then solder the other wire to the diode's anode side (with the stripe). Trim the end of this wire to compensate for the added diode length, then connect it to the positive battery contact.

To smooth over the wire connections in the battery compartments so the batteries slide in more easily, we soldered small pieces of copper tape to the ends of the wires.

Use a rotary tool or file to cut a small notch in the side of the battery compartment door, to route the wires through when the door is closed (Figure A). Attach the solar cell to the back of the remote with double-sided tape, and cover the cell's ends with electrical or foam tape, just over the light-colored parts. We taped over our diode as well.

Finally, load your rechargeable batteries (Figure B). Our remote takes 4–5 hours to charge its batteries fully, but it never gets low as we leave it by the window.

+ Get the kit at makershed.com/solarremote

Amy Parness and Ariel Churi work at Sparkle Labs (sparklelabs.com). Amy is a digital artist and product designer who pets her cats while working with new technologies. Ariel eats chapati and creates "hi-tech, hi-touch" toys for all ages.

FAUX FLAMES

Make realistic fake fire, using fabric, a fan, and some LEDs. By William Gurstelle

Disneyland's Pirates of the Caribbean ride, the last attraction that Walt Disney himself had a hand in designing, is well known among stagecraft buffs for its use of extravagant, realistic faux fire. At the climax of the tale, the village of Isla Tesoro is set aflame by drunken animatronic pirates led by the fearsome Hector Barbossa; the entire scene is filled with red-orange, glowing flames mimicking a five-alarm fire.

With so many people moving through the exhibit, those flames could not be real. Instead, they're created out of whole cloth, so to speak — cloth as well as a frame, stage lighting, and moving air.

From the time of the ancient Greeks, stagecraft experts have been relied on to control or simulate fire safely on stage — anything from a cozy stove or fireplace to giant flames shooting up from the gates of hell.

Exploring fire effects made from silk, moving air, and colored lights is an interesting and accessible entrance into the world of theatrical stagecraft. This project combines simple electronics with some mechanical know-how to create amazingly realistic faux fire.

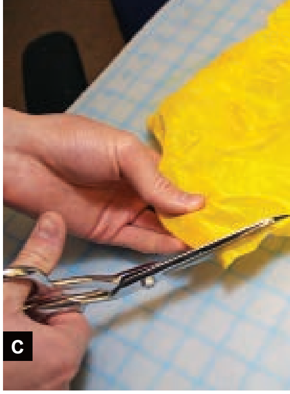
To simulate a campfire, this project uses 12-volt electronics, inexpensive colored LEDs, and lightweight, brightly colored cloth. The voltages involved are low and non-shocking, the LEDs emit very little heat, and the "muffin" fan is too small to constitute much of a hazard to fingers. This makes it easy to experiment with different fire shapes, LED arrangements, and geometries.

Use these instructions as a departure point for your own, perhaps more elaborate, faux fire projects. This project can be completed in an afternoon or two.

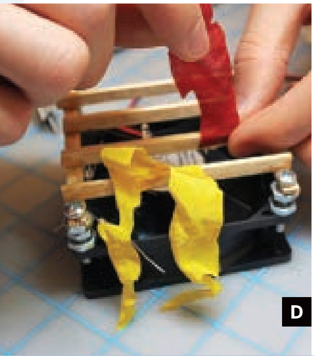
Photograph by Sam Murphy

A

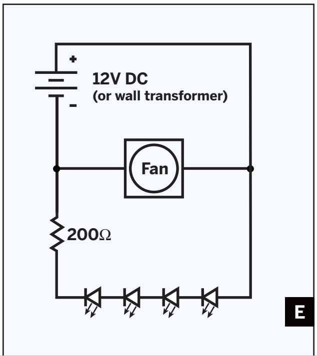
B



C



D



E

F

Fig. A: Glue together a wooden lattice. Fig. B: Mount the lattice to the front of the fan. Fig. C: Cut fabric into sharp flame shapes. The bigger the fan, the bigger the flames. Fig. D: Glue the fabric flames to the lattice.

Fig. E: Schematic diagram for the entire circuit, including LEDs and fan. Fig F: Solder the LEDs together in series with the resistor, or just twist the connections together if you don't feel like soldering.

Photography by Ed Troxell, William Gurstelle (B)

MATERIALS AND TOOLS

Fan, 12V The build shown here uses a computer fan, but larger "squirrel cage" fans also work. The more air the fan moves, the larger and more realistic the flames; an 18 CFM fan will simulate small fires, and a 60 CFM fan can simulate larger ones.

12V battery or power supply

Ultrabright LEDs (4) in red, yellow, or orange These LEDs produce 5,000 to 10,000 millicandelas (mcd) of light. Many online retailers sell them, including Amazon, Parts Express, and Solarbotics.

Resistor, 200Ω

Hookup wire, 22 gauge

Silk or silk-like fabric, red and yellow, 1 sq. ft. each The fabric must be light and sheer.

Fast-drying glue

Tape

Machine screws, #8-32×1" (4)

Nuts, #8-32 (10)

Wood strip, ¼"×⅛", about a 2' length

Wood strips, 5/16" or ⅜" wide, about 4" long (2)

Hardware cloth, a piece about 3"×10"

Small saw

Drill and 5/32" bit

Wire cutters to cut hardware cloth

1. Prepare the fan.

First, fabricate a lattice out of the wood strips, using fast-drying glue. Figure A shows one way to build the lattice. You may need to improvise based on the dimensions and characteristics of the fan you use.

Drill holes in the lattice frame to match your fan's mounting holes, then secure the lattice to the fan using machine screws and nuts (Figure B).

2. Create the silk flames.

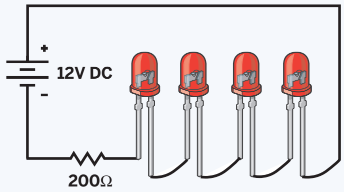
Cut the red and yellow fabric into flame shapes (Figure C). The more powerful your fan (i.e., the higher its CFM rating), the longer the pieces can be. Glue or tape the fabric flames to the lattice (Figure D).

3. Light the flame.

The schematic diagram (Figure E) shows how to wire the LEDs and the 200Ω resistor in series. You can solder the connections (Figure F) or just twist them. Figure G shows how to derive the resistor value, assuming LEDs with a forward voltage (usually denoted VF in specifications) of 2V to 4V, and a normal current draw of 20 to 30 milliamps (mA).

Recall that the longer of the LED's 2 legs is the positive leg. Wire the 4 LEDs, positive to negative, as shown in Figures E and G.

WHICH RESISTORS FOR 4 BRIGHT LEDs?



Why a 200-ohm resistor? An LED must have a resistor connected in series to limit the current that passes through it; otherwise it will burn out almost instantly. For a system using 4 red ultrabright LEDs, the required resistor value (R) is given by:

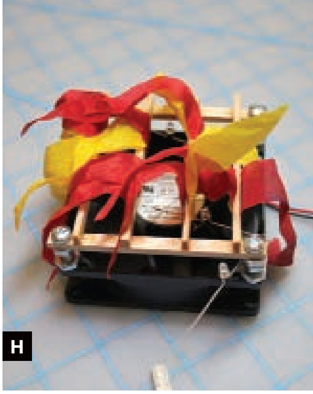
$$R = (VS - (VL1 + VL2 + VL3 + VL4)) \,/\, I$$

Where VS = supply voltage; $VL1$, $VL2$, $VL3$, and $VL4$ = LED forward voltages (typically 2V apiece); and I = LED current (e.g., 20mA).

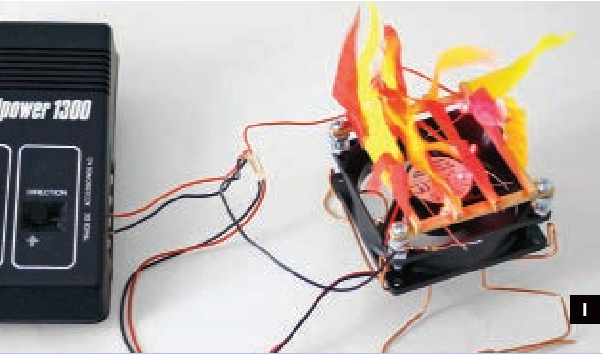
Therefore,

$$R = \frac{(12V - (2V + 2V + 2V + 2V))}{0.02A} = 200\Omega$$

G



H



I

J

Fig. G: Derivation of the resistor value (200Ω) needed for 4 ultrabright LEDs connected in series with 12V DC. You can use this formula to build bigger and better fake fire projects. Fig. H: LEDs attached into the lattice.

Fig. I: Fan and LEDs powered up. Fig. J: Enhance the realism of your faux fire by putting it on a stand hidden behind wood logs.

Attach the LED/resistor assembly to the lattice that you mounted to the fan housing (Figure H).

4. Hook up the battery.

The LED assembly is wired in parallel with the fan, and the power supply powers both simultaneously. Hook them up following the schematic diagram in Figure E. When everything is wired correctly, the fan will spin and the LEDs will light (Figure I).

5. Adjust and tweak.

When the fan turns and the LEDs light up, your faux fire may not look very realistic initially. Don't worry; a little bit of tweaking will work wonders. Your fake fire may require the following adjustments:

5a. The light emitted by the LEDs is very directional. Most of the light is aimed straight up, and little goes out radially from the sides. Use this to your advantage by bending the LED wires to aim the light onto the fabric to produce the most realistic lighting effects. Bend the leads slowly and gently; too much bending will cause them to break.

5b. You can adjust the "flutter" of the flames by re-cutting the fabric into different shapes. Long, skinny shapes flutter more than shorter, flatter designs, although shapes that are too long will sag rather than wave. Once you determine the maximum length of fabric that your fan can handle, attach the longest shapes near the center of the lattice, and the shorter, flatter shapes near the edges.

6. Enhance.

6a. You can improve the realism by placing the entire assembly on or near a bed of dry ice. The vapor from the dry ice will simulate smoke.

6b. You'll get maximum airflow by keeping your fan inlet uncovered so that air can enter freely. To build a simple stand, bend the hardware cloth into a circle and secure it with tape or wire, by folding the edges over. Place your faux fire upon it. This will allow better airflow and improve performance.

6c. For the most realistic effects, hide the fan and its stand behind wood logs (Figure J).

6d. You can add a second row of LEDs if you wish. Wire the second row in parallel to the first.

William Gurstelle is a contributing editor of MAKE.

Photography by Ed Troxell, William Gurstelle (J)

MAKE MONEY

Scrabble By Tom Parker

Sometimes it costs more to buy it than to make it from the money itself.

Photograph by Tom Parker

$14.95

Scrabble game from Hasbro.

$1.00 (a $B_3U_1C_3K_5$!)

Scrabble game made from pennies.

Invented by architect Alfred Butts in 1938, the popular board game Scrabble is now sold in 121 countries, with an estimated 150 million sets in use worldwide. The basic version sold by Hasbro Inc. for $14.95 consists of a game board marked with a 15-by-15 grid and 100 tiles, 98 of which are marked with a letter and a point value ranging from 1 to 10.

For my MAKE Money version of Scrabble, I used 100 pre-1982 pennies (exactly a $B_3U_1C_3K_5$!) made of 95% copper. Using a ¾" Forstner "web-foot" bit in a drill press, I bored a tightly spaced grid of 100 very shallow, flat-bottomed registration holes in a smooth plank of rock maple. Then I drilled a ¼" pilot hole through the center of each registration hole to the backside of the board.

By clamping the board to a workbench and attaching a Shop-Vac to the underside using a plenum fashioned of cardboard and duct tape, I was able to use suction to hold all 100 pennies in place face-down while I sanded their backsides smooth using an orbital sander.

Then, using a hammer and anvil and two sizes of metal stamps, I punched the appropriate letters and score values onto each shiny coin, leaving two "blanks" just like in the real game.

For my playing surface, I salvaged a piece of peg-board from a dumpster. To define the Scrabble grid and label the "premium squares," I cut up and glued down pieces of (get this) paper coin wrappers! You can get these for free from your bank and they just happen to come in colors that approximate those used for the original game: dark red for "triple word," pink for "double word," dark blue for "triple letter," and light blue for "double letter" squares.

BUILD NOTES

This locked gift box guides recipients to the only place on Earth where it will open.

Reverse Geocache Puzzle Box

BY MIKAL HART

WHAT FOLLOWS IS A TALE OF ROMANCE, mystery, friendship, and mechatronics. A couple of years ago my friend Chris, who first introduced me to the Arduino, amazed us with the news that he was moving to France to marry his girlfriend Christèle.

Once I recovered from the surprise, it occurred to me that putting together an Arduino-based wedding present might be a uniquely fitting thanks for his years of friendship and for launching me on my many delightful microcontroller-based journeys.

I decided to make them a box that presented a puzzle — and with that in mind, I'd like to put you in their shoes and ask you to pretend for a moment that you never read the title of this article. All you know is that you've received a wooden box from a friend as a gift at your wedding.

It's an exotic-looking thing, about the size and shape of your mom's old jewelry box, with elephants carved around its perimeter. Curiously, the lid sports a sleek blue LCD display and a silvery push button. A shiny power socket protrudes suspiciously from a side panel. You want to pick this mysterious box up, and you really want to push that button (Figure A).

So you do it. You push that shiny button. And in response, a series of puzzling messages appears on the display (Figure B):

Good morning, (your name)!
This is attempt 1 of 50.
Distance 391km
Access Denied
Powering Off…

Push the button again and you'll get a similar series, only this time it's "attempt 2 of 50." Can you decipher these clues?

To spell it out, here are 3 basic truths about the Reverse Geocache puzzle:

1. To open the box, you must take it somewhere else.
2. Each button push tells you how far your current location is from that "somewhere else."
3. You only have 50 button pushes to get it there.

The box is latched from the inside, where an electronic lock stands guard, waiting to be transported to a certain place. You see, it's a smart box that knows both where it is and where it wants to be, and it's programmed to stay locked tight until someone takes it there. Analytically minded adults who figure this out often draw circles on a map (representing locations that are the advertised distance from each reading) to see where they intersect (Figure C). Kids working the puzzle tend to approach the solution more organically, Marco Polo-style (Figure D).

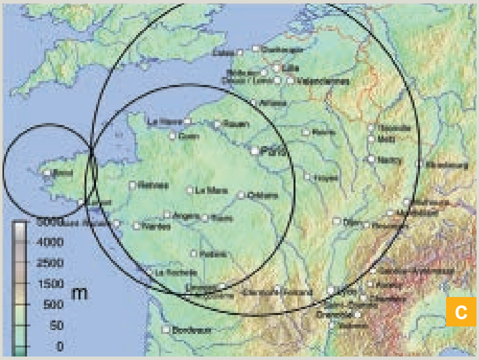
For my newlywed friends I programmed the box to open within 2,000 meters of exactly 48.8469°N by 2.9986°W — that is, anywhere on Île-de-Bréhat, the charming little island off the coast of Brittany where the newlyweds first fell for each other shooting Chris' film *The Luminiferous Aether* (2009).

Since creating this gift, the "elephant box," I've made many more Reverse Geocache puzzles (Figures J–L, page 147), including the one my 12-year-old son employs when we go weekend puzzle-boxing to "magic destinations" like the barbecue restaurant up the road. Here's how to do it.

Select and Cut the Box

The first step in assembling a Reverse Geocache is to select and cut an appropriately sized box. The box needs to be large enough to accommodate the electronics and the "treasure" (Figure J). Tiny boxes are attractive, but not if they don't leave enough room for decent treasure. On the other hand, a trunk brimming with booty wouldn't be much fun to lug up Mount McKinley or through the Amazon jungle.

All the box's electronics will be mounted inside its lid. You'll need to drill a hole for the button (in the top of the box or the front of the lid if it's deep enough)



MATERIALS

Box A wooden box, with approximately ¼" walls and a hinged lid at least 1¼" deep, is recommended, although other styles will work. You can make your own or buy one (see "Select and Cut the Box"). I bought my elephant box at Cost Plus World Market.

Arduino Uno microcontroller item #MKSP4 from the Maker Shed (makershed.com), $35; or the older Arduino Duemilanove

Arduino shield (plug-in mini PCB) I designed a custom shield for this project, which you can buy from me for $10 at info@arduiniana.org. Or you can use a Proto-Shield for Arduino Kit, Maker Shed #MKAD6, $15.

GPS receiver, 20-Channel EM-406A SiRF III with antenna #GPS-00465 from SparkFun Electronics (sparkfun.com), $60

LCD display, 8×2 char, blue backlight #LCD104B6B from Seeed Studio (seeedstudio.com/depot), $5

Servomotor Hitec HS-311 or HS-55 #LXDEL5 or #LXTX42 from Tower Hobbies (towerhobbies.com), $8 or $10, respectively

Push button momentary switch with blue LED, metal Seeed #SWT101A2B, $7

Push button power switch module (LV) #750 from Pololu (pololu.com), $7

5V boost voltage regulator NCP1402 Pololu #798, $5

Ribbon cable, 16-pin IDC for the LCD, #H3AAH-1606G-ND from Digi-Key (digikey.com), $2

Wire cable, 4-pin dual-female jumper Seeed #CAB104C4O, $3 for 5

Connection header, 6-pin horizontal for the GPS, Digi-Key #455-1806-1-ND

Headers, straight pin, 40-pin breakaway (2) (optional) SparkFun #PRT-00116, $3 each. You'll need these only if you're using my custom shield. Break them into 7 header rows with pin counts of 9, 8, 8, 6, 6, 4, and 3.

Headers, right-angle, 40-pin breakaway SparkFun #PRT-00553, $2. You'll need one 3-pin row.

Batteries, AA (2)

Battery holder, 2x AA #270-408 from RadioShack (radioshack.com), $2

Resistor, 220Ω (⅛W, ¼W, or ½W) RadioShack #271-1313

Capacitors, 0.1µF (2) RadioShack #272-135, $2 each

Wood dowel, ⅜" diameter, 4"–5" long

Lumber, 1×1 stock, 2"–3" long

L-hook screw Its length should measure just less than the inside height of your box.

Paper clip for motor-to-dowel linkage

Wood screws the right depth for the walls of your box

TOOLS

Handsaw for light woodworking
Rotary tool or jigsaw
Drill and drill bits
Soldering equipment
Glue gun and hot glue
USB cable
Computer with internet connection

Photography by James Giroux and Mikal Hart; map copyright David Monniaux

This locked gift box guides recipients to the only place on Earth where it will open.

and use a rotary tool or jigsaw to cut 2 rectangular holes to expose the display and the microcontroller's USB connector (Figure E). (The latter will be used for emergency power if the batteries have been discharged or the 50 attempts consumed.)

Assemble the Latch

I use different latch mechanisms in my boxes. For an engagement ring box (Figure J), I used a nail and a latch hook connected to a tiny "feather" servo.

In the elephant box, the servo is wired via a bent paper clip to a wooden dowel, all mounted inside the lid (Figure F). As the servo arm rotates, it drives the dowel laterally through a conduit, moving it into one of 2 positions — engaged or disengaged. When engaged, the dowel runs underneath the crossbar of an L-hook that's screwed into the base to extend up and over the dowel. The dowel snags the L-hook, preventing the box from opening.

To make the latch mechanism, drill a hole down the center of a 2"–3" long 1×1 block of wood, just wide enough for a dowel to slide through freely. Mount a sturdy L-hook into the box's base so it extends nearly to the roof. Cut the dowel sufficiently long so that it engages this hook when deployed.

Drill a tiny hole through one end of the dowel, about ¼" down from the end, so it can be linked to a servo arm with a piece of rigid wire or paper clip. Mount the latch assembly (Figure G) into the lid.

Solder the Electronics

The brain of a Reverse Geocache is the wonderful little Arduino microcontroller. Its friendly programming language makes it simple and fun to control its rows of I/O pins. The trick, of course, is to mate those pins to the sensors and actuators that play well with it, and that's why I built a custom Reverse Geocache "shield" PCB (Figure H) that plugs right into the Arduino's headers. You just solder a few header pins and connectors to the shield, and then attaching this project's GPS, button, display, and motor literally becomes a snap. See makezine.com/25/puzzlebox for a diagram of all component connections and wiring.

Whenever batteries are mounted inside a locked box, it's important to conserve power. That's why my current designs use the Pololu LV power switch, which enables the circuitry to turn itself off completely. After each "turn" the system shuts down, drawing virtually zero power until the next button push. With this power management, the batteries in the elephant box should last for years.

I mounted all the electronics inside the lid using hot glue and a couple of small screws. The shield sports a 3-pin right-angle header for the servo to plug into, a 4-pin female jumper that connects to the illuminated push button, a 6-pin horizontal connection header for the GPS module to plug into, and a 16-pin IDC ribbon cable that feeds the display (Figure I).

Upload the Software and Treasure

Once the physical construction is complete, it's time to turn your attention to software. You can download my basic puzzle box sample source from makezine.com/25/puzzlebox and the Arduino software from arduino.cc/en/Main/Software. From within the Arduino environment, you'll want to personalize the greeting text in the source, specify your own "magic destination," and tweak the motor settings. Plug a USB into the Arduino board, upload the modified code, and your new puzzle box is ready to go!

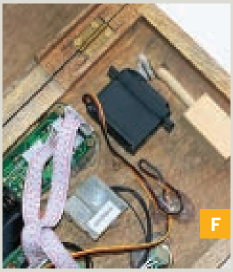
Or rather, not quite. One important task remains: deciding what treasure to hide in your cache. Consider your recipient carefully. For a fulfilling and meaningful adventure, three things — recipient, destination, and contents — should resonate with each other. Choose memorably. Choose beautifully.

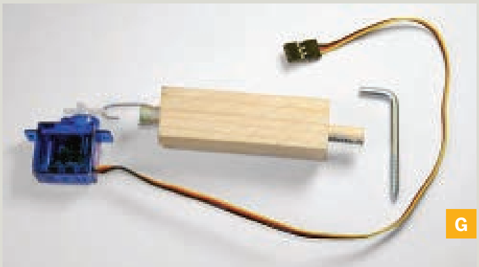
+ For the connection diagram and Arduino code, visit makezine.com/25/puzzlebox.

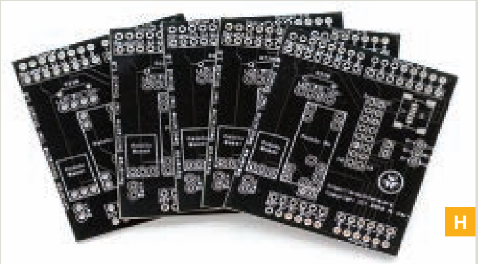
I've heard many more beautiful puzzle box-related tales, some of which I've posted at arduiniana.org/projects/more-puzzle-box-tales.

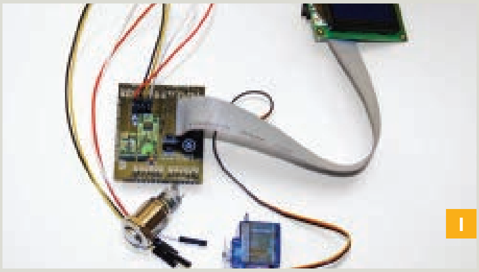
NOTE: I encourage DIYers to make their own puzzle boxes, but note that the technology is patent pending and the software is protected by copyright and licensed only for noncommercial use. So you shouldn't make one for sale. Also, "Reverse Geocache" is my trademark, so please don't use that name.

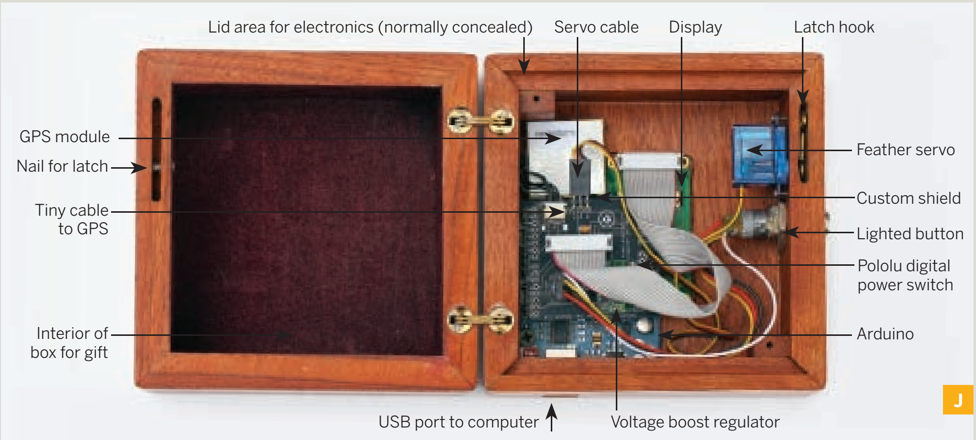
Mikal Hart (arduiniana.org) is a member of the Arduino 1.0 advisory team and a senior software engineer at Intel. He spends his days developing software and his nights playing French horn and tinkering with microcontrollers at his studio in Austin, Texas.











Photography by James Giroux and Mikal Hart

By Charles Platt

A Trippy Crystal Nightlight

Create any color, using 3 LEDs and the trick of pulse-width modulation.

» On a shelf above my bed stands something highly unusual: a giant, translucent, crystalline chunk of salt (Figure A), given to me by a friend who imports gemstones from Pakistan.

The salt was drilled with a hole to accept a 15-watt light bulb, but I decided to convert it using red, green, and blue LEDs that would create a mysterious, ever-changing spectral glow (Figure B). Hairy people wearing tie-dyed T-shirts might have described this as "trippy!" rather a long time ago.

Mysteries of Color Mixing

Why use red, green, and blue? Because by varying the intensities of these colors, we can create almost all the other colors in the spectrum. This is known as additive color mixing, and every color TV or computer monitor depends on it.

Most of us are familiar with subtractive color mixing, which is what happens when we blend colored paints or inks on paper. White light falling on the artwork penetrates the colors, which absorb some of the frequencies before the light is reflected back to the eye. The more colors you add, the more light they subtract, and the darker the mix becomes.

When we deal with colored light sources, the more sources we add, the brighter the combination becomes. Mix red, green, and blue, and they can create white. Mix just red and green, and you see yellow. The setup in Figure C can be put together quickly, with fascinating results. LEDs don't have exactly the right spectra and don't project an even spread of light, but you'll still see the possibilities.

How can we vary the intensity of each LED to create a full range of mixed colors? Unfortunately LEDs don't respond linearly to changes in voltage, but we can pulse them on and off. Long pulses with short gaps between them will make an LED seem brighter. Short pulses with long gaps will make it look dimmer. Naturally, the pulses must be fast enough to exceed the persistence of vision of the human eye.

MATERIALS

Components can be found at mouser.com except where noted.

Breadboard
Hookup wire, solid, 24 gauge

For the 555 timer version:
555 timer ICs, 1,000mW minimum (3) TLC555IP, TS555IN, or similar
Switching diodes (3) 1N4148 or similar
LEDs, 5mm Lumex high-brightness (3) SSL-LX5093XRC/4 red, SSL-LX5093UEGC green, and SSL-LX5093USBC blue (1 each)
Lens caps (3) Keystone 8665, to diffuse light
Resistors, minimum ¼ watt: 1kΩ (6), 20Ω (2), 100Ω (1)
Capacitors, minimum 12V, electrolytic or ceramic: 47μF (3), 0.01μF (6), 100μF (1), 0.1μF (1)
Potentiometers, 100kΩ linear (3) Vishay M63S104KB40 for board mount or Alpha RV170F-24-20K-B15-3 for panel mount, or similar
5V DC power supply Use any AC adapter with a 5V DC output, or use 3 AA or AAA batteries in series.
50kΩ photocells or photoresistors (optional) Hard to find? Try eBay.

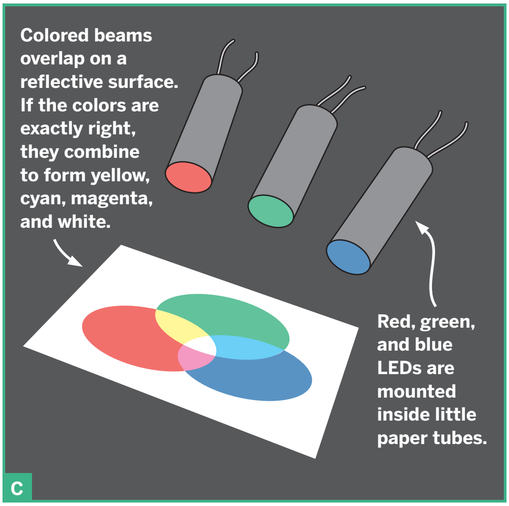
For the PICAXE version:
PICAXE 08M microcontroller SparkFun part #COM-08308, sparkfun.com
USB programming cable SparkFun #PGM-08312
3.5mm stereo audio jack SparkFun #PRT-08032
PICAXE Program Editor (Windows) or AXEpad (Mac/Linux) software free from rev-ed.co.uk/picaxe/software.htm
Resistors: 10kΩ (6), 22kΩ (3), 20Ω (2), 100Ω (1)
Capacitors, minimum 12V: 100μF (1), 0.1μF (1)
LEDs same as 555 timer version
Perforated board (optional) Twin Industries 7100-45 or similar; you cut it to size.
Sockets (optional) Mill-Max 801-93-036-10-012000; you snap to fit.

TOOLS

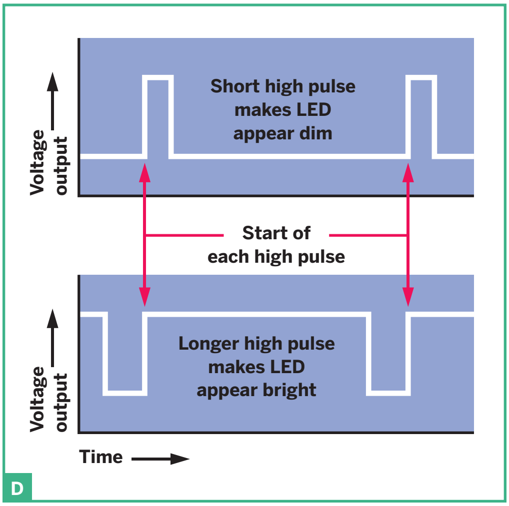
Wire strippers
Needlenose pliers
Optional: magnifier, multimeter, solder and soldering iron for perf board version

A

B



C



D

Fig. A: By day, an 8" crystalline chunk of salt. Fig. B: By night, a mysteriously trippy nightlight! Fig. C: Regrettably, not all high schools take the trouble to demonstrate additive color mixing. It's well worth the trouble. Fig. D: The perceived brightness of an LED can be adjusted by using longer or shorter rapid pulses. The frequency is measured from the start of one pulse to the start of the next, and remains constant.

This trick is called *pulse-width modulation*. By repeating the on-off pattern fast enough, we make the LED look as if it's on constantly, and by modulating the pulse width, we make it look dimmer or brighter.

To get a near-linear response, the on-pulse and the off-pulse should add up to a constant duration. In other words, the pulse frequency (the sum of on and off pulses) should remain constant, as in Figure D.

There are two easy ways to do this — with a 555 timer, and with a PICAXE microcontroller.

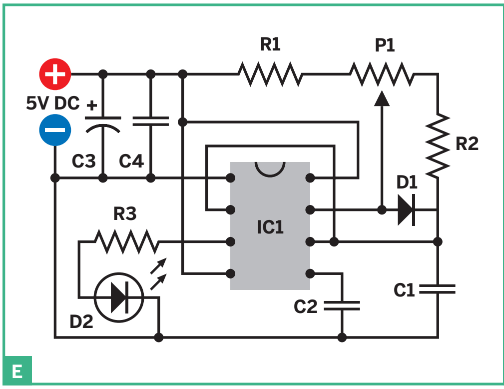
Your Father's Trippy Color Mix

As in my previous columns, I'll start with an "old school" setup: a simple 555 timer, although I have wired it in a novel manner. Check the schematic in Figure E (following page).

The "on" cycle of the timer is determined by the time it takes for positive voltage to charge capacitor C1. Current passes through R1, then through the left-hand side of potentiometer P1. It can now make a shortcut, through diode D1, to the capacitor.

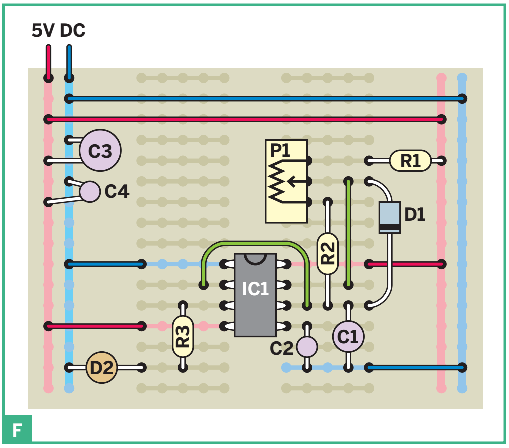
During the "off" cycle, C1 discharges to pin 7 of the timer. D1 blocks the current in this direction, forcing it to pass through R2 and the right-hand side of

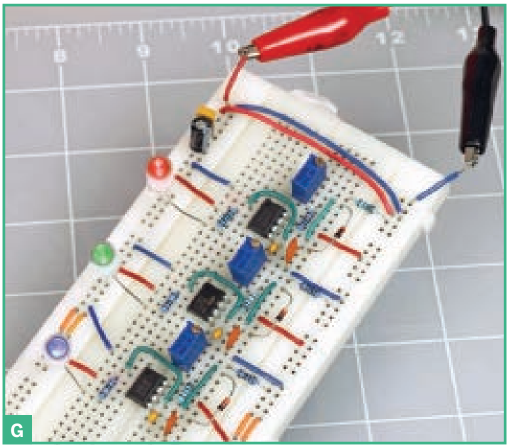
Photography and diagrams by Charles Platt



COMPONENT VALUES
D1: 1N4148 switching diode
D2: LED, see Materials list
IC1: 555 timer
R1 and R2: 1K
P1: 100K trimmer
R3: 20Ω for green or blue LED, 100Ω for red
C1: 0.01µF for operation, 47µF for testing
C2: 0.01µF
C3: 100µF
C4: 0.1µF

Fig. E: In this circuit, a 555 timer can generate pulses of varying length while maintaining a near-constant frequency. Capacitors C3 and C4 smooth the power supply and need not be duplicated for additional copies of the circuit.





P1. Since R1 and R2 are fixed (they're included just to protect the chip when P1 is at either end of its scale), the left side of P1 establishes the charge time and the right side establishes the discharge time.

Because the left side and the right side of P1 always add up to the same resistance, the cycle length will be constant while the charge and discharge times vary. You can use this circuit anytime you want to create varying pulses within a fixed frequency.

A breadboard layout is shown in Figure F, using a trimmer for P1 that plugs directly into the board. The first time you try it, use a 47µF capacitor for C1. The pulses will be very slow and easily visible as you adjust the potentiometer. When you're happy that the circuit works, substitute a 0.01µF capacitor for C1. The LED will now flash beyond your persistence of vision, and the potentiometer will function as a dimmer.

You'll need 2 more copies of the circuit to drive the remaining 2 LEDs. Figures G and H show a completed breadboard.

I used high-output Lumex LEDs that withstand more current than most, and need smaller series resistors (R3) than you might expect. Note that the green and blue LEDs in this series require different

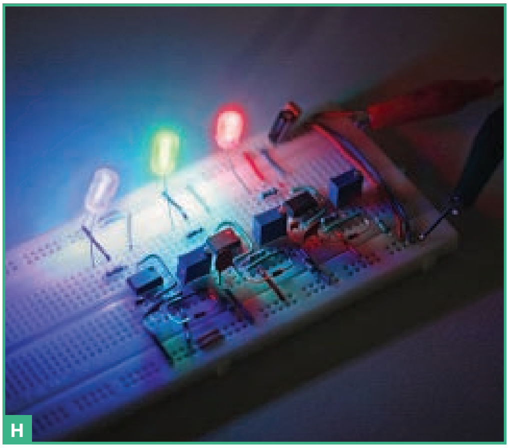
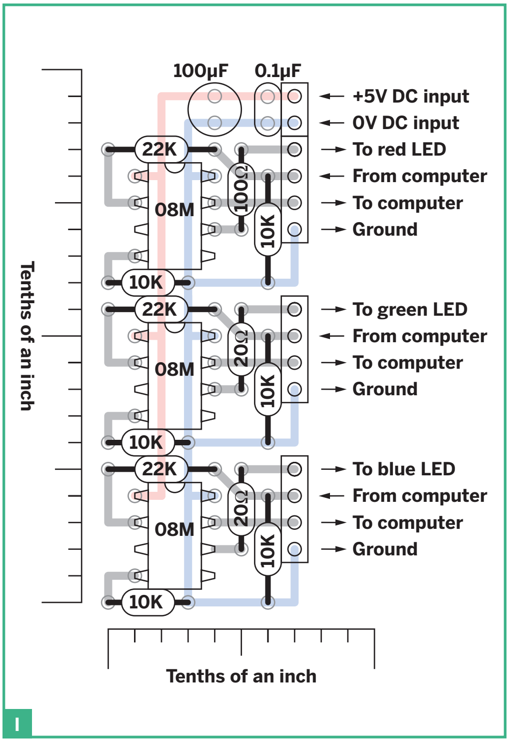


Fig. F: Breadboard layout of the schematic in Figure E, using the same components. Interior functionality of P1 is shown for clarity. Pale-colored conductors are hidden inside the breadboard.
Fig. G: Breadboarded version of the schematic in Figure E, in triplicate to drive 3 LEDs.
Fig. H: The color-mixing circuit in action.



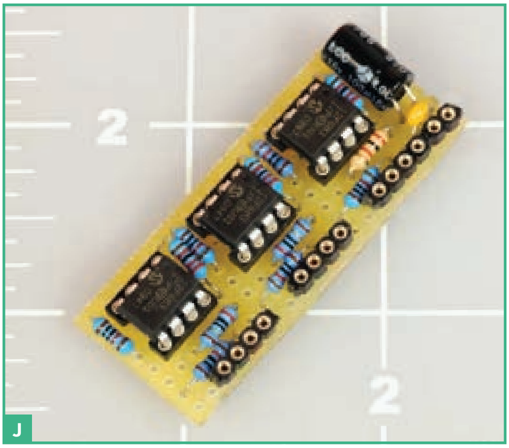


Fig. I: Mounted on perforated board, the microcontroller circuit can measure less than 1"×2", good for a possible jewelry project. It can run with 3 AAA batteries. Components are on the front of the board, pale-colored conductors on the back. Refer to the downloadable schematic from makezine.com/25/electronics for more details.
Fig. J: Fabricated version of the layout in Figure I. Sockets are used to receive power, to attach the LEDs, and to reprogram each microcontroller.

resistors from the red LED (see Figure E).

If you don't happen to have a giant salt crystal lying around, you can put your LEDs behind a piece of clear polycarbonate plastic, and sand it well to create a translucent screen. Use a crinkled sheet of foil to reflect the light before it reaches the screen, and the patterns should be pleasing. Or add multiple LEDs, since each 555 timer can drive at least 7.

Twiddle the knobs on your potentiometers to set the color that matches your state of mind each night.

Enhancements? For fun, substitute two 50K light-sensitive resistors, also known as photoresistors or photocells, for each half of each potentiometer. Mount them in a "keypad" under a little desk lamp, some distance from the colored LEDs. The shadow from your finger will change the photocell resistance, and will "magically" modify the colors.

Of course it would be more interesting for the colors to vary automatically and randomly. Since "randomly" is always a challenge for old-school components, it's time for some microcontrollers.

Nightlight on Full Auto

I'll use PICAXE microcontrollers because they're cheap and easy. The little 08M, which is sufficient for this project, costs about $3 each. See my book, *Make: Electronics*, or my previous column for more about the PICAXE, or visit rev-ed.co.uk/picaxe.

You'll need three 08Ms, because a single one won't process instructions fast enough to control all the LEDs. You'll find a schematic, a breadboard layout, and a program listing available online at makezine.com/25/electronics.

Copy and paste the program into the program editor and download it into each PICAXE. It will send pulses varying in length from about 1 to 10 milliseconds, allowing 10 levels of brightness, plus zero brightness when the LED is off.

To make the LEDs fluctuate out of sync with each other, edit the program to change the value for the `colorinc` constant for each PICAXE. Comments in the listing tell you how to do this.

You can squeeze the whole circuit onto a piece of perforated board smaller than 1"×2" (Figures I and J). Run it from 3 AAA batteries wired in series, enclose everything in a locket, and you have some attention-getting electronic jewelry.

Lastly, for a mega mood light, you could use three 28X1 PICAXEs, each driving 8 LEDs from its 8 output pins on a pseudo-random basis. This way, the brightness of each color would be determined by how many LEDs are switched on.

Charles Platt is the author of *Make: Electronics*, an introductory guide for all ages. A contributing editor to MAKE, he designs and builds medical equipment prototypes in Arizona.

By Lee D. Zlotoff

Zombie Attack!

The Scenario: You've noticed a few underground reports on the web of a spreading epidemic causing hordes of undead to rampage through suburban neighborhoods in search of new victims. You dismissed the postings as just hype to promote some new horror movie. But now you're home at night relaxing with a friend when you hear screams of panic. You look outside to see a throng of, well, zombies terrorizing your street — a pack of which is shuffling hungrily toward your house. *Good God, the internet was right!*

Grabbing your cellphone and laptop, you quickly retreat to the garage and lock yourself in, trying to shut out the methodical pounding on your walls and the moaning cries of "*Brraaaiiinnnnssss!*" Furiously surfing the web and calling, you learn that the police are overwhelmed but have secured a school two miles from your house from which they can evacuate the uninfected by helicopter. Too bad your car is at the repair shop tonight.

The Challenge: Trustworthy sources say that the mindless zombies can be kept at bay by flames of sufficient size and — worst come to worst — incapacitated by a forceful blow to the head. But it's clear from the shaking of your garage door that you have at most an hour before you face those suckers head-on. Bottom line: You're going to need whatever's in the garage to fight your way out and through the zombies to get yourself to that school on foot!

What You've Got: You have your set of tools (hammers, saws, screwdrivers, wrenches, etc.), a working sink, two towels, a first-aid kit with adhesive bandages and hydrogen peroxide, an acetylene torch, a chainsaw (out of gas), a propane grill (with a ¾-full tank), 50' of Class 315 PVC pipe, a framing nail gun with a box of 500 3½" framing nails, 3 full cans of oil-based ceiling paint, a bottle of turpentine, a case of 10W-40 motor oil, a six-pack of empty beer bottles, your camping gear (backpack, tent, sleeping bag, and a handful of lighters and matches), a disassembled lawn mower (half full of gas) and 3 bottles of 30W lawn mower oil, a cricket bat, and of course, some duct tape and your Swiss Army knife or Leatherman.

It's time to show those zombies what happens when they cross a real maker. Good luck and good hunting!

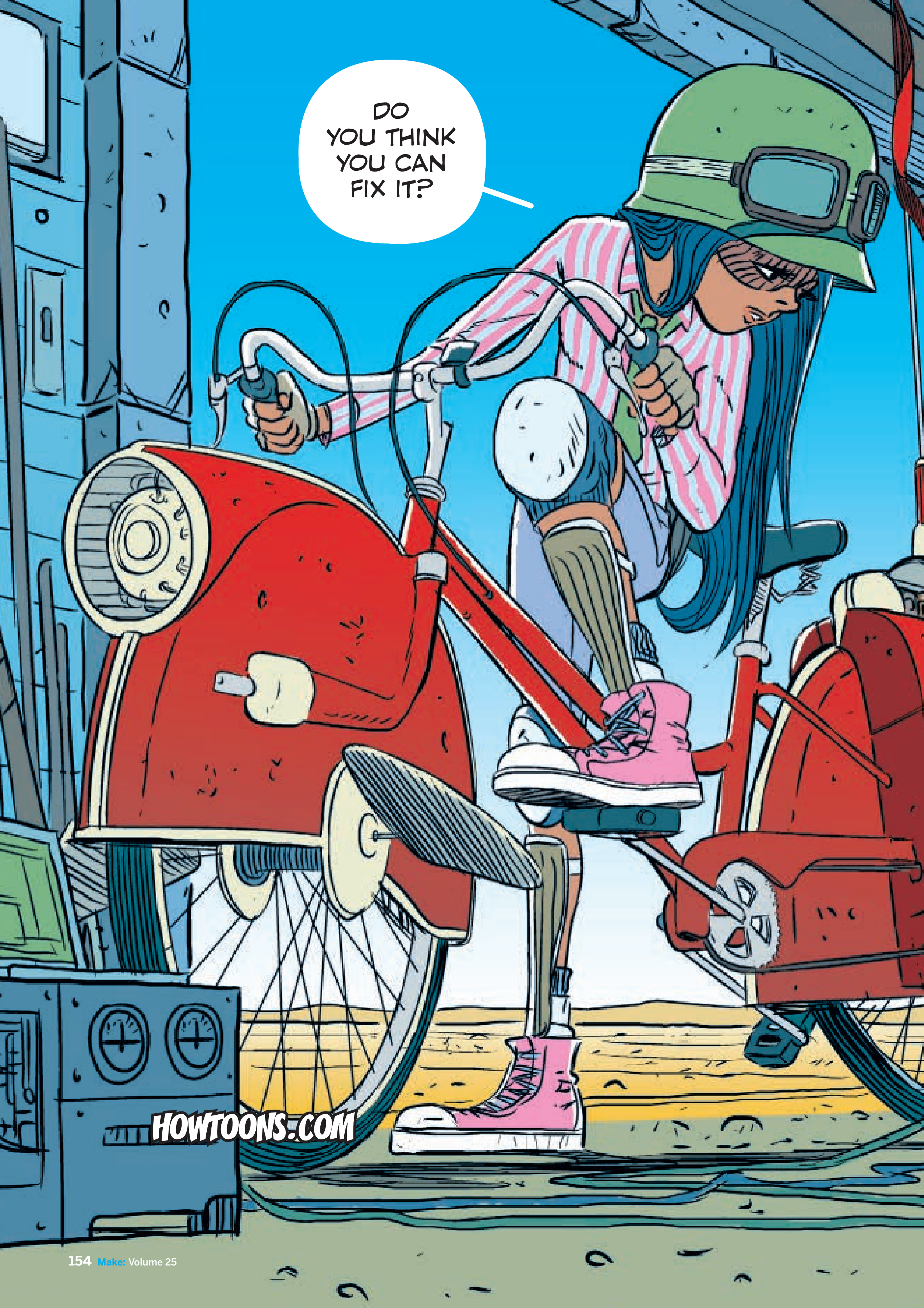
Send a detailed description of your MakeShift solution with sketches and/or photos to makeshift@makezine.com by June 3, 2011. If duplicate solutions are submitted, the winner will be determined by the quality of the explanation and presentation. The most plausible and most creative solutions will each win a MAKE T-shirt and a *MAKE Pocket Ref*. Think positive and include your shirt size and contact information with your solution. Good luck! For readers' solutions to previous MakeShift challenges, visit makezine.com/makeshift.

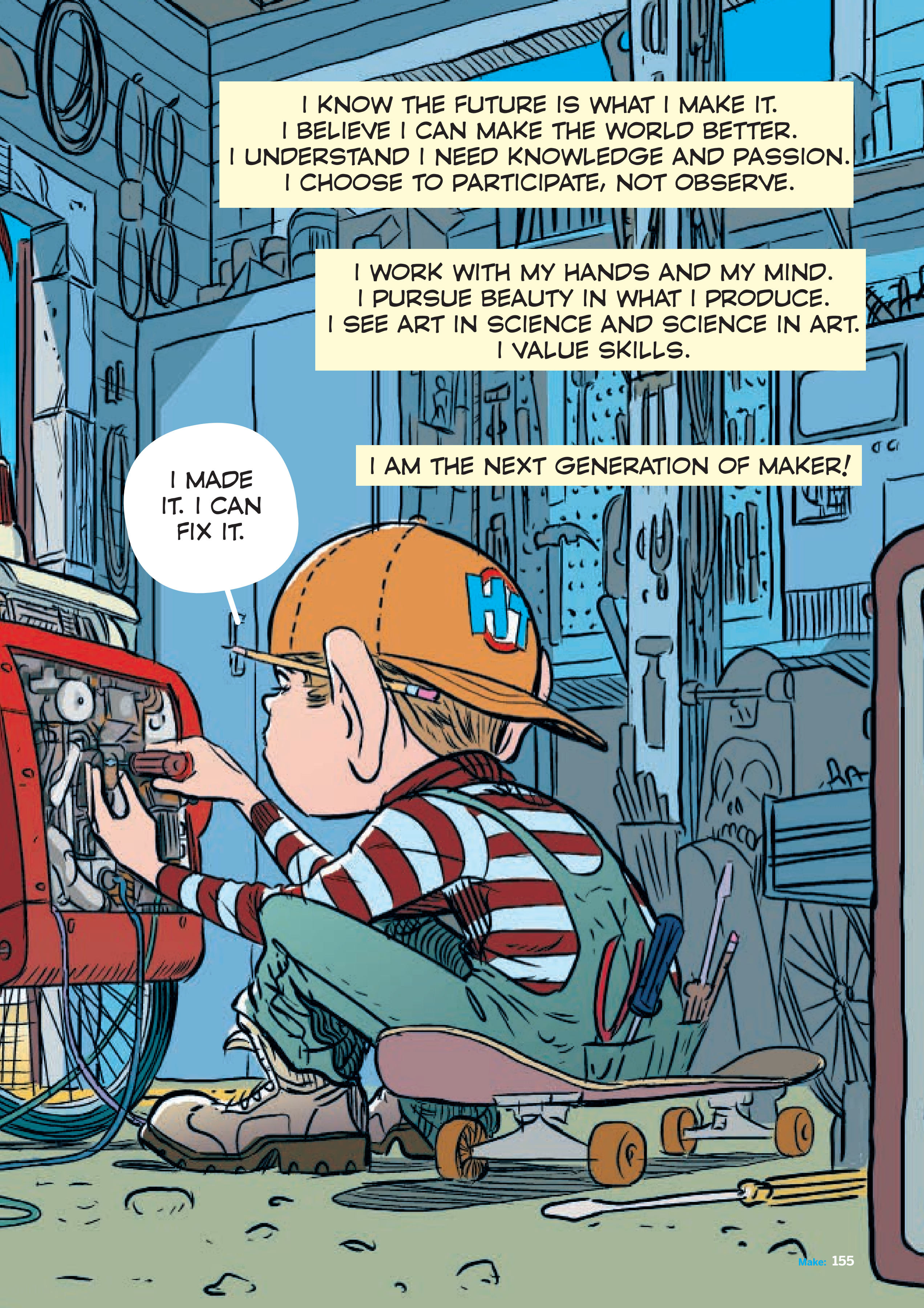
Moving Online

We've been posting the winning, reader-submitted solutions online at makezine.com/makeshift since we started the MakeShift competition more than 5 years ago. And now we're doing an online upgrade, offering everyone the chance to vote on the best entries, making it easier to submit your own solutions, and highlighting the analysis our authors and editors give after each episode of MakeShift. Be sure to check it out this summer, when MakeShift goes all online, all the time, at makezine.com/makeshift!

Lee David Zlotoff is a writer/producer/director among whose numerous credits is creator of *MacGyver*. He is also president of Custom Image Concepts (customimageconcepts.com).

Photograph by Jen Siska


DO YOU THINK YOU CAN FIX IT?
HOWTOONS.COM


I KNOW THE FUTURE IS WHAT I MAKE IT.
I BELIEVE I CAN MAKE THE WORLD BETTER.
I UNDERSTAND I NEED KNOWLEDGE AND PASSION.
I CHOOSE TO PARTICIPATE, NOT OBSERVE.
I WORK WITH MY HANDS AND MY MIND.
I PURSUE BEAUTY IN WHAT I PRODUCE.
I SEE ART IN SCIENCE AND SCIENCE IN ART.
I VALUE SKILLS.
I AM THE NEXT GENERATION OF MAKER!
I MADE IT. I CAN FIX IT.

Build a village, clock your speed, wear sunglasses at night, craft a canoe, and geek out your garb.

TOOLBOX

Grizzly G0516 Lathe/Mill Combo

$1,195 grizzly.com

I'm sure there are a lot of tools that would like to claim the title of the most important tool in a maker's workshop, but I believe that right after a good imagination and some free-ranging creativity come heavyweight tools like lathes and milling machines.

Due to my small workshop and limited budget, I chose a combination machine from Grizzly Industrial. The G0516 is a good representative of this type, and I can report that, while there are limitations with a small combo machine, once you learn its true capabilities, you'll be surprised at what you can do. I've been running this little beauty since summer and for the most part have been quite happy with it.

The lathe uses a series of belt-driven stepped pulleys and change gears to vary the spindle speeds and carriage feeds. Although not as convenient as a gearbox, I quickly learned how to set the required speeds and feeds to create smooth cuts. The milling head has its own variable-speed milling motor with a high/low gearbox, and uses a slotted accessory table that replaces the tool post for milling operations.

One of the notable advantages of the Grizzly G0516 is that it comes with an impressive list of standard accessories that are usually considered extra cost options. My machine came with both 3- and 4-jaw chucks, faceplate, drill chuck, dead centers, 4-way tool post, quick-release tool post, quick-change collet set, and milling vise.

The only things you'll need to add are a set of indexable carbide tool bits and some end mills. After that you can make chips to your heart's content.

—Chris Singleton

Want more? Check out our searchable online database of tips and tools at makezine.com/tnt.
Have a tool worth keeping in your toolbox? Let us know at toolbox@makezine.com.

USBCell

$18 for 2 usbcell.com

These nifty batteries look like standard AA cells, but if you pop off the end cap, a USB plug is revealed for recharging. You can monitor its progress via an LED indicator. It lights up when charging, flashes after a 90% "quick charge," and goes dark when done. Although not as robust as a lithium battery, the USBCell's 1.2V/1,300mAh output is comparable with most rechargeable batteries. —*John Baichtal*

RSK Mk 1 Pocketknife

$120 dougritter.com

The RSK Mk 1, designed by Doug Ritter for Benchmade, is the most sane "survival" knife I own. It's a 3.44" bladed folder (including the usual Benchmade mechanism) with a S30V stainless steel blade and a Zytel grip. Ritter's a smart man. To him, a good survival knife is the one that will always be in your pocket; then, when something bad happens, it's there to help you survive.

On this principle, he designed the RSK series to be really useful pocketknives: no special blades, no serrated edges; just a good, tough, generalist blade with a deep belly and a good point. I've carried mine for about four years now and use it daily for normal cutting chores.

One of the few special features is that the knife is completely ambidextrous. Not only can you open it one-handed with either hand (nice if you break an arm or are holding something), you can also swap the pocket clip for left-handed carry. But anyone with a working thumb can open the blade.

—Frank Landis

Liberator Open Source CEB Press

$8,000 openfarmtech.org/wiki/Compressed_Earth_Brick_Press

If I were allowed only one tool with which to build a village, I would want Open Source Ecology's new CEB press, the Liberator v2.0.

The Liberator makes compressed earth blocks (modular construction unit) using the dirt at the building site (ubiquitous raw material). Laying these blocks is an easy way to construct just about any building or feature in a homestead or village. With a little mortar, they can form walls, floors, patios, decorations, and even roofs.

The machine's design is open source and accessible, allowing for full customization. You can adjust block size, compression ratio, rate of output, and anything else, based on your local conditions and materials.

The controls use Arduino and RepRap components, and everything is built from off-the-shelf items, making repairs and modifications a piece of cake. You could maintain this machine for generations. It's a must-have tool for anyone in construction.

—*Abe Connally*

Pocket Radar

$199 pocketradar.com

Speed radar technology hasn't changed much since its inception in 1954. The bulky "gun" you're used to seeing is pretty much standard — until now. Invention Planet, an R&D organization run by three engineers, tackles previously ignored disruptive tech. Enter Pocket Radar.

Though it looks more like a remote control than a gun, it's accurate within 1mph, fits in the palm of your hand, and purportedly gets 10,000 readings on one set of AAA batteries. The tech is "game changing" indeed, considering its obvious sports training applications. Personally, I love having it in my jacket pocket at the ski area. No more empty bragging rights! *—Goli Mohammadi*

Cirago NUS1000

$45 cirago.com

This networked attached storage box, slightly larger than a deck of cards, features an Ethernet port and four USB 2.0 ports on the back. It's essentially a hub that can be plugged into your network, integrating any USB-compatible hard drive. Similarly, you can do all the usual USB hub things like streaming music through the integral iTunes server or printing to a network printer. That alone is pretty cool, but the Cirago packs robust security software, too — and it's plug-and-play out of the box. *—JB*

Samson Go Mic

$50 amzn.to/usbgomic

This neat little USB condenser microphone is designed for netbook- or laptop-driven creativity. It features rich, professional sound more commonly seen in full-sized mics, combined with excellent portability.

When not in use, the Go Mic folds up and stores in an elegant bundle that will surely find a place in your laptop bag; it comes with a zippered case for added protection.

But when you need it, it deploys into a robust dual-pattern mic, totally killer for ad hoc recording — podcasts, interviews, VoIP chats — facilitated by the onboard headphone amp that reduces latency.

The base of the Go Mic is a plate of die-cast zinc that's solid enough to keep the mic stable. However, the base doubles as a spring-loaded clip that lets you attach the mic directly to a laptop. In the latter case, be sure to invest in a shorter USB cable because the 8-foot cord that comes with the mic is probably far longer than you need. *—JB*

Dremel Trio

$100 dremel.com

With more power and better stability than a standard rotary tool, the new Dremel Trio offers a number of features and capabilities — including routing, sanding, and circle cutting — that make it useful for the creative maker in all of us.

The basic Trio kit includes a combination edge guide/circle-cutter jig, a plunge-cut bit, a ¼" router bit, and a drum-sanding arbor with sleeves.

Notably, this tool has a larger-diameter shaft than a regular rotary tool to better withstand the heavier side-loading it'll be exposed to.

Its wide base and standard cutter bit makes clean plunge cuts with ease, and the supplied edge guide/circle cutter helps make some of the cleanest router cuts I've ever seen. The tool's light weight and dust-collection capability has turned it into a great choice when I'm doing light routing and edge sanding on plywood, plastic, and foam. *—CS*

KID REVIEW

AR.Drone

$300 ardrone.parrot.com

Parrot incorporated some amazing technology into the AR.Drone. The quadricopter (which has four propellers as opposed to one) is easy to navigate thanks to two cameras — one facing forward, the other facing down. The drone is able to tell when it flips over by sending out ultrasound waves that "see" the ground.

One of the most remarkable things about this device is that it can be controlled by an iPod touch, iPhone, or iPad — although I advise you to master it indoors before you take it outside and remove the indoor hub with its propeller protectors.

—Robert M. Zigmund

Vuzix Wrap 310XL

$250 vuzix.com

The Vuzix Wrap 310XL heads-up display is undeniably cool. It's basically a pair of opaque glasses with mini 428×280, 24-bit color LCD displays sitting right over your eyes — the manufacturer claims that it's equivalent to viewing a 55-inch 16:9 TV from a distance of 10 feet. You can use the unit to watch a TV show on the train without nosy passengers looking over your shoulder or catch a movie at night while your partner is asleep. You can even plug them into the VGA port of your computer and use the glasses as a monitor. (RCA and iPod/iPhone/iPad connectors come in the box; USB and VGA adapters are available from the manufacturer's website.)

—JB

Alibre Design Personal Edition

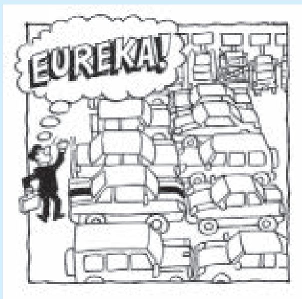
$99 alibre.com

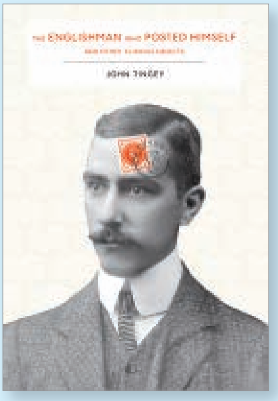
Most 3D parts begin life as 2D sketches that are pushed and pulled into 3D models on your computer screen. Some programs use a kind of wire-mesh frame to create objects (Blender, Google SketchUp, Rhino), and others use a more direct mathematical language (OpenSCAD), but the best ones use actual solid shapes (Alibre Design, SolidWorks, Inventor, Pro/Engineer).

What really differentiates the solid modeling programs is their ability to create assembly files that include multiple parts, which relate to each other just the way they relate in real life. This way, one part can represent an off-the-shelf motor, another part can be exported for 3D printing, and another can be made into a drawing to send to a laser cutter.

Of all the solid modeling programs, the only one that offers a commercial version (not a student license with watermarked parts) for less than $4,000 is Alibre Design, and it's way less — just $99. Alibre PE has all the stuff you need and leaves out the stuff you don't. Unless you're a power user, you'll never miss the fancy stuff.

—Dustyn Roberts





Return to Sender

***The Englishman Who Posted Himself and Other Curious Objects* by John Tingey**
$25 **Princeton Architectural Press**

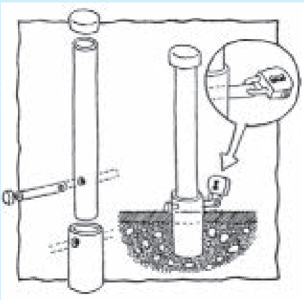
This slim but gorgeously produced book is the biography of W. Reginald Bray, arguably the father of mail art. Long before punks and art hipsters in the 1980s and 90s started using the world's postal systems as a conversational art medium, this unassuming Englishman, an accountant, was pushing the envelope (you know I had to go there) of the Royal Mail.

At the turn of the 19th century, Bray got ahold of the *Post Office Guide*, detailing all postal regulations. He decided to "hack" the system — to see what shenanigans he could get away with. Soon, he was mailing produce (onions, turnips); a crocheted letter/envelope his mother had made for him; postcards with rebus-written puzzle addresses; letters to politicians with just their photos as the address; letters to lighthouses, trains, and ships at sea; and letters to "Any Resident of London."

Most cards and letters had some way of alerting him of success, the results of which were meticulously recorded in a ledger as only an accountant might manage. Bray even mailed his bicycle, his Irish terrier, and as the title states, himself.

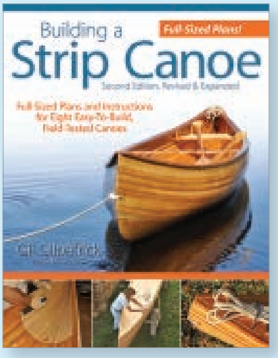
His story is charming, the reproductions of the work are beautiful, and the book art is what you'd expect from Princeton Architectural. I was involved in mail art in the 1990s and mailed some absurd stuff myself, but I'd never heard of this man nor known how old this fun form of self-expression was.

—Gareth Branwyn



Paddle Your Own Canoe

***Building a Strip Canoe, 2nd Edition* by Gil Gilpatrick**
$25 **Fox Chapel Publishing**



My dad and I decided to build a strip canoe in our garage for a winter project.

Having never attempted such an in-depth build, we turned to Gil Gilpatrick's book. His 30+ years of experience teaching and building canoes is displayed throughout the book. This revised and expanded edition provides pages of wisdom and guidance.

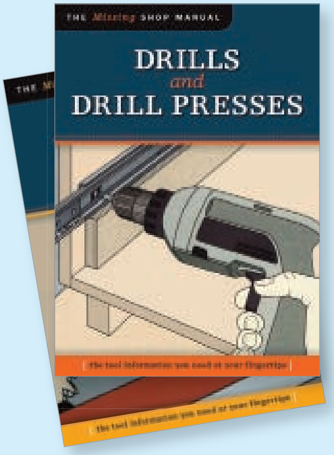
The beginning chapters cover hand and power tools you'll need for the project; in the back of the book are eight full-sized templates that Gilpatrick has refined over the years to help you get started.

Beyond the construction of the strip-built canoe, Gilpatrick also provides valuable information about repairs and maintenance, paddle construction, and canoe seat building and weaving.

What sets this book apart from other instructional canoe build books are the large, colorful pictures and detailed diagrams that complement the text. For me, the pictures are the most important component of an instructional book, as I tend to spend a lot of time focusing on the images and schematics during the actual building process.

This book will not only help you in the build process, but it's one you'll want to keep with you throughout the lifetime of your canoe.

—Nick Raymond



Serious Series

***Back to Basics* series**
$20 **Fox Chapel Publishing**

***The Missing Shop Manual* series**
$10–$13 **Skills Institute Press**

I've had a passion for instructional books for more than a decade. I've bought out the how-to section of nearly every secondhand bookstore in North America, and my attic is weighed down by bookshelves of "How To Do Anything" books.

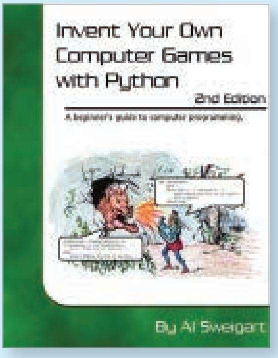
Far from jaded, I still think there's room for improving this genre. My new standard has been set by two series: *Back to Basics* and *The Missing Shop Manual* series.

Back to Basics includes titles like *The Woodworker's Guide to Joinery*, *Fundamentals of Sharpening*, and *Setting Up Your Workshop*.

The *Missing Shop Manuals* are handy little booklets on everything from your router to your table saw. Their clear — almost comic-book — images are perfect. They run the fine line just this side of condescending and didactic, to successfully make a range of manuals that help even the most proficient craftspeople learn a little more about their tools and what's possible.

Both series cover woodworking tools brilliantly — I love the tool sharpening guide most of all. I only hope they're so successful that they can eventually cover every tool out there. I recommend both series to anyone. I'd like the complete set for my library (soon to be on iPad, I hope).

—Saul Griffith



Snake Charmer

Invent Your Own Computer Games with Python, 2nd Edition
by Al Sweigart
$25 **inventwithpython.com**

Python continues to grow in popularity among makers because it's open source, cross-platform, and well maintained.

What dissuades many from learning it as a second (or third, or fourth) computer language is finding a text with a good learning curve. Either they oversimplify and don't adequately cover complex concepts, or they take a theoretical approach and fail to maintain reader interest.

This book avoids both of those pitfalls, presenting an intuitive Python 3 tutorial. While geared toward the absolute beginner, the presentation of concepts moves quickly from "Hello, world," through debugging, to the implementation of AI, and culminates with the Pygame animation library used in a graphics-based game.

Throughout the process, Python functions and syntax are presented in a concise, logical pattern with a full explanation of all code examples. The text is supported with a comprehensive website featuring free downloads of the book's text and source code.

I've previously coded in other languages, and I found this book a convenient way to get up to speed with Python 3. The concepts presented inspired me to think beyond the text to explore code libraries for network gaming. It's a great way to learn a cool hacker language.

—L. Abraham Smith

MAKER SARTORIAL

The clothes that make the maker. By Gareth Branwyn

OUR READERS may not be a particularly fashion-conscious bunch, but ask any maker what he or she is wearing and carrying, and you'll likely get a very long, precise explanation of functionality, durability, and the operating methods of everything.

Geeks may not be less particular about clothing, accessories, and personal items after — they're just more focused on functionality over style. On Make: Online, we asked our community to tell us something about what they wear and carry and why. Here's a sampling. (You can read the rest at makezine.com/go/sartorial.)

Ben Davis Shirts

Keith Hammond, MAKE's managing editor, writes: "I'm a longtime fan of Ben Davis short-sleeve shirts (bendavis.com, $25) — bombproof, grease-resistant work fabric, cut loose, with not one but two shirt pockets, one with a pencil slot, so my little notebook's always at hand.

"I like the dark colors for dirty jobs, and for bike commuting — chain grease disappears and then washes out easily later.

"Bonus: the OG (original gorilla) on the label."

Outerwear

MAKE contributor John Baichtal loves the Scott eVest Ultimate Hoodie Microfleece (scottevest.com, $70): "It has 11 pockets, special channels for gadget wires, loops for earbuds, magnetic pocket closures, and a secret pocket!"

Cargo Pants

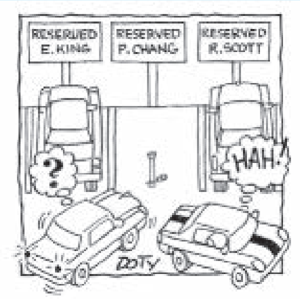
A lot of people said they're fond of cargos. MAKE pal Kent Barnes swears by Blåkläder pants (blakladerusa.com, $45–$80): "They take kneepad inserts, which are very important to me."

M-51 Engineer's Field and Laptop Bag

Jason Schlauch writes: "I'm still on a quest to find the perfect man-bag. I've come one step closer with this laptop bag ($30 on amazon.com). It has two dozen pockets that hold and organize my flashlight, multi-tools, cables, notebooks, and more. I still have a few compartments empty!"

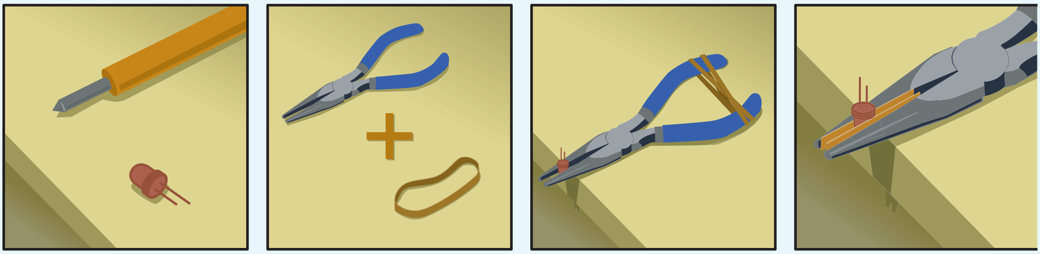
Combat Boots

Not surprisingly, a lot of people recommended black, military-type combat boots. I have a trusty pair of Doc Martens myself. Mark Adams, of HacDC, says: "Indestructible and really comfortable. With a good pair of wool socks and a cotton pair as a liner, you can withstand the cold and walk for miles. You have to take care of 'em, but they'll last forever."



Tricks of the Trade By Tim Lillis

Cool your clamp.



Andrew Lewis, journeyman, maker, and creator of the Upcraft.it blog, shared this great trick to make a DIY vise/clamp with heat sink properties.

You'll need some pliers, and a rubber band. Yup, that's it. Maybe some tape.

You can use this to hold your work, and it's especially useful if you're generating heat and don't have a "third hand" holder.

Add some copper tape in the jaws if you want a smoother surface that will still conduct heat away, or regular tape if you just want a smoother surface.

Have a trick of the trade? Send it to tricks@makezine.com.

Carhartt Double-Front Work Dungarees

Men's and Women's: $45–$50 carhartt.com

I found out the hard way that regular old blue jeans are ill-suited for working around fences and old barns — there are too many nails and edges to catch and rip. My cousin the carpenter counseled, "You need Carhartt Double-Fronts!"

These dungarees are *tough*! Made of heavy cotton duck material with triple-stitched main seams, they're exceptionally comfortable despite the heavy weight (the women's sizes even have a fitted waistband — no gaping). They have scads of useful pockets for everything from your space pen to your hori-hori.

There's a double panel of fabric in the front, from the thigh to just below the knee — right where you need the reinforcement. You can even buy "permanent" kneepads to insert.

I originally balked at the price, but finally decided to try a pair — and I've been wearing them on the ranch every single day since.

—Terrie Schweitzer

John Baichtal is a contributor to MAKE, makezine.com, and the GeekDad blog on wired.com.

Abe Connally is an off-grid adventurist based in Mexico. velacreations.com

Saul Griffith designs electric cargo tricycles at onyacycles.com.

Frank Landis is an environmentalist and writer living in Southern California.

Tim Lillis is a freelance illustrator and DIYer.

Nick Raymond is one of MAKE's awesome engineering interns.

Dustyn Roberts is a doctoral student in mechanical engineering and founder of Dustyn Robots (dustynrobots.com).

Terrie Schweitzer lives in a yurt on a goat ranch in Sonoma County, Calif., and digs permaculture.

Chris Singleton is a product designer, inventor, writer, and maker living in Cambridge, Ontario, Canada.

L. Abraham Smith works with open source hardware and software development every chance he gets.

Robert M. Zigmund is 12 years old, loves his subscription to MAKE, and has a blog at eztechreviews.blogspot.com.

Have you used something worth keeping in your toolbox? Let us know at toolbox@makezine.com.

TOY INVENTOR'S NOTEBOOK

Amazing Maker Marble Maze

INVENTED AND DRAWN BY
Bob Knetzger

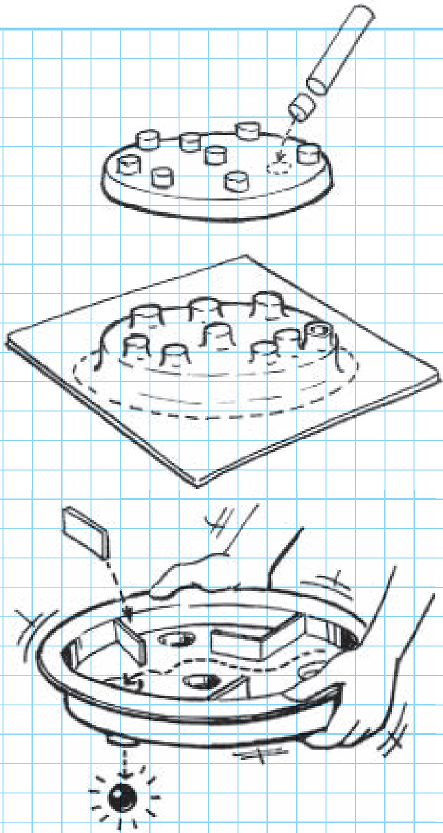
In MAKE Volume 16, I showed you how to mold thermoformed plastic parts with the "Kitchen Floor Vacuum Former." (Here's the how-to and video on Make: Projects: makezine.com/go/vacuform.) You can use that MAKE project to make this project!

Make a form by cutting a 12" disk out of ½" plywood or particleboard. Cut a ¾" wood dowel into ¾"-long pegs. Arrange the pegs on the disk to create a marble maze obstacle course. Each vertical peg on the male form will create a marble pit on the inverted vacuum-formed part.

Vacuum-form the plastic, then trim it to make a rimmed tray. Cut off the bottom of one pit to make a "finish" hole. Glue on small strips of plastic to create a path of maze walls around the pits.

Test it: tilt and tip the tray to get your marble across the maze, avoid the pits, and go down the hole. Too hard? Reposition the walls until it's just challenging enough!

Make up your own games. Race against the clock. Mark each hazard pit along the path with ascending point values — the further you get, the higher your score. Use more than one marble at a time. Mark the pits with different colors — get colored marbles into the matching pits to win.

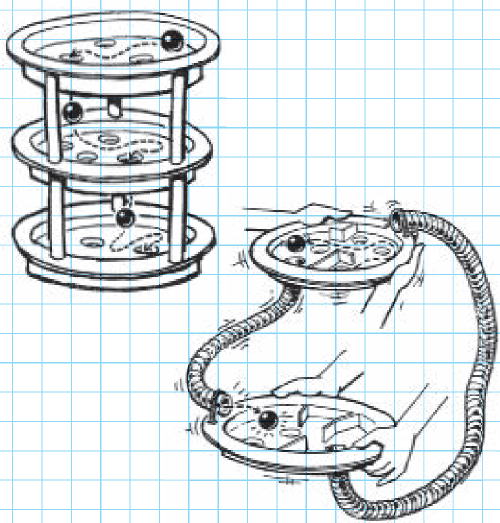


Take It Further

Make a multi-level stacked maze!

Or, for multi-player marble maze action, connect the trays with long, flexible tubes (I used scrounged air tubes from a hospital ventilator). Connect one end of the tube to the finish hole of one maze, then zip-tie the other end to the rim of the other maze. Get your marble through the maze and down the hole, and then *lift* your maze to send the marble through the tube and out into the other maze for the next player.

Connect more mazes in series to make a giant, multi-player relay maze! Or connect 2 mazes with 2 tubes for back-and-forth action. Each player gets 3 marbles: first one to get all their marbles into their opponent's maze wins!

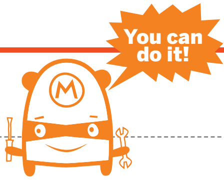


Bob Knetzger is an inventor/designer with 30 years' experience making all kinds of toys and other fun stuff.

DANGER!

Lick a 9-Volt Battery

By Gever Tulley with Julie Spiegler



Taste electricity.

HOW-TO

You're about to give yourself a tiny shock. It won't exactly hurt, but it will feel strange.

1. Hold the battery in your hand, with the terminals facing up.

2. Stick out your tongue.

3. Take a deep breath.

4. Think about kangaroos hopping around in a field of flowers. This is really just to heighten the anticipation — you can think about anything you want.

5. Quickly touch the metal terminals of the battery to your tongue.

6. Try again, but hold it there for 1 full second.

How would you describe the sensation to someone who has never done it? Does it have a taste, or is it something else?

If you'd like to experience something similar, chew on a wad of aluminum foil for a few seconds (be sure not to swallow any!). The foil will create a weak electric current when it contacts the acid in your saliva. If you have any fillings, you may experience an odd tingling in your teeth as the metal in the fillings conducts the electricity to the nerves nearby.

REQUIRES
9-volt battery

Duration: Short **Difficulty: Easy**

WARNING

Do not hold the battery to your tongue for more than a few seconds at a time.

SUPPLEMENTARY DATA

Normally, the nerves in your tongue are activated by tiny chemical receptors in your taste buds. The surface of your tongue is divided into different specialized regions that are tuned to notice specific flavors. The battery has no specific flavor of its own, but the electrical current that runs between the terminals activates a random collection of nerves on your tongue, causing you to experience a sensation of exaggerated, but nonspecific, taste.

The tongue may be one of the earliest sense organs to evolve. You can imagine how important it might be, to an early multicelled creature floating around in the primordial soup, to be able to taste things before eating them. In humans, the tongue is wired directly to the brain through little holes in the skull, bypassing the spinal cord completely.

Early batteries may have been made in clay pots with copper and lead plates. Evidence of these was found in the ruins of a 2,000-year-old village near modern Baghdad. Archeologists suggest that lemon juice could have been used as the electrolyte, and recent reconstructions confirm that it would have generated electricity.

Illustration by Gever Tulley

Excerpted from *Fifty Dangerous Things (You Should Let Your Children Do)* by Gever Tulley with Julie Spiegler (fiftydangerousthings.com). Gever is the founder of Tinkering School (tinkeringschool.com), a camp where kids get to use power tools and be trusted.


Spinneret Web Server
The Spinneret Web Server (#32203; $49.99) is a low-power, low-cost embedded web server development platform. Use the Spinneret to connect most any electronics project to a local Ethernet network or to the internet. With an onboard multicore Propeller microcontroller it can be directly programmed to perform many tasks, all while consuming less than one watt of power. A microSD card socket and real-time clock controller are built in, for ample file storage of web page content, files, and logs. Visit www.parallax.com/go/spinneret for details.
open source
Spinneret Web Server Design Contest
$5,500+ in cash prizes! Design a project using the Spinneret Web Server that performs a useful function and demonstrates your technical abilities.
Last day to request a Project ID is 05/31/11.
Details: www.parallax.com/go/spinneret
PARALLAX INC
www.parallax.com
"ParallaxInc" on Twitter, Facebook, & YouTube
Order the Spinneret Web Server at www.parallax.com/go/spinneret or call our Sales department toll-free at 888-512-1024 (M-F, 7am-5pm, PST).
Friendly microcontrollers, legendary resources.™
Prices are subject to change without notice. Parallax, the Parallax logo, Propeller and Spinneret are trademarks of Parallax Inc.


LASER IT!
Cut it. Engrave it. Mark it.
Laser Engraving, Cutting and Marking Systems from Epilog Laser
From creating and personalizing 3D models, to engraving photos on keychains, to marking high-tech gadgets, our laser systems create the products you see here and more!
zing
LASER
Moooo!
Laser Systems Starting at $7,995
Visit epiloglaser.com/make.htm for more information and to receive your brochure kit with engraved and cut samples!
EPILOG LASER
1.888.437.4564
sales@epiloglaser.com
MADEINUSA

HEIRLOOM TECHNOLOGY
By Tim Anderson

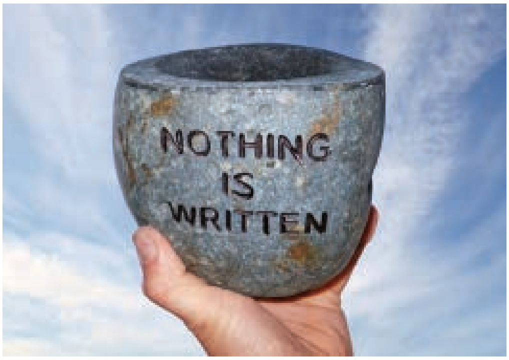
Carve Inscriptions in Stone

Send a message to eternity.

“Look on my works, ye mighty, and despair!” So read the words carved in stone in the poem “Ozymandias.” Based on an actual Egyptian statue, that carved inscription outlasted the statue and indeed the empire that created it.

Do you have a message you'd like to pass on to future generations? Here's how to carve it in stone. The same techniques work to carve inscriptions and designs in glass, ceramic, and metal.

It's amazingly easy and cheap. The rotary tool and burrs you'll need (shown in Figure A) can cost as little as $20 online.



MATERIALS AND TOOLS

- Printed paper pattern of your design
- White glue
- Stone to carve
- High-speed rotary tool such as a Dremel
- Diamond burr, ball-shaped
- Diamond burr, round-edged disc aka wheel
- Respirator
- Safety goggles
- Hearing protection
- Towel
- Wire brush
- Permanent marker (optional)

1. Print a pattern.

Print out your design in various sizes and figure out where you'll carve it. I'm inscribing the stone bowl that I carved in my previous column (*see MAKE Volume 24, page 168*).

“Nothing is written” is a quote from the movie *Lawrence of Arabia*. It means that fate is not destiny. And what, you may ask, does that mean? Perhaps future generations are destined to solve it.

2. Apply the pattern.

Smear white glue all over the back of your pattern so it's good and wet. Glue it to your stone (Figure B). The water-based glue softens the paper and lets it conform to the irregularities of your stone.

3. Carve the straight strokes.

Put your stone on a folded towel so it won't roll around. Put on your goggles, respirator, and hearing protection, and brace your hands well against the workpiece so the cutter doesn't jump (Figure C).

For my carving, the only burrs I needed were the round-edged diamond wheel and a ball burr.

Use the wheel to carve the straight strokes of the letters. Then use the ball burr to carve the ends of the straight strokes that the wheel couldn't get to.

4. Carve the curved strokes.

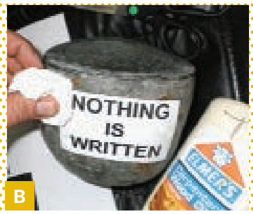
The cutting direction matters: the ball burr cuts better and jumps less in some directions. As you work, rotate the stone so you have a good angle for each cut. Again, brace your hands well, as shown in Figure C.

Having the paper pattern glued on the stone protects the stone from gouges if the cutter jumps.

5. Remove the pattern.

Soak the pattern and scrape it off. Use a wire brush to get the remains of paper and glue off.

Photography by Tim Anderson



6. Embellish (optional).

Use a permanent marker to darken the text for better contrast. Some stones have good contrast already and won't need this.

Magic markers have many uses. I just saw a worker in an antique rug store marking an entire rug to make the colors more vivid.

And that's it. The next time someone tells you, "It's not written in stone," you can surprise them! Future archaeologists may be surprised as well.

Tim Anderson (mit.edu/robot) is the co-founder of Z Corp. See a hundred more of his projects at instructables.com.

By William Gurstelle

The Chinese Windlass

» In the summer of 1860, the Second Opium War was reaching its climax. Three years earlier, Britain and France had invaded Canton, China, in order to expand their trade in such unsavory commodities as opium and indentured servants. Chinese Emperor Xianfeng had resisted the outsiders, and a low-intensity war continued in fits and starts.

The European nations, tiring of the expense and anxiety of fighting a war so distant, launched an invasion fleet carrying nearly 18,000 men. After landing at the port of Beitang, the soldiers fought their way westward into Peking (now called Beijing) in October 1860. Not long after, a treaty was signed and an uneasy peace was restored.

Some British soldiers, encamped on the outskirts of Beijing in the aftermath, took note of an ingenious device being used to raise and lower drawbridges throughout the city.

The device consisted of a lifting hook suspended from a broad windlass made from cylinders of unequal diameters. The lifting hook was connected by an enormous length of rope, which came off one side of the windlass, went around the hook, and was coiled onto the other side when the mechanism was operated.

The soldiers observed that this windlass was capable of lifting huge loads with little effort. It soon became well known among Western engineers as the Chinese windlass.

How It Works

The Chinese windlass, also known by its more descriptive name, the differential windlass, is easy to construct and produces enormous mechanical advantage.

The lifting power comes from the way the rope is wound around 2 drums of slightly different diameters, on the same axis. When the handle is turned to lift the load, the rope is paid off the smaller drum and onto the larger (Figure A).

The larger drum winds up a bit more rope than is unwound from the smaller. It's by this small

British soldiers, encamped on the outskirts of Beijing after the war, took note of an ingenious device being used to raise and lower drawbridges.

difference in length, divided by 2, that the load is raised with each turn of the handle. So, while raising the load is very slow and requires many turns of the handle, what's lost in speed is gained in power. With drums of slightly differing diameter, even the smallest person can lift a very heavy load.

For the engineer or physicist, there's a simple formula that computes the mechanical advantage or "purchase" that the windlass provides:

$$\frac{R}{r} * \frac{C}{D} * 2 = P$$

Where

R = Radius of the crank handle
r = Radius of the large barrel
C = Circumference of the large barrel
D = Difference in circumference between the large and small barrels
P = Purchase (mechanical advantage gained)

Suppose we built a Chinese windlass with a 6" crank handle and 2 steel-pipe barrels, one with a 2¾" diameter and the other 3" in diameter. Could we lift a 600lb engine block?

$$\frac{6}{3} * \frac{3\pi}{3\pi - 2.75\pi} * 2 = 48$$

Yes, we could lift a 600lb block with 12½lbs of force, or just 1⁄48 the effort of getting underneath and using our backs. Quite a bit easier indeed!

MATERIALS AND TOOLS

PVC pipe (2 pieces): 3" diameter, 10" long; 2" diameter, 10" long You can substitute ABS pipe and fittings for the PVC.
PVC end caps: 3" (1), 2" (1)
PVC reducer fitting, 3" to 2" (1)
¼" bolts: 2½" long (1), 2" long (1)
Pulley
Lifting hook
Iron pipe, ½" diameter, 28" long both ends threaded
Iron pipe fittings, ½": elbows (2), nipples 4" long (2), end caps (2)
Cord, ⅛" or 3/16" diameter, 20'
Pine boards, 2×6: 36" (1) and 24" (2) for the base and 2 uprights. You can substitute 1×6 boards.
Plywood, ½" thick, cut into triangles 6" on a side (2) for the gussets
Wood screws, 1" long (6)
Deck screws, 2½" long (6)
Cordless drill
Drill bit, 7/32"
Spade bits: 1", ⅞"
Thread tap, ¼"

Build a Chinese Windlass

If you have access to a metal turning lathe, you can machine a high-capacity windlass from a 3"-diameter steel bar mounted on bearings, with one end turned to a slightly smaller diameter.

But if you don't have a lathe, you can easily construct a Chinese windlass demonstration model from PVC and iron pipe, pipe fittings, and a few odds and ends commonly found around the workshop or the hardware store.

1. Drill a ⅞" hole in the center of each end cap (Figure B).

2. Assemble the windlass barrel as shown in the barrel diagram (Figure C), using PVC primer and cement.

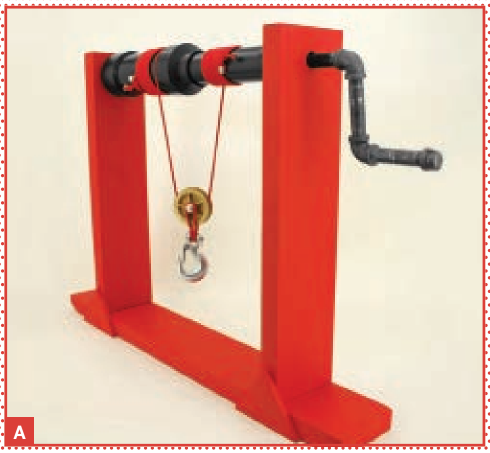
Be sure to read and follow label directions.

3. Drill two 7/32" holes in the PVC pipes as shown in Figure C. Take care to drill them straight into the center of the pipe, because the ¼" bolts that go in them, in Step 8, will act as setscrews against the iron pipe inside, allowing the windlass assembly to rotate when the crank is turned.

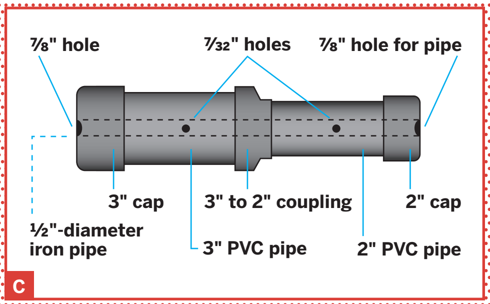
Tap the holes with a ¼" tap.

4. Drill a 1" hole in each wooden upright as shown in the windlass assembly diagram (Figure D). Fasten one of the uprights to the base using three 2½" deck screws.

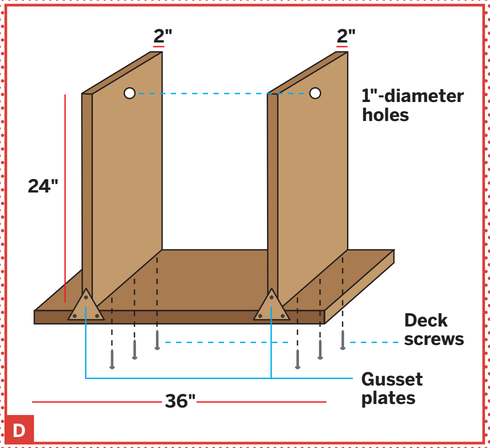
Photography by Ed Troxell; illustrations by Katie Wilson


A


B



C



D


E


F


G


H

5. Insert the iron pipe into the PVC windlass barrel through the holes in each PVC end cap. Center the barrel on the pipe.

6. Insert one end of the pipe into the hole in the upright attached to the base. Slide the other upright into place and attach it to the base using three 2½" deck screws.

Attach the wooden gusset plates to the base and uprights (Figure E) using 3 wood screws per gusset.

7. Screw the 2 iron pipe elbows and 2 nipples onto the pipe to form a crank. Add end caps to the open pipe ends (Figure F).

8. Insert the ¼" bolts into the tapped holes on the windlass and tighten them gently until they bottom on the iron pipe (Figure G).

9. Attach the cord, pulley, and hook as shown in Figures A and H, tying off the cord on the ¼" bolts. Your Chinese windlass is ready for use!

Troubleshooting

Be sure the ¼" bolts set firmly against the iron pipe, otherwise the crank won't turn the windlass. If the bolts slip, use a metal file to make flat spots on the pipe to help them hold better.

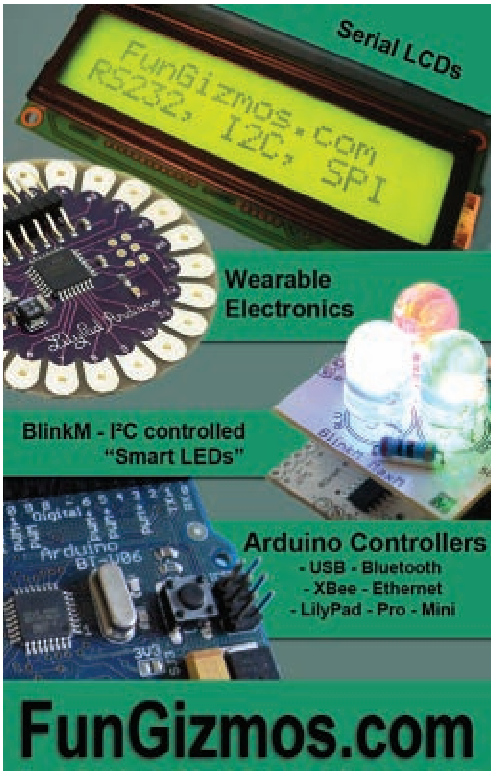
If they still slip, or if you want to lift more weight, put a pencil through the bolt holes to mark the iron pipe, then drill and tap holes in the iron pipe as you did the PVC, and screw the bolts into the iron pipe.

Make certain the rope is wound as shown in Figure H. The system won't provide mechanical advantage if the rope isn't arranged correctly on the windlass.

William Gurstelle is a contributing editor of MAKE.


Automotive
Cables
Components
Converters
Data Loggers
Displays
Embedded
Industrial PCs
Software
Test Equipment
Video
Wireless
Saelig .com
unique electronics


FOR THE
TOUGHEST
JOBS ON
PLANET EARTH®
GORILLA GLUE
STRONGER-FASTER
GORILLATOUGH.COM
1-800-966-3458 Made in U.S.A.
© 2010 Gorilla Glue Company


Serial LCDs
FunGizmos.com
RS232, I2C, SPI
Wearable
Electronics
BlinkM - I²C controlled
"Smart LEDs"
Arduino Controllers
- USB - Bluetooth
- XBee - Ethernet
- LilyPad - Pro - Mini
FunGizmos.com


" TOOLS MADE FOR MAKERS "
ISO-TIP
MODEL# 7975
Flameless Butane
Heat Gun
MODEL# 7971
Butane
Soldering Iron
MODEL# 7700
Rechargeable Battery
Soldering Iron Kit
10 % coupon code: maker10
www.iso-tip.com

HOMEBREW

My Own Visible Computer

By Kevin Quiggle

Photograph by Kevin Quiggle

Building PCs for myself, friends, and family has been my hobby for many years. With each PC I build, I try to be a bit more creative than the last time. For my most recent build, I wanted to show the working parts as much as possible. I considered several alternatives — completely transparent cases and side-window designs have already been done, and "theme" case designs hide the parts, which defeated my purpose.

I settled on an aquarium-style case design using transparent acrylic everywhere but the side pieces, with the internal parts laid out next to each other (as opposed to the standard method of stacking them), sacrificing compactness for visibility. The final design was drafted in OpenOffice Draw by configuring the components in breadboard fashion and taking appropriate measurements.

The sides and legs were cut from red oak in an Art Deco-inspired design. The side pieces attach to each other via six stainless steel rods, hand-threaded on each end to hold the sides at a specific width. I really like the look of the stainless steel, but if I had it to do over again, I'd use brass — it took 15–20 minutes to cut each thread (a total of 12). I also polished the stainless rods by spinning them with a hand drill and buffing them with emery cloth.

The transparent top and bottom pieces rest on two top and two bottom rods, and the other two rods form an internal frame for mounting the components. The full-sized front and back windows sit in grooves cut into the sides and slide up for quick access to the inside (the front window was removed in this photo to eliminate reflections). A hidden side port provides access for power and monitor cables.

A cooling fan is mounted to the top, in addition to the power-supply cooling fan, the power and reset switches, case lights, and a vertically mounted CD/DVD unit, which was a small project in itself: I removed the metal case and built a transparent one. It's really fun to watch a DVD load and spin, with the laser light faintly visible through the disk!

The bottom is perforated to enable airflow, with a dust filter covering the perforations. Also mounted to the bottom is a "front panel" extension (with case removed to show the internals) including USB, speaker, and microphone ports, plus a digital card reader. After many enjoyable hours spent designing and building, the final result is now at work and on display in my living room.

Kevin Quiggle has been working with computers long enough to remember FORTRAN card decks.

Mitchell Baker. Don Chadwick. Bill Gates. William McDonough. Toshiko Mori. Elon Musk. Pierre Omidyar. Stan Ovshinsky. Carroll Shelby. Lyn St. James. Alfred Taubman. Martha Stewart. Gordon Moore. Steve Wozniak. Charles Jones. Bob Metcalfe. Jim Hall. Charles Elachi. Andrea Donnellan. **Dean Kamen.**

Inspiring Innovation

At The Henry Ford, we believe that innovation is a state of mind, a culture that can be taught and nurtured. It was America's passion for problem-solving and creative thinking that inspired Henry Ford to found this institution more than 80 years ago. Today, we continue to spark and spur the spirit of innovation for new generations.

What are you making?

Join forces with The Henry Ford and some of America's greatest innovators by sharing a glimpse into your workshop, garage or studio. As a contributor to *America Invents* your work will be a source of inspiration to classrooms nationwide through our Innovation 101 Curriculum.

Share Your Passion

For more information, participation benefits and to post your videos visit **OnInnovation.com/AmericaInvents**

OnInnovation
Visionaries thinking out loud.™

Powered by the Henry Ford

US $14.99 CAN $17.99
ISBN: 978-1-449-39398-4
51499
9 781449 393984