

# Make:



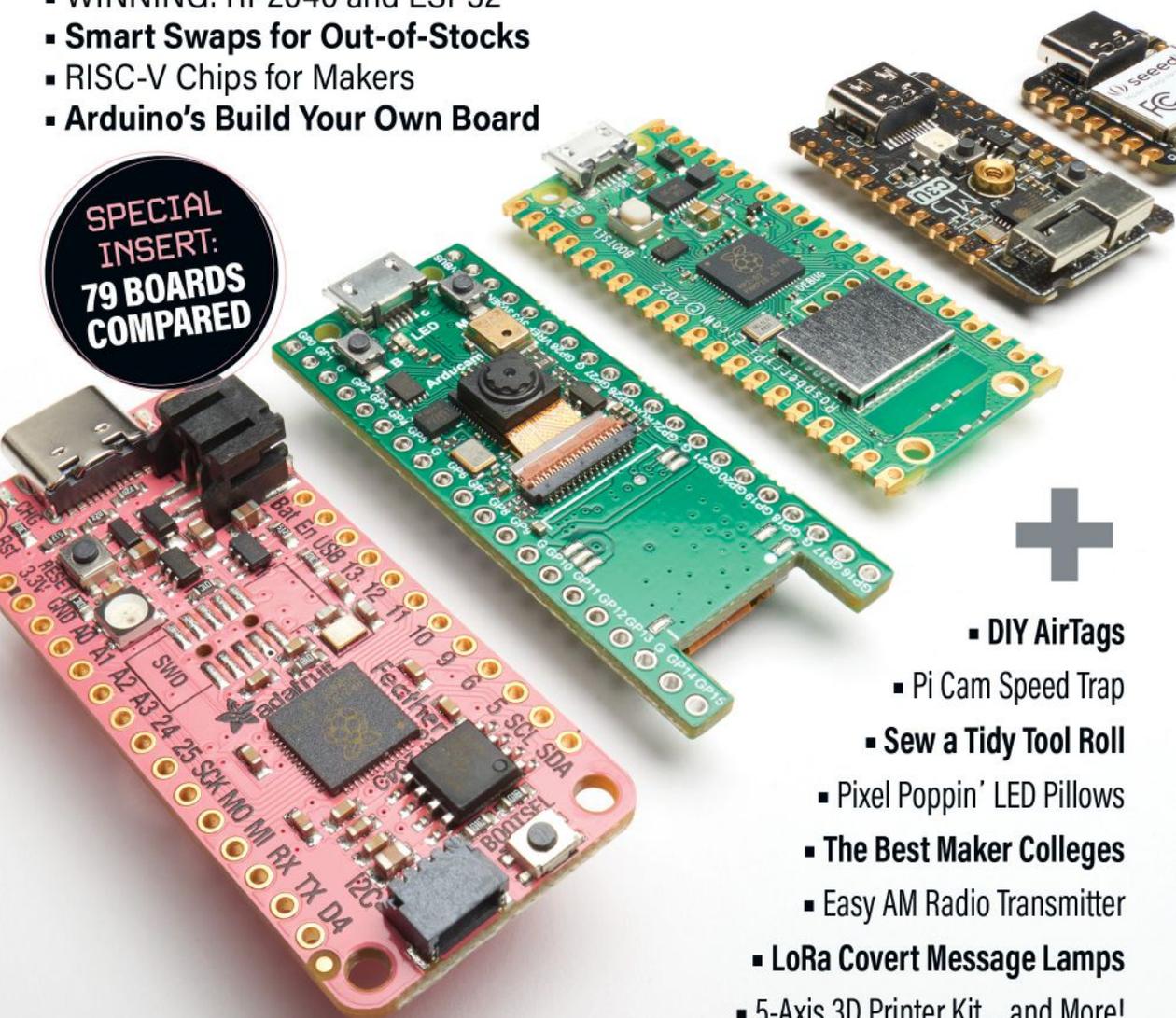
**PIXELBLAZE:**  
FAST & EASY  
RGB LED  
PATTERNS

## 2022 BOARDS GUIDE

WHO STEPPED UP WHEN THE CHIPS WERE DOWN

- **WINNING:** RP2040 and ESP32
- **Smart Swaps for Out-of-Stocks**
- **RISC-V Chips for Makers**
- **Arduino's Build Your Own Board**

**SPECIAL  
INSERT:  
79 BOARDS  
COMPARED**



- **DIY AirTags**
- **Pi Cam Speed Trap**
- **Sew a Tidy Tool Roll**
- **Pixel Poppin' LED Pillows**
- **The Best Maker Colleges**
- **Easy AM Radio Transmitter**
- **LoRa Covert Message Lamps**
- **5-Axis 3D Printer Kit... and More!**

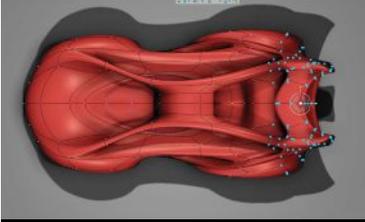
# INTRODUCING



## FUSION MAKER

- Laser engrave, cut, and inlay wood
- Customize projects for added value
- 24" x 12" x 7" work area
- IRIS™ Camera for artwork layout
- Affordable pricing for entry-level
- Made in the USA

# FROM DIRT TO DESIGN



## 3DEXPERIENCE SOLIDWORKS for Makers is Jason Pohl's "digital clay"

**Dirt bikes, tree forts, 4-wheelers — Jason Pohl grew up in the dirt, and remembers vividly the way rivers cut through the land surrounding his rural home.** He soon learned to harness the same carving and sculpting prowess that he revered in water as he pursued a fine arts degree. And it is something he continues to command every day thanks to 3DEXPERIENCE SOLIDWORKS for Makers.

A self-proclaimed "preengineer," Pohl's desire to make his designs "work" led him from the polygonal world of video game modeling and animation to the powerful fabrication and visualization toolkit provided by 3DEXPERIENCE SOLIDWORKS for Makers.

And make it work he did when he joined the Orange County Choppers team as Lead Designer, where he helped realize the wildest vehicular fantasies of the most demanding custom motorcycle enthusiasts. Using tools like xShape, Jason sculpted his client's imaginations into reality, then created lifelike renders to share their vision, and finally realized them physically with CNC and 3D printing.

Jason likens 3D printing to "teleporting" his designs into real life after using xShape to create them from simple blocks that he calls "digital clay." His projects include creating WWE championship belts and a unique fishing lure, although his favorite is always "the next one!"

Asked why he relies on 3DEXPERIENCE SOLIDWORKS for Makers for all of his DIY projects, Jason explained "for the longest time you had to sit down with sketches and drafts and sweeps and extrudes — it was very boring — 3DEXPERIENCE SOLIDWORKS for Makers has a surface modeling tool that allows you to be an

artist ... it's so fast, no waiting for it to calculate ... the software adapts to me!"

Jason is a 3DEXPERIENCE SOLIDWORKS advocate and avid proponent of 3DEXPERIENCE SOLIDWORKS for Makers. He operates Jason Pohl LLC from an elaborate home workshop, where he continually involves his four children in his DIY projects. See Jason's work at [@pohlie](#) on Instagram and his personal portfolio [JasonPohl.com](#). Watch Jason model a concept car at [youtu.be/vf-vZUmB7\\_M](#).



3DEXPERIENCE SOLIDWORKS for Makers is available for your hobbies and personal projects. For just US \$99/year or US \$9.99/month you'll get the same intuitive cloud-connected CAD modeling tools that the professionals use, along with:

- Fully online design solutions you can access from any web browser
- An online community that lets you connect with worldwide makers from fab labs, makerspaces, and influencers, all ready to share their designs, ideas, and expertise
- Access to an expanded professional ecosystem to rapid prototype your parts, or receive engineering services via the 3DEXPERIENCE Marketplace
- Support to help you get the most out of 3DEXPERIENCE SOLIDWORKS for Makers.



**SOLIDWORKS**



Get all the details and sign up at  
[discover.solidworks.com/makers](#)



**34**



**24**



**30**

**ON THE COVER:**

From left to right: Adafruit Feather RP2040, Arducam Pico4ML, Raspberry Pi Pico W, M5Stack M5Stamp C3U Mate, and Seeed Xiao RP2040

Photos: Mark Madeo and ElectroMage

**COLUMNS**

**From the Editor's Desk 05**

Tips, tributes, tidings, and tweets from readers like you and members of our worldwide community.

**Welcome: Tilling the Fields of Scarcity 06**

Flexibility is built into the maker mindset — perfect for weathering today's supply chain woes.

**MacGyver Challenge: Wildfire Escape 07**

Can you MacGyver your way out of a hot spot?

**Made on Earth 10**

Backyard builds from around the globe.

**FEATURES**

**The Amazing Maker Awards:**

**The Winner's Circle 14**

Meet the prize winners and standout projects from our inaugural Amazing Maker Awards — the Makeys.

**The Amazing Maker Awards:**

**Community Maker Award 20**

Given to a person or organization whose work expands the maker community, this year's award goes to Noelle Conover for her groundbreaking work developing Matt's Maker Space.

**Best Maker Schools 2022 24**

College and university makerspaces and programs are on the rise. Here are this year's standouts.

**A Perfect Circle 28**

An LED "supergroup" collaborates to create the tiny, mesmerizing Lux Lavalier pendant.

**California Fresh 30**

Three California women makers tell why they launched new businesses building and selling kits.

**NOW BOARDING**

**Now Boarding 34**

Need a board? Buckle up for a bumpy ride through all that's new in 2022.

**When Chips Are Scarce 36**

Maker companies share how they've been impacted by shortages, and what the future looks like.



36



48



56



70



84



86



92



110

### ARISC-V Revolution 42

Free and open-source silicon is poised to take the maker ecosystem by storm.

### The Replacements 46

Unsatisfied with back-orders for your favorite Pi or Arduino? Consider these alternatives.

### Build-a-Board Synth-Pop 47

Get an exclusive sneak peek at Arduino's back-to-basics Uno kit!

## SPECIAL INSERT

### Make: Guide to Boards 2022

Check out specs and hands-on reviews of the latest microcontrollers, single-board computers, and FPGAs — in dazzling *augmented reality*. 79 boards for you to compare!

## PROJECTS

### Pixelblaze Pillows 48

Use full-color LED strings and a Pixelblaze controller to make these cuddly animated throws.

### Waxed Canvas Tool Roll 56

Stitch a stylish and sturdy roll-up tool carrier that will last for many tough jobs.

### 1+2+3: LED Pendant 61

Build a glowing, wearable circuit sculpture.

### Power Through Thin Air 62

Learn how to transmit and receive audio signals on an AM radio carrier wave.

### LED Love! 70

Cover the LED basics and create amazing light patterns with addressable LEDs and controllers.

### Pi Cam Speed Trap 76

Build a motion recorder that analyzes video to catch speeding vehicles on your street.

### Make a Mac Mini Mobile 84

Hack that powerful Mini M1 with an iPad Mini to make a badass portable computer.

### Kit Cards 86

Modern model method: 3D print the parts on an old-school "sprue" then snap them off and build.

### The Fifth Dimension 88

Convert your 3D printer to 5-axis, and print the impossible.

### One-Shot Wonders 92

Design simple mechanisms with moving parts that print in place — no assembly required.

### Long-Distance Emoji Lamp 94

Build an interactive lamp that sends emojis to its partner — miles away — using LoRa radio modules.

### Remaking History: Christiaan Huygens and the Pendulum Clock 100

Build the ingenious gravity-driven mechanism that kept the world's time for 270 years.

### Electronics Fun & Fundamentals:

#### Analog Reborn 104

Before the digital revolution there was analog computing — and it's making a comeback.

#### Toy Inventor's Notebook:

##### Bubble Plane 108

Rock crushes scissors, scissors cut paper — but bubbles rule them all.

## SKILL BUILDER

### Find My: DIY AirTag Tracker 110

Understanding, [ab]using, and extending Apple's offline finding protocol.

### RaZberry Rescue 118

Take control of smart home gadgets — and revive those zombie bulbs — with a Z-Wave hat for Raspberry Pi.

## TOOLBOX

### Toolbox 124

Gear up with the latest tools and kits for makers.

## MAKER FAIRE AROUND THE WORLD

### Maker Faire Hannover Returns 128

After a two-year hiatus, Maker Faire returned to Hannover, Germany. Here are a few of our favorite moments.

Mark Madeo, Timothy Hursley, Alpenglow Industries, Kevin Walseth, Debra Ansell, Becky Stern, Lee Wilkins, Scott Yu-Jan, Loyal Moses

## PRESIDENT

**Dale Dougherty**  
dale@make.co

## VP, PARTNERSHIPS

**Todd Sotkiewicz**  
todd@make.co

## EDITORIAL

EDITOR-IN-CHIEF  
**Keith Hammond**  
keith@make.co

SENIOR EDITOR  
**Caleb Kraft**  
caleb@make.co

COMMUNITY EDITOR  
**David J. Groom**  
david@make.co

PRODUCTION MANAGER  
**Craig Couden**

CONTRIBUTING EDITORS  
**Tim Deagan**  
**William Gurstelle**

CONTRIBUTING WRITERS  
Rhett Allain, Debra Ansell,  
Fabian Bräunlein, Noelle Conover,  
Emma Fletcher, Charlyn Gonda,  
Gareth Halfacree, Freddie Hong,  
Edward C. Hume III, Nico Jan,  
Fredrik Jansson, Bob Knetzger,  
Sean Nolan, Nancy Otero,  
Marshall Pirots, Charles Platt,  
Bryan Serinese, Becky Stern,  
Carrie Sundra, Lee Wilkins,  
Scott Yu-Jan, Lee David Zlotoff

CONTRIBUTING ARTISTS  
Mark Madeo

## MAKE.CO

ENGINEERING MANAGER  
**Alicia Williams**

WEB APPLICATION DEVELOPER  
**Rio Roth-Barreiro**

## DESIGN

CREATIVE DIRECTOR  
**Juliann Brown**

BOOKS  
BOOKS EDITOR  
**Michelle Lowman**  
books@make.co

## GLOBAL MAKER FAIRE

MANAGING DIRECTOR,  
GLOBAL MAKER FAIRE  
**Katie D. Kunde**

GLOBAL LICENSING  
**Jennifer Blakeslee**

## MARKETING

DIRECTOR OF  
MARKETING  
**Gillian Mutti**

PROGRAM COORDINATOR  
**Jamie Agius**

## OPERATIONS

ADMINISTRATIVE  
MANAGER  
**Cathy Shanahan**

ACCOUNTING MANAGER  
**Kelly Marshall**

OPERATIONS MANAGER  
& MAKER SHED  
**Rob Bullington**

LOGISTICS  
COORDINATOR  
**Phil Muelrath**

## PUBLISHED BY

MAKE COMMUNITY, LLC  
**Dale Dougherty**

Copyright © 2022  
Make Community, LLC. All rights reserved. Reproduction without permission is prohibited. Printed in the U.S. by Schumann Printers, Inc.

Comments may be sent to:  
editor@makezine.com

Visit us online:  
make.co

Follow us:  
@make @makerfaire  
@makershed  
makemagazine  
makemagazine  
makemagazine

Manage your account online, including change of address: [makezine.com/account](http://makezine.com/account)  
For telephone service call 847-559-7395 between the hours of 8am and 4:30pm CST. Fax: 847-564-9453. Email: [make@omeda.com](mailto:make@omeda.com)

**Make:**  
Community

Support for the publication of *Make:* magazine is made possible in part by the members of *Make:* Community. Join us at [make.co](http://make.co).

## CONTRIBUTORS

*What's your New Year's Maker Resolution?*



**Charlyn Gonda**  
San Francisco, California  
(1+2+3 LED Pendant)  
To set aside time to hone new skills. There are deeper topics where I really need to sit down, not produce art, and just learn. It's been a maker upgrade every time I do it.



**Jennifer Blakeslee**  
Oakland, California  
(Made on Earth, Best Maker Schools)  
To improve my laminated pastry dough, restart making ceramics, learn to weld, continue to create and support projects through my art collective, Flux Foundation.

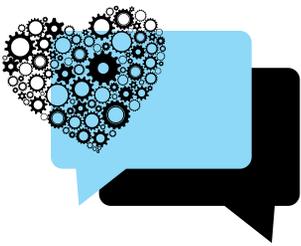


**Gareth Halfacree**  
Bradford, UK  
(RISC-V Revolution)  
To hopefully see the back of the industry's supply chain issues, to un-stall so many pending projects!

Issue No. 83, Winter 2022. *Make:* (ISSN 1556-2336) is published quarterly by Make Community, LLC, in the months of February, May, Aug, and Nov. Make: Community is located at 150 Todd Road, Suite 100, Santa Rosa, CA 95407. SUBSCRIPTIONS: Send all subscription requests to *Make:*, P.O. Box 566, Lincolnshire, IL 60069 or subscribe online at [makezine.com/subscribe](http://makezine.com/subscribe) or via phone at (866) 289-8847 (U.S. and Canada); all other countries call (818) 487-2037. Subscriptions are available for \$34.99 for 1 year (4 issues) in the United States; in Canada: \$43.99 USD; all other countries: \$49.99 USD. Periodicals Postage Paid at San Francisco, CA, and at additional mailing offices. POSTMASTER: Send address changes to *Make:*, P.O. Box 566, Lincolnshire, IL 60069. Canada Post Publications Mail Agreement Number 41129568.

**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. Make Community, LLC, 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.





# FROM THE EDITOR'S DESK



## THE NOSE KEEPS ON KNOWIN'

David Tseng from Taiwan reports that he and CAVEDU Education ([cavedu.com](http://cavedu.com)) put together a great tinyML workshop based on Benjamin Cabé's **AI Nose** (*Make*: Volume 77, page 32, [makezine.com/projects/second-sense-build-an-ai-smart-nose](http://makezine.com/projects/second-sense-build-an-ai-smart-nose)) hosted by the National Taiwan Science and Education Center. Nice job!

## MAKING MORE IN THE CLASSROOM

As a newcomer to the maker movement and the engineering practices from the Next Generation Science Standards, attending the **Make: Education Forum 2022** was an invaluable pedagogical experience that will indubitably rejuvenate my classroom practice.

I felt like a kid in a candy store. One, I was able to network with the most knowledgeable and influential professionals in the maker movement. Two, gain insight into the relevance of maker skills in cultivating new engineered technologies. Three, learn ready-to-use hands-on strategies I can immediately integrate into my STEAM instruction. Four, demystify essential skills such as soldering, coding, physical computing, and robot building ... Five, learn how to integrate the latest technologies in the maker movement in classroom projects.

I walked away feeling confident, knowing that I can do this!

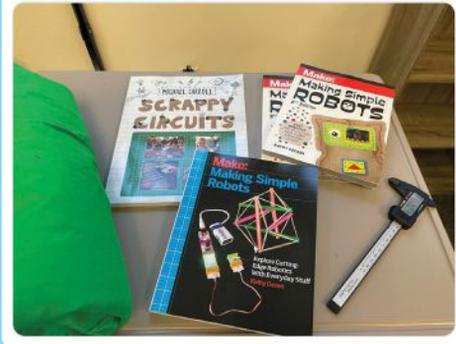
— Alecia Redway, New Rochelle, New York

## SHOW AND TELL US!

Send your favorite tips, maker moments, project pics, or thoughts about *Make*: to [editor@makezine.com](mailto:editor@makezine.com), subject line "Reader Input." If we select your note for this page, we'll send you a *Make*: hat!

## FROM TWEETSPACE

 **Mr. Rojas**  
@MrRojas14  
So glad to be starting the school year with these great resources in hand by @KathyCeceri and @ScrappyCircuits! 🧠 ⚡



## ON THE MAKE: TUBES

Wow! I made it to *Make*: magazine! Thank you! This community is great! ["Babybelts: The \$100 tiny DIY belt 3D printer," [youtu.be/6lQrY-1w5fU](https://youtu.be/6lQrY-1w5fU)]

—Rob Mink, Kilmarnock, Virginia



I thought I was done with YouTube today! Then *Make*: goes and does this! Follow recommendation: @make #DIY #maker ["Maker Mondays interview with Peter Brown," [youtu.be/szhYgEi-1sw](https://youtu.be/szhYgEi-1sw)] — Jason McFeetors, Edmonton, Alberta, Canada



# Tilling the Fields of Scarcity

by David J. Groom, Community Editor at Make:

It's finally fall in Michigan: the leaves are turning, there's a chill in the air at night, and endless fields heaving with copious pumpkins for sale are suddenly materializing along rural roads. Driving around the Metro Detroit area, these fields of abundance are not uncommon at this time of year.

But there's another bumper crop — one not so expected — that has caught our eyes lately. It might be a field, it might be a parking lot, but it's always a wide-open space, and it's filled with ... brand new cars.

These are not ad-hoc car dealerships, or the collections of overzealous car enthusiasts. They are millions of dollars of automotive liability, paralyzed by pennies' worth of parts. With dozens of tiny microcontrollers built into even the most basic modern vehicle, controlling everything from cabin climate to advanced collision avoidance systems, the global chip shortage has resulted in countless cars lying fallow as they wait for that one unobtainable chip.

Makers have been similarly scathed by scarcity, with a dearth of dev boards leaving innumerable projects uncultivated. But they are adapting in the face of adversity: Instead of designing in

abstract only to find the shelves bare when it's time for production, products are being based on the availability of parts, given known shortages and ETAs. Boards are being laid out with multiple footprints, so that, for example, any of three different voltage regulators can be employed

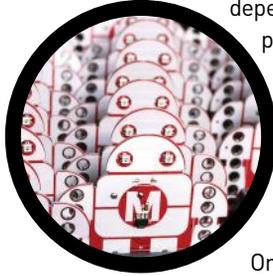
depending on what's in stock. Makers are pivoting to abundant microcontrollers, like the RP2040 and ESP32. They're stepping out of their comfort zones and trying less well-known single-board computers, or running their Python code on microcontrollers instead when SBCs aren't plentiful.

Or even moving to new instruction set architectures like RISC-V when more familiar processors are out of arm's reach.

Hopefully the automakers will get their hands on the chips they need, and future archeologists won't misinterpret these vehicles' hulking remains as a vain offering to the gods of silicon.

But ignoring potential automotive relics, we observe with elation as our community breaks free of constraints, crashes through supply chain issues, and evolves new skills and methods (which you can learn all about in this issue!).

To paraphrase *Jurassic Park's* Dr. Ian Malcolm: Makers, uh, find a way! 🍀



**STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION** (required by Act of August 12, 1970: Section 3685, Title 39, United States Code). 1. MAKE Magazine 2. (ISSN: 1556-2336) 3. Filing date: 10/1/2022. 4. Issue frequency: Quarterly. 5. Number of issues published annually: 4. 6. The annual subscription price is 34.99. 7. Complete mailing address of known office of publication: Make Community, LLC 150 Todd Road Ste. 100, Santa Rosa, CA 95407. 8. Complete mailing address of headquarters or general business office of publisher: Make Community, LLC 150 Todd Road Ste. 100, Santa Rosa, CA 95407. 9. Full names and complete mailing addresses of publisher, editor, and managing editor. Publisher, Dale Dougherty, Make Community, LLC, 150 Todd Road Ste. 100, Editor, Keith Hammond, Make Community, LLC, 150 Todd Road Ste. 100, Santa Rosa, CA 95407, Managing Editor, N/A, Make Community, LLC, 150 Todd Road Ste. 100, Santa Rosa, CA 95407. 10. Owner: Make Community, LLC, 150 Todd Road Ste. 100, Santa Rosa, CA 95407. 11. Known bondholders, mortgages, and other security holders owning or holding 1 percent or more of total amount of bonds, mortgages or other securities: None. 12. Tax status: Has Not Changed During Preceding 12 Months. 13. Publisher title: MAKE Magazine. 14. Issue date for circulation data below: Fall 2022. 15. The extent and nature of circulation: A. Total number of copies printed (Net press run). Average number of copies each issue during preceding 12 months: 53,388. Actual number of copies of single issue published nearest to filing date: 53,388. B. Paid circulation. 1. Mailed outside-county paid subscriptions. Average number of copies each issue during the preceding 12 months: 39,941. Actual number of copies of single issue published nearest to filing date: 36,893. 2. Mailed in-county paid subscriptions. Average number of copies each issue during the preceding 12 months: 0. Actual number of copies of single issue published nearest to filing date: 0. 3. Sales through dealers and carriers, street vendors and counter sales. Average number of copies each issue during the preceding 12 months: 3,418. Actual number of copies of single issue published nearest to filing date: 2,520. 4. Paid distribution through other classes mailed through the USPS. Average number of copies each issue during the preceding 12 months: 0. Actual number of copies of single issue published nearest to filing date: 0. C. Total paid distribution. Average number of copies each issue during preceding 12 months: 43,359. Actual number of copies of single issue published nearest to filing date: 39,413. D. Free or nominal rate distribution (by mail and outside mail). 1. Free or nominal Outside-County. Average number of copies each issue during the preceding 12 months: 273. Number of copies of single issue published nearest to filing date: 460. 2. Free or nominal rate in-county copies. Average number of copies each issue during the preceding 12 months: 0. Number of copies of single issue published nearest to filing date: 0. 3. Free or nominal rate copies mailed at other Classes through the USPS. Average number of copies each issue during preceding 12 months: 0. Number of copies of single issue published nearest to filing date: 0. 4. Free or nominal rate distribution outside the mail. Average number of copies each issue during preceding 12 months: 653. Number of copies of single issue published nearest to filing date: 656. E. Total free or nominal rate distribution. Average number of copies each issue during preceding 12 months: 926. Actual number of copies of single issue published nearest to filing date: 1,116. F. Total free distribution (sum of 15c and 15e). Average number of copies each issue during preceding 12 months: 44,285. Actual number of copies of single issue published nearest to filing date: 40,529. G. Copies not Distributed. Average number of copies each issue during preceding 12 months: 11,103. Actual number of copies of single issue published nearest to filing date: 13,059. H. Total (sum of 15f and 15g). Average number of copies each issue during preceding 12 months: 55,388. Actual number of copies of single issue published nearest to filing date: 53,388. I. Percent paid. Average percent of copies paid for the preceding 12 months: 97.91%. Actual percent of copies paid for the preceding 12 months: 97.25%. 16. Electronic Copy Circulation: A. Paid Electronic Copies. Average number of copies each issue during preceding 12 months: 3,723. Actual number of copies of single issue published nearest to filing date: 3,451. B. Total Paid Print Copies (Line 15c) + Paid Electronic Copies (Line 16a). Average number of copies each issue during preceding 12 months: 47,082. Actual number of copies of single issue published nearest to filing date: 42,864. C. Total Print Distribution (Line 15f) + Paid Electronic Copies (Line 16a). Average number of copies each issue during preceding 12 months: 48,008. Actual number of copies of single issue published nearest to filing date: 43,980. D. Percent Paid (Both Print & Electronic Copies) (16b divided by 16c x 100). Average percent of copies paid during preceding 12 months: 98.07%. Actual percentage of copies paid for single issue published nearest to filing date: 97.46%. I certify that 50% of all distributed copies (electronic and print) are paid above nominal price: Yes. Report circulation on PS Form 3526-X worksheet 17. Publication of statement of ownership will be printed in the Winter 2022 issue of the publication. 18. Signature and title of editor, publisher, business manager, or owner: Todd Sotkiewicz, Business Manager. I certify that all information furnished on this form is true and complete. I understand that anyone who furnishes false or misleading information on this form or who omits material or information requested on the form may be subject to criminal sanction and civil actions.



# MacGyver Challenge: WILDFIRE ESCAPE!

CAN YOU MACGYVER YOUR WAY OUT OF A HOT SPOT? By Lee D. Zlotoff and Rhett Allain

**Editor's note:** MacGyver TV creator Lee David Zlotoff wrote the popular *MakeShift* challenge in this magazine from 2005 to 2011 ([makezine.com/tag/makeshift](http://makezine.com/tag/makeshift)). Since then, Mac's moniker has officially become an English verb in the Oxford and Merriam-Webster dictionaries! Now we're bringing Mac back to help you think — and make — your way out of emergencies and disasters. As Lee wrote in *Make: Volume 82*, "We are all MacGyvers now."

Watch for the next MacGyver Challenge on the *Make: blog* ([makezine.com](http://makezine.com)), Twitter (@make), and Facebook ([makemagazine](http://makemagazine)) and enter your solutions for a chance to be featured in these pages and win *Make: goodies!*

## THE SCENARIO

You're just returning home from an extended business trip by cab to discover your neighborhood is now in the path of a wildfire and everyone has been ordered to evacuate — and they are fleeing in droves because emergency services have made it clear they will not be coming back for anyone. You race into the house to gather up whatever crucial documents and valuables you can, packing your small late-model, automatic economy car with as much as it can hold. But as you jump in to start it, you discover ... the battery is dead!



**LEE ZLOTOFF** is an award-winning writer, producer, and director of film and TV, including *MacGyver* (1985–1992). His new production, *MacGyver: The Musical*, casts a different audience member as Mac at each performance. [macgyver.com](http://macgyver.com)

## THE CHALLENGE

By now, there's no one around to help you jump-start it. And just as you begin to realize how serious this might be, all the power in the area goes out! So, how are you going to get your car started and escape before the fire gets to you?

## WHAT YOU'VE GOT

And it's *all you've got*:

- A smartphone
- A two-stroke weed trimmer, pull start, with fuel
- Chemicals: paint thinner, acetone, paint (latex based), wasp spray
- Copper pipe, zinc nails, steel screws, nuts/bolts
- Basic tools: screwdriver, wrenches, corded plug-in electric drill, saw, pliers, clamps, hammer
- Ratchet straps, tarps, and a tow rope
- Floor jack
- A mountain bike
- A gas lawnmower (push type, no drive)
- 2 battery flashlights with D cells (6 total)
- An old Walkman with 2 AA batteries
- A TV remote with 2 AAA batteries
- Scuba tank with 1,000psi of air
- 2 extension cords
- A beach umbrella

## TURN THE PAGE FOR SOLUTIONS!



**RHETT ALLAIN** teaches physics at Southeastern Louisiana University. He was technical consultant for the *MacGyver* reboot (2016–2021) and an advisor for *MythBusters*. He blogs about physics fun at [rhettallain.com](http://rhettallain.com).



## MacGyver Challenge: WILDFIRE ESCAPE!

### OUR SOLUTION

The biggest problem is the dead battery. You might think you could replace a 12V car battery with eight 1.5V batteries (like a combination of D cells and AA batteries). Well, you can't. Although 8 of these batteries would indeed make 12 volts, these smaller batteries are not able to deliver an electric current of 50 amperes or more to turn the starter motor.

But as Miracle Max would say: "It just so happens that your battery here is only *mostly* dead. There's a big difference between mostly dead and all dead. Mostly dead is slightly alive."

That means that it's possible to use these smaller alkaline batteries to recharge the car battery. If you connect 8 of them in series and then to the positive and negative terminal of the battery, they will indeed charge it — at least a little bit. Even with just 10 minutes you should be able to get enough energy in the battery to start up that car and get out of there. Hopefully that fire isn't spreading too fast.

### MOST CREATIVE SOLUTION

Submitting his comments on the *Make* blog, **Bob West** provides not only a creative solution, but an excellent narrative to fully describe the situation. His method of starting the car involves the weed trimmer connected to the serpentine belt in the car (likely a difficult task). But here's the most creative part — he uses the scuba tank to turn the two-stroke engine into a compressed air motor. This still might not work, but it seems like something MacGyver would do!

### HONORABLE MENTION

Three readers submitted a version of this solution: **Michael Beaveridge** (blog), **Scott Monaghan** (Twitter) and **Douglas Coleman** (Facebook). Their solution is to charge the car battery with the electric drill and the weed trimmer. The idea is to use the trimmer motor to rotate the motor of the drill to make it an electric generator.

It's a brilliant idea — but it might not always work. There are different kinds of motors in electric drills:

- A corded drill, as in our challenge, most likely has an AC motor — you probably can't use this (without serious modification) to charge your battery. Sorry, guys.
- Older cordless drills will have a DC motor. This is what you want. Remove the battery and connect the terminals from the drill to the battery (check positive and negative). Now connect the weed trimmer to the drill shaft and get it to rotate. There's your generator.
- The last kind of motor is the modern DC brushless motor. These are quite awesome motors — but they can't be used as a generator. ❌

### Read all the entries at:

- [makezine.com/article/maker-news/macgyver-wildfire-escape](https://makezine.com/article/maker-news/macgyver-wildfire-escape)
- [twitter.com/make/status/1569795396724989952](https://twitter.com/make/status/1569795396724989952)
- [facebook.com/makemagazine/photos/a.389005918189/10159721300928190/](https://facebook.com/makemagazine/photos/a.389005918189/10159721300928190/)

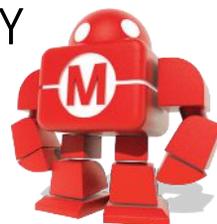
# Maker Faire®



INSPIRING AND ENCOURAGING LIFELONG  
CREATIVITY AND CURIOSITY

Coming to a community near you!

[makerfaire.com](http://makerfaire.com)



# MADE ON EARTH

**Backyard builds from around the globe**

Found a project that would be perfect for Made on Earth?

Let us know: [editor@makezine.com](mailto:editor@makezine.com)





Courtesy of musicMagpie

# FACING E-WASTE HEAD ON

[EDENPROJECT.COM](http://EDENPROJECT.COM) | [JOERUSH.COM](http://JOERUSH.COM)

E-waste is a bittersweet subject for the maker community. Our love of tech and electronic experimentation generates e-waste, while at the same time we're committed to ideas like the right to repair, upcycling, and creative reuse. Overshadowing any discussion of the topic, however, is the staggering mountain of actual tech trash on the planet — the UN estimates 53 million tons are generated globally each year. With its ubiquity eclipsed only by the endless new product cycles, how do we draw attention to the ever-growing hazards of electronic waste?

Enter *Mt. Recyclemore*, the tongue-in-cheek yet deadly serious installation created by lead artists **Joe Rush** and **Alex Wreckage** in partnership with UK tech reseller musicMagpie. With the help of 15 collaborating artists/makers and over 20,000 pieces (12 tons) of e-waste, the pair led the construction of the monumental piece over six weeks in Rush's South London studios.

Erected in Cornwall for the 2021 G7 Summit held there, the sculpture depicts the busts (à la South Dakota's Mt. Rushmore) of G7 leaders writ large, each 3m×1m head constructed entirely out of old toasters, phones, speakers, screens, and other electronic detritus. Although its timing and location (it could be seen across the water from where the summit was held) were a pointed statement to the leaders of countries that collectively produce much of the world's e-waste, Rush noted to *The Guardian* at the time that, "It's not just a politician's problem; it's a problem that the human race has to deal with."

Following its mini tour to Manchester, *Mt. Recyclemore* has returned to Cornwall to be housed for a year at the Eden Project, an eco-park and educational charity dedicated to building relationships between people and the natural world to demonstrate the power of working together for the benefit of all living things. Amidst the park's iconic biomes, it stands as a shiny testament to technological triumph and the folly of conspicuous consumption. —*Jennifer Blakeslee*



## TEA TIME FIT FOR MIDDLE EARTH

[YOUTUBE/S\\_OEZ1EY204](https://www.youtube.com/watch?v=OEZ1EY204)

When you glance at this diorama, you probably see a corner of a beautiful Elven village. The flying buttresses, the sweeping lines, the mossy overgrowth are all exquisitely executed to look like Rivendell out of *Lord of the Rings*. When he made this, **Samuel Gidoïn** had that very imagery in mind, but he also had a bigger plan. This quaint village is also an articulated machine that prepares the perfect cup of tea.

Underneath the laser-cut exterior of Therrarium: The Ancient Tea Machine — which shows no evidence of its mechanical undercarriage — you'll find an Arduino Mega, most of the components that would typically reside inside a commercial coffee machine, and a handful of other actuators and lights. Gidoïn spent 4 years perfecting the machine, designing 3D printed parts for the mechanisms and learning laser cutting at FabLab Moebius in Barbizon, France.

When it's tea time there are multiple options available, for different blends of tea and temperatures for steeping. With a quick press of a button, the Elven village hops to life measuring out tea leaves, depositing them into the tea ball, heating water and dispensing it into the cup, dunking the tea ball for the prescribed amount of time, then depositing it on a tiny coaster for disposal.

At the end of the process, Gidoïn is left with a perfect cup of tea, and a view that is absolutely wonderful. —*Caleb Kraft*

Samuel Gidoïn

# JUBILANT JUNK [YOUTUBE.COM/C/BILLMAKINGSTUFF](https://www.youtube.com/c/billmakingstuff)

Humanity's collective consciousness is peppered with useful pearls of wisdom. Among this catalog of aphorisms, the saying that "One man's trash is another man's treasure" is common across many languages, and perhaps no one has taken it more seriously than **Bill Mullaney** of the YouTube channel Bill Making Stuff.

Born in the East London town of Dagenham, Bill grew up with an eye and mind for filming but never felt he had enough creative control over his finished products while working as a filmmaker. When the company he worked for dissolved during Covid, he decided to get back into making. His creations of choice are miniatures made entirely from recycled junk: cardboard boxes, old toys, googly eyes, plastic knickknacks, and other items that most would label trash. Bill's creative process often starts with doodling in a sketchbook for inspiration and fiddling with his chosen bits and bobs. Once he has an initial structure, he glues everything into position, adds more layers for extra detailing, and finishes the build with a careful paint job and surrounding diorama. "I never plan meticulously," Bill says. "I'd say 90% of my builds are happy accidents, but ... the more I build, the more happy accidents I seem to have, so maybe it's more than an accident?"

As faithfully documented on his YouTube channel, Bill's miniatures are original, often dystopian or sci-fi inspired designs. His most recurring concepts are the "beadbot" (28mm robots made from beads and armature wire) and an ever-growing apocalyptic shantytown named Respite. Although he puts a lot of care into crafting the builds and their world, "The video itself for me is the real craft and the thing I treasure most at the end," Bill says. He likes to incorporate sound effects, hand-drawn transitions, live-action skits, and stop-motion animations into his videos. The editing process may take much longer, but he loves the creative freedom.

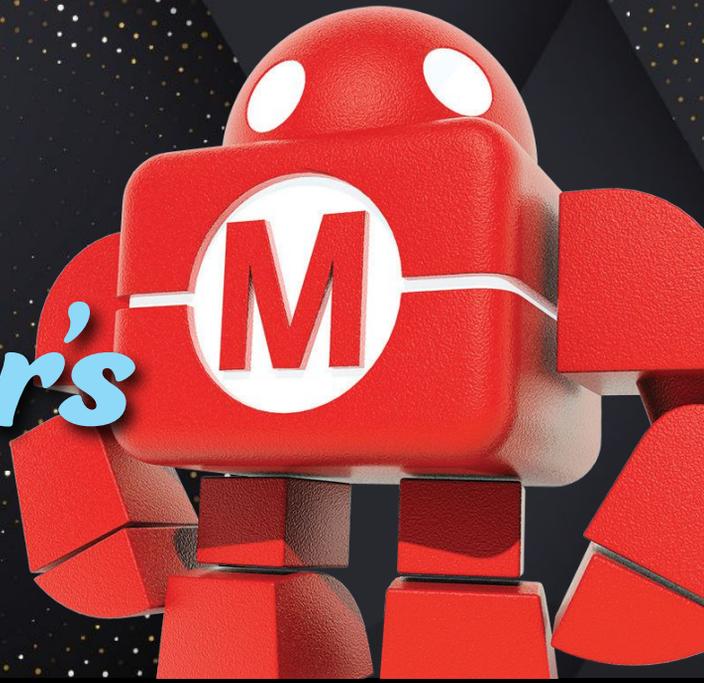
Bill urges those interested in making things on YouTube to learn filmmaking first. His advice? "Sound and lighting are just as important as visuals, have fun with editing, and stick at it. It takes a long time to get anywhere." You can keep up with Bill's builds on YouTube, Instagram, and Patreon, under the name Bill Making Stuff or [@billmakingstuff](https://www.instagram.com/billmakingstuff). —*Marshall Piro*s

Bill Mullaney



# The Winner's Circle

Written by *Make*: Editors



Adobe Stock-Avector

Here are the winners and standouts from our inaugural **Amazing Maker Awards**

✦ ✦ ✦ ✦ ✦  
**W**e are happy to announce the winners of this year's Amazing Maker Awards.

Every maker who shares a project is amazing, and the same can be said collectively of our maker community, which supports and values the work of makers. We hope these awards help gain recognition in the broader society for the creative and technical contributions of makers.

The Amazing Maker Awards celebrate what makers do — for different reasons and in different areas of endeavor. The winners are an eclectic group of makers, just like what you'd see at a Maker Faire. What they have in common is that each one taps into a deep human urge to create and make real something that starts out as an idea. We are grateful that they shared their work with us. We truly believe in the power of sharing, which motivates others to create and make.

In the pages that follow, we present the 26 winners by category, as well as specialty award winners, who will split a \$22,000 prize pool. Check out the showcase of winners online to get a fuller understanding of these projects and their makers, along with video demonstrations. I hope maker educators and parents will share this showcase with

students as a way to encourage them to generate their own ideas and start making.)

We are particularly grateful to our many judges\* who reviewed the projects. David Wells who used to run the makerspace at the New York Hall of Science wrote: "It was so nice seeing all the cool projects. It is inspiring to see people doing stuff!!" Judges Gary Rohrbacher and Ann Filson, authors of *Designing for CNC* and professors at the University of Kentucky, wrote: "Gary and I just finished judging our projects. What a nice way to end the day! So inspiring." Our team at *Make*: reviewed the rankings produced by judges and selected the top winners.

We are proud to have launched the Amazing Maker Awards, aka the Makeys — and we encourage you to enter your project for the 2023 awards.

—Dale Dougherty, CEO of *Make: Community*

**\*JUDGES** — **Matthew Wettergreen** (Rice University), **Debra Ansell** (maker/educator), **Lydia Cline** (author), **Joan Horvath** (author), **Ryan Spurlock** (Human Made makerspace), **Ann Filson and Gary Rohrbacher** (authors/educators), **Jen Fox** (Microsoft), **Suchit Jain** (Solidworks), **Cathy Chen** (Fab Lab El Paso), **Robert Kundel** (Restorer Tools), **Julie Darling**

(author/educator), **Hideo Tamura** (O'Reilly/Make: Japan), **Julie Legault** (author/educator), **David Wells** (museum educator), and **Nicole Shuman** (AmeriCorps VISTA maker).

## Top Prize & 1st Prize Education Category

\$5,000 prize

### 1 SHE BUILDS ROBOTS Christina Ernst

She Build Robots is the kind of website that creator Christina Ernst wanted when she was younger: Building engineering skills with a focus on creativity over academics.

Currently numbering around nine projects, [shebuildsrobots.org](http://shebuildsrobots.org) is a free educational resource for learning robotics and e-textiles. Designed to appeal to teen girls, it features color-changing skirts, musical cupcake toppers, tea-brewing robots, and more. As Ernst put it: “Less *Big Bang Theory* and more what-if-Coco-Chanel-knew-about-conductive-thread.”

Working directly with teachers in the Chicago area to get kits into classrooms and after-school programs, Ernst’s ultimate goal is to give girls confidence that there’s a place for them in STEM.

“Students absolutely have the capacity to be fascinated by topics they once perceived as boring,” she says. “Additionally, tone matters a lot when introducing students to a subject that may intimidate them; colloquial and approachable usually beats lofty and academic.”

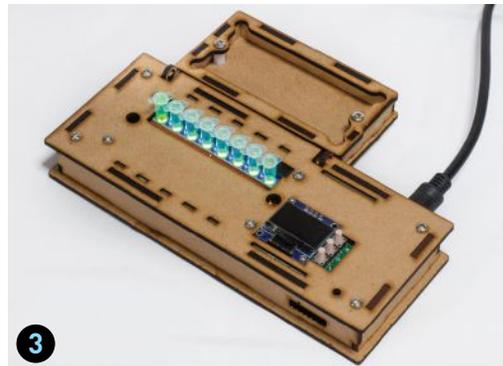
## 1st Prize, Artistic Category

\$2,500 prize

### 2 ❤️❤️❤️ — AN ARTISTIC EXPLORATION OF CONTEMPORARY COURTSHIP

#### Kenzie Housego

Contemporary dating has a near inescapable digital component, whether it’s online dating profiles, taking the perfect selfie, or trying to parse sometimes cryptic emoji responses to texts. With ❤️❤️❤️, Kenzie Housego set out to explore digital courtship language and symbols using five 2D pieces, augmenting traditional embroidery and craft with LED displays, motion sensors, and interactive chatbots. The effect is engaging and “highlights how romantic communication is transmitted, interpreted, and misinterpreted through technology.”



Christina Ernst, Kenzie Housego, Shingo Hisakawa

## 1st Prize, Technical Category

\$2,500 prize

### 3 QNINJA — REAL-TIME PCR

#### Shingo Hisakawa and Mariko Hisakawa

Academia likes to tell us that “legitimate science” is done in labs and needs high-level degrees, but there’s a rich history of punk scientists getting down ‘n’ dirty in home labs with DIY equipment. Inspired by Covid-19, qNinja continues the tradition as a low-cost, open source PCR gene analyzer that you can build and use at home.

A polymerase chain reaction, or PCR, is a



4



6



5



7

Caleb Kodama, Daniel Springwald, Selin Alara Ornek, Alana Balagot

method for turning a small amount of DNA into a large amount of DNA so you have more material to analyze. By adding fluorescent dye and watching how samples react to light, you can identify whether genes, like Covid-19, are present.

For the Hisakawas, getting their 2-year labor of love working was a global effort, from an optics expert in Kathmandu to a Chinese company in Guangdong helping with custom heaters and optical filters. “Everyone over the world can solve gene-related problems by making tools, and designing biosensors by themselves in the open-source circle.”

**1st Prize, Social Impact**

\$2,500 prize

**4 CODED BREATH — A NON-INVASIVE DIAGNOSTIC APPROACH Caleb Kodama**

Inspired by a childhood bout with a difficult-to-diagnose case of fungal pneumonia, high school student Caleb Kodama is developing an AI-powered diagnostic tool to help others diagnose the condition. An artificial lung simulates real-world breathing while an array of sensors and a trained AI model act as an

electronic nose to sniff out the simulated fungal particles (aka essential oils during testing).

“Effectively, my artificial lung breathes in the synthesized fungal pneumonia and breathes out to the E-Nose that confirms a fungal pneumonia diagnosis.”

**Community Choice Award \$1,500**

**5 NIKOLA TESLA INTERACTIVE ANIMATRONIC BUST Daniel Springwald**

Intrigued after reading a biography of Nikola Tesla, Daniel Springwald decided to build this interactive animatronic bust to spread the word about the ahead-of-his-time inventor. Servomotors control movement of the head, eyes, and mouth, while a webcam and microphone capture questions. An AIML chatbot sends responses to a text-to-speech program to keep the conversation going.

**Young Maker Award \$1,000**

**6 IC4U3 Selin Alara Ornek**

IC4U is a prototype robot guide dog to help people with visual impairments. Currently in its third iteration from 16-year-old Selin Alara Ornek,

the four-legged robotic companion accepts voice commands for directional movement and for sitting, lying down, and standing up. But a guide dog that blindly walks its human into traffic isn't exactly helpful, so IC4U3 uses sensors to detect sounds and focus in on them with the onboard camera for object detection. As an added bonus, you can chat with it via Dialogflow.

## Artistic Category

2nd Place, \$250

### 7 FOUR MUSES

Alana Balagot and Federico Tobon

Four experimental robotic instruments are controlled wirelessly with a MIDI keyboard and LED matrix display in four different modes.

3rd Place, \$100

### 8 MUSIC DERBY Tetsuji Katsuda

Seven robots assigned to musical notes race to music to see who can reach the finish line first.

## Technical Category

2nd Place, \$250

### 9 AUTOMATIC TABLE SAW FENCE, ROUTER LIFT, AND JOINERY MAKER Richard Want

This digitally controlled table saw fence and router table lift is capable of making customized joints and performing common woodworking processes with high precision.

3rd Place, \$100

### 10 WORLD'S LIGHTEST FOLDABLE COROPLAST TANDEM CANOE Hong Wong

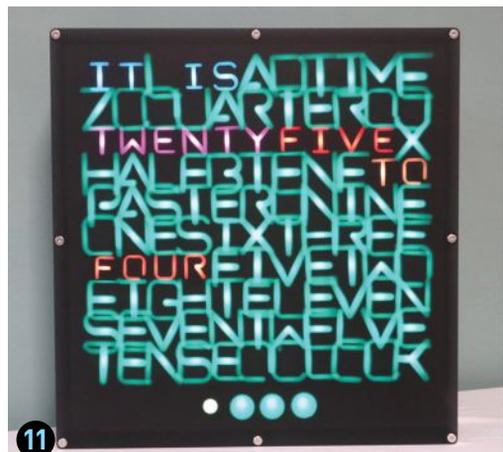
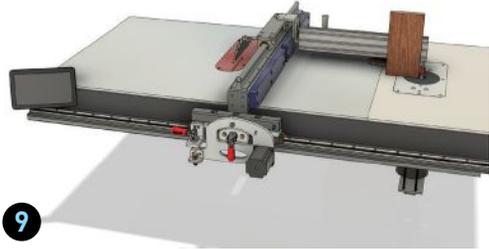
A DIY, super lightweight, foldable 10-foot-long Coroplast canoe for two.

Over the Top Award, \$250

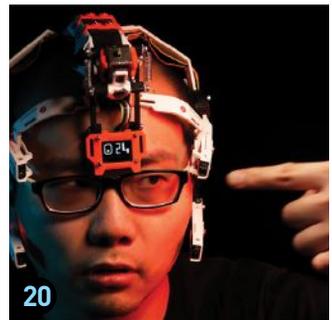
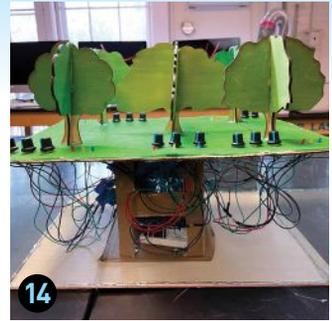
### 11 SERVO WORDCLOCK

Moritz von Sivers and Fabian Thum

In this variation of the popular word clock, letters are projected from the back onto a screen. Each letter is connected to a linear actuator that can be moved back and forth by a servo so that the projection changes size and focus.



Tetsuji Katsuda, Richard Want, Hong Wong, Moritz von Sivers



*Playful Maker Award, \$250*

**12 GIGANTY'S BOUNCING INFINITY GADGET**

**Sarah Gonsalves**

A 20-foot-tall, hand-cranked machine that tosses 5,000 superballs into a bounce chamber where they careen around and bounce off musical cymbals to make analog music.

*Education Category*

**2nd Place, \$250**

**13 TAPEBLOCKS: CREATIVE CIRCUIT MAKING FOR ALL** **Kirsten Ellis**

TapeBlocks is a toolkit to learn about and create circuits. Each block is easy to make and acts as a component in a circuit using conductive tape instead of wires.

**Science Fair Awards, \$100**

**14 HOW PLANTS SHARE NUTRIENTS**

**Sofia Egan**

An electronic simulation of how trees share nutrients through mycorrhizal networks.

**15 AIR QUALITY POP-UP PAVILION: RETHINKING SHARED SPACES**

**Aye Couloute**

The Air Quality Pavilion uses an internal sensor to monitor and then trigger various reactions to CO<sub>2</sub> levels, including opening and closing motorized shutter walls, an automatic sliding skylight, a ventilation fan, and color-coded warning lights.

Sarah Gonsalves, Kirsten Ellis, Sofia Egan, Aye Couloute, Lorraine Underwood, Yoshiharu Kawamura, Andrew Rudolph, Joshua Umansky-Castro, Chen Lu

## Social Impact Category

2nd Place, \$250

### 16 RGB LED SCROLLING FACEMASK

Lorraine Underwood

A facemask that uses two 8x8 RGB LED grids and speech-to-text software to scroll what you're saying across your mask.

3rd Place, \$100

### 17 BUDDY-KUN — COMMUNICATION ROBOT FOR NURSING HOME

Yoshiharu Kawamura

Buddy-kun is a robot that talks, instructs and guides singing and gymnastics, and plays musical instruments in kindergartens and nursing homes.

DIT Community Build Award, \$250

### 18 BRIGHTBIKES BOOMBIKE

Andrew Rudolph, Ian Cole, Mike Bakula, Harry Armstrong, Bryan Maier, Allen Paschel, Jeff Driscoll, Mike Gellatly, Andrea Gellatly, Tracy Lunquist, Steven Erickson, Rey Lim, Mike King, and Bob Houston

When the pandemic hit and makers from MakerFX couldn't get together indoors for big events, they brought the party outside with their giant mobile boombox bike!

Makers in Space Award, \$250

### 19 ALPHA CUBESAT — REACHING FOR THE STARS

Joshua Umansky-Castro, Andy Tan, Lauren Greenhill, Eleanor Glenn, Gillis Lowry, Andrew Filo, and Alex Burke

Alpha is a 1U CubeSat designed to deploy a spin-stabilized, free-flying light sail fitted with ChipSats. Its goal: to demonstrate that light sail technologies are suitable for interstellar propulsion.

Chindogu Award, \$250

### 20 STOP FACE TOUCHING DEVICE

Chen Lu

Early in the pandemic, health officials suggested refraining from touching your face in case it was a way of transmitting the virus. So Chen Lu built this fully functional — yet preposterously impractical — device to help him stop.

## Honorable Mentions

3D HANGMAN GAME Lochlan Fitzgerald

CRAFTIN' THE TELECRAFTSMAN Ray Rumore

GHOSTBUSTER PKE METER REAL Moonmakers

SMART CITY OF VUKOVAR Davor Sijanovic

EDUCATIONAL ROBOT ARM Isaac Venezia

Thanks to all who participated this year! 🌟



See more about all of our winners at [make.co/amazing-maker-awards/2022\\_winners\\_circle](https://make.co/amazing-maker-awards/2022_winners_circle).

## Thanks to Our Sponsors!



3DEXPERIENCE SOLIDWORKS for Makers





**NOELLE CONOVER** is the founder and executive director of Matt's Maker Space, Inc., which she and her husband, David, founded in memory of her late son, Matt. Noelle and David live in Pennsylvania near their two adult daughters.



# Community Maker Award

## Noelle Conover of Matt's Maker Space

*The first-ever Community Maker Award is given to that special person whose work expands the maker community by providing access to those who might not otherwise have the opportunity to experience the joy of making and learning. Noelle Conover is the 2022 recipient of the Community Maker Award, and when you read the story she wrote here, you'll understand why she deserves this recognition. Noelle started Matt's Maker Space in schools in the Pittsburgh area, and then went on to start 33 makerspaces in schools, children's hospitals, and psychiatric clinics.*

—Dale Dougherty

**H**ave you ever watched children in a makerspace? They come alive! Especially the children who never had a chance to use their hands or their minds in creative ways. My own children, now in their 20s and 30s, love visiting makerspaces. They never had a woodshop class or sewing in school. In the 1990s, schools removed the hands-on classes, replacing them with honors and advanced placement. There was talk of replacing recess with an extra half hour of math! Kids sat in their seats all day, reciting back the same things that their classmates recited. Every project they brought home was the same as the project their sibling made the year before.

And every snowman had to have an orange carrot nose!

My children are now part of our management team at Matt's Maker Space. They visit our 33 spaces and become children again. They marvel at the inventions of these young children and their creativity.

Makerspaces don't just ignite kids' learning. They are judgment-free zones. Mistakes not only can be made, but they help kids learn. And I think that's the human part of it. That is the human story. We're allowed to make mistakes that we can learn from.

## Who Was Matt?

I know that Matt, our son, would have just loved a makerspace. When he was ill and in the hospital, the only pastime was a craft cart that stopped by once a day to his room. He wanted to ... but it wasn't cool for a 12-year-old boy, so he passed. If that cart had been full of Lego bricks, robots, and other maker items, he would have thought that was the greatest.

When Matt was diagnosed with cancer, the whole community just wrapped itself around us, as many communities do. Nobody can really do anything when a child is sick, except maybe feed you and do your laundry and grocery shopping. We had all that.

Matt passed away from cancer at the age of 12. He would have been a maker. He was a tinkerer and a creator. He never built what was on the Lego box — he built what was in his mind.

And when Matt died, everyone surrounded us with support. We spent many years trying to find ways to make meaning out of Matt's life. And we realized that the community that helped us so much gave us the answer.

## "What's a Makerspace?"

When our youngest was graduating and we no longer would have kids in the schools, we met with the principal of the elementary school they had all attended. "Have you ever heard of makerspace?" the principal asked us. We had not. It opened our eyes to a world where our philanthropy could match our passion: *We can remember Matt while helping to change kids' lives.* That was in 2016.

We told Mt. Lebanon School District that we would build a makerspace in their school. They said, "That's great, except that we have seven elementary schools." So our gift to one became a gift to seven, and Matt's Maker Space was born with our donation of \$175,000 to a public school system.

Many people shook their heads when we donated to a public school system since it is not common. But we knew it was an investment that would launch something bigger.

We associated these spaces with libraries in all the schools, based on a suggestion from Carnegie Mellon University that no one group should "own" them. Each space was able to determine their design and what equipment to include. We only asked that each space appoint a "maker" or "STEAM" teacher who would be responsible.

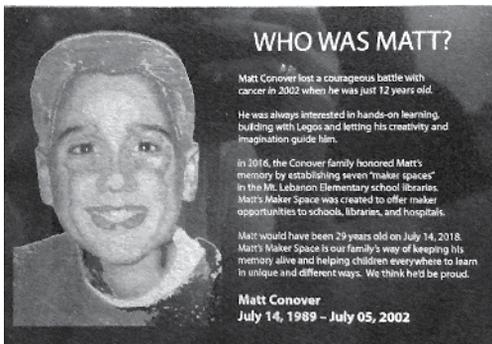
Each of these first seven spaces is completely unique. One focuses on "low tech" learning, using sewing machines and fiber materials; another focuses on 3D printing and laser engraving. They all focus on developing teacher expertise and not on the stuff. If you give a teacher a 3D printer without adequate training, we know it will eventually just be used as a coat rack.

We were able to design and create these makerspaces through collaboration with Carnegie Mellon and with the Children's Museum of Pittsburgh, particularly Anne Fullenkamp, their senior director of creative experiences.

## "I Was Supposed to Do This"

In 2017, we became our own 501(c)(3) organization. I never in my wildest dreams thought I would start a nonprofit. I had two jobs, kids, and an elderly father who lived with us.

I like to say it was the 501(c)(3) that started me.





Matt's Maker Space

down. Their program is now seamless from kindergarten through high school. If a student wants to be a maker or follow a STEAM path, they are able to do it there.

We have returned to many of our spaces to check in on them and ask if they have any further needs. We have provided training funds, programming, and equipment when needed. Once the spaces are up and going, we encourage them to work with their PTAs or to work with outside granting organizations to ensure their future.

### *Makerspaces in Hospitals*

Makerspaces can go just about anywhere! I basically wanted to put makerspaces into places that had touched our family, places we had experienced. One of those places was UPMC Children's Hospital of Pittsburgh, where I happened to work.

In my day job, I work with pediatric cancer survivors. I started thinking a makerspace would be really cool to have in a hospital. The Children's Museum of Pittsburgh had already worked closely with UPMC Children's Hospital and they understood the environment. We worked closely with the Child Life department of the hospital, who are responsible for making sure that kids can still be kids in spite of their medical problems.

Makerspaces can help kids with medical needs in many ways. A child facing surgery might begin working on a project with a child life specialist. He may say "I'm really scared. I'm having surgery tomorrow." While working on the project together, they can explore his feelings. Or perhaps a child is having open heart surgery. The surgeon and the child can make a model of a heart on the 3D printer and see what the procedure will entail.

In July 2018, we opened our space there. It is staffed by a full-time child life specialist who was also a teacher. The makerspace is used by all children in the hospital as part of the creative and expressive arts program. Getting to go to the makerspace can help to achieve therapeutic objectives like getting out of bed or walking. Next year, we hope to open another Matt's Maker Space in a pediatric hospital in Chicago.

I was supposed to do this.

Gregg Behr from the Grable Foundation gave us our first grant and he gave me my first mentoring advice. Sitting in his office, I said, "I have this story and I have this desire, but I'm not trained like you guys, what do you think I should do? And he just said, "Jump in — jump in with both feet. You have this desire to give ... just follow that and let Matt's story lead you." And that's really what I did.

Matt's Maker Space began in the more affluent Mt. Lebanon School District, but we decided to pursue Title 1 schools, which are typically underserved. We want to make sure that we're giving to kids who otherwise wouldn't have opportunities.

The rule of thumb for each space is that we give them about \$25,000 and we give them a wish list of what we think could go into a makerspace. It includes design services and training.

We recently awarded Mt. Lebanon School District more funds to build middle school maker spaces. Students wrote to me, saying that they were now in middle school and were looking for their Matt's Maker Space? We couldn't let them

Many people shook their heads when we donated to a public school system, since it is not common. But we knew it was an investment that would launch something bigger.

### *Making Can Be Therapeutic*

In the spring of 2021, the pandemic was taking its toll on everyone, especially our youth. When our son was diagnosed with cancer, everybody came around us and took care of us. If Matt had suffered from depression, anxiety, or schizophrenia, or had tried to commit suicide, our experience would have been quite different. And that breaks my heart. Today, so many kids have mental health problems.

We know that making can help in a behavioral health setting. If you sit a teenager down and you say, "OK, today I want to talk to you about your feelings. Will you share them with me?" — you know that they aren't going to say a word. But if we start working on a robot together, or we start building a cardboard bridge, we engage them and provide an opportunity that they will open up.

The demand for mental health needs had increased at our sister hospital, UPMC Western Psychiatric, and it was bursting at the seams with new inpatient youth. We decided that having a makerspace in a mental health hospital could not only be therapeutic but would meet the increased needs of this inpatient population. With the Children's Museum staff, we designed a wonderful space for the outpatient building. In the fall of 2021, we opened our first Matt's Maker Space in a mental health facility. This outpatient building contains a school environment where the patients go prior to being fully discharged back to their home schools.

This past spring, we opened a space in the inpatient hospital as well. It will serve ages 3 to 93. They have youth, adolescent, geriatric, and schizophrenic units and treat some pretty difficult

cases. All of the therapists were trained at the Children's Museum, where they went to Maker Educator Boot Camp and learned how to facilitate maker activities. They bring small groups to the space and do group therapy while they make. It is really amazing.

### *Makerspace: The Bridge to Healing*

I found very soon after we lost Matt that people did not want to talk to us about Matt. They didn't. But now they are interested in talking about the makerspaces.

When a parent loses a child, the key to moving along in their grief is making meaning out of that loss. David Kessler worked closely with Elizabeth Kübler-Ross, the psychiatrist and researcher who wrote about the five stages of grief. In his own book *Finding Meaning*, Kessler says there's actually a sixth stage. After you've been through the fifth stage, which is acceptance, the sixth stage is finding meaning. This is our way to get up in the morning after losing a child. This is what we're doing with Matt's Maker Space.

We are able to celebrate Matthew and tell the world about him. We didn't name the spaces the Matt Conover Memorial Makerspaces. We didn't want that, but calling it Matt's Maker Space always evokes the question anywhere we go. "Who's Matt?"

And the story goes on. Over \$1 million given, 33 spaces and equipment, teacher training and a bright hope for the future. All because of a sweet boy named Matt who loved to tinker!

—Noelle Conover



Learn more at [mattsmakerspace.org](http://mattsmakerspace.org)



# BEST MAKER SCHOOLS 2022

**Learning by doing:**  
The maker's path in  
higher education

Written by Dale Dougherty and Jen Blakeslee

Last year we introduced the Best Maker Schools list, a non-ranked survey of colleges that offer maker programs and makerspaces for students. Providing access to physical space and tools is the most basic form of a makerspace, but access to training and a diverse community of makers are equally important.

Fortunately, the number of colleges and universities with makerspaces continues to grow, so rather than print the listing here in the magazine, we've created a directory of Best Maker Schools at [make.co/maker-schools](https://make.co/maker-schools). (Soon we might have a much shorter list of colleges and universities that do not have a makerspace!)

For some time, we've heard that the makerspace is one of the stops on the college tour for prospective students, and surveys indicate that a makerspace is a reason for students to choose a school. Just as important is that students are arriving on campus understanding what they might do in a makerspace. It is our hope that the growth of makerspaces in higher education causes high schools and middle schools to provide makerspaces for their students.

One reason universities support makerspaces is that they represent an opportunity for interdisciplinary learning and collaboration among students from different parts of the university. Crossing outside of a field or meeting others from a different discipline can spark new ideas. Makerspaces have also become places to promote the practice of innovation, serving as an incubator for new projects that can lead to the creation of new companies. A makerspace, like a library, can be used for many different purposes, including as a social space for students.

Large universities have managed to find funding for makerspaces from alumni, and some are housed in brand-new buildings. Larger universities also have many makerspaces on campus, which creates the need to standardize training so that students can move from one space to the next while also allowing spaces to have their own characteristics and purpose. The **University of Wyoming**, for example, has started to manage a network of makerspaces in universities, local libraries, and other locations throughout the state.

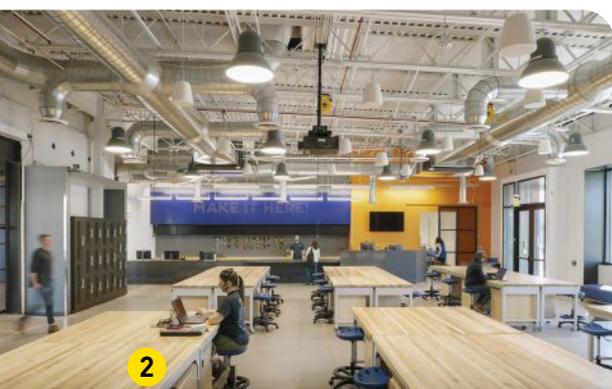


One of the challenges faced by Fab Labs and makerspaces was how to open the makerspace to the local community. Some have addressed this problem by partnering with outside organizations who manage the space, balancing the needs of students with those of a broader community. Such collaborations also help connect students to maker communities off campus.

The value of a makerspace need not be defined by its size, the extent of its tools, or its budget. There are modest makerspaces in small community colleges that also deserve recognition. They reach many students who may never have had the opportunity to create something in a makerspace.

Earlier this year, I met students from **Moorpark College** (pictured above), a community college in California. Their makerspace is called MakeShop, “a cross-disciplinary hub of entrepreneurship run entirely by students under the guidance of discipline faculty.” Mia had designed a red robot toy. Angela had built a kit for kids to make an animal design. Arylon had created a kiln furniture stamp, a marking solution for the school’s ceramics studio.

The best maker schools are those that encourage more and more students to do real-world work, develop useful skills, and gain experience solving problems or creating products that they care about.



## SOME HIGHLIGHTS FROM THE 2022 LIST ...

### 1 Monash University

Melbourne, Victoria, Australia

Imagine having the space, tools, and guidance to build the next big thing in autonomous driving systems, or to design processes and products that reduce and reuse plastic pollution. Through a robust Student Teams Initiative, a state-of-the-art makerspace, and a startup generator that connects R&D to real-world problems, Monash University helps makers to dream big.

To boot: As co-host of an expanded Maker Faire Monash (this year's is on December 4), MU connects to and elevates the creative culture of Melbourne, a city that prides itself on the vibrant, independent art/maker/entrepreneur scene that winds through its alleys. [monash.edu](https://www.monash.edu)

### 2 GateWay Community College

Phoenix, Arizona

September 13 marked the grand opening of Phoenix Forge, the largest makerspace in Arizona and the culmination of years of work by big thinkers from nearby GateWay Community College — the first technical college in the state — who worked with the city and Arizona State University to bring the project to life. Housed in an abandoned 1929 warehouse downtown, the “for everyone” purpose of the space is evident in its diverse offerings — there’s even an Arduino gardening class! Open to the public, the new makerspace connects GateWay’s diverse learners to the broader community in Maricopa County, hands-on workforce development (a stated goal), and the expanding maker culture in Phoenix. [gatewaycc.edu](https://www.gatewaycc.edu)

### 3 Coppin State University

Baltimore, Maryland

The first-of-its-kind collaboration between an HBCU, West Baltimore's Coppin State University, and Open Works, the city's innovative community makerspace, is about more than providing a place for making — although it has done that for students and neighbors alike. Rather, its goal has been to foster a “sense of trust and shared economic destiny” in a city with a legacy of segregation, crime, political corruption, and persistent poverty. To identify why and how this works — and help others benefit from those lessons — the Coppin State Ingepreneurship program produced the study “Turning Makerspaces Into Greater Places,” assessing Open Works’ impact on local business, workforce development (in which the college is deeply rooted), and institutional trust. [greaterspacesandplaces.com](http://greaterspacesandplaces.com)



### 4 Rural Studio

Auburn University, Georgia

### 5 DesignBuildUTAH

University of Utah

Where some schools invest in high-tech tools and collaborative spaces, others support making in a more nuanced way. At Auburn University's Rural Studio and the University of Utah's DesignBuild-UTAH program, prospective architects not only get lessons in design theory and CAD, but put their ideas into practice while learning engineering and construction skills in the field.

Importantly, these programs are community focused, “educating citizen architects” (as Rural Studio puts it) and building housing where it is most needed — for Utah's program in Bluff, that's the adjacent Navajo Nation. [ruralstudio.org](http://ruralstudio.org) and [designbuildutah.org](http://designbuildutah.org)



## Explore the List

The list of Best Maker Schools in higher education can be found at [make.co/maker-schools](http://make.co/maker-schools). Explore your options across the U.S. and the world.

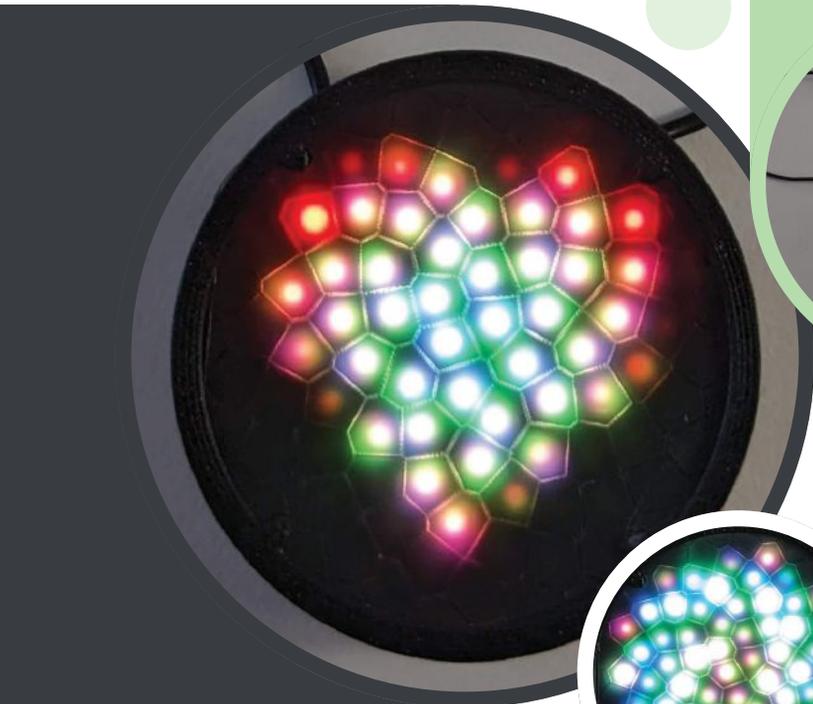
You can also submit a school for consideration at [make.co/maker-school-survey](http://make.co/maker-school-survey).



**DALE DOUGHERTY** is the founder and publisher of *Make:* and the president of Make: Community.



**JEN BLAKESLEE** runs the Global Maker Faire program and lives in Oakland, Calif. She has worked in events, radio, writing, education, big data, and public policy.



# A PERFECT CIRCLE

An LED “supergroup” collaborates on the Lux Lavalier **Written by David J. Groom**

**M**ad Season, Cream, Prophets of Rage, the Traveling Wilburys — what do they have in common? Perhaps it’s that you’ve never heard of them, but they are also all supergroups — successful artists teaming up to create something new and exciting. In the world of LEDs, instead of Layne Staley, Tom Morello, and Tom Petty, more familiar names might be Debra Ansell, Jason Coon, and Ben Hencke. So you can

imagine our excitement when we learned they were cooking up something together. Let’s meet these stars and learn how their solo projects led to an epic magnum opus — the tiny, mesmerizing LED pendant necklace known as Lux Lavalier.

## **BUSINESS IS BLAZING**

**Ben Hencke** (Figure A) is a multifaceted entrepreneur. He writes software, creates



**DAVID J. GROOM** loves writing code you can touch. If he’s not hacking on wearables, he’s building a companion bot, growing his extensive collection of dev boards, or hacking on 90s DOS-based palmtops. Find him on Twitter at @IShJR

hardware, manufactures products, stuffs boxes, runs the pick-and-place, manages support, and occasionally finds time for social media. His **Pixelblaze series of LED controllers** have taken the LED art community by storm, as they offer easy pattern creation via web browser with no coding required (learn more about LED controllers on page 70).

### LED BURN AND CAULDRON BUBBLE

**Debra Ansell** (Figure **B**) used to juggle the many hats of the entrepreneur as well, but prefers unbridled creativity to tax law, so has put aside bookkeeping to give herself fully to whatever inspiration strikes — **frequently wearables, always LEDs**. She describes her brain as a “chaotic cauldron of ideas” where she focuses on whichever project she is most passionate about, and pushes others down until they bubble back up again (sew her Pixelblaze Pillows on page 48).

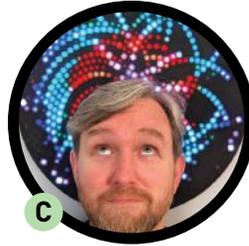
### GOOD MEETS EVIL

**Jason Coon** (Figure **C**) is a mild-mannered professional enterprise software developer by day. But at night, he assumes his alter-ego as the mad villain behind Evil Genius Labs. Jason started making LED art for fun, but soon became inundated with requests to buy his delightful *objets*, leading him to create the brand and its signature **Fibonacci spiral LED matrixes**. As an entrepreneur he describes himself as somewhere between Ben and Debra — he doesn’t want to make thousands of something but definitely more than one.

### STRANGE BREW

The ingredients to this *mélange* may seem obvious now, but the recipe evolved over time from the diverse experience of its chefs. For Debra it was wearables all along — she started with sewable LEDs, learned about Arduino, and graduated to a fiber optic dress that garnered a lot of attention. Over time she became aware of Jason’s Fibonacci boards and Ben’s Pixelblaze line, and as both started offering smaller versions Debra got excited about incorporating them into wearables.

Meanwhile, Jason and Ben had been collaborating for some time, Ben assembling



classic Fibonacci256 boards and Jason adopting Ben’s board to create the Fibonacci256 Pixelblaze (obviating the need for maintaining his own firmware). When it came to making a wearable, Debra was top of mind as co-conspirator and they kept sending her successively smaller prototypes, like RGB Matryoshka dolls.

### STARTING SMALL

As the prototypes shrank, the challenge became how to power the device on the go. Despite advances in battery tech, the mAhs required to move this many electrons for a sustained period are still substantial. Debra had already been playing with a behind-the-neck battery holder, which shifted the weight burden away from the device and out of sight behind the wearer. Her final design appears similar to a typical battery holder except for one weird trick: the circuit is completed by attaching a magnet to the negative terminal, making the necklace itself a conductive strand that provides power to the device.

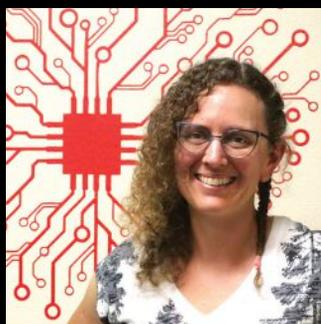
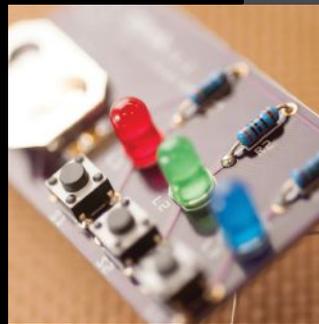
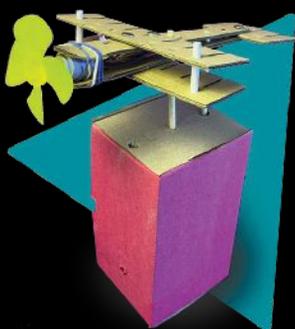
### CONVIVIAL COLLAB

Other challenges? Debra’s chief complaint was that Ben and Jason are too nice! They also had difficulty naming the project — with a Twitter poll resulting in the predictable “Pixel McPixelface” — before settling together on the final name: Lux Lavalier.

Parting thoughts? Ben, like many *Make*: readers, started out blinking LEDs on a dev board, and fell in love with that simple yet delightful aesthetic. Jason quips that he never moved past that stage, and Debra gladly proclaims, “Let there be light!” 🌟

# CALIFORNIA PRESS

Three California women makers who have started new businesses building and selling kits



**CARRIE SUNDRA** is the founder of Alpenglow Industries in San Luis Obispo, California. She grew up on the island of St. John in the U.S. Virgin Islands, then attended Harvey Mudd College where she proudly scored a 19 on her first electrical engineering exam.

**NANCY OTERO** is a Mexican-American entrepreneur, dancer, mother, and educator in Oakland, California. She loves to make projects and research the learning that happens with each one. The founder of KitCo, she is also working on creating AI-powered tools for making projects.

**EMMA FLETCHER** is a software engineer and hardware maker living in Sacramento, California, where she tinkers in both the digital and physical worlds. Learn to Solder Kits blend her love of engineering and entrepreneurship.

# #1

Alpenglow Industries, est. 2018

## "I Can Make Something Better" By Carrie Sundra

*Why am I even doing this?* These words go through my head late at night as I'm staring at yet another bill of materials, trying to figure out how to wrangle another dollar or two of parts cost out of a new kit. I mean, I could be making bank working for pretty much anyone, designing circuit boards or getting a new product ready for manufacturing. I could even do it from home now, thanks to Covid normalizing remote work. Instead, I'm trying to figure out how to sell soldering kits for \$10 or \$20 and make enough money to employ a few people and live a modest life in California. Yep, livin' the dream.

But I know why I do this. After 20-plus years of working for others in tech and designing circuit boards, I'm sick and tired of being the only technical woman in the room. The outlier in an overwhelmingly toxic monoculture that tries to normalize rude and disrespectful behavior by claiming that's just how nerds are, tries to prove worth by one-upping "cleverness" and losing all sight of practicality. They gatekeep their own little castles by purposefully speaking in jargon and then condemning anyone who dares to ask the meaning of a word, and generally put down anyone who just might have different priorities or opinions, or Bob forbid, want to look feminine.

I look at this terribly awful and broken thing and think, *I can make something better*. I'm not the first to think this, and I won't be the last. This better thing might not be big, it might not single-handedly change the world, but maybe if enough of us start making better places and support and encourage newcomers, maybe together we can start making a dent. Maybe we can at least create a small breath of fresh air, a place where it's OK to be a n00b at something, to experiment without the fear of judgment, to ask the meaning of words, and to make something pretty.



So I started a small business called Alpenglow Industries and we try to do all of those things. Encouraging people to pick up soldering irons and learn through-hole and SMT soldering skills is one thing we do through soldering kits that are a little out of the ordinary. Our weather-themed badge series is cute and approachable, while our SMT kits feature artwork that's creative and sassy. We're not afraid to put our values into our products — Foxy Pride is an SMT kit that allows you to pick your own LGBTQ pride colors in LEDs, and we're working on a Ruth Bader Ginsburg-themed board (RBG RGB!) and another called the FUturus.

We livestream and feature a variety of guests who are technical and creative, many of whom make great kits themselves. We make videos and write posts about electronics and programming, often sharing something we've recently dug into or something new we've learned.

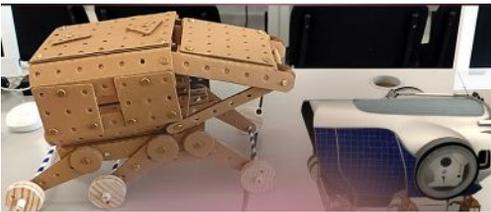
It's all about community and making a welcoming space, even if it's just a corner of the internet. *Ah, maybe if I switch out that 0.1" header and shunt combo for a switch, I can save 50 cents. Perfect!*

# #2

KitCo, est. 2019

## A Modular Canvas for Projects

By Nancy Otero



KitCo brings project-based learning into homes and classrooms through “challenges” linked to unique ideas that are worth learning. Our starting point is a cardboard kit that invites even technology-skeptical teachers and students to create projects. Once they feel like it, they can add textiles, paint, robotics, augmented reality (AR), or artificial intelligence (AI).

I love technology; I was a software engineer in my previous life, but my passion has always been learning, so I’ve been dedicated to education in the last decade. I have worked with educators from China, Brazil, Mexico, the United States, and Spain. I created professional development programs that trained hundreds of teachers in New York City, and built Fab Labs and makerspaces in all five boroughs. I co-founded a nonprofit in Mexico that implemented innovation labs with underserved high schoolers while researching their experiences with Stanford University. I was the founding director of learning

at Portfolio School, director of learning at Make:, and a fellow at OpenAI.

When the Covid crisis started, I was worried about children having so much screen time and not enough opportunities to externalize what was happening in their world. So I began KitCo, to support schools in implementing remote project-based learning, a type of learning that could give children agency — something so needed when we were all witnessing the world change without our say.

The design had a few constraints: I wanted to create something comfortable that didn’t scare anyone, not a STEM kit, but a modular canvas for projects. It had to be cheap, able to mix with recyclable materials, and able to grow into robotics or AR.

For a long time, it was me, a laser cutter, and many cutoffs of cardboard boxes. I dived into pieces I designed for old workshops, analyzed other kits, and conducted mini tests with friends and family. The real learning was the first time I implemented it in an elementary school, where I observed teachers and helped more than 300 children play and create with the kit.

Our cardboard kit is compatible with the most common motors and servos used for Arduino, micro:bit, and GoGo Board. The holes in our design are great for wooden sticks or pipe cleaners, and we have wooden pulleys that give projects smooth movement. We have pieces made for structural support, others for storytelling and wearables, and a set for mechanical functions.

After the first year, we were happy and astounded by all the different projects children created for our challenges. For our Defy Gravity challenge, a team of second-grade girls from Sunnybrae Elementary School created a solar system with robotics. For the same challenge, a kindergarten made a wearable that launched a ball through a lever mechanism. Today our subscription model includes a magazine that showcases some of those ingenious solutions children send us.

Next month we’ll be launching Hands and Dots, a series of experiences co-designed with extraordinary educators who understand how to operate in underserved and complex contexts.

Into the Wonder is our first learning experience in the series, and it's co-created with educator Carlos Espinosa. Into the Wonder questions the exploration and commercialization of space, and invites learners to design their space mission and create a rover using AR and robotics.

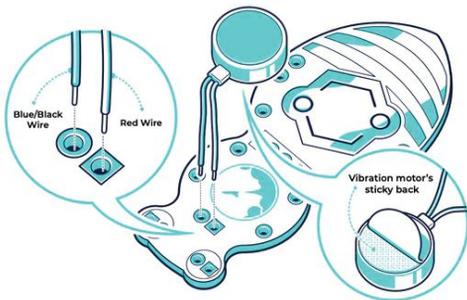
My dream is to amplify and scale the voice of extraordinary teachers who support their students in creating open-ended projects. I want to open-source our design so everyone can cut it and upcycle cardboard into learning materials, because learning through projects doesn't require fancy, expensive materials and step-by-step instructions.

# #3

**Learn to Solder Kits, est. 2015**

## Share The Love of Making

By Emma Fletcher



In 2015 when I volunteered at an all-day engineering outreach conference, an event for local high school girls, I wanted to share my love of making with them. I led a workshop on soldering, and I was sure I would inspire at least a few girls to become electrical engineers by the end of the day. That wasn't how it turned out.

The event organizers had purchased some low-cost soldering kits with many components. It was a challenging task to teach safety, explain soldering techniques, and expect the students to finish it all within an hour. By the end of the workshop, not a single kit was working. Half were finished but had broken during assembly, and the other half were incomplete. Worst of all, I had not been able to share the joy of building electronics

with the students.

There had to be a better way to teach soldering. I decided that if the kit I wanted didn't exist then I would create it. My requirements were that it would be simple enough to complete in a 30- to 60-minute workshop and that every kit would work at the end. I made a simple design of three pushbuttons and three LEDs. It was interactive and every student could walk away with a feeling of accomplishment.

I ordered my first run of printed circuit boards, a total quantity of 60. Sixty boards seemed like a lot but that was the minimum quantity I could order and still get the bulk discount. How would I find 60 people who wanted to purchase simple soldering kits? I decided not to worry about it and instead focused on sourcing parts. Once everything arrived, I packed up the first five dozen kits and printed some labels designed in MS Paint. The first run of Blink Learn to Solder Kit was ready to go.

I posted on Facebook, reaching out to other makers and engineering educators. Within the first day, I sold 25. All 60 were gone by the end of the week. I placed another board order and set up a small table for kit assembly. Before I knew it, I had schools asking for bulk discounts, and the Learn to Solder Kits brand was born.

From there, I decided to expand the line of kits we were offering. I partnered with Andy Colborn and created two more kits. The Hue kit features a large RGB LED and three potentiometers. Jitterbug is a fun bug-shaped board with vibration motors and two glowing red eyes. We opened an online store and made all the kits certified open source hardware.

Our kits are now shipped all over the United States and even internationally. Last year we shipped over 8,000 kits. This year in March I made the decision to leave my job as a software engineer and go full-time on Learn to Solder Kits. It is still early days for me in this transition but so far I am loving it! Not only do I get to focus on something I love each day but the business is growing. We're on track this year to double the number of units shipped annually. While I may not have created any electrical engineers at my first workshop, that failure ignited a business that has had a far further reach than I could have imagined. 🚀

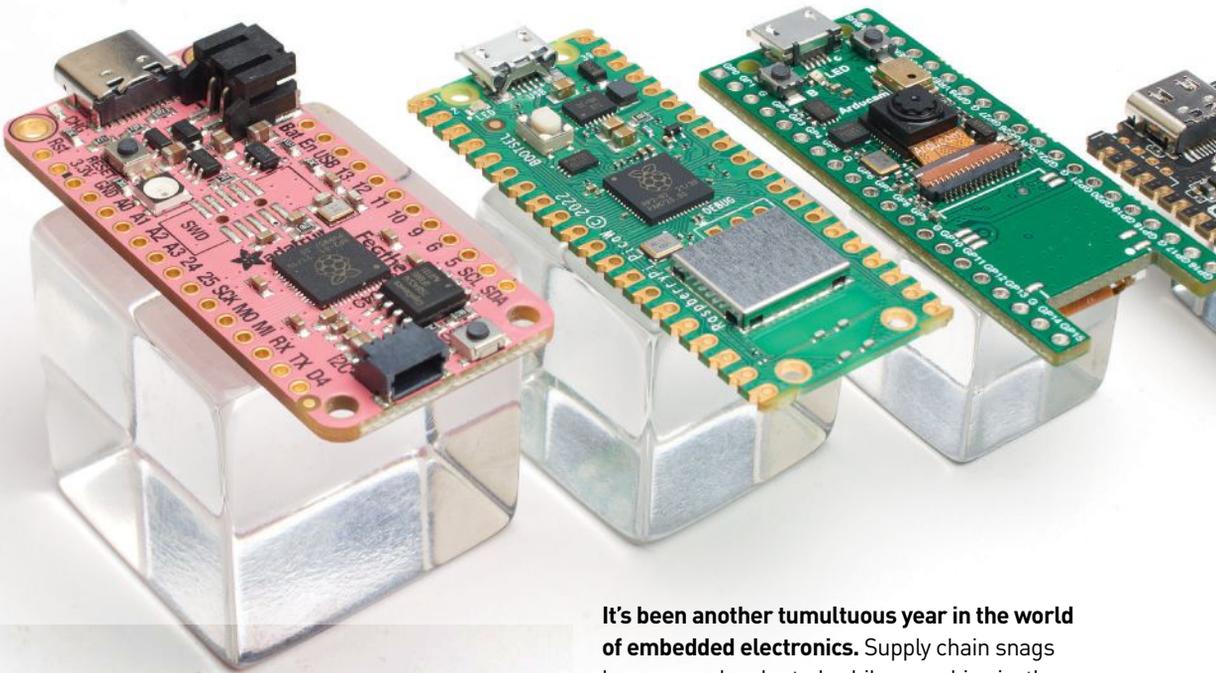
# NOW BOARDING

Need a board? Buckle up for a bumpy ride through all that's new in 2022

Written by David J. Groom



**DAVID J. GROOM** loves writing code you can touch. If he's not hacking on wearables, he's building a companion bot, growing his extensive collection of dev boards, or hacking on 90s DOS-based palmtops. Find him on Twitter at [@IShJR](#)



**It's been another tumultuous year in the world of embedded electronics.** Supply chain snags have scarcely relented, while new chips jostle for position as the maker's go-to. Meanwhile, lots of exciting new boards, software, and other technologies let us continue to innovate. Let's survey the maker landscape and see who stepped up when the chips were down!



## PUTTING THE NO IN ARDUINO?

Arduino Week in March was a brilliant virtual event, showcasing amazing new products — but frustrated makers couldn't get their hands on many items being showcased because supply issues prevented them from being manufactured.

But even when there's no hardware to hack on, software development can continue, which led to the release of the new **Arduino IDE 2.0** in September. The complete ground-up rewrite ditches 1.x's Java-based Processing origins for the Eclipse Theia framework, familiar from Microsoft's wildly popular and extensible Visual Studio Code IDE. Read more about it on our blog at [makezine.com/go/arduino-ide-2](https://makezine.com/go/arduino-ide-2) — and flip to page 47 in this section to sneak an exclusive peek at Arduino's exciting new **Make Your Uno kit!**

## PI NOT ALWAYS EASY

The Raspberry Pi Foundation is finding great success in the microcontroller market, as its Pico and new **Pico W** (wireless) microcontroller dev boards find their way into a plethora of projects, not to mention the **RP2040 chip** that powers them. Meanwhile the trend of porting Python projects to microcontrollers using Adafruit's

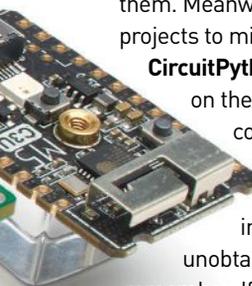
**CircuitPython** platform — instead of relying on the grunt of single-board computers — continues to prove popular, benefiting the whole RP2040 ecosystem.

But not everything is so sweet in Pi world, with SBCs practically unobtainable for as long as we can

remember. If you check [rpilocator.com](https://rpilocator.com) every day, you may be lucky enough to get your hands on a CM4 compute module, which you can pair with a carrier board like Timonsku's Piunora, but the stalwart Pi 4 is often only available via secondary markets, typically for several times list price.

## NO NEW NORMAL

Some experts predict we'll see things back to "normal" or at least stabilizing by 2023. In the meantime, we revel in the novel solutions and workarounds we see in the community, and we hope this special section, and our annual *Make: Guide to Boards*, help make your making a little easier this year!



## IN THIS SECTION:

### • **WHEN CHIPS ARE SCARCE** page 36

Scarcity continues to pinch makers and manufacturers alike. We interviewed a range of stakeholders from the supply chain spectrum, from distributors and big-name board makers down to individual maker pros, to ask how they're affected and when — or if — supplies of components will return to normal.

### • **RISC-V REVOLUTION** page 42

If you can't get the chips you want, why not make your own? Open silicon is evolving at an incredible pace, with RISC-V chip architecture expanding from experimental FPGAs into mainstream boards like Espressif's ESP32-C3. PINE64 is using RISC-V MCUs in its Pinencils and working on a 64-bit RISC-V single-board computer. Even laptops and cyberdecks are going RISC-V, with ClockworkPi offering an R-01 spec of its popular DevTerm deck using Allwinner's D1 64-bit single-core RISC-V chip.

### • **THE REPLACEMENTS** page 46

Makers are gonna make, whether there's a shortage or not. We've lined up some recommendations for when your favorite board turns unobtainium — use our infographic to see what substitutes might work for your next project!

### • **MAKE: GUIDE TO BOARDS 2022**

Included in the bag with this mag, you'll find our 12-page Guide to Boards, new and updated for 2022! We touch on top trends and dive into details of 79 of the hottest microcontrollers, single-board computers, and FPGAs — with an emphasis on boards you can get right now.

We also went hands-on with a dozen of our favorite new and notable boards, which you'll find described in the guide as well as in our exclusive video review. Don't forget to grab the Digi-Key AR app so you can watch the video and see the featured boards come alive before your eyes! 📺

**NOW BOARDING:** Supply Snafus



# WHEN CHIPS ARE SCARCE

Maker ingenuity in uncertain times

Written by David J. Groom



## The past few years have been ... unprecedented.

Supply chain failures manifest themselves visibly around us, from parking lots in Detroit full of cars that can't be finished for lack of a single part, to immense backlogs of container ships at the Port of Long Beach. From consumers all the way up to foundries, it feels like no part of the electronics supply chain has escaped the impact of material shortages and logistical quandaries.

We reached out to folks in the various strata of the maker community to hear their stories, how they've been impacted, and what the future looks like from their vantage point.

## KEY SUPPLIERS

**Digi-Key Electronics** started out as makers themselves, and have grown over the past five decades to become one of the world's largest distributors of electronic components. Today they supply parts for prototyping and production to everyone from the novice maker to some of the largest engineering companies on the planet.

But even at this level — or perhaps particularly at this level — supply chain issues can hit hard. In the past, the world at large has occasionally disrupted the world of components — the Great Recession of the late 2000s, the odd run on Raspberry Pis or ATmega328 DIPs — but during this latest shortage Digi-Key has been continually unable to attain their desired stock levels for many high-running parts. Not just silicon either, but connectors, and even some passives like SMT capacitors.

Paradoxically these shortages have often resulted in an increase in sales, as board designers over-buy instead of assuming parts will be available for the next re-spin. While the RP2040 chip has gained huge traction with makers, larger and more conservative customers have stuck with, for example, STMicro chips, but from different silicon suppliers. When even those suppliers weren't able to come through, Digi-Key ran into a unique issue: their systems aren't built to handle three-digit-month lead times!

Despite the tension between supply and demand, Digi-Key have been able to keep prices relatively stable for customers — it would be easy to tack \$10 onto each Raspberry Pi when they do get more in stock, but with an eye on long-term



relationships Digi-Key are doing everything in their power to help customers out. Thankfully, however, from their perspective the pinch is starting to plateau — and may slowly improve in 2023. Innovation isn't going away, nor are the fundamentals of electronics — more fabs will be created, but it takes time for the fruits of their efforts to trickle down the chain.

## HERO D'ITALIA

Even the mighty of the maker macrocosm are unable to avoid the impact of shortages. At their annual celebration in March, **Arduino** showed off a number of new products and projects, only to disclose uncomfortably that the featured board was out of stock or unavailable.

The company informs us that they'd planned to announce even more products but chose not to, due to their inability to get them produced. Furthermore, they're reworking some existing products to eliminate dependence on hard-to-find parts.

Consequently, the inability to source parts has caused a sea change in Arduino's product





## NOW BOARDING: Supply Snafus

planning and strategy. Instead of just hiring more hardware engineers, the company has invested in expert analysts to help forecast and meet demand. New boards are being designed to permit easy substitution of parts with similar footprints or even multiple footprints to accommodate availability, so they can easily switch production back to the original/full-featured product should a missing component suddenly become available again.

Since Arduino control their whole ecosystem, part substitutions — like switching a 9-axis IMU to discrete 6- and 3-axis components — can often be made transparently by updating the board definition or library. Sometimes, however, the required changes are more drastic, and necessitate a product variant such as the Portenta Lite, which lacks the NXP security chip found on the original. Fortunately, thanks to their mission as a learning community, Arduino occasionally receives favors from suppliers who would rather see what few chips they can source end up in schools rather than buried in less virtuous applications.

Other than a few advanced processes (for example 8-layer PCBs), most Arduino products are still made in Piedmont, Italy, near their headquarters in Turin. In order to provide the highest level of support for all these variants, Arduino have worked serial numbers into their products, for traceability from PCB to release, as well as making boards easily identifiable visually so that support staff can immediately confirm the specifics from a user-supplied picture. Arduino's significant investment in automated testing of software and hardware also facilitates hardware variants, since they can ship with confidence that a slight mutation from the original product will still work with all the different libraries and sketches.

Thanks to their adaptations, Arduino have actually increased deliveries compared to pre-pandemic levels. Although Arduino, and even their suppliers, continue to have difficulty forecasting the supply chain, they assure us they'll keep fighting to bring us the products we want and love. [For an example, check out the brand new Make Your Uno kit on page 47.]



## PI VERSUS .PY

**Adafruit Industries** have been so impacted by the chip shortage that they wrote a little jingle and created a web series about it! Having encountered supply issues early on, they were able to quickly pivot to an availability-first design methodology for new products, instead of designing for manufacturing and relying on just-in-time inventory.

Another advantage is Adafruit's significant investment in Python as a hardware programming language. Not only does their extensive support for CircuitPython make getting into hardware development more accessible, but it also means incredible portability across boards and architectures. With their Blinka library, even single-board computers can get in on the act, with full CircuitPython API support meaning that the same code that runs on your Feather and QT Py can easily move back and forth to Raspberry Pi, BeagleBone, Jetson, and ClockworkPi.



Watch Adafruit's Chip Shortage series and get the theme tune stuck in your head: [youtube.com/hashtag/chipshortage](https://www.youtube.com/hashtag/chipshortage)

David Ray

## BOARD BIZ BLUES

David Ray of **Cyber City Circuits** gives us a fascinating view from another stratum of the shortage. His company in Augusta, Georgia, provides PCB design and prototyping services for small businesses, which gives them unique insight into the challenges faced across a range of projects.

One interesting facet that didn't come up in other conversations was the prevalence of counterfeiters and scalpers in the supply chain. It turns out that if a hard-to-find microcontroller like an ATmega328 mysteriously becomes available at a single random source, chances are it's a fake. But not so fake that it's immediately obvious. Basic functionality like blinking an LED will have parity with the genuine article, but then using the ADCs will reveal that they're of disastrously low quality, or an attempt to use interrupts will reveal that they simply do not exist.

And it's not just suppliers who are faking it. Overseas brokers are just as likely to burn customers and come back the next day with a different name and Skype account. If they claim to be able to get that one part that nobody else can — be wary.

As other vendors mentioned, it's not just components that are more expensive, but basic materials like solder paste, as well as staff, electricity — everything. In addition to scarce SAMD21s and ATmega328s, voltage regulators like the LMR23610 and even proprietary parts like Cinch connectors have caused designs to have to be started from scratch when supplies evaporated. Asked when things will get back to normal, David makes a subtle Deltron 3030 reference: "Crisis precipitates change."



David Ray (left) and Chris Williamson (right) in front of CCC building.

## PREDICTIONS: WHEN WILL THINGS BE BACK TO NORMAL?

- **Kevin Walseth, Digi-Key**  
Technology's not changing — a new car has over 10,000 capacitors, and that's just cars! [We're hearing predictions of a] slow ramp up in 2023, then a slow decline in lead times.
- **Alessandro Ranellucci, Arduino**  
Even suppliers seem to have trouble forecasting. [Arduino is] going to continue to fight!
- **Phillip Torrone, Adafruit**  
We've been preparing for all this, at least in the ways one can — we're now "designing for purchasing" versus "designing for manufacturing" and/or "just-in-time inventory."
- **David Ray, Cyber City Circuits**  
Crisis precipitates change. Lots of small companies [will] get bought up ... things we like get deprecated and not reissued ... [we] don't know what normal is anymore!
- **Diego Fonstad, Lectrify/Imagination Supply Co.**  
I think it's kind of like "When is Covid going away?" I don't think it's ever going back to normal; I hate [the phrase] "new normal" but the extreme flexibility we were used to [is past] — we'll have to do more planning, be more flexible. It's going to be more local.
- **Joey Castillo, Oddly Specific Objects**  
[I'm] adapting the way I work and design things [and] will probably keep that mindset. Simplify! Simplify!! Make it as easy on yourself as you can.





Keith Hammond, Ben Sklar, David Ray

# NOW BOARDING: Supply Snafus



## MAKEY: BIT BY BIT

Diego Fonstad of **Lectrify/Imagination Supply Co.** makes kits for students K-12 that teach electronics, engineering, and coding (and delightfully, all of them are Lego-compatible!). Lectrify partnered with *Make:* to create the Makey:bit Adventure Board — a non-rectangular, robot-inspired micro:bit expansion board who’s fun to be with (find it at [makershed.com](http://makershed.com)).

As operations have expanded, Diego has increased his in-house production capabilities in San Francisco, including a Neoden pick-and-place machine and reflow oven to facilitate custom PCB fabrication — technology that would have been prohibitively expensive not long ago. This lets him build small batches just in time versus sending out for thousands of boards at once; it also eliminates the overhead of communication over specs and BOMs, and facilitates short-run experiments and tweaks. Instead of sitting on finished inventory, Diego can stockpile common components and manufacture whatever’s selling. Smaller production runs also open up opportunities like scrounging around for partial reels which can be re-taped and fed to save a few bucks.

In terms of sourcing, it’s not the more complex parts that have been difficult; simple components like SMD ceramic caps have sometimes had 3-month lead times, and a critical voltage divider could only be found on eBay this side of 2025. Even low-tech hardware like nylon screws, and commodities like wood for laser-cut designs, or

specific box sizes, have presented challenges. The flexibility afforded by in-house production, however, means that multiple parallel footprints can be developed to fit whichever component is available, and parts don’t need to be committed to boards until the last moment.

On top of all the sourcing issues, tariffs have been another great hurdle, adding a surprise half grand or more to the already high price of parts and shipping. Diego doesn’t see a future where we return to the extreme flexibility we once took for granted; instead we’ll need to plan much further in advance, and when possible, focus on small-batch runs that facilitate flexibility.

## KEEP IT SIMPLE, WHIZ KID

For burgeoning maker pro Joey Castillo, the focus is on simplification. Joey’s hardware engineering practice **Oddly Specific Objects** makes gadgets that explain themselves and how they were made, unlike so many seemingly magic black boxes out there. Joey believes that since these are all devices made by people, they can be understood, and he does an amazing job of helping people to understand, from his detailed process tweets as [@josecastillo](https://twitter.com/josecastillo) to his boards’ own silkscreens which contain extensive self-documentation.

Joey’s first two products are the Sensor Watch — a hackable replacement PCB for the venerable Casio F-91W — and the LCD FeatherWing, which lets makers add a watch-inspired LCD to their projects based on the Adafruit Feather ecosystem. But these weren’t his first attempts at

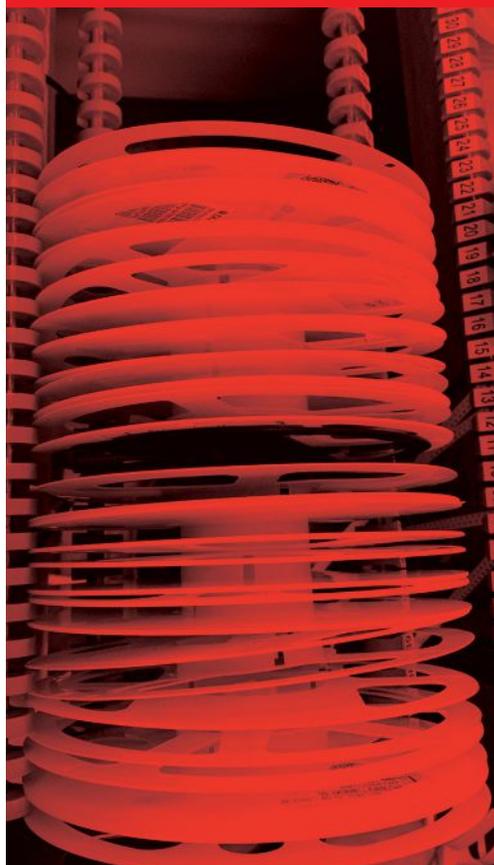


creating a product; his SAMD51-based PyCorder sensor gadget was quickly stifled when he found that obtaining more than the 15 SAMDs he'd used for prototyping was not going to be possible until 2023. Until then, Joey had relied on just-in-time ordering, knowing that whatever he desired was just two days away via Digi-Key.

For his Open Book project, Joey has stopped chasing SAMDs and switched to the Raspberry Pi Pico, which can be soldered straight to his updated board thanks to its castellated edges. And instead of worrying about battery charging ICs, he redesigned the Book to use ordinary AAA batteries. In addition to embedding multiple footprints to accommodate equivalent available parts, Joey has begun exploiting every peripheral of his microcontrollers before reaching for a separate dedicated component. Forged in the flames of the pandemic, Oddly Specific Objects now designs their products, like the LCD FeatherWing, with minimization of supply chain impacts in mind. In an uncertain and scarce world, it's best to keep things simple.

## SURVIVAL OF THE FLEXIBLE

Predicting the future was never easy, but the challenges of recent years have made even reasoning about tomorrow impossible at times. The overarching sentiment of the folks we spoke to was that nobody has a firm grasp on what the future may hold, and the best approach is to adapt within the parameters that you can control and bend with the changing winds. 🍀

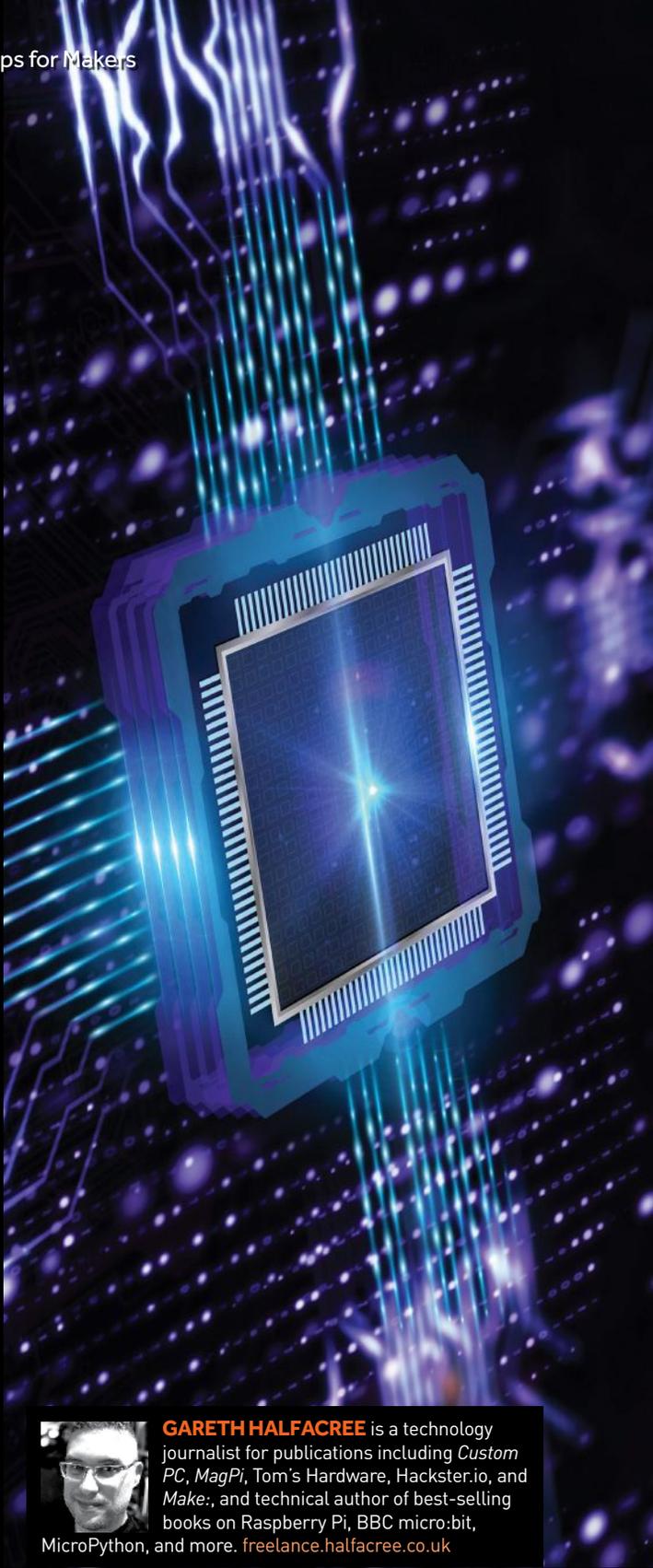


**DAVID J. GROOM** loves writing code you can touch. If he's not hacking on wearables, he's building a companion bot, growing his extensive collection of dev boards, or hacking on 90s DOS-based palmtops. Find him on Twitter at @IShJR

# A RISC-V REVOLUTION

The rise of free and open-source silicon for makers

Written by Gareth Halfacree



**GARETH HALFACREE** is a technology journalist for publications including *Custom PC*, *MagPi*, Tom's Hardware, Hackster.io, and *Make.*, and technical author of best-selling books on Raspberry Pi, BBC micro:bit, MicroPython, and more. [freelance.halfacree.co.uk](http://freelance.halfacree.co.uk)



**Makers are well-used to the benefits of free and open-source hardware, from being able to study and iterate on designs by others, to the promise that if a design goes out of production you can always make replacements yourself.**

Open-source silicon, though, is not so well understood — but thanks to the success of the RISC-V (“risk five”) project, that’s changing.

The idea of an open processor, designed around the concept of reduced instruction set computing (RISC), from which others could learn isn’t new: The Berkeley RISC project, which launched in 1981, famously published its research papers for all to read, providing a direct inspiration for Sophie Wilson and Steve Furber to create the Arm architecture that so many chipmakers pay to license today.

### ARM ALTERNATIVE

The Berkeley RISC project inspired numerous processor projects, some open and others closed, but it was the launch of the RISC-V project in 2010 that truly lit a spark.

Born at the same university as the original, RISC-V is Berkeley’s fifth-generation RISC architecture — and by far its most successful. Originally pitched by Krste Asanović as a “short, three-month project” aimed at graduate students before attracting Berkeley RISC alum David Patterson, RISC-V has leapfrogged its forebears to become the most successful free and open-source architecture in history.

### FREEDOM IN SILICO

There’s an important distinction to note: “free” as well as “open source.” Sun’s microSPARC processor was open source, but not free: Anyone wishing to make modifications needed a commercial license. RISC-V, by contrast, is free-as-in-speech: Anyone can implement RISC-V either as-written or with as many tweaks, modifications, and extensions as they desire — and never have to pay a cent in royalties or license fees.

What began as a 3-month university project now ships millions of cores a year: RISC-V implementations are found in commercial products including smartwatches, fitness bands, storage products, and graphics cards, where the



**ARM IN ARM:** Inspired by Berkeley’s 1981 RISC project, Sophie Wilson (far left) and Steve Furber (third left) created the Arm microprocessor architecture that’s now ubiquitous.

allure of true freedom — plus a bundle saved on license fees — has won out against the desire to keep proprietary IP suppliers on-side.

Unsurprisingly, RISC-V has also been making inroads into the maker sector — slowly at first, but gaining momentum with each passing year. Low-power microcontroller parts came first, with application processors soon following. Server-class hardware, including proposed 128-bit chips designed to accompany existing 32- and 64-bit parts, is right around the corner.

### FROM FPGAS TO CHIPS

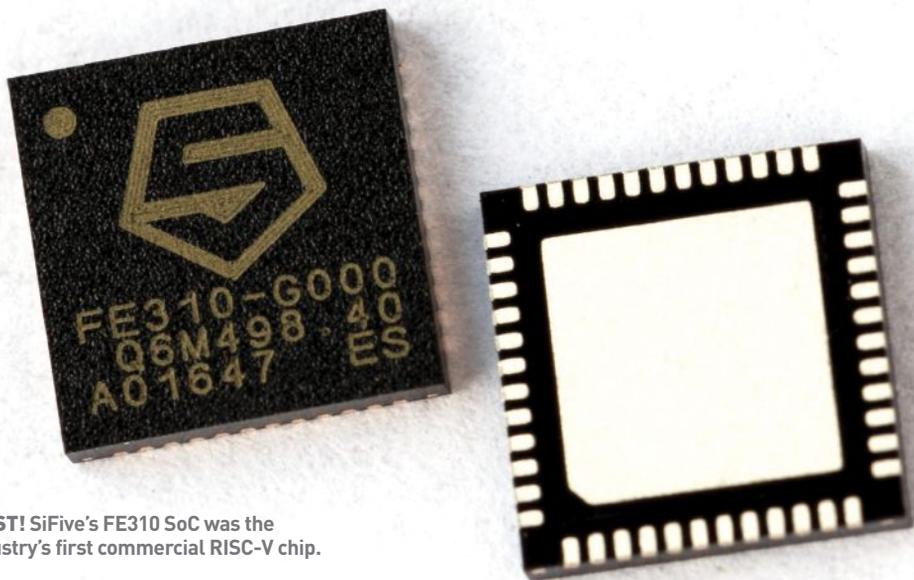
At first, experimentation was a challenge. Few RISC-V designs had been committed to silicon, as the specification had yet to be ratified, so if you wanted to develop for RISC-V you needed to use field-programmable gate array (FPGA) hardware to run soft-core implementations — or even emulate RISC-V in software on a mainstream Arm or x86 chip.

SiFive, co-founded by Asanović himself, was



**RISC REWARD RATIO:** SiFive, co-founded by RISC-V project leader Krste Asanović, was first to market with RISC-V chips and Arduino-compatible boards in 2016, and Linux boards in 2018. Today they’re designing a 12-core CPU for NASA’s new High-Performance Spaceflight Computing (HPSC) processor.

Adobe Stock-puftlov\_denis and arthead, Trevor Johnson



**FIRST!** SiFive's FE310 SoC was the industry's first commercial RISC-V chip.

SiFive, Espressif Systems, StarFive, PINE64

one of the first to offer actual silicon chips — microcontrollers initially, then Linux-compatible cores on a single-board computer. Others followed: Today, you can buy RISC-V chips from Allwinner, Bouffalo Lab, GigaDevice, GreenWaves Technologies, and StarFive, with companies including Alibaba, Google, Seagate, and Western Digital developing chips for in-house use.

## MAKING A SPLASH WITH MAKERS

The biggest news for the maker market, though, came from Espressif Systems when the company announced that not only was it launching RISC-V based products into the ESP32 microcontroller family but that it would concentrate solely on

RISC-V architecture from now on, ditching proprietary alternatives like the Cadence Tensilica Xtensa architecture.

So all future ESPs will be RISC-V? “Yes, it is true,” Teo Swee Ann, Espressif chief executive and president, confirmed on LinkedIn. “Unless we have some special needs for something else that I don’t see now.”

What about the maker’s favorite mini computer? “The main things holding RISC-V back in the traditional Raspberry Pi/[Arm] Cortex-A market,” says Eben Upton, Raspberry Pi CEO, “are a lack of available high-end licensable cores — I don’t think I can go out and get anything that’s competitive with the Cortex-A72 in Raspberry Pi 4, for example — and a lack of software maturity in the Linux userland.”

“The barriers may be a little lower in the microcontroller/Cortex-M space, as the software stacks and core design space are simpler. If/when RISC-V really takes off,” Upton predicts, “that’s where it will happen first.”



**CHIPS ALL IN:** Espressif CEO Teo Swee Ann has committed to RISC-V architecture for all future chips. The bare ESP32-C3 chip can be had for about \$1; the dev board for \$8.

## SINGLE-BOARD SYSTEMS

There are numerous single-board computers that aim to offer a RISC-V alternative to Arm-based devices like the Raspberry Pi range. At first, choices were limited: SiFive’s boards offered



impressive performance but priced themselves out of the maker market, while devices based on Allwinner's D1 chip struggled with poor performance. Gadgets like Microchip's PolarFire SoC Icicle Kit offer RISC-V cores, too, but they play second fiddle to the board's FPGA resources.

But now, the StarFive VisionFive — a dual-core 64-bit single-board computer running Linux — offers a reasonably affordable entry point, with its follow-up, the VisionFive 2, boosting performance and slashing costs.

PINE64, best known for its Pinebook and Pinephone ranges, is also getting in on the act: After putting a RISC-V microcontroller inside the Pinecil and Pinecil 2 soldering irons, the company is now preparing to launch the Star64, an open-source single-board computer built atop the same StarFive JH7110 chip as the VisionFive 2.

## BUILDING YOUR OWN CHIPS

For many makers, architecture will always take a back seat to features when it comes to choosing a chip. But RISC-V, and other free and open-source silicon efforts, provide a whole new playing field for the curious maker: The ability to get down and dirty with the architecture itself in a way that previously would have required a decade of education and a job application to Intel, AMD, Arm, or the like.

"RISC-V essentially gives you the freedom to implement and customize the processor core to your needs," explains Stefan Wallentowitz, who sits on the board of RISC-V International to represent community members. "While the average maker will probably not build chips at a commercial scale, there are efforts like the Open Multi-Project Wafer for fully open-source chips," he adds, referring to a Google-funded project that lets designers of open-source silicon have their chips built at SkyWater or GlobalFoundries fabrication facilities at absolutely zero cost — something never before possible. "Free and open-source silicon makes learning digital design and computer architecture accessible and fun."

Even Intel, which has a vested interest in pushing people toward its own proprietary x86 architecture, would seem to agree: In August the company launched Pathfinder for RISC-V, a development environment for RISC-V systems-



**FRESH LINUX SBSCS:** StarFive's quad-core, 1.5GHz RISC-V JH7110 processor is at the heart of their new VisionFive 2 single-board computer (top) and also PINE64's new Star64 (bottom) — both with the familiar Raspberry Pi-format GPIO header. Debian and Fedora Linux distros are already being ported to the JH7110.

on-chips, with a free-of-charge Starter Edition which Intel has specifically pushed to the hobbyist, academic, and research communities.

"Intel Pathfinder for RISC-V represents our ongoing commitment to accelerate the adoption of RISC-V," claimed Intel's general manager for RISC-V ventures Vikay Krishnan at the launch, "and catalyze the ecosystem around an open source and standards-based vision." 🗳️



Learn more about RISC-V at [riscv.org](https://riscv.org)

# THE REPLACEMENTS

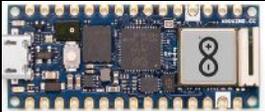
## Don't let supply chains slow you down

Written by David J. Groom

We can't hardly wait for the chip shortage to be over and dev boards to be abundant again, but instead of waiting for somebody to fix the supply chain, we've gathered a list of near substitutes and potential replacements for your next project. Unsatisfied with back-orders for your favorite Raspberry Pi or Arduino? Use the infographic below to find an alternative — maybe it'll turn out to be your new favorite thing!

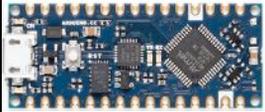


**ARDUINO NANO 33 BLE SENSE**



**Arduino Nano RP2040 Connect**

Same form factor but RP2040 (Dual M0+) instead of nRF52840 (M4)



**Arduino Nano Every**

Same form factor but 8-bit ATmega4809 vs. 32-bit nRF52840



**BBC MICRO:BIT V2**



**Adafruit CLUE**

Same form factor, similar nRF chip, but 240px LCD display instead of 5x5 LEDs



**ElecFreaks Pico:ed V2**

Same form factor but RP2040-based and larger/denser 7x18 LED matrix



**PJRC TEENSY 4.0**



**PJRC Teensy 4.1**

Larger form factor but adds more flash, mSD slot, Ethernet pins



**SparkFun MicroMod Teensy Processor**

Same chip, in SFE's MicroMod ecosystem (requires MicroMod carrier)



**RASPBERRY PI ZERO 2 W**



**MangoPi MQ Pro**

Same form factor, Allwinner D1 RISC-V core



**Banana Pi M2 Zero**

Same form factor, quad-core Cortex A7 Allwinner

**OUT OF STOCK**



**RASPBERRY PI 4, MODEL B**



**Rock Pi 4 Model C+**

Pi form factor, 64-bit Rockchip RK3399 (dual-core Cortex-A72 + quad Cortex-A53)



**Khadas VIM4**

64-bit Amlogic A311D2 (quad-core Cortex-A73, quad-core Cortex-A53), Pi-compatible 40-pin header



**DFRobot LattePanda 3 Delta**

64-bit Intel Celeron N5105 (quad-core x86-64); own form factor and tons of I/O



**BeagleBone family**

Stalwart SBC range — own form factor, varied capabilities

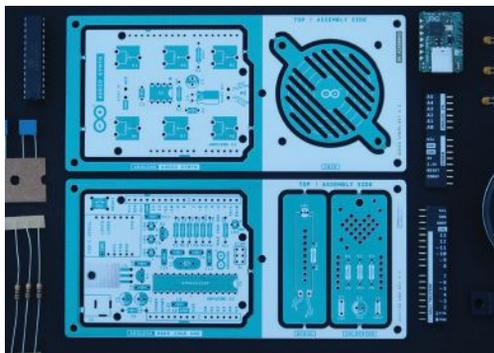


**PINE64 Quartz64 Model B**

Pi form factor, 64-bit quad-core Rockchip RK3566

*NOTE: Non-Pi options other than BeagleBone and PINE64 may lack community/support.*

## BUILD-A-BOARD SYNTH-POP



David J. Groom

### An exclusive sneak peek at Arduino's back-to-basics Uno kit!

Early Arduinos mostly consisted of a simple PCB, an ATmega8 and a handful of simple components, plus a DB9 or USB connector for programming and debugging. Today's most powerful Arduinos, such as the Portenta X8, feature multiple cores running 100 times the speed of the original. But sometimes it's fun to go back to where it all started, which is what Arduino has done with their new **Make Your Uno Kit**, announced at Maker Faire Rome as this issue went to press!

This exciting release comes as a box of parts that are easily solderable by most novices, thanks to choices like an assembled USB-C module, as opposed to the tricky surface-mount FTDI chips of those early days.

The Uno form factor popularized the original shield ecosystem, so it's only logical that the kit includes a DIY add-on for the full Uno experience! And what an exciting shield this is, allowing you to transform the kit and its packaging into a tiny little electronic audio synthesizer! 🎵



**DAVID J. GROOM** loves writing code you can touch. If he's not hacking on wearables, he's building a companion bot, growing his extensive collection of dev boards, or hacking on 90s DOS-based palmtops. Find him on Twitter at @iShJR



# Pixelblaze Pillows

Written and photographed by Debra Ansell

Use robust outdoor LED strings and the awesome Pixelblaze controller to make these cuddly, full-color animated throws



**DEBRA ANSELL** is a maker and educator who will never stop demonstrating that LEDs improve everything.

It's a profoundly human compulsion to change our environment to suit ourselves, and the quarter-trillion-dollar U.S. home décor industry presents a playful (and profitable) example: We bedeck our living spaces to express ourselves and illustrate our individuality. Colorful light displays decorate busy event spaces like concerts and sports arenas, but they can also function as enchanting home ornaments, like my LED Rainbow Weather Station (*Make*: Volume 69) and Inner Glow Edge-Lit Heart (Volume 71).

## ILLUMINATING INTERIOR DESIGN

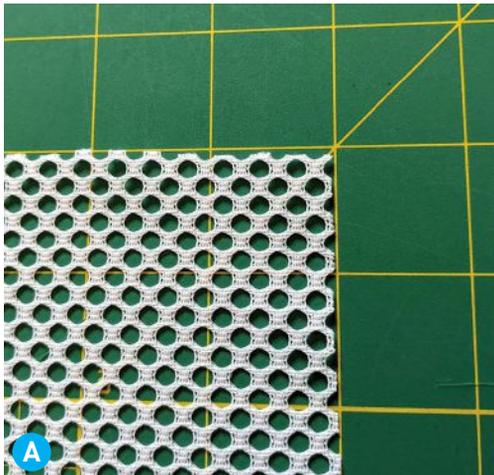
This LED matrix pillow project is not only pretty but practical as well. The electronics attach to a removable DIY pillowcase, so the outer decorative cover and stuffed pillow insert may be removed for cleaning. You can change display colors and patterns, controlled by the codable Pixelblaze controller board, for both visual embellishment and entertainment.

There is something unexpectedly delightful about embedding electronics in soft, cuddly objects, and these sparkly pillows emphatically spark joy inside my home.

### 1. CUT FABRIC AND MARK A GRID

Cut six rectangles from the mesh fabric: two 21"×21" and four 13"×21". Cut the edges parallel to the rows and columns of holes in the mesh, to make aligning the matrix easier (Figure A).

Fold a 21" square piece in half and make a dot with the Sharpie at one end of the midline (Figure B on the following page). Unfold the



**TIME REQUIRED:** 10–15 Hours

**DIFFICULTY:** Intermediate

**COST:** \$150–\$175 Pair

## MATERIALS

- » **Pixelblaze V3 Standard Controller** \$35 from [shop.electromage.com](http://shop.electromage.com). Works with virtually all types of individually addressable LEDs.
- » **Throw pillow inserts, 20"×20" (2)** e.g. Amazon B08Z3HRKD3
- » **Decorative pillow covers, 20"×20" (2)** in translucent fabric, like white shag or velveteen, e.g. Amazon B09FXJD43H or B098T48DLF
- » **Quilt batting, lightweight (about 1/2" thick), cut in 20"×20" squares (2)** Amazon B0017058TU
- » **Outdoor RGB LED strings, 10 meters (2)** with addressable LEDs, 10cm spacing, and transparent insulated wiring, e.g. AliExpress 3256803638380385 or 3256802720232748. These have 3-pin JST-SM connectors on both ends, but many similar strings have an attached USB controller instead, in which case you'll need to cut off the controller and solder a 3-pin JST SM connector in its place.
- » **JST-SM 3-pin connectors (2) (optional)** if your LED strings don't already have them, Amazon B07787L74H
- » **JST-SM 3-pin extension cable(s)** e.g. Amazon B07G6PRDBQ. Quantity and length will depend on how far apart you wish to place your pillows.
- » **Polyester mesh fabric, with 1/8" holes (1 1/2 yards)** Amazon B00N9I806Q
- » **Twist ties, double wired, white, 10cm long (pack of 100)** Amazon B08816GSYF
- » **DC wall adapter, 5V 3A (or higher amperage)** with DC jack connector and extra DC plug breakout with screw terminals, e.g. Amazon B078RXZM4C or B01GCI9G4O
- » **Thread, white**

## TOOLS

- » **Wire cutters or flush cutters**
- » **Wire stripper**
- » **Fabric scissors**
- » **Sewing machine** or needle and thread
- » **Ruler**
- » **Sharpie pen, dark color**
- » **Sewing clips** such as Amazon B08KGTFR6T, or small binder clips
- » **Pliers**
- » **Small screwdriver**
- » **Soldering iron (optional)** if you need to attach screw terminals to the Pixelblaze board

## PROJECTS LED Matrix Pillows

square and lay it on a flat surface with scratch paper underneath. Now use the Sharpie to draw two vertical marks that are 2.5cm on either side of the midline. Continue to draw vertical marks at 5cm intervals on both sides until there are 10 symmetrically spaced marks (Figure C).

Next, using a ruler as a straightedge that is aligned with the mesh holes, draw lines to extend the marks to the opposite side of the mesh square. Then turn the mesh fabric square 90 degrees, and follow the same procedure to draw identical lines running perpendicular to the first set. Now you have a 9x9 grid of 5cm squares, centered in the mesh square (Figure D). Repeat this process with the second 21" square.

### 2. SEW THE PILLOWCASES

Lay one 21" mesh square on a flat surface with the marked side up. Place two 13"x21" rectangles over it so that their long edges align with opposite sides of the square. The rectangles will overlap each other by 2½" in the middle. Use sewing clips or binder clips to secure the edges of all three pieces around the perimeter (Figure E).

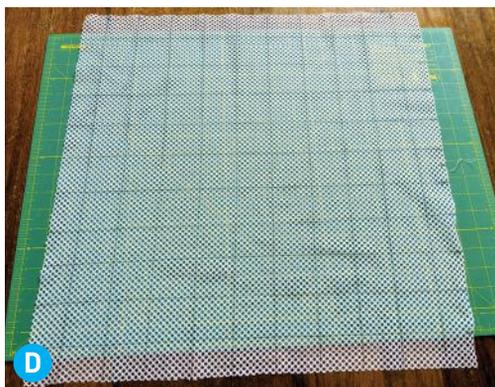
Use your sewing machine or a needle and thread to sew a seam with a ½" allowance around the entire perimeter of the sandwiched fabric pieces. Reinforce the seams by back-stitching near the corners and the places where the rectangular layers overlap; or if you're sewing by hand, use small, tight stitches to secure the fabric well and knot the thread ends tightly.

Once the mesh pillowcase is sewn, use sharp scissors to trim the four corners at a diagonal, just outside the seam (Figure F). Turn the case inside out (Figure G), pushing the corners out from the inside with your fingers. Insert a pillow insert into the mesh case, plumping and tugging the pillow corners to fill the case without slack.

### 3. MAKE THE LED MATRIX

The grid drawn on the cases has a 5cm spacing but adjacent LEDs on the strings are 10cm apart. We'll reconcile this difference by weaving the LED string into a pattern of two perpendicular zig-zag grids that overlap perfectly to place one LED at each grid intersection.

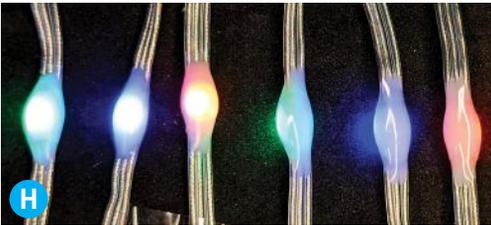
Each of the resin-encased LEDs has a distinct front and back side (Figure H). The back is a bit





flatter and noticeably dimmer than the front. Be careful to place the LEDs front side up.

To attach the matrix to your mesh pillowcase, you'll use the double-wire twist ties. Don't use standard twist ties — they'll twist around and poke through your pillowcase. Using wire cutters or flush cutters, cut about 50 of the 10cm ties into approximately equal fourths about 2.5cm long (Figure I). Then, take the input end of the LED string and lay it along one of the border grid lines, aligning the LEDs with the intersections.



You'll secure the LED string to the mesh case at regular intervals by wrapping it in a cut twist tie near one side of each LED. Don't tie both sides of the LEDs because there must be enough slack to run part of the string underneath itself. Slide a cut twist tie through two adjacent holes which border the Sharpie line so that the tie is perpendicular to the string (Figure J). The tie is slightly wider than the holes, so you may have to pull the mesh taut and wiggle it a bit to slip it through.



Using pliers, sharply fold about one-quarter of the tie over the LED string (Figure K), then fold the opposite side of the tie extending from underneath the LED string to cover the first folded section (Figure L). With the pliers, bend and crease the remaining overhanging twist tie section back underneath the string and tie, pressing it firmly to grip the wires inside. Tuck the final short segment into the mesh under the string (Figure M).



Proceed this way down the entire first row, placing one twist tie just before each LED (Figure N on the following page). At the end of the row, bend the string into a U-shaped turn and place the next LED in the closest adjacent row so that it is offset by one column from the previous LED (Figure O). Proceed to fill all the rows using this technique until half the LED string is attached to the case in a zig-zag pattern with one folded twist tie near each LED. The LEDs should sit at



## PROJECTS LED Matrix Pillows

alternate grid intersections, so that the LEDs in each row are offset by one from their neighbors (Figure P). Double check that all LEDs have front sides oriented upward, away from the pillow.

Next, weave the second half of the string in and out of the attached rows. Start by turning the pillow 90 degrees, then lay the next loose portion of the string across the closest perimeter row. Fasten the next LED at the intersection nearest the pillow corner as shown in Figure Q.

Continue connecting the LED string along the rows like before, sliding the loose portion of string under each of the LEDs in perpendicular columns (Figure R). When you're finished, the LED string input/output connectors should extend from opposite pillow corners and the string will be woven into a pattern that looks like Figure T. You can use more ties to affix any parts of the string that seem loose.

After the first LED matrix is secured, repeat the process with the second pillowcase. This step can take hours for two pillows; you might want to cue up a podcast or TV show for distraction.

### 4. SET UP PIXELBLAZE AND POWER

Next you'll connect the Pixelblaze controller and power supply to one of the LED matrices. Both the Pixelblaze and LED string operate at 5V and can be powered with the same source. Don't run the power for the LED string through the Pixelblaze USB micro-B connector, as it is not designed to carry that much current.

The specific wiring layout will depend on whether your LED string connector has extra power wires broken out. If so, it is straightforward to attach those extra power wires to the barrel-jack screw terminal connector as shown in Figure S, then connect an extra 3-pin JST-SM cable to the LED string input, and connect that cable's Power, Data, and Ground wires to the corresponding screw terminals on the Pixelblaze.

If your connectors don't have extra power wires broken out, you'll need to splice hookup wire to the JST connector's 5V and GND wires and attach those hookup wires to the DC jack's screw terminals to supply power.

Once the Pixelblaze and the power adapter have been connected to the input end of one matrix, use a JST-SM extension cable to connect the first



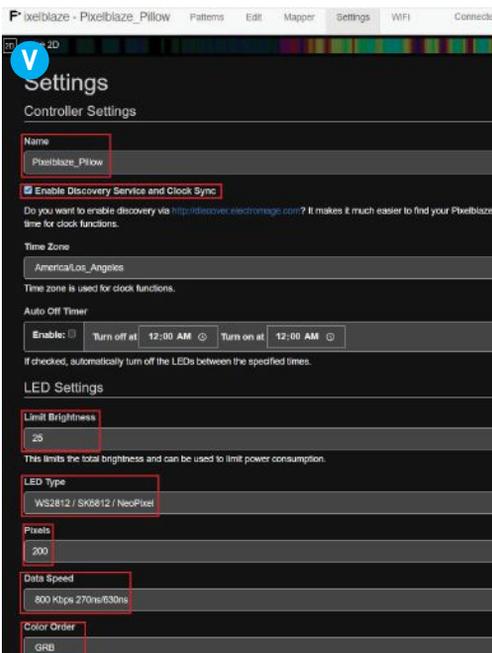
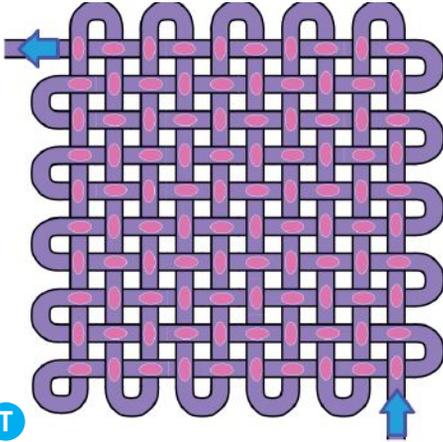


pillow's output to the second pillow's input. If you want to move the pillows further apart, you can insert a second (or third) extension cable between them. With wiring completed, plug in the power source into a wall jack to turn it all on.

When powered, a new Pixelblaze board will automatically boot into access point mode with its own unsecured Wi-Fi network starting with the prefix *Pixelblaze*. Connect a Wi-Fi-enabled device to the access point network, then visit <http://192.168.4.1> in the device's browser to configure the Pixelblaze Wi-Fi settings with your local network name and password. Be sure to select the "Enable discovery service" checkbox.

Once configured, the Pixelblaze will reboot and connect to your network. To find it, visit the Cloud Discovery Service at [discover.electromage.com](http://discover.electromage.com), which displays all Pixelblaze boards on your network (Figure U).

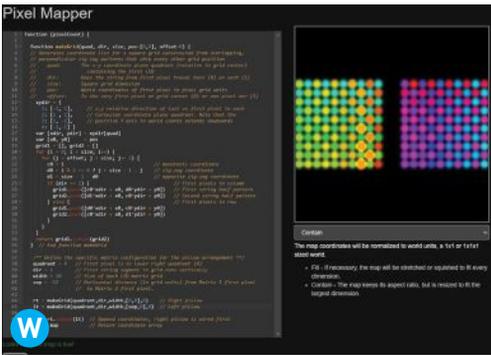
Selecting the board will open the Pixelblaze web app in your browser, where you can configure the LED setup and display patterns. First, select the Settings tab to configure your LED string characteristics (Figure V). Assign your board a descriptive name. Set the LED Type to WS2812 and the Data Speed to 800Kbps. If you're not sure of the Color Order, select GRB — if the colors in some patterns look wrong, you may have to experiment with this setting to find the correct option. Set the number of pixels to 200, for the two LED matrices. It's a good idea to limit LED brightness in the settings to reduce power consumption and heat output. Specifying a maximum 25% power is usually more than adequate to generate a bright display.



## 5. MAP THE LEDS

The Pixelblaze app stores LED configurations independently from the animation code, so the same pattern runs without modification on different layouts. To map the pattern to a specific pixel arrangement, you need to give the app the relative position of all the LEDs. We specify LED positions in the Mapper tab. Any coordinates entered there will be automatically scaled by the app to a range of **[0, 1]** for standardized use in code for different LED patterns.

The text box in the Mapper tab will accept a formatted list of Cartesian coordinates, but it can



also generate pixel locations from a JavaScript program (Figure **W**). Download the code file *PixelPillowMapperCode.txt* from the project page at [makezine.com/go/pixelblaze-pillows](http://makezine.com/go/pixelblaze-pillows), then paste its contents into the Mapper text box. This code specifies pixel locations for the two pillows using variable parameters to define their orientations and separation. Click the Save button to retain the new mapping. You'll know the map is correct when an affirmative message in green text appears below the mapper box and the image showing the LED positions resembles the physical layout of your pillows' LED matrices.

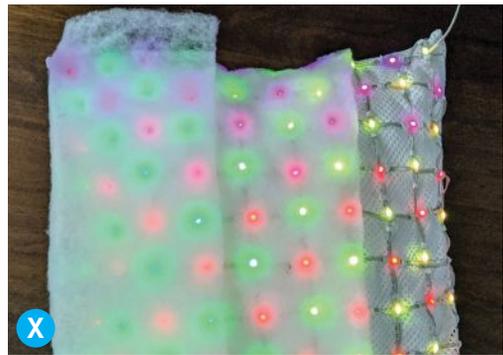
You may also want to experiment with the Fill and Contain options, which stretch the pattern to eliminate empty space at the periphery or keep empty space to maintain the pattern's aspect ratio, respectively. This choice is strictly aesthetic.

## 6. ATTACH DIFFUSER AND COVER

A 20"×20" square of quilt batting over the LEDs helps to diffuse emitted light and prevents twist tie wires from poking out. Figure **X** shows the difference in appearance of LED light shining through (from right to left) no diffusing material, a single layer of batting, and a double layer.

Place a 20" square of batting over each matrix, then insert the pillow and batting into the cover. Plump the pillow and tug the corners to ensure they fill the cover evenly. Smooth the batting if it has bunched up, then zip the covers closed, leaving just a small gap for the wiring to emerge.

**TIP:** The fuzzy white shag fabric is an amazingly good light diffuser as well, so not only is it a fun texture, it's a functional element.



Arrange your pixel pillows on a sofa or bed, tucking the wiring out of the way to hide it. If you change the pillow orientation and position, be sure to update the Pixelblaze Mapper.

## 7. PICK YOUR PATTERNS

This last step is by far most fun to execute. The Pixelblaze controller comes with several dozen animation patterns pre-installed. Switching between them is as simple as clicking the pattern name in the app's Patterns tab. The pillow display will update immediately.

You can view and modify the pattern code in the app's code editor. Select the Edit button next to a specific pattern in the Patterns tab to bring its code into the editor window. You can also access the pattern editor directly via the Edit tab and create new patterns there or upload them from pattern files in the *.epe* format.

Only a few lines of code are required to create your own original pattern, like this Bullseye pattern that sets the pixel colors based on their distance from an oscillating central point. The oscillations around the center are created easily in the code with the Pixelblaze editor (Figure **Y**).

When making or modifying patterns you can verify that your color settings and pixel map are correct by comparing the editor pattern preview window to the actual pillow display (Figure **Z**). If you're not comfortable with jumping into coding your own patterns, making small changes to existing code and watching the real-time display response is a great way to learn.

The Pixelblaze pattern editor has far too many powerful and useful features to describe here, but you can explore them in greater depth at

[electromage.com/docs](https://electromage.com/docs). There's an open source library of Pixelblaze patterns from various contributors at [electromage.com/patterns](https://electromage.com/patterns); those with a "2D" designation in their name will generally look the best on the pillow display. And programmer "Zranger1" has created a wonderful collection of intricate 2D patterns in his Pixelblaze repository ([github.com/zranger1/PixelblazePatterns](https://github.com/zranger1/PixelblazePatterns)) that are worth trying out.

## SPARKLY JOY

These pillows are a DIY project to display proudly. They morph from innocuous home accessories to something far more beautiful and interesting with the flick of a power switch. It's hard for still photos to convey the dynamic beauty of constantly changing LED animations, but these pictures are a small example of what is possible.

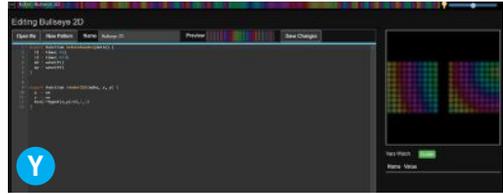
They're also extremely touchable and will stand up to a fair amount of abuse, because the weatherproof outdoor LED strings flex easily. I'll stop short of recommending them for a pillow fight, but they will easily survive the use that normal throw pillows get (as long as nobody yanks hard on the wiring). And the covers unzip and can be removed for washing, so kids can absolutely touch them, hug them and enjoy them.

## NEXT STEPS

If you're enjoying your LED matrix pillows, here are a few ideas to take this project further:

- Use the Pixelblaze app's Playlist feature to auto-play your favorite patterns
- Add a Pixelblaze sensor board for sound or motion reactivity (see page 74)
- Create additional pillows to connect in series for an even larger decorative display
- Sew a matching LED throw blanket
- Switch to battery power. I ran the pillows continuously at 25% brightness and the power draw stayed below 2 amps the entire time, so any portable phone charger capable of 2A (many of the larger capacity ones can do that) will work fine for portability.

I've had more fun than usual creating and tinkering with this LED build and I hope others will enjoy its accessible construction and versatile display. It is home décor that is literally brilliant. 🌟



### More Resources:

- Quick Start guide for Pixelblaze V3: [electromage.com/quickstart-v3-standard](https://electromage.com/quickstart-v3-standard)
- The ElectroMage Forums are a great place to find inspiration and get questions answered: [forum.electromage.com](https://forum.electromage.com)



# Waxed Canvas Tool Roll

Stitch a stylish and sturdy roll-up tool carrier that will last for many tough jobs

Written and photographed by Becky Stern



**BECKY STERN** has authored hundreds of DIY tutorials about everything from microcontrollers to knitting. She is an independent content creator and STEM influencer living in New York City. Previously she worked as product manager at Instructables (Autodesk), director of wearable electronics at Adafruit, and senior video producer for *Make*:. She enjoys riding on two wheels, making YouTube videos, and collecting new hobbies to share with you. [beckystern.com](http://beckystern.com)

**TIME REQUIRED:** An Afternoon

**DIFFICULTY:** Easy

**COST:** \$25–\$30

## MATERIALS

- » **100% cotton canvas, 1 yard** such as Robert Kaufman “Big Sur” 9oz canvas, Amazon B07FCQT6J7
- » **Matching sewing thread**
- » **Metal zipper, 10" (25cm)** such as Amazon B07JQ8VVJR
- » **Fabric wax** such as Fjällräven Greenland Wax, Amazon B000M5GBNS; or blend your own, a 1:1 ratio of paraffin to beeswax

## TOOLS

- » **Sewing machine**
- » **Sewing pins and/or clips**
- » **Scissors**
- » **Thread snips**
- » **Heat gun or hair dryer** or an iron you don't mind using on crafts
- » **Scrap cardboard** for protecting your work surface from wax
- » **Printer and paper** for pattern; or paper to draft your own
- » **Tape**
- » **Double boiler and wax pitcher (optional)** such as Amazon B00J2BGZPM, for blending wax
- » **Paintbrush (optional)** for wax application

**This tool roll is an easy and useful sewing project that can be whipped up in an afternoon.**

It's a fun little portable kit that's perfect for my motorcycle tools, but also great for drawing, electronics, or anything where you need to bring a variety of different-sized hand tools.

I provide a pattern that you can easily modify to fit your needs. I'll show you how to customize it to hold your specific tools, no matter what you make. And then I'll take you through the waxing process to add durability and water resistance.

The waxing is an optional step that stiffens up the canvas and makes it more durable and resilient. This is a satisfying process where the melted wax soaks into the fabric with the application of heat. The cool thing about waxed canvas is that it only gets better with age.

When I was a kid, bags were the first thing I learned to sew on a machine. They're easy because the shapes are pretty geometric and ultimately success is defined by utility — either it holds stuff or it doesn't — unlike sewing clothes, which then have to fit and be flattering.

Look for 100% cotton canvas in the colors of your choice. Pick up some matching thread and a zipper — metal is preferred, because we're going to heat this thing up.

## SEW YOUR WAXED CANVAS TOOL ROLL

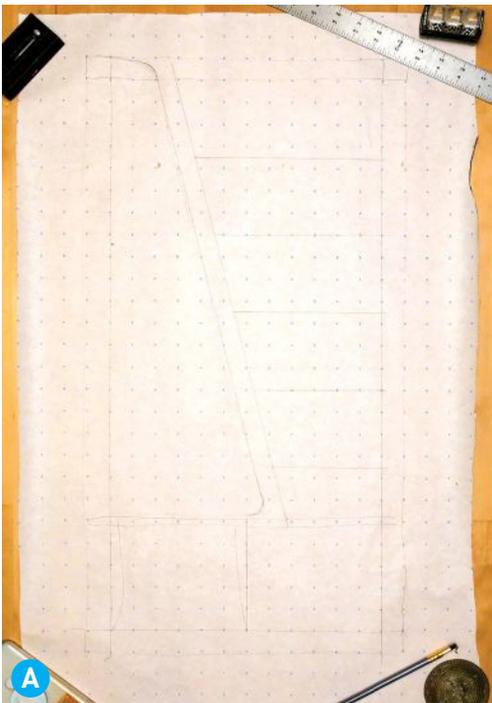
Before you start cutting and sewing, check out my project video at [youtu.be/TdPUkeLv9Ac](https://youtu.be/TdPUkeLv9Ac) to see how the whole thing goes together.

### 1. PRINT THE PATTERN

You can download and print my pattern from [beckystern.com/2022/04/03/waxed-canvas-tool-roll](https://beckystern.com/2022/04/03/waxed-canvas-tool-roll), or draft your own. It's got one big rectangle for the main body, two pieces for the zipper pocket, two pieces for the tool holder area, and a strap to tie it up.

I designed the pattern to be easily printable on seven sheets of Letter sized paper. Assemble it by lining up the registration marks and taping the sheets together (Figure A). I've included the printable PDF along with the Illustrator file in case you want to make changes (CC BY-SA-NC).

To draft the pattern, I laid out the tools I wanted to hold, and loosely traced around them on a large





piece of paper. Then I photographed my paper pattern and traced over it in Adobe Illustrator.

## 2. CUT FABRIC PATTERN PIECES

Iron the fabric and cut out the pattern pieces (Figure B). I made two of these tool rolls at once, so I cut the pieces out of two colors and mixed and matched the pieces. All the pieces fit on a folded length of 45"-wide fabric less than one yard long, but half a yard is too little. Take care about right sides and wrong sides if your fabric has them (my canvas is the same on both sides). You should have the following pieces (Figure C):

- Main body rectangle
- Two trapezoidal pocket/flaps
- Two rectangular pieces for the zipper pocket (one skinny, one wide)
- One long piece for the strap

## 3. IRON AND TOPSTITCH THE INNER LAYERS

Next up there's a little prep ironing for some of the raw edges. The top flap gets a double-folded edge since we'll see both sides of it, while the tool holder piece only needs one fold at the top and the short end. The strap also gets ironed with edges to the center and then in half once more.

On the tool flap, iron the three loose edges over twice and then topstitch the foldover to create a nice edge.

On the tool pocket, iron the top and left edges over once and topstitch. These edges will not be captured by a future seam, but we won't see the other side of them either.

Also topstitch along the length of the tie strap (Figure D).

## 4. INSTALL THE ZIPPER

Install the metal zipper (Figure E) between the two rectangular pocket pieces, sewing right sides together before ironing the seams open and topstitching along the zipper to finish it up. I use a special zipper foot on my sewing machine that can stitch up close to the zipper.

**TIP:** If you've never sewed a zipper before, refer to Mikaela Holmes's excellent guide at [instructables.com/Sewing-Zippers-and-Buttons](https://www.instructables.com/Sewing-Zippers-and-Buttons).

## 5. FINAL ASSEMBLY

Now all the parts are ready to be assembled. First, attach the front to the back. Pin/clip the pieces together, right sides facing, and stitch around the whole perimeter (Figure F).

Then turn everything right side out, iron, and topstitch around the whole thing, including across the top of the zip pocket (Figure G). You don't have to topstitch around the outer edges of the zip pocket if you don't want to (this does reduce the pocket capacity slightly).

Next, stitch the short edge of the tool pockets (Figure H). The distribution of the tool dividers is completely up to you — you can customize the pocket widths to accommodate your specific tool needs.

Lastly, stitch the strap in place on the outside wherever you like, depending on which way you prefer to roll it up (Figure I). I sewed the midpoint of the strap to align with the top seam of the zip pocket.

## 6. WAX ON!

Adding wax to the fabric makes it water-resistant and more durable. It's an optional step depending on how you want to use your tool roll. For motorcyclists, it's super important to keep your tools dry. For drawing, maybe you don't need the wax to protect your pencils.

There are two ways to apply fabric wax. Either way, use a scrap piece of cardboard to protect your work surface from the wax. If you buy the bar type, you can just rub it on and melt it in with a heat gun (Figures J and K). This method is a little slow and provides an arm workout. You build up the desired amount of wax with multiple applications.

The other way is to melt the wax in a double





boiler and paint it on before melting it in with a heat gun (Figures **L** and **M**). This method is faster, but provides less control over the amount of wax that is applied — you'll get a full soak on the first application. This method also allows you to make your own blend of wax. I made a mix of paraffin and beeswax at a 1:1 ratio.

Applying heat to the waxed fabric helps it soak in and become evenly distributed. It's so satisfying to watch!

Once waxed, the tool roll is stiffer, darker in

color, and has a water-resistant finish (Figures **N** and **O**). I added a few patches to my tool roll before the waxing, since it's easier to sew the unwaxed fabric (Figure **P**).

## USE IT!

Load up your new tool roll and get out there and make something!

This project also makes an excellent gift for any makers in your life — it's fun to make a bunch of these (Figure **Q**) in whatever colors you like. 🍷

# 1+2+3 LED Pendant

Written and photographed by Charlyn Gonda



**TIME REQUIRED:** 1 Hour + Design Time

**DIFFICULTY:** Easy/Moderate

## MATERIALS:

LED sequins (3), brass discs (2), thin brass rod, CR2032 battery holder, double-sided foam tape, solder, flux, small paintbrush, soldering iron, flush cutters, transparent double-sided tape. **Optional:** circle bending pliers, kapton tape



**CHARLYN GONDA** is a coder by daylight, maker by moonlight, from San Francisco. She loves to create delightful (often glowy) things.

**Wearable glowing accessories — we often associate them with Halloween costumes or a packed concert.** While those are definitely fun, I wanted to make elegant jewelry, fit to wear on a night out with friends or a casual day at work. There's something magical about LEDs, and I want to carry that magic with me wherever I go.

To keep it sleek, I designed this pendant as a *circuit sculpture* — it uses thin metal rods both to conduct electricity and to provide structure and decorative shape. I started with a thick brass disc to conceal the battery holder, and took a lot of inspiration from art deco motifs.

## 1 DRAW THE DESIGN ON PAPER

Measure the LED sequins and brass discs, and build a simple line-based design around these shapes. Draw or print your final design in a horizontally mirrored orientation so that the solder joints will end up behind the pendant. Make sure the printed size is correct.



## 2 BEND AND CUT THE RODS

Place transparent double-sided tape over the design to secure the parts while soldering (credit to Jiri Praus at Remoticon 2020!). Using a flush cutter, cut and bend the rods to your design. Try jeweler's circle-bending pliers for clean bends, and Kapton tape to stabilize small parts.

## 3 SOLDER AND ASSEMBLE

Dab a little water-based flux on each joint using a paintbrush. Touch the brass rod with the soldering iron to get it up to temp (might take a second longer than regular wires), and melt solder onto it. The solder should flow in between the rods and solidify soon after removing the iron.

Finally, use double-sided foam tape to secure the brass disc to the battery holder, and attach to a long necklace chain using some jumper rings.

This pendant-making method is versatile and can suit a variety of personal styles. I'm excited to see more glowy jewelry like this out in the world. 🌟

**NOTE:** The frame of the pendant is the circuit itself. This means that a portion of it needs to connect to the negative side of the battery and a portion of it to the positive — and the two should be bridged by the LEDs.



# Power Through Thin Air

Written by Fredrik Jansson with Charles Platt

**Radio is still a mysterious phenomenon. Learn how to transmit and receive audio signals on an AM radio carrier wave**



**FREDRIK JANSSON** is a researcher in physics and weather modeling and coauthor of the *Encyclopedia of Electronic Components Volumes 2 and 3*.



**CHARLES PLATT** is the author of the bestselling *Make: Electronics*, its sequel *Make: More Electronics*, the *Encyclopedia of Electronic Components Volumes 1–3*, *Make: Tools*, and *Make: Easy Electronics*. [makershed.com/platt](http://makershed.com/platt)

When radio was first invented in the late 1800s, people thought of it as a mysterious, magical phenomenon — and in some ways, it still is.

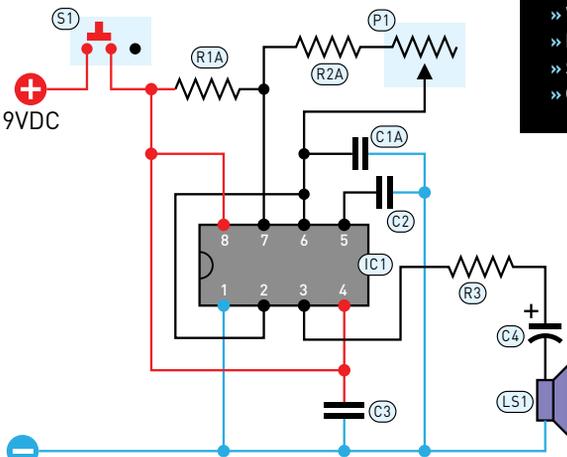
We tend to imagine electric current as a stream of electrons flowing through a wire, yet a radio signal can pass through the air, without any electrons to carry it. Our first experiment, from our book in progress, *Make: Radio*, will enable you to see this for yourself, as you create a tiny AM radio transmitter on your table or workbench.

## ALL ABOUT AUDIO

You may be surprised to see that we're using that most ancient chip, a 555 timer (actually the 7555 version, because the 555 creates voltage spikes which would be a problem in this project). We chose the timer for two reasons: (1) It's such a common, low-priced item, you may already have a few, and (2) it's the easiest way we could think of to generate radio waves. We're willing to bet you never imagined using it for this highly unusual purpose!

The purpose of many radio transmissions is to transmit audio, so we'll start with an audio oscillator circuit that you may have seen before. The schematic diagram is shown in Figure A, and the breadboard layout is shown in Figure B on the following page.

We turned the breadboard on its side because we wanted it to match the schematic. In audio and radio circuits, most people show positive power at the top and the signal passing through



A A basic circuit to generate audio frequencies.

**TIME REQUIRED:** 1-2 Hours

**DIFFICULTY:** Easy

**COST:** \$15-\$50

## MATERIALS

No kit exists yet, for this work in progress, but you should find the parts easily on eBay or Amazon, or at the big online electronics suppliers.

**For the transmitter module:**

- » 7555 timer integrated circuit (IC) chips (2)
- » 22-gauge jumper wire (7 feet) for breadboard and optional receiver
- » Electrolytic capacitor, 100µF
- » Ceramic capacitors: 100pF (1), 10nF (3), and 0.1µF (1)
- » Resistors: 100Ω (1), 330Ω (1), 2.2kΩ (2), and 10kΩ (2)
- » Speaker, 8Ω impedance, 2" diameter or larger lowest cost available
- » Trimmer potentiometer, 500kΩ
- » SPDT slide switch, 0.1" pin spacing

**For the optional receiver module:**

You can use a cheap handheld AM radio, or build your own receiver with just six components:

- » Schottky diode, BAT48, or germanium diode
- » Ferrite rod, 1/2" diameter and 5" long, or larger
- » Tuning capacitor, 200pF also known as a variable capacitor. Use model 223P (223F looks similar but won't work). If it has a plastic tuning wheel, it will be easier to use. See Figures L and M on page 68.
- » Resistor, 10kΩ
- » Earphone, high impedance often sold "for crystal radios." Quality of these earphones is hit-and-miss, but you can obtain reliable ones, in our experience, from [www.protechtrader.com](http://www.protechtrader.com).
- » Terminal block (optional) to join 6 wire pairs

## TOOLS

- » Solderless breadboard
- » Wire cutters/strippers
- » Pliers
- » Small screwdriver
- » Oscilloscope (optional) but useful

Components	
S1	Slide switch, SPDT
R1A	Timing resistor, 10K
R2A	Timing resistor, 10K
R3	Current limiting, 100 ohms
C1A	Timing capacitor, 10nF
C2	Bypass capacitor, 10nF
C3	Bypass capacitor, 0.1µF
C4	DC blocking, 100µF
IC1	7555 timer chip
LS1	8-ohm speaker, 2" minimum

from left to right.

The specification for 7555 chips varies slightly from one manufacturer to another, but so far as we know, all chips of this type will work in the circuits in this experiment.

In Figure B, the square blue object with a white circle on it is a 500K trimmer potentiometer — that is, a variable resistor whose resistance you can adjust from zero to 500,000 ohms by turning the screw built into it. The pink dots show the locations of pins under the trimmer.

**TIP:** Some trimmer potentiometers are not well designed for use with breadboards. Their little pins have kinks in them, which may be difficult to insert fully, and the trimmer may tend to rise up out of the board. You can overcome this tendency by using pliers to straighten or flatten the pins.

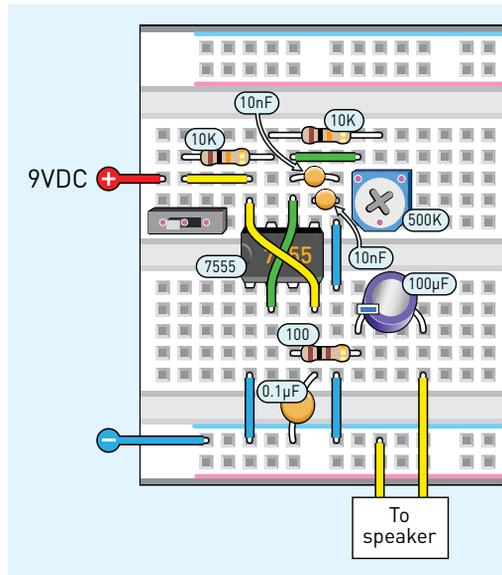
Because we'll be referring to some of the pins on the 7555 timer by name, we have summarized them in Figure C.

As soon as you apply 9VDC to the circuit and switch it on, the timer emits a stream of pulses from its Output pin. In the circuit, the pulses pass through a 100-ohm resistor (to limit the current) and a 100 $\mu$ F electrolytic capacitor (which blocks DC) on their way to a speaker. The duration of each positive pulse is determined by three resistances: R1A, R2A, and P1. The gaps between pulses are determined by R2A and P1. The size of capacitor C1A also determines the duration of the pulses and the gaps between them. Higher-value resistances and/or higher capacitance will generate longer pulses and gaps, so the potentiometer will adjust the pulse stream.

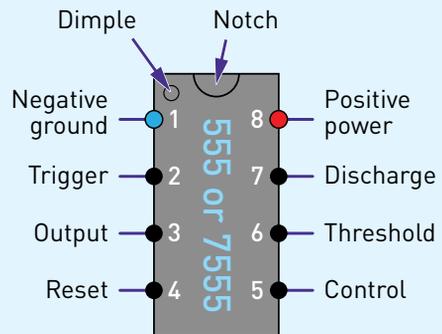
**TIP:** If you want to know how to calculate the results of resistance and capacitor values when using a 555 or 7555 chip, search the web for "555 timer calculator." Many calculators are available online, and the values for a 555 timer will create exactly the same frequencies as a 7555 timer. The 7555 just does the job better.

## FREQUENCY AND WAVELENGTH

Figure D shows the actual output from the 7555 chip, displayed on the screen of an oscilloscope. You don't have to own an oscilloscope to build and test the circuits here, but it will be extremely useful, and some oscilloscopes are now almost



**B** The breadboarded version of the circuit shown in the schematic, Figure A.



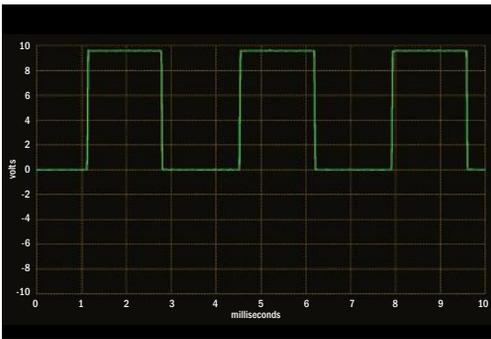
**C** The names of pins on a 555 or 7555 timer chip.

as cheap as multimeters.

Even though the output in Figure D consists of straight lines, it's known as a *waveform*. (This one happens to be a square wave.)

The scale on the left in Figure D shows that the output from the 7555 ranges from almost exactly 0V to 9V, when you are using a 9VDC power supply. This voltage will be pulled down somewhat when you add a load to the Output pin of the chip, such as a speaker. The oscilloscope trace shown here was measured without a load.

The *period* of the output is a measurement of time, from the start (or end) of one pulse to the



**D** A square-wave output from a 7555 timer running in a stable mode.

start (or end) of the next pulse. In other words, the period is the duration of the pulse plus the duration of the gap between it and the next pulse. In the example shown here, the period is measured in milliseconds.

When the signal from the 7555 passes through the speaker, the cone of the speaker (also known as its diaphragm) vibrates and converts each pulse into a pressure wave in the air. When a membrane in your ear responds to these pulses, it enables your brain to experience them as sound.

The *frequency* of a signal is the number of pulses per second. This is expressed in *hertz*, named after the electrical pioneer Gustav Ludwig Hertz, and is abbreviated Hz (the H is capitalized, because it refers to his name). A frequency of 100 pulses per second is written as 100Hz, while 1,000 pulses per second would be 1 kilohertz, written as 1kHz, and 1,000,000 pulses per second are 1 megahertz, written as 1MHz. The capital M means *mega*. A lowercase m would mean *milli*, so you must avoid getting them mixed up.

Human hearing can resolve sounds ranging from around 20Hz to a maximum of 20kHz, although elderly people may have difficulty hearing sounds higher than 10kHz, and people who have damaged their hearing as a result of environmental noise (or rock concerts) may have a limit as low as 5kHz.

A soundwave, consisting of alternating regions with high pressure and low pressure, travels through the air at approximately 1,125 feet per second, at sea level. (At higher altitudes, sound travels more slowly.) The distance from the start of a high-pressure region to the start of the next

high-pressure region is called the *wavelength*.

If  $f$  is the frequency of a sound, measured in Hz, and  $p$  is the period in seconds, and  $w$  is the wavelength in feet, and  $s$  is the speed in feet per second, these values are related by two simple formulas, where the asterisk is a multiplication sign:

$$f = 1 / p$$

$$s = w * f$$

You can see from the scale at the bottom of Figure D that the period of this soundwave is about 3.5 milliseconds, or 0.0035 seconds.

Using the first formula,  $1 / 0.0035 = 286\text{Hz}$  (approximately). That's the frequency of this sound.

If you rewrite the second formula as  $w = s / f$ , you can see that the wavelength of the sound is  $1,125 / 286 =$  about 4 feet.

## THE CARRIER WAVE

In theory, you could disconnect the speaker from your circuit and substitute a piece of wire, which would function as a transmitting antenna. This would radiate a small amount of power, which you would be able to pick up by using another circuit functioning as a receiver.

In practice, higher frequencies are better able to transmit more power over longer distances. In any case, if all radio stations in the world transmitted audio frequencies, we would have no way to separate them and listen to just one.

The answer to both problems is to add the audio frequency to a much higher frequency, known as a *carrier wave*, and assign a different frequency of carrier wave to each radio station in its area. If you look at the AM tuning dial on a radio, you'll see numbers ranging from 540kHz to 1,600kHz, which are carrier frequencies.

To add your audio soundwave to a carrier wave, all you have to do in this experiment is connect a second 7555 timer to the circuit that you just built. A 7555 can run at up to 2MHz (which is the same as 2,000kHz), so it is quite capable of transmitting a signal in the AM waveband, even though it is never normally used for that purpose.

First remove the speaker from the circuit that you just built. You can also remove resistor R3

# PROJECTS AM Radio Transmitter

and capacitor C4 if you wish. Then add the extra 7555 timer, as shown in Figures E and F. The components that you placed on the breadboard previously are still there, but they are grayed out.

The new component values in this circuit will generate a frequency around 800kHz. The circle of yellow wire in Figure E, connected through a 330-ohm resistor with the output from IC2, is a loop about 2" in diameter, which will function as your transmitting antenna. The 330-ohm resistor prevents IC2 from being overloaded.

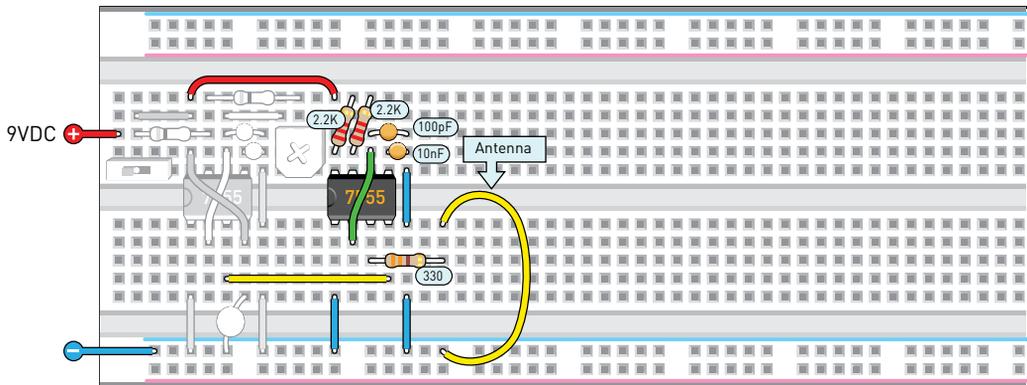
Here's how the circuit works. Pin 4 of a 555 or 7555 timer is the Reset pin, which puts the timer on hold when the pin voltage is near negative ground, but allows the timer to run when you apply a voltage near the power supply. Notice the horizontal yellow wire in Figure E which connects the Output pin of IC1 with the Reset pin of IC2.

This means that the output of IC1 switches IC2 on and off. IC2 doesn't mind being switched on and off quickly, even at audio frequencies as high as several kilohertz.

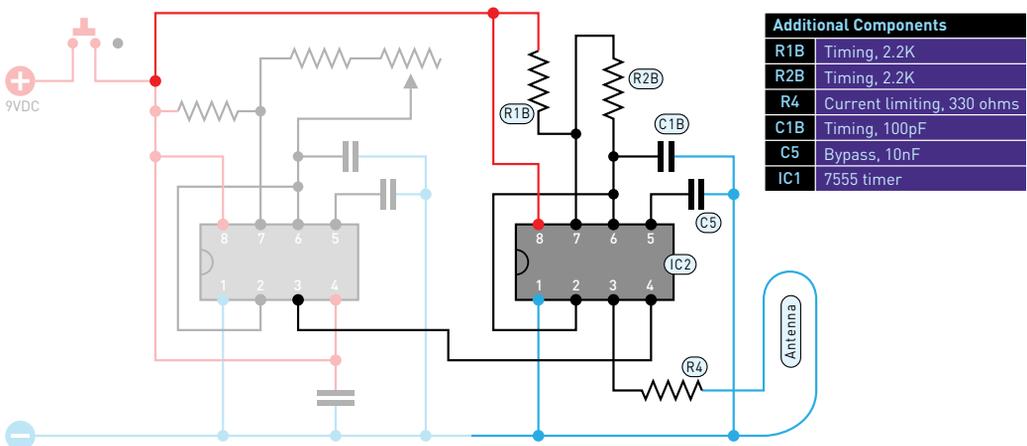
Notice that C1B, which you can see in Figure F, is 100pF. That's picofarads, not nanofarads. Be careful not to get your p's and n's mixed up.

When you apply power to the circuit, IC2 generates a carrier wave of around 800kHz, which is switched on and off by IC1. In Figure G, an oscilloscope was attached to the output from IC2 so that you can actually see how this happens. (In reality, there would be as many as 1,000 oscillations inside each burst, rather than the dozen or so shown in Figure G. The oscilloscope downsamples them so that you can get an idea of what is going on.)

But what if you don't have an oscilloscope?



**E** Adding a second timer that runs at a radio frequency.



**F** The schematic version of the breadboard circuit in Figure E.

How will you know that IC2 is actually doing anything?

If you have an AM radio, try tuning it to the middle of its range, and hold your radio close to the yellow loop antenna on the breadboard. If you adjust the radio carefully, and turn it from side to side, you should be able to hear the audio frequency.

Congratulations! You just demonstrated a radio transmitter — and sent power through thin air.

## LOCAL RECEPTION

Wouldn't it be even more interesting to build a receiver of your own? This is easily done.

Two main types of transmission are used by radio stations today: *frequency modulated* (abbreviated as FM) and *amplitude modulated* (AM). The earliest radio transmissions were AM, meaning that the loudness of a sound coming out of the radio was proportional with the *amplitude*, or voltage, of the signal.

In our book *Make: Electronics*, we included a radio in Experiment 28, titled “One Radio, No Solder, No Power.” You could use this radio to pick up your audio tone from your 555 transmitter circuit, but building it is a bit of a chore, as you need to wind many turns of wire around a large object such as an empty multivitamin bottle. We now suggest a better option, if you don't mind acquiring a ferrite rod, which you should be able to find for a very modest cost.

Ferrite conveniently intensifies the effect of a coil. You can use everyday 22-gauge hookup wire to wind 63 turns around the rod. To make them fit, you may need to wind the turns in two layers, one on top of the other, as in Figure H. Make sure that all the turns go in the same direction, and tape them in place. We show a file card curled around the rod, inside the coil, so that you can slide the coil up and down the rod to adjust reception. For your basic test, though, this isn't necessary.

Figure I shows the schematic for your ultra-simple radio, while Figure J suggests how you can twist the wires together so you don't need a breadboard. The green wires go to the coil, while the black wires go to the earphone. The diode is the blue component with a black end.

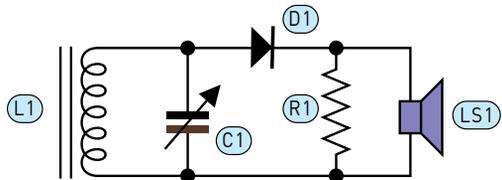
Figure K on the following page shows how



G Output from the second timer. The oscilloscope has downsampled the high frequencies to make the oscillations individually visible.



H A coil of 63 turns of 22-gauge wire, in two layers on a ferrite rod.



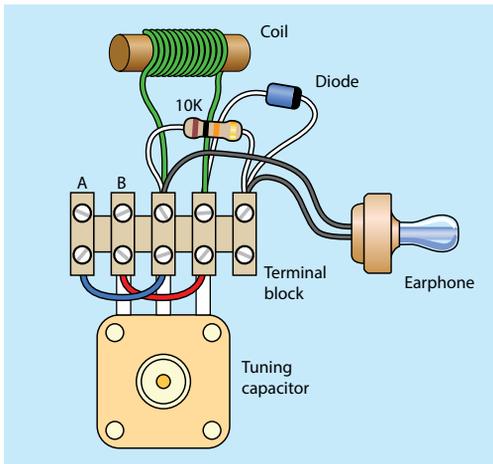
### Components

L1	Coil, 63 turns on ferrite rod
C1	Tuning capacitor, 200pF
D1	BAT48 Schottky diode or similar
R1	10K resistor
LS1	High-impedance earphone

I The schematic for an ultra-simple AM radio receiver.



J Connecting the coil (green wires), earphone (black wires), resistor, and diode.



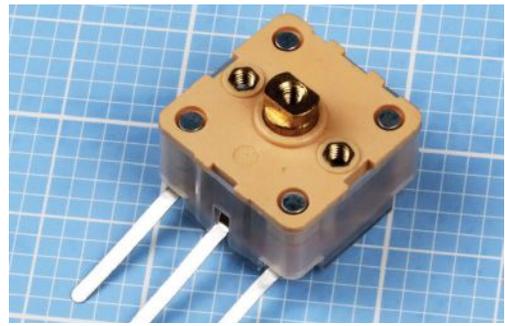
**K** All the components of your minimal AM radio receiver. Terminals A and B are available for ground wire and antenna.

everything can be connected through a terminal block, but if you don't have one, you can use alligator clips to link the upper wires with the variable capacitor at the bottom. The variable capacitor (Figures **L** and **M**) actually contains a pair of capacitors, which is why it has three tabs linked as shown, connecting them in parallel.

Terminals A and B can be connected with ground and an external antenna if you want to pick up real radio stations. But for your test circuit, this won't be necessary.

Place your radio circuit near the loop antenna on your breadboard circuit, adjust the variable capacitor, and you should easily hear the tone from the 7555 oscillator.

But what is your radio actually doing, here? Flip back to Figure G, which shows the output from



**L** Tuning capacitor, also known as a variable capacitor. Graph squares are tenths of an inch.



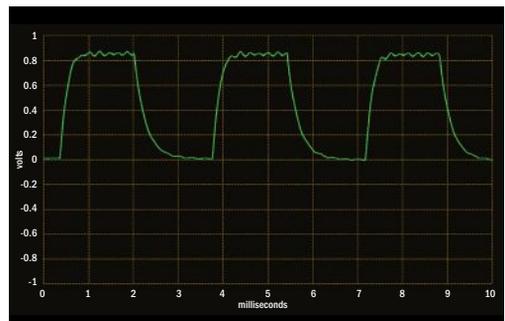
**M** Tuning capacitors are often sold with an additional wheel.

IC2. This is the signal that goes into the loop of wire, which is your transmission antenna. The oscilloscope trace shows that the output of IC2 varied between 0V and approximately 7.5VDC.

Now take a look at Figure **N**. This trace was measured across the two wires from the receiving coil in your simple receiver. Somehow the coil picks up the transmission and turns it into a voltage that varies between  $-0.35\text{V}$  and  $+0.35\text{V}$ .



**N** Voltage across the receiving coil in the AM radio receiver.



**O** Voltage across the earphone in the receiver, after the signal passes through the diode to block negative fluctuations. The audio signal "rides the peaks" of the carrier frequency.

Why is this? Because the coil is not connected in any way with the transmitter circuit. It cannot “see” the 9V supply and 0V negative ground on the breadboard. All it knows is that the high frequency fluctuates up and down, interspersed with intervals of no voltage.

What would you hear if the diode wasn't in the circuit? Nothing! You can test this yourself by shorting out the diode with a piece of wire.

The diaphragm in the earphone cannot vibrate at the carrier frequency because the oscillations are much too fast — and even if the earphone could reproduce them, they are too fast for your ear to resolve them as sound. Moreover, the oscillations are equally positive and negative, so they add up to an average of zero.

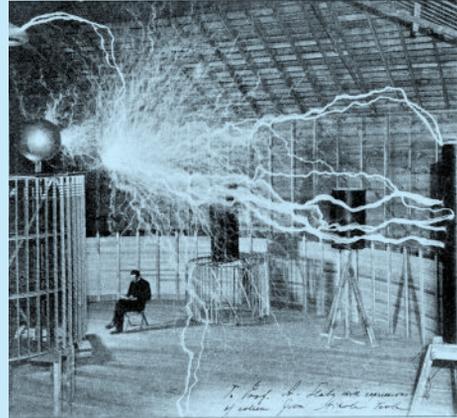
But when you pass the output from the coil through a diode, the negative half of each oscillation is blocked, and only the positive part flows through. This is known as *rectifying* the signal.

You can think of all of the tiny, rapid, positive pulses of the carrier frequency pushing the diaphragm of the earphone in one direction. Then there's a pause (lasting a fraction of a second), and the diaphragm relaxes. This happens at the audio frequency set by IC1. You can see this in Figure 0, which is an oscilloscope trace taken from the wires to the earphone. The earphone — the audio signal — “rides the peaks” of each burst of carrier frequency. 🔍



This article is adapted from our book in progress, *Make: Radio*, forthcoming in 2023 and available for pre-order now at the Maker Shed, [makershed.com](https://makershed.com).

## A WORD ABOUT TESLA



It's easy to imagine how Nikola Tesla made an intuitive leap and wanted to crank up a radio transmission so that it would broadcast serious electrical power. His most ambitious installation, in Colorado Springs, used a coil 100 feet in diameter which created a potential of up to 12 million volts. It broadcast sufficient power to light a fluorescent tube equipped with an antenna, 25 miles away. It also drove horses crazy by inducing electric shocks in the iron shoes on their hooves.

This was a time when many aspects of electricity and physics were not fully understood, and there were competing theories. Unfortunately, Tesla pinned his hopes on some theories that turned out to be wrong. He believed empty space wasn't really empty, but was filled with an invisible, mysterious substance known as “ether.” He hoped, incorrectly, that if he tapped into the ether at the right frequency, infinite power was available. This turned out not to be the case.

If he had devoted less time and effort to his dream of infinite electrical power, probably he could have won the race to transmit a radio message across the Atlantic. Marconi ended up with that achievement, but Tesla had a more powerful transmitter, a larger antenna, and a more sophisticated receiver. He just didn't think that radio messages were as interesting as free electricity.



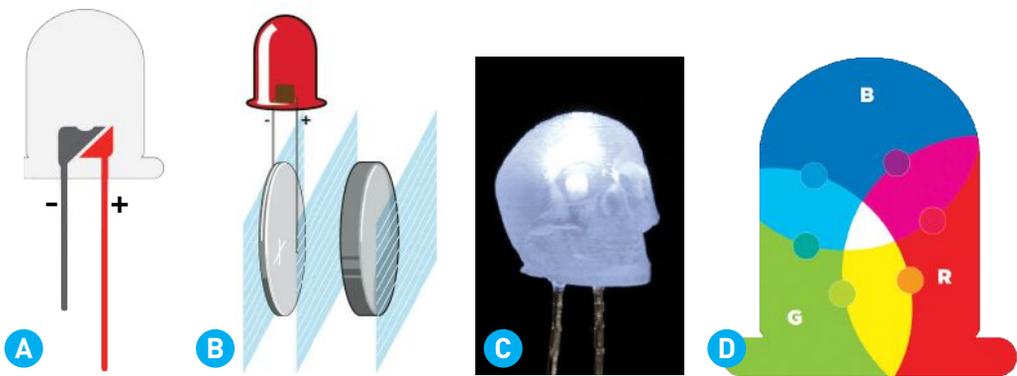
# LED Love!

Written by Lee Wilkins

All about LEDs,  
and how to  
create amazing  
light patterns  
with addressable  
LEDs and  
controllers



**LEE WILKINS** is an artist, cyborg, technologist, and author of our new "Squishy Tech" column in *Make*: looking at technology and the body and how they intertwine. Follow them on Twitter @leeborg\_



**LEDs make everything better, but sometimes it's hard to get all the information you need to make the super-cool light-up costume of your dreams. Today, we'll go from basics to pixel mapping!** One of my favorite things about LEDs is that they're a gateway to creating all kinds of fashion and accessories, like the responsive LED umbrella tutorial I'll share here.

## THE BASICS

There are countless shapes of LEDs, but we can start with the two types of packaging you'll encounter: *through-hole technology (THT)* and *surface mount technology (SMT)*. These each come in multiple sizes, shapes, and form factors. You're probably most familiar with a through-hole LED (Figure A), which (usually) has two legs, and LED strips, which are how we most often see SMT components, soldered flat onto them.

All LEDs are *polarized* components, meaning that they have a positive and negative lead. By applying a specific voltage to the positive leg, and connecting the negative leg to ground, we can turn on the LED! No matter the shape or size or color, at their core, they're all working like this. I'll go over a few configurations.

Often people get started with LEDs by hooking up a 3mm through-hole LED to a CR3032 coin cell battery (Figure B), making a "throwie" ([makezine.com/projects/Extreme-LED-Throwies](http://makezine.com/projects/Extreme-LED-Throwies)). They're so much fun — add a coin-shaped magnet to make them stick to stuff when you throw them! In a basic through-hole LED, the longer leg is usually positive and the shorter one is negative. I like to remember this by reminding myself that I want "more positivity" (or less negativity?) — because keeping track of the polarity is very important.

If you've already twisted, trimmed, or soldered the legs of your LED, sometimes through-hole

LEDs are big enough to look inside and see that each leg is connected to a piece of metal, and one piece is hanging over the other. The one hanging over is the negative leg — because "hangovers are negative." Easy to remember!

Now, all through-hole LEDs aren't created equal. Every green LED you buy won't give the same color of green light, because of a variety of factors including the materials used to produce the LED, or the shape of the epoxy lens. Some have tinted epoxy casings, some are 3mm, 5mm, or even 10mm in diameter, and some cast different shadows or have different light qualities. I find the LEDs with clear epoxy to be the brightest; they're often labeled "super bright." You can find LEDs in a variety of shapes — dome, square, cylindrical — which can be useful depending what you're doing, but some of my favorites are the **dLUX-dLITE LEDs by Unexpected Labs**, which come in all kinds of cool shapes from skulls to spikes to spheres (Figure C).

You won't find LEDs in every single color, but thanks to *additive color theory*, you can mix two or more LEDs to get the color you want. Red and green make yellow, green and blue make cyan, blue and red make magenta. By mixing different amounts of these, you can achieve the full color spectrum.

**RGB LEDs** are full-color LEDs which have four legs, one each for red, green, blue, and ground. These are basically three small LEDs in a single package, and you can make any color by mixing different amounts of each color (Figure D).

## IDENTIFYING LEDs

If you want to add more than a few LEDs, you'll need to look into other form factors, such as panels and most commonly LED strips. This is where things get a bit tricky, and some vocabulary

## PROJECTS Squishy Tech

can help you get the right LEDs for your project. I'll refer to these mostly as strips, but you can get all kinds of panels, flexible matrixes, string lights, rings, or other creative form factors.

**Non-addressable LEDs** come in a few types. **Single color** LED strips are as they sound, a single color. The whole strip can be on, off, or fading, but you can't control each individual light. These are great for simple environmental lighting or just to add light quickly to a project. They usually have 2 wires, red and black, as well as a voltage rating.

If a strip is described simply as **RGB**, it will usually have 4 wires, one each for red, green, blue, and ground, and you can make the entire strip any color. **RGBW** strips have an additional fourth channel for pure white LEDs and a fifth wire to control it.

Speaking of white LEDs, you'll find that there is a range of warm and cool white. This **color temperature** is measured in degrees Kelvin, and goes from around 2700K (warm) to 6000K (cool) with 4000K being considered roughly neutral.

**Addressable LEDs** are the type where you can control each LED individually, as opposed to the whole strip (or matrix) behaving the same way. Using addressable LEDs, you can make patterns, images, and all kinds of complex effects that aren't possible with plain RGB LEDs.

You may have heard these commonly referred to as **NeoPixels** (Figure E), which is a brand name in the Adafruit ecosystem. Not all addressables are NeoPixels, and understanding the part numbers can help you buy LEDs from various suppliers. NeoPixels are WS2812 LEDs (and their clone, SK6812) or the older version with an external IC, WS2811. The benefit of WS2812 is that if there's a broken LED, the sequence continues after it instead of breaking the rest of the line, unlike the WS2811. There is also WS2813, which has Data In and Data Out pins broken out, instead of a single data line.

APA102 or SK9822 are called **DotStar** by Adafruit. They're much faster, and require two pins to control (data and clock). DotStars are great for things that require speed, like POV displays or fast animations.

LED strips also have dust/waterproofing



**ingress ratings** so you'll know if you can use them outside. A strip rated IP30 has no waterproofing; IP65 has a rainproof silicone covering. Strips are also often described by **density** — how many LEDs there are per meter.

The individual LEDs within a panel or strip also come in various **package sizes**. The most common, and brightest, is the 5050 packaging, which is 5mm×5mm square. Plain RGB, RGBW, addressable, and many other kinds of LEDs come in 5050 packaging. Other common sizes are 3528 and 5630, which are generally not as bright (or big), and 3535, which Adafruit calls NeoPixel Mini. Figure F shows a 5050 (left) and 3535 (right).

Phew, that's a mouthful but thankfully this article is always here for reference!

## CONTROLLING LEDs

You can control addressable LEDs in a variety of ways depending on how customized you want the pattern. Some LEDs come **pre-programmed** with patterns or sequences that you can't change; this will usually be indicated as "auto flashing" or similar. But it's more fun to use controllers to change up the patterns and sequences.

**Generic controllers** often come with LEDs you can buy off the shelf. Usually these have an IR remote that can set simple patterns or preset colors, although some have buttons that cycle through pre-set patterns, like your holiday lights!

To get a bit more advanced, you can program your own patterns using an **Arduino**, no additional

hardware required. The two most popular LED libraries for Arduino are **Adafruit NeoPixel library** and the **FastLED library**. Make sure to verify your part numbers before you start programming to be sure you're compatible. Both are excellent choices, depending on your style of programming, but there's a lot more information online about the NeoPixel library, including some great Adafruit tutorials. One of my favorite places to get patterns is **Tweaking4All**, which provides a great variety in both libraries: [makezine.com/go/tweaking4all](http://makezine.com/go/tweaking4all).

There are a ton of **dedicated LED controller boards** out there that can help you map full video onto your matrix, program responsive designs, or do a variety of wacky things, but my favorite is the **Pixelblaze** by ElectroMage (Figure G) because it's easy to get started and it's pretty powerful! There's also a great community online, and it's packed with features right out of the box. The Pixelblaze V3 uses an ESP32 and has 12 GPIO pins for additional inputs (there's also a tiny Pico version without the GPIO). You connect it to a Wi-Fi network and access an interface through your browser, which allows you to use Javascript to code your own patterns and access a whole bunch of existing ones as well (Figure H). There is also a sensor breakout board that has an onboard microphone, accelerometer, light sensor, and co-processor that performs audio frequency analysis, making your interactive LED projects extremely compact and fast!

Other honorable mentions for best LED drivers include the **WOW Pixel Driver** by Elec-Tron, which has a web interface and ESP8266 support; it's available assembled or as a kit ([wow.elec-tron.org](http://wow.elec-tron.org)). And although it's discontinued, the **Fade Candy** by ScanLime is great for integrating with programming languages like Processing. I also like the **nLiten Tech Controller** used by my pals at **MakeFashion** — although I haven't used it myself, I have seen great things and it is made to get started quickly with beautiful effects ([nliten.tech](http://nliten.tech)).

All of these techniques can give your LED projects a smart look. Don't underestimate the power of a single LED, or the light of a basic strip. There's usually no need to for expensive high-density strips. You'd be surprised at what you can achieve with some creative diffusion!

## LIGHTING TECHNIQUES

It's best practice in the LED world to not just put bare LEDs on your project. You can use a variety of techniques to make your lighting projects more interesting:

- You can **diffuse** light by placing fabric or other semi-opaque white materials in front of your LEDs (Figure I); this makes the individual lights less visible. The trick with diffusion is getting distance between your LED and material. I recommend using things like plastic bags or sheets, cotton balls, fabric — get creative!
- **Backlighting** can be really useful to highlight an opaque element, like a sign or PCB. You place LEDs behind the material and allow light to spill out on the edge (Figure J).
- You can also try **edge lighting**: placing an LED or strip directly against the edge of a piece of clear plastic. If the plastic is etched, it will catch the light and illuminate your patterns (Figure K).
- Other options include **reflecting** light off mirrors (Figure L), or **refracting** it through prisms or gems (Figure M).



# Make a Motion- and Rain-Responsive Light-Up Umbrella With the Pixelblaze

We're going to make a simple but smart light-up umbrella that responds to both movement and the sound of rain. To do it, we'll use the Pixelblaze LED controller and its sensor breakout board, making use of the sensor board's existing accelerometer as well as connecting a piezo mic to the board's analog input pins.

## 1. SET UP THE LEDS

Place the LED strip around the umbrella's edge (Figure N) by sliding them under the umbrella arms. Secure with double-sided tape if needed. Point the LEDs inward, toward yourself, so you get a reflecting effect that illuminates the user.

Make sure you have enough wire on your LEDs to reach the umbrella handle by wrapping around the arms, so you can connect your circuit later.

## 2. ATTACH THE MIC

Solder two leads to the piezo microphone. I use silicone wire because it is very flexible, and as you open and close the umbrella you'll want to be sure the wire itself doesn't break.

Attach the piezo to the wall of the umbrella. I just slid it tightly under an arm (Figure O) but you can glue or tape it in place as well. The only requirement is that it is flush against the wall of the umbrella so it can sense vibrations.

## 3. CONNECT IT UP

Solder the piezo wires to pins A0 and GND on the sensor board (Figure P). Connect that board to the Pixelblaze header pins, and the LED strip to the Pixelblaze screw terminals (Figure Q). Finally, connect the battery bank (Figure R) and attach it all to the handle of the umbrella.

Set up your Pixelblaze with your local Wi-Fi network, and select the correct type of LEDs in the Settings tab of the browser interface. You'll access the Pixelblaze by navigating to the IP address, or through the device manager screen that pops up once you've connected.



**TIME REQUIRED:** 1-2 Hours

**DIFFICULTY:** Easy

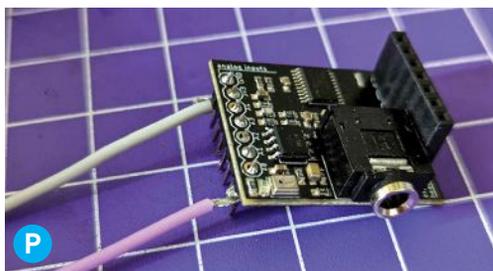
**COST:** \$100-\$120

### MATERIALS

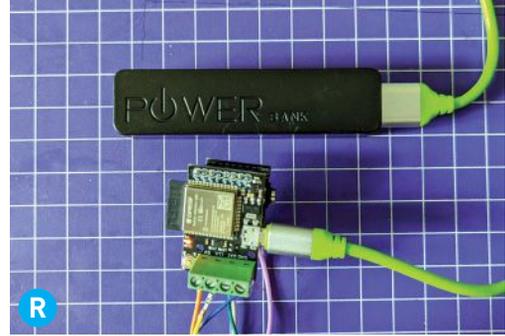
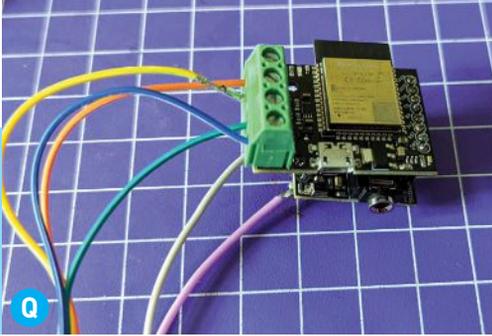
- » **APA102 LED strip with IP65 covering, about 2 meters** or other compatible LEDs such as SK9822, DotStar, etc.
- » **Pixelblaze V3 Standard LED controller** \$35 from [electromage.com](http://electromage.com)
- » **Pixelblaze Sensor Expansion Board** \$29
- » **USB battery bank, ~1000 mAh**
- » **Umbrella** white, black, or clear
- » **Piezo microphone**
- » **Silicone wire** or other stranded-core wire
- » **Double-sided tape**

### TOOLS

- » **Soldering iron and solder**



Dushan Mlicic, Lee Wilkins



```

6  sparkHue = accelerometer[0]      // Set the hue for each spark
7  sparkSaturation = 1 // Set the saturation for each spark (0 = white)
8  numSparks = 10 + floor(pixelCount / 100) // Scale number of sparks based on # LEDs
9  decay = .99 // Decay their energy/speed. Use .999 for slower
10 maxSpeed = .6 // The maximum initial speed of any spark / firefly
S  newThreshold = .01 // Recycle any spark under this energy

```

```

42 export function render(index) {
43   sparkHue = accelerometer[0] // hsv taken from accel axis
44
45   if(analogInputs[0] < 0.3){ // if the piezo is hit
46     v = pixels[index] // pick a point
T     hsv(sparkHue+random(0.1), sparkSaturation, v * v * 10) // display the color

```

#### 4. CODE SOME PIXELS!

The Pixelblaze is programmed using JavaScript, and what's great about it is that there are a whole bunch of existing pieces of code you can modify. I downloaded the FireFlies example from [electromage.com/patterns](http://electromage.com/patterns), opened it in the Pixelblaze interface, and began to modify some of the variables that were available to better understand how to control the pattern (Figure S).

The effect I wanted was that when the rain hits the umbrella, there should be a sparkle of randomly generated firefly light. So I knew my pattern had to be bright and fast. I played with variables like **maxSpeed**, **decay**, and **sparkSaturation** until I liked my sparkle effect.

Then, I added sensor input. I wanted the Y axis of the accelerometer to modify the colors, and the analog 0 pin to read the piezo mic data. To do this, I added these two lines at the top which create arrays that hold the input data. These are built into the Pixelblaze code and will automatically grab the variable data from the accelerometer and the input pins.

```

export var analogInputs = array(5)
export var accelerometer = array(3)

```

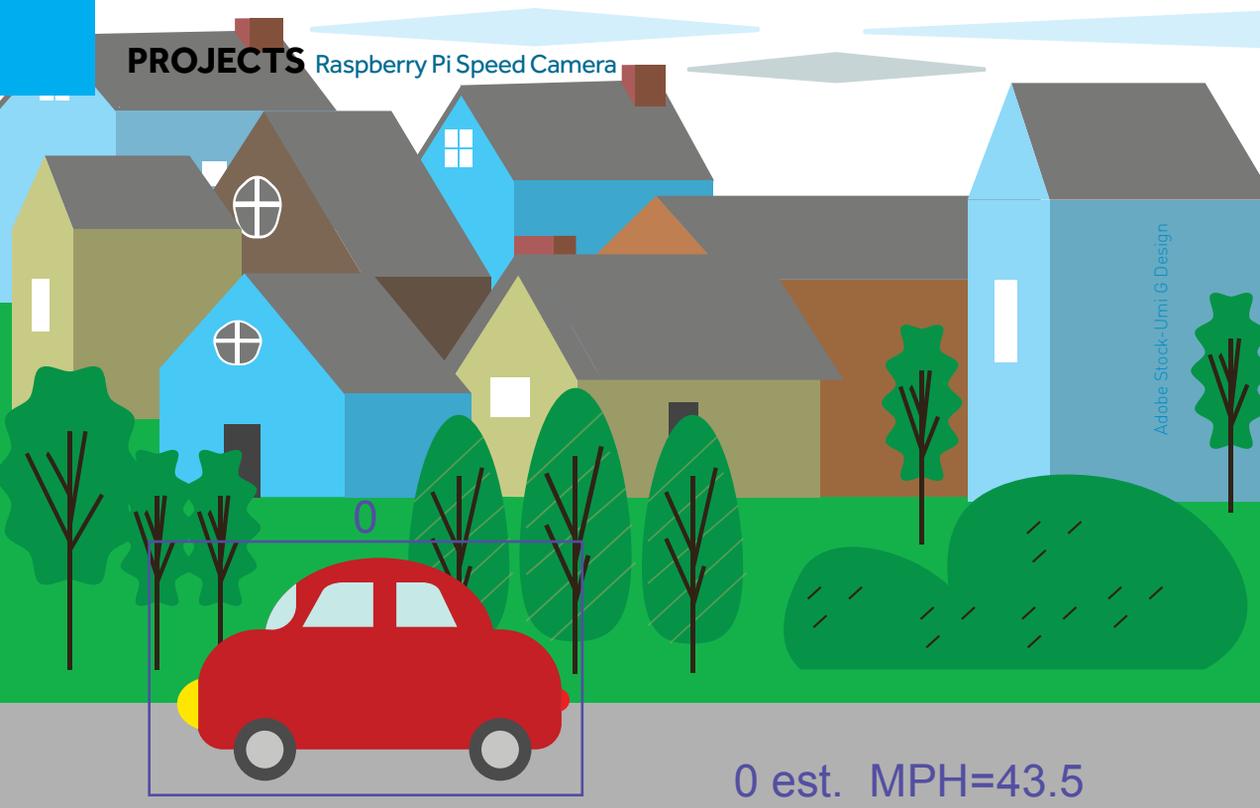
Next I modified the **render** function, which is responsible for displaying the pixels. I added an **if()** statement which checks the value of analog pin 0 by checking the array in position 0 with **analogInputs[0]** (Figure T). I set a threshold value of 0.3, but this might be different for you. You can monitor the value of input variables on the right-hand side of the interface, so I checked to see what the value was when I tapped on the piezo to simulate rain.

I also used HSV (hue, saturation, vibrance) instead of RGB (red, green, blue) values to manipulate the colors, which made it easier to use the **accelerometer[0]** value to change the color as the user moves the umbrella.

The resulting code shows a sparkle only when the piezo picks up a sound, and the hue of the sparkle is based on the umbrella's position! You can grab my code, and modify it as you wish, from [github.com/LeeRobot/accel\\_umbrell/blob/main/pixels.js](https://github.com/LeeRobot/accel_umbrell/blob/main/pixels.js). 🍌

**More about Pixelblaze**

- Read "A Perfect Circle" on page 28
- Sew your own Pixelblaze Pillows, page 48
- 2022 Boards Guide, included with this issue



# Pi Cam Speed Trap

Written and photographed by Ed Hume

## Build a motion recorder that analyzes video to catch speeding vehicles on your street

**How would you like an inexpensive camera you could point out the window (or at the fridge) and record video whenever motion occurs?** A great project — but let's take it further. If your window faces the street, would you like to know the speed of vehicles and pedestrians going by?

This project takes motion detection code examples found on the web and combines them with more sophisticated tracking and speed estimation. It only takes a little familiarity with Python for you to customize the software for your

own setup. Take a look at the screen capture (Figure A) or this video clip on Flickr ([flickr.com/photos/edhume3/52008654721](https://www.flickr.com/photos/edhume3/52008654721)) to get a feel for the speed trap we'll build here.

The Python app is a single file which can run on the Raspberry Pi 4B single board computer (SBC) using the Pi V2 camera module, or on a Linux system using a webcam. It also runs on my Jetson Xavier NX, which greatly outperforms the Pi — 21fps at 720×480 — but the Raspberry Pi is the best choice for a dedicated setup.

## MOTION DETECTION & TRACKING

Here's the high-level concept for motion detection and speed estimation:

- The camera provides a sequence of image frames. Each frame is compared to an average of past frames to detect *differences*.
- Analysis methods from the Open Source Computer Vision library (Open CV) are used to *blur and expand* the differences, for example to make a blob from moving legs merge with a blob from a moving torso.
- The **findContours** method is used to outline the blobs, and then to simplify matters, *bounding rectangles* that surround each blob are computed. So for each frame, there is set of rectangles that represent changing portions of the image. By correlating the rectangles from one frame to the next, we can track moving objects.

If a car is moving across the image from left to right, first a smaller rectangle is seen on the left side of the frame where the car enters the camera view. As more of the car comes into view, the rectangle expands in width, as there is a difference of the car image to what was an empty street. The leading edge of the rectangle, the right side, is seen at larger values of the *x* coordinate which is a count of pixels from the left frame edge. The difference in the *x* position of the leading edge from frame to frame is an approximate measure of the car's speed.

For the most part the frames are evenly spaced in time. However, there are occasionally lost frames, resulting in the time interval between frames being 2x or more than normal. So a call to **time.perf\_counter()** is a better measure of time than a count of frames.

## SPEED CALCULATION

The essence of speed tracking is to calculate the difference in *x* position per difference in time ( $dx/dt$ ) using the current frame versus the prior one, and to analyze the sequence of  $dx/dt$  values versus *x* for the moving object across the frame.

- The  $dx/dt$  values tend to vary in a regular way for a car that moves at steady speed along a street that may be angled slightly across the view, or is gently curved.

**TIME REQUIRED:** 2–8 Hours

**DIFFICULTY:** Intermediate

**COST:** \$130–\$160

## MATERIALS

- » **Raspberry Pi 4B single-board computer (SBC), 8GB RAM** or other Linux system
- » **Raspberry Pi Camera Module V2** or webcam for a Linux system
- » **MicroSD card, 32GB or larger**
- » **Aluminum heatsinks** for Pi 4B from pishop.us
- » **Ventilated case for Pi 4B** such as the PiShop HighPI case, pishop.us, with my ventilated replacement cover at [printables.com/model/178257](http://printables.com/model/178257)
- » **Flex cable for Pi Camera, 200mm** or suitable length for your installation
- » **Camera case and mounting brackets** You can 3D print my window shade camera case, [printables.com/model/172019](http://printables.com/model/172019), or tripod-mounted case, [printables.com/model/107623](http://printables.com/model/107623), or search Thingiverse and Printables for other cases.

## TOOLS

- » **3D printer (optional)** for DIY cases and/or brackets
- » **Computer** for setting up the Pi, not needed after setup
- » **Project code** Download the Python code file *piMotion.py* from the project page at [makezine.com/go/pi-cam-speed-trap](http://makezine.com/go/pi-cam-speed-trap).



Two kids on motorbikes, clocked at 24.8mph by the Pi Cam Speed Trap.



**ED HUME** lives in Austin, Texas, and is well known in the “live steam” hobby as the author of *A Climax Class A Live Steam Locomotive Model* and various magazine articles on metalworking.



Hardware for the window mounted speed cam.

- Next, the changing angle of view of the camera causes the  $dx/dt$  values to be slightly larger at the center of the view than at either end where the view of the moving object is slightly angled.
- On top of these regular variations in  $dx/dt$  there is less regular variation caused by changes in lighting and background color variation, changes in the actual speed of the object, and partial blocking of the image from tree trunks, vegetation, etc.

We cut through all this complexity by fitting the  $dx/dt$  versus  $x$  values to a **best-fit line** and interpolating the observed values to compute  $dx/dt$  at the center of the frame image. Most of the calculations for the best-fit line are done incrementally as each frame is analyzed, so the overhead of speed estimation does not impede real-time video capture.

If the tracked object motion is fairly steady, the  $dx/dt$  value calculated for the center of the frame is a decent estimate of the object speed at the center of the frame. The  $dx/dt$  value is converted to mph by multiplying it by a factor found by calibration runs. In my case, the street is about 90 feet away from the garage window, and slightly angled so that the left side is farther away. Driving by at 19.5mph resulted in a  $dx/dt$  value of 150.5, so my conversion factor for the garage camera is  $0.130 \text{ mph}/(dx/dt)$ .

## THE HARDWARE

Our app is somewhat resource intensive and uses the CPU heavily so a Raspberry Pi 4B with 8GB memory and some provision for heat dissipation is recommended. I've replaced the top cover of the HighPi case purchased from [pishop.us](http://pishop.us) with



Hooded camera case on window blind.

a 3D-printed cover that's perforated to provide venting. I also suggest adding inexpensive stick-on aluminum heatsinks from the same source (Figure **B**).

Video files captured with the Pi Version 2 Camera Module can range in size from 100MB to 3GB on a windy day. The app has logic to open new output files every midnight and to limit the number of files to a configurable number of days. A 32GB MicroSD card is adequate for several days' storage, and my practice is to move video files off the Pi to a different workstation for review. To save weeks of video on the Pi, get yourself a 64GB card or larger.

For a dedicated setup there is no monitor and keyboard connected to the Pi and all access is through the home network. By logging in with **ssh -X**, we enable X-Windows and see a decorated camera view plus another view of the computed blobs. The app uses console commands to optionally show or dismiss the two windows; unless they're being viewed, they're dismissed to reduce lost frames, CPU usage, and network load.

A good network connection is important to avoid lock-ups. The Pi's built-in Wi-Fi had difficulty keeping a good connection with my mesh routers, so I tried a USB Wi-Fi adapter that had a larger antenna. I soon replaced that with an additional router with a wired connection to the Pi. Once the Pi is using the wired LAN, the built-in Wi-Fi and Bluetooth can be disabled using **raspi-config**.

## TOO MUCH SUN

If the sun shines directly on the camera lens, the images get washed out. So I designed and



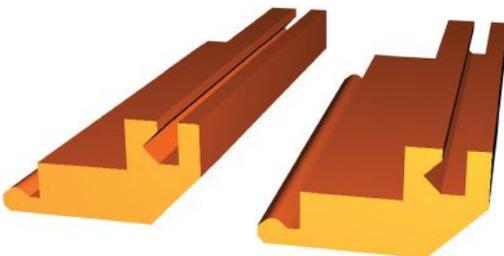
Pi case mounted on interior of window blind.

3D printed a camera case with a built-in hood (Figure C). You can download this case design, and the brackets that support it on a wooden slat of the window blinds, from [printables.com/model/172019](https://printables.com/model/172019). The lens hood should be printed in black filament or painted matte black to reduce light reflection. The brackets feature button heads spaced 40mm apart to support the Raspberry Pi case on the shade interior side (Figure D).

A non-shaded camera case that rests on wood blinds using the same brackets is seen in Figure E; you can download the 3D files from [printables.com/model/107623](https://printables.com/model/107623).

## NO-FASTENER CAMERA CASE

We are makers, so let's talk about the interlocking tongue-and-groove joints that slide into place on the camera cases. An experienced designer recognizes up front that several iterations of varying the dimensions, printing the items, and testing the fit may be needed to settle on the final dimensions. To economize on time and materials, a small test piece is designed and two of them are printed for test fitting, instead of the full-size items. So I've included a STEP CAD file and 3MF print file for the tongue-and-groove joint in the shaded case downloadable files (Figure F).



Plain camera case on window shade.

**TIP:** Here's something you may not realize: Most 3D CAD packages can import a STEP file and let you edit it, scale it, and then merge it with your own design. This can be a valuable shortcut from having to create a joint as a new feature with explicit dimensions for an extrusion. You can review the dimensions of the tongue-and-groove joint pieces by using the STEP file with the dimensioning features of your CAD software.

The Printables site also has project files for the ventilated case top mentioned above.

Plan to secure and protect the Pi and camera such that the camera has a fixed position, or is easily moved to a known position, when you're observing the app display. For example, a camera bracket could be fastened to a window frame using wood screws.

## SOFTWARE SETUP

Once your Pi system is up and running Raspberry Pi OS, and on the network, you can install the speed trap software. Python3 is likely already installed, and support for the camera and for an older but adequate version of Open CV (3.2.0) exists in the default software repositories. So this sequence of commands is likely all that's needed:

```
sudo apt update
sudo apt-get install python3-opencv
python3-picamera
```

Download the project code file, *piMotion.py*, from the project page at [makezine.com/go/pi-camera-speed-trap](https://makezine.com/go/pi-camera-speed-trap). Then you can run it at a command prompt of the `ssh -X` login window:

```
pi@raspberrypi:~/pycode $ ./piMotion.py
```

Here we are using a directory *pycode* for the app file and its output.

When reviewing or editing Python code, it helps to use an IDE that highlights the language keywords and can prompt you to fix syntax errors. Microsoft's Visual Studio Code is available from the default software sources and can be installed with: **sudo apt install code**

Let's hit some highlights of what a casual user may want to customize in the program code. The program is ready to run as-is on the different hardware platforms. There are different camera classes defined depending on the value of `platform.machine()`. There's a lot more detail in the code than I can present in this limited article, but you don't need to understand it all to configure and use it.

### CLIPPING LIMITS & NOISE SOURCES

We make the Pi's job easier with *clipping limits* that tell the motion detection logic to ignore image areas at the top of the view, such as moving tree branches, and at the bottom of the view, such as moving shadows. The Pi camera frame is 640px wide by 480px high. The y coordinate is 0 at the top of the frame, so setting `yTopClip` to 220 ignores blobs whose y centers are from `y=0` to `y=220`. The lower limit is configured similarly.

Within the image area of interest, what if there's a flashing light, or a waving tree branch? The motion detection logic knows to ignore these *noise sources* after several frames because they do not move away from their initial locations. However, their presence distracts the limited set of *tracker* instances. A list of *noise\_tuples* can be configured to define rectangular areas where smaller blobs will be ignored.

We've mentioned blurring and expanding the areas of *blobs* done by Open CV method calls; these act uniformly across the view. Additional logic tries harder to combine blobs that are seen at street level. For example, this can cause a man walking a dog on a leash to be seen as one moving object instead of two. So in addition to the clip limits, there are configurable bounds to define a *street level*, and there is a *gap size* value to influence the extent of merging.

### THE INVENTION OF BLOCKERS

Next, there is some algorithmic voodoo with the idea of *blockers*: As a walking person disappears behind a tree, the detected blob is reduced and then disappears for a few frames. Without some kind of software correction, a large tree trunk can cause an object that has disappeared to be seen as a new object when it reappears, or split what should be one big blob into two. A blocker may also cause the apparent speed to be reduced as the leading edge is hidden, and then to jump as it suddenly reappears.

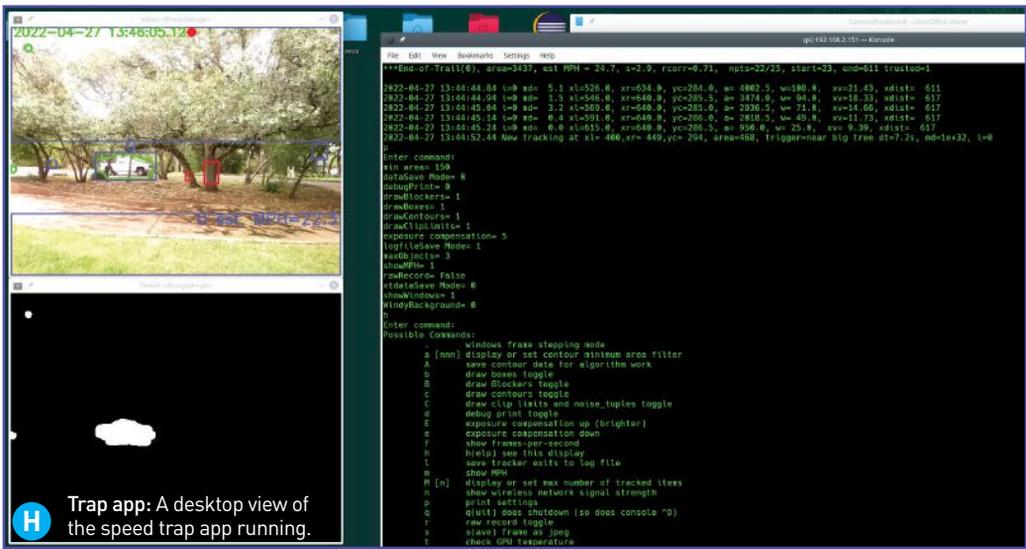
A list of blockers can be configured as rectangular areas that block the view of the moving objects. When the motion of a tracked object is predicted to intersect a blocker, the software creates fake blobs for the predicted area of intersection, and these fake blobs are merged with the observed blobs. The net effect can be seen in video clips — a blue rectangle indicating a tracked car or person moves smoothly through the large tree trunk (Figure G), like the camera has X-ray eyes.

### INTERACTIVE CONTROL

The app has keyboard commands to display or hide the configured clipping limits, noise sources, and blockers. This feature helps with checking the configuration or with re-aiming a camera to a previously configured position. Commands can always be entered at the console window using the Enter key. If X-Windows are displayed, single keystroke commands can be directed at a window without using Enter. Entering `h` (as in help) displays the available commands.



Big block: The speed trap uses predictive software to "see through" view blockers like large trees.



**H** Trap app: A desktop view of the speed trap app running.

The screen capture of Figure **H** shows a fully-decorated view that includes the clipping limits, blockers, and noise sources as configured for my setup. The clipping limits are the horizontal blue lines that span the image width. The bright red squares on the trees depict blockers. Small blue rectangles outline pesky moving branches that are configured as noise sources. The upper image also shows a bright green contour which is the outline of the blob created by the moving pickup truck, and there is a blue rectangle showing the limits of the blob. The truck has been tracked for several frames and there is a speed estimation label on the lower right. The bright blob in the lower image shows how effective the logic is at detecting the moving vehicle.

Speed estimation works best for a relatively clear view of traffic that basically moves horizontally across the frame. By default, the software is configured to only track the three largest blobs in each frame. This limit reflects performance and complexity considerations; Pi CPU usage is typically 80% with a video framerate of 10 or 12 frames per second. The system works well for a typical residential street, but not for a busy highway.

## SPEED CALIBRATION

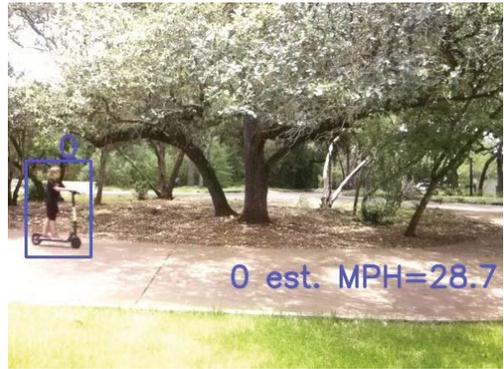
Once the clipping limits and blockers are configured, you'll make calibration runs to adjust the speed estimation for your street monitoring

setup. Simply drive through the entire camera view at a safe, steady speed. Then stop the program and play the recorded video file to see the estimated speed. Compute the ratio of your actual speed to the estimated speed. Find the **speedfactor** value in the source code and multiply it by your ratio.

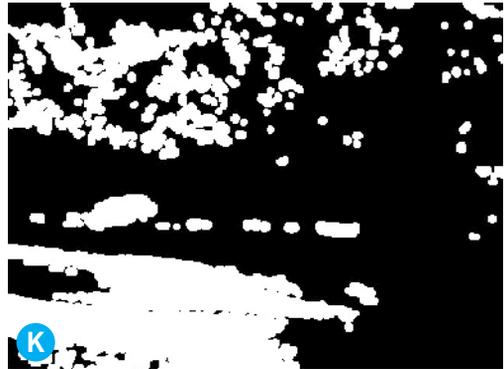
Based on my setup, the app is coded to save JPEG screen captures of any vehicles going by at 28mph or greater if the speed estimation passes various quality checks and there's enough daylight to yield a decent photo (Figure **I**). These criteria are in the **trail\_complete** Python method, which also does routine logging of the vehicle speeds to a text file. You can have fun adjusting the criteria for your deployment. The text logging doesn't consume much storage and



**I** Busted! A white SUV clocked at 28.3mph.



Back off, man: Speed is overestimated if moving objects are closer than the calibrated street.



Wind screen: A special command recalculates the background to cancel motion noise caused by the wind.

could be raw input data for your own analysis of dog-walking and traffic patterns.

## WIND TALES TAILS

So far it's all been happy talk. Then comes a windy day with moving clouds changing the light, waving branches, moving tree shadows, and leaves blowing around. These image changes adversely impact the analysis and tracking logic. While the program is running, a *minimum blob area* can be adjusted with a console command to ignore blobs that are smaller than the specified value. For my setup, the minimum area defaults to 150 pixels. A car's area might be 3,000 pixels, and the area of an SUV or truck might be 5,000 or 8,000. Setting the minimum area to 2,000 will still track cars and trucks while eliminating a lot of wind noise.

An even larger noise correction exists with the **W** command, which changes the *background calculation* so that the current frame makes a contribution to the background image used for

comparison before the comparison is made.

This is a big change from the usual behavior of only making a contribution to the background image when the scene is idle or mostly idle.

But the **W** command causes tracked vehicles to have tails (Figures **J** and **K**)! The newly revealed street at the rear of a moving vehicle contrasts with the vehicle image that has been part of the background for several frames. The tail phenomenon explains why the speed estimation is based only on the movement of the leading edge, and why the logic does not pay much attention to changes in the area of tracked objects.

A recent improvement was made to greatly reduce the recording of noise. Instead of saving each frame as motion is detected, the frames are queued to a buffer and file writing occurs after a delay. During the delay, the tracker instances decide if the motions have developed into something interesting enough to be recorded.

## WHEN BLOBS COLLIDE

A dramatic visual feature of the tracking software is the ability to track objects that *collide* and separate, for example two cars passing each other in front of the camera. The video clip on Flickr mentioned previously shows a vivid example.

The logic to handle colliding objects is similar to the logic that handles blockers. Every tracked object has a predicted position for the current frame. If two objects are predicted to collide, and an observed blob is consistent in size and location with the expected merged blob, then the logic continues to update the tracked coordinates to the predicted positions. This can continue for several frames until the objects pass by each other and are again tracked as separate blobs. Before adding this feature, the tracking logic would lose one of the objects and compute a distorted position sequence for the other.

## ARE WE THERE YET?

If you're interested in the software you can pore over the source code for more detail. The Open CV library code provides methods for motion detection and image manipulation, which are key features of this application, but at least 90% of the app is custom logic to organize and analyze the detected motions. An idea that's easy to express in casual speech, such as whether the scene is "mostly idle," may take dozens of lines of code to implement. The code to match observed blobs to tracker instances is elaborate and involves computing scores to determine the best match and whether a blob represents a new object or the continuation of a tracked object.

Some techniques in the code are very useful for any project, such as using a worker thread and command queue for console input, so that the main thread remains free for image processing and optionally displaying windows. You'll also find credits and links to earlier projects by Adrian Rosebrock and John Beale that were a starting point for the motion detection and tracking. Thank you to these gentlemen, and to Paul McWhorter whose YouTube series got me started with OpenCV and video analysis software. 🚗



Collision insurance: The speed trap correctly tracks two cars whose images collide and separate again.



**TIME REQUIRED:** 1 Day

**DIFFICULTY:** Easy

**COST:** \$15-\$30

#### MATERIALS

- » Mac Mini computer with M1 processor
- » iPad Mini 6 tablet
- » 3D printed case in PLA or ABS. You can download the free files for printing at [thingiverse.com/thing:5320501](https://thingiverse.com/thing/5320501), or buy a case on Etsy (search "portable mac mini").
- » Machine screws, M3×30mm socket head cap, with nuts Amazon B07VPGQ1RZ
- » Cable, USB-C to USB-A, 18" long Amazon B07PFHB3R4
- » Cable ties, 4" aka zip ties, Amazon B07P7L5169

#### TOOLS

- » 3D printer
- » Hex wrench, 2.5mm or 3/32" aka Allen key

# Make a Mac Mini Mobile!

**Hack that powerful Mini M1 with an iPad Mini to make a badass portable computer**

Written by Nico Jan and Scott Yu-Jan  
Photographed by Scott Yu-Jan

**As someone obsessed with modding and tinkering to get the most out of what I own, I've always thought my Mac Mini was not being used to its full potential.** With its impressively small footprint and light weight, I wondered why no one had attempted to turn it into a portable setup.

Meanwhile, I've always been disappointed by laptops, whether it's due to the short-lived batteries, overheating, or expensive price tags. Finally, as I don't work on-the-go often, my ideal setup has always been something between a laptop and a desktop computer: something that can be portable when I need to work from another location once or twice a month but without compromising performance.

After a year of loving the M1 Mac Mini, I was drawing on my iPad Mini 6 when I realized they're *exactly* the same width, making the iPad Mini the perfect monitor in a portable setup, with the aid of software like Sidecar or Duet.

After some brainstorming and quick sketches, I began designing a 3D-printable rig that could not only bond these two devices physically but also house a USB-A-to-C cable to connect them, as public Wi-Fi usually isn't consistent enough to allow the iPad to wirelessly act as a reliable monitor. Ultimately I designed the rig to embrace the familiar laptop-esque form so that the screen of the iPad can close against the Mac Mini, giving the setup the slimmest profile possible while also protecting the screen (Figures **A** and **B**).

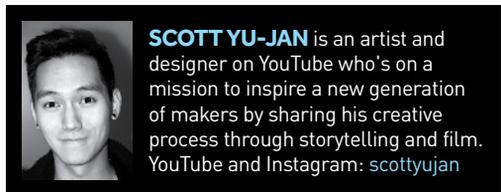
Amazingly, the Duet app and Sidecar both support Apple Pencil and touch as inputs for macOS, so a mouse is optional. When it's time to travel, I use Apple's Magic Keyboard for its Bluetooth connectivity, long battery life, and slim and lightweight design.

I know this design isn't for everyone and I definitely don't recommend going through the effort to make this rig if you often work on-the-go, as this does not prioritize mobility. The combined cost of an iPad Mini 6 and M1 Mac Mini is around the price of an M1 laptop, which has built-in batteries, a trackpad and keyboard, and a much thinner profile.

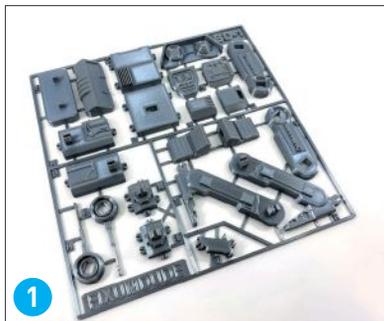
Nevertheless, if you're like me — already have the Mac Mini and maybe the iPad, don't work mobile enough to justify purchasing a laptop, but still want the option to work remotely given



access to wall plugs — then there's no setup more perfect. Making this project made me realize that this is the purpose of being a maker: to create the solutions for our own very unique problems and fill the needs that cannot be addressed by companies with mass-produced, one-size-fits-all solutions. 🛠️



Project video: [youtu.be/XFZ5l9SalVw](https://youtu.be/XFZ5l9SalVw)  
How to boot up Mac into Duet: [tinyurl.com/pmmduet](https://tinyurl.com/pmmduet)



# Kit Cards

Written by Caleb Kraft

**Modern model method: 3D print the parts on an old-school “sprue” then snap them off and build**

## TIME REQUIRED:

**1–2 Hours Print, 1–2 Hours Build**

## DIFFICULTY:

**Easy**

## COST:

**\$5–\$10**

## MATERIALS

- » 3D printing filament
- » Model glue
- » Model paints (optional)

## TOOLS

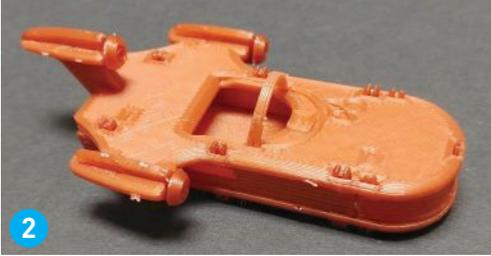
- » 3D printer
- » Hobby knife

## Model building carries positive memories for many makers.

Snapping those parts off that weird frame, cleaning them up, carefully assembling and then finally admiring the finished model. The experience could be so cathartic. An interesting trend in 3D printing has captured some of that nostalgia and repackaged it as a new method of making things.

*Kit cards* are 3D-printed models that come on a sprue — the plastic frame that used to be the result of injection molding — so that you can cut the pieces free and assemble your model. These little cards are a clever way of giving someone a 3D printed experience in a tidy, portable package, so that they too can enjoy the catharsis of building their model.

Here are six kit cards for you to try out.



2



4



3



5



6

### 1 STAR WARS BD-1 DROID

by Michael Grugel

[printables.com/model/188568](http://printables.com/model/188568)

Little buddy bot BD-1 may not be the best-known droid in the *Star Wars* universe but this kit card is very impressive in the quality of the finished model.

### 2 STAR WARS LANDSPEEDER

by Michael Grugel

[printables.com/model/80917](http://printables.com/model/80917)

Another stunning piece of work by Grugel, aka Fixumdude, this speeder shows how you can cleverly clip pieces together to get nice shapes with real depth. It doesn't even require glue! He's got lots more kit cards at [printables.com/social/98664](http://printables.com/social/98664).

### 3 T-REX

by Marco Melzi d'Eril

[www.thingiverse.com/thing:4902649](http://www.thingiverse.com/thing:4902649)

Going back to the basics, this *Tyrannosaurus rex* model is very reminiscent of a model many of us had as kids. Mine was glow-in-the-dark and if I use the right filament, this one can be too!

### 4 DIY MINI GUILLOTINE

by Thomas Quick

[printables.com/model/127813](http://printables.com/model/127813)

Frankly, I'm not sure what office doesn't need a guillotine.

### 5 JOHNNY 5 (NOVA ROBOTICS S.A.I.N.T)

by Thomas Quick

[printables.com/model/135284](http://printables.com/model/135284)

Johnny 5 from the *Short Circuit* movies looks fantastic in this kit. The fact that the neck and eyebrows were designed to be poseable is fantastic and adds so much life to the model.

### 6 FIDGET KIT BUSINESS CARD

by Yuval Dascalu

[www.thingiverse.com/thing:1956539](http://www.thingiverse.com/thing:1956539)

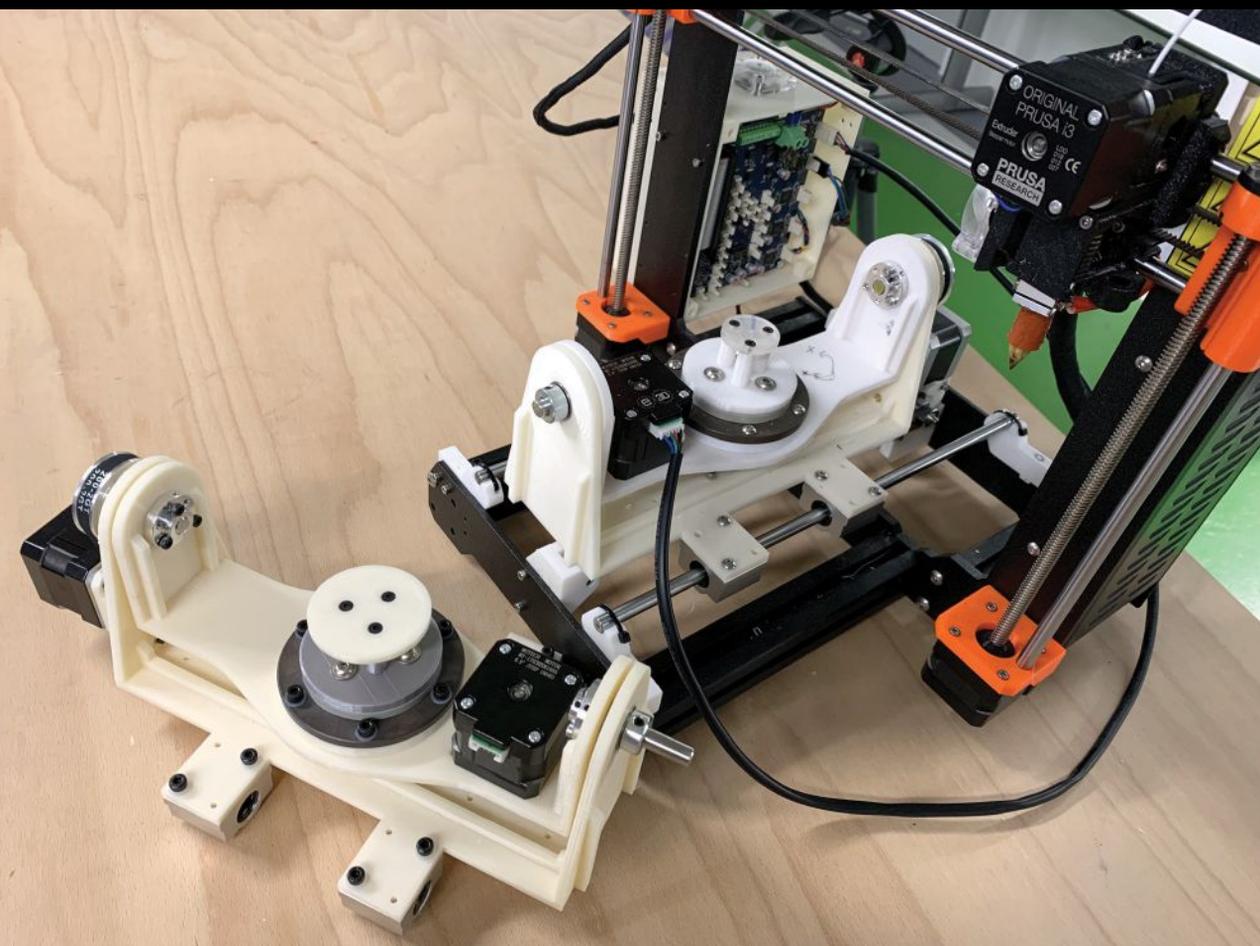
This clever little business card fits together to have functional gears. Customize it in CAD with your own text! 🚀



**CALEB KRAFT** is senior editor of *Make:* and has been 3D printing since 2013.

# The Fifth Dimension

Convert your 3D printer to 5-axis, and print the impossible Written and photographed by Freddie Hong



**FREDDIE HONG** is an artist and researcher who lives and works in London. Being interested in how digital technology and manufacturing dictates our relationship with materials, he finds new ways of designing and crafting artifacts through various inventive instruments in a research-led practice.

**TIME REQUIRED:** A Week

**DIFFICULTY:** Moderate

**COST:** \$480–\$500 for Upgrade

## MATERIALS

- » **Compact stepper motors, NEMA 17, 0.9° step angle, 40mm body length (2)** [e3d-online.com/products/motors](http://e3d-online.com/products/motors)
- » **Pulleys, GT2: 60 tooth, 8mm bore (2) and 20 tooth, 5mm bore (2)**
- » **Timing belts, GT2, 6mm wide: 224mm/112 tooth (1) and 200mm/100 tooth (1)**
- » **Brass nozzle, 0.4mm, extra long nonplanar.** [xyz/product-page/0-4mm-non-planar-nozzle](http://xyz/product-page/0-4mm-non-planar-nozzle)
- » **3D-printed parts: print bed, rotating gantry, etc.**
- » **Various bearings and hardware** See the full list of materials and vendors at [github.com/FreddieHong19/Open5x](http://github.com/FreddieHong19/Open5x).

## TOOLS

- » **3D printer** to print the hardware parts for this conversion
- » **Computer with Rhino/Grasshopper CAD software** The current development of the conformal slicing runs on Rhino/Grasshopper, which is not a free license software. We hope to develop and implement the software on a more accessible CAD platform.
- » **Pliers**
- » **Wire cutters**
- » **Hex keys**
- » **Table clamps or vise (optional)** useful during assembly

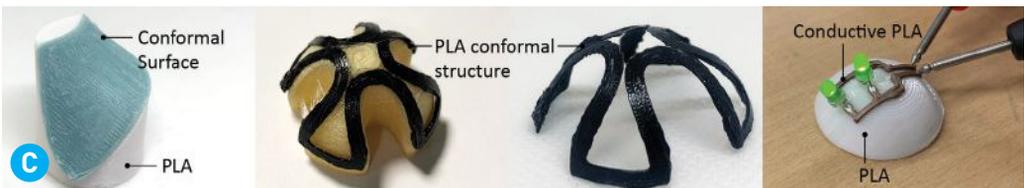
Five-axis 3D printing has many advantages over conventional planar 3-axis 3D printing, but when we searched for an affordable 5-axis 3D printer for a desktop environment, we couldn't find a suitable machine. (We did find one multi-axis CNC machine with filament extruder attached, but this was quite industrial and unable to use any of the open-source firmware such as RepRap.) So we decided to build one ourselves, and to share it so you can try it too.

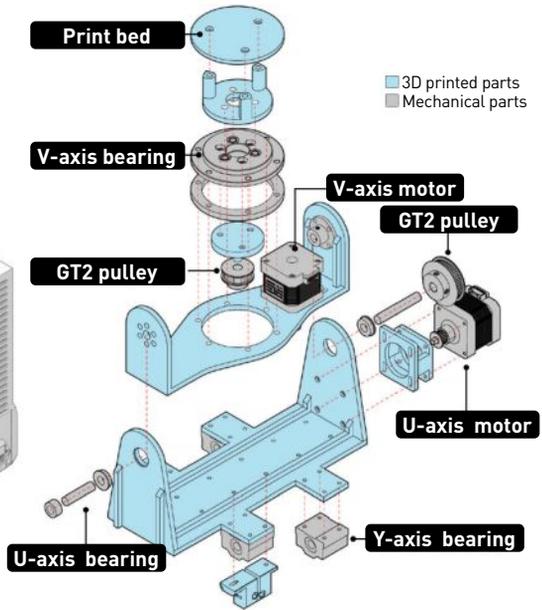
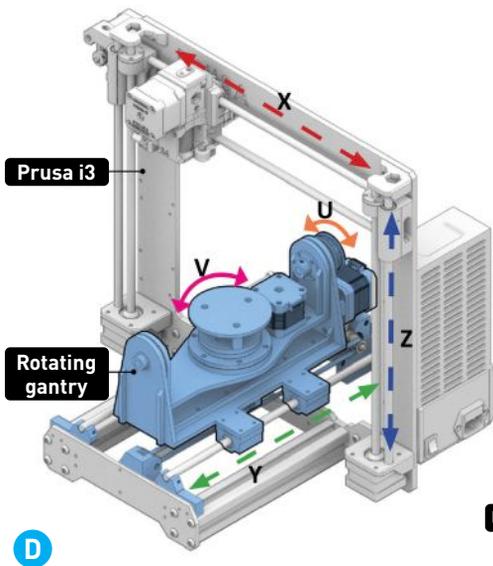
Why 5-axis? By adding two rotating axes of motion, you can do all kinds of new tricks: print without support (Figure A), do curved layer deposition for optimal mechanical strength (Figure B), achieve smooth, conformal surface finishes without the “staircase” effect, and even add colors and patterns on the surface cleanly without oozing or stringing (Figure C).

Instead of building entirely new 5-axis hardware, we decided to convert an existing Cartesian 3D printer — the Prusa i3. It's the most commonly owned 3D printer, and it's also entry-level hardware, which means that those who have moved onto more complex 3D printers may own unused i3's that they can experiment with.

The primary aim of our Open5x project was to develop a conversion kit (3-axis to 5-axis) that is highly affordable, easy to follow, and customizable (to other 3D printers) to invite more hobbyists and makers to try 5-axis 3D printing. We've also developed a GUI-based conformal slicer, integrated within a popular CAD package (Rhino/Grasshopper). This CAD software is not free license, which may limit some users, but we aim to implement the slicer onto open-source platforms going forward.

Excluding the Prusa i3 and the cost of 3D printing the kit parts, it cost us roughly £400 (\$500) to buy all the mechanical and electrical hardware for this 5-axis upgrade. It's a great project for someone who has built open-source 3D printers before, and has some knowledge of G-code and custom slicing.





Freddie Hong/Open5x

## ELECTRONICS

For our 5-axis conversion, we replaced the original Prusa i3 electronics board with the RepRap firmware-compatible 3D printing board called Duet 2. Duet 2 supports the addition of an expansion board called Duex 5 that allows up to 5 additional stepper motors.

Replacing the board is straightforward: you simply rewire the existing electromechanical elements from the original Prusa i3 board to corresponding pins on the new Duet 2 board. We share full connection details in our Open5x repository ([github.com/FreddieHong19/Open5x](https://github.com/FreddieHong19/Open5x)).

We found the Duet Web Control interface very user friendly to monitor and control the 5-axis system. Duet also has an active community forum where we sought advice when we needed to.

## BUILDING 4TH AND 5TH AXES

Our video at [youtu.be/C3vKHuxUmjc](https://youtu.be/C3vKHuxUmjc) shows a detailed walk-through of how to assemble the two rotary axes, known as the *U-axis* and *V-axis*. The exploded isometric drawing (Figure D) shows the overall assembly guideline. Parts colored in blue are 3D-printed parts and the parts colored in gray are consumer hardware.

You can 3D print the STL files from our repository, and we also share Rhino and Fusion files so you can modify them to fit different hardware. We tried using both ABS and PLA with

solid infill settings, and found that solid PLA has better rigidity.

The trickiest part of assembling the current iteration is mounting the U-axis stepper motor, but this will be changed in the next iteration. It's also tricky to fit the Y-axis belt, but not much harder than ordinary Y-axis tensioning.

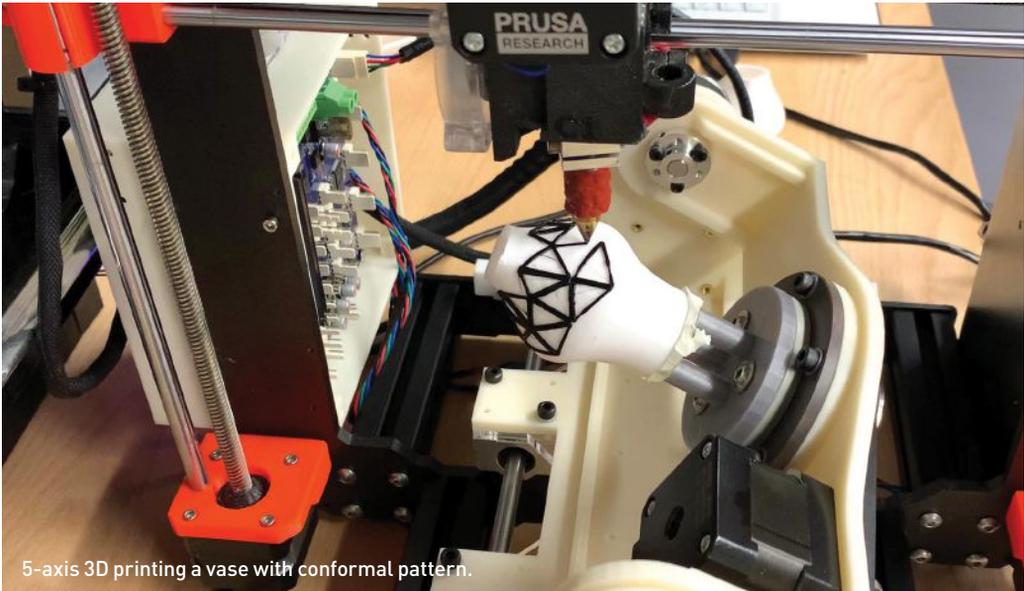
## TOOLPATH GENERATION

With extra freedom of design, 5-axis 3D printing also requires more considerations by the user. For example, you need to decide the direction of the filament extrusion, select supportless structures, and revise the toolpath to avoid collisions.

First you import the 3D model into Grasshopper and select the surface or a geometry that you want to slice and 3D print in 5-axis. Then you decide on printing parameters such as layer height, infill direction, and so on. In our current version of the slicer we can preview the toolpath before proceeding to produce the G-code.

## G-CODE GENERATION AND SIMULATION

When ready, you can start the G-code production in Grasshopper by clicking a button for slicing. Bear in mind that slicing for 5-axis takes much longer than conventional slicing, due to the two



5-axis 3D printing a vase with conformal pattern.

extra axes and the continuously rotating Z-axis! The small examples shown in this article took around 5–15 minutes to slice.

Users can also decide on the printing speed, nozzle temperature, and extrusion rate. Since the long brass nozzle takes quite a while to heat up, we recommend waiting about 2 minutes after the temperature is reached before starting the print.

In Rhino the user can simulate the printing process by moving the simulation slider. This is very important because in 5-axis printing, the nozzle may collide with the rotary bed, so you need to visually assess that there is no collision in the simulation. You can also adjust the retraction height to ensure that there is no collision.

To minimize the contact angle between the nozzle and the substrate we employed an extra long brass nozzle made by Nonplanar.xyz.

### 3D PRINTING WITH A TWIST

The primary aim of the Open5x project was lowering the entry barrier of 5-axis 3D printing. For this, we developed a 5-axis conversion kit that can fit onto the Prusa i3 but can also adapt to many other printers such as the Ender-3. Most multi-axis 3D printers are out of reach for individual makers because they're too expensive or need too much space and structure to operate; we wanted to create hardware that could invite more makers into 5-axis 3D printing research.

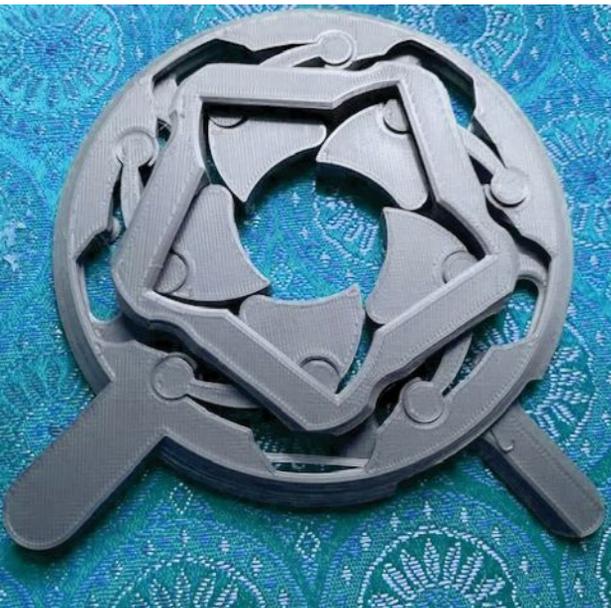
In addition to hardware, we understand that for many users the absence of a 5-axis slicer is the bigger barrier. Therefore, we also focused on creating a 5-axis slicer that looks similar to the existing planar 3D printing slicers with Graphical User Interfaces (GUIs). The Open5x project is still at an early stage of development and is not completely automated, but already we can showcase that it's possible to do 5-axis 3D printing on the desktop.

Going forward, we're working on improving the current slicer and implementing some new hardware. We are very happy to invite interested makers to contribute to the Github repository. We hope this project invites you into 5-axis to produce some unprecedented 3D printed objects. 🚀

**OPEN5X TEAM: Freddie Hong, Steve Hodges, Connor Myant, and David E. Boyle.** 2022. *Open5x: Accessible 5-axis 3D printing and conformal slicing.* 2022 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, [doi.org/10.1145/3491101.3519782](https://doi.org/10.1145/3491101.3519782)



Check out our YouTube playlist at [makezine.com/go/open5x-playlist](https://makezine.com/go/open5x-playlist) for more details on the whole project.



# One-Shot Wonders

**Design moving parts that print in place — no assembly required**

Written by Caleb Kraft

Something I find magical about 3D printing is that you can design gadgets that have some kind of mechanical motion — hinges, gears, pivots that rotate or spin — that print all at once and simply pop off the print bed ready to be used, no assembly required. These are often referred to as *print-in-place* mechanisms.

There are many ways to pull this off, but the easiest one has to be the *vertical hinge or pivot point*. Let me show you how this works, and how best to create one of your own.

Think about how a simple hinge works, standing vertically. You have a center pin and an outer sleeve. To print a hinge, you connect one side of your design to the outer sleeve and connect the other side to the inner pin. In Figure A you can see what I mean.

If you printed this design with a gap between the two parts of about 0.5mm, most printers would successfully create two parts that do not impede each other. If your printer is very precise, you can reduce this gap even smaller. (To figure out the precision of your printer for print-in-place, try printing the “tolerance coin” to the right.)

There are some issues with the basic design though. For example, if you were to print the basic hinge shown at left, it would simply fall apart as you removed it from the 3D printer. There’s nothing holding the two pieces together.

The simple solution to this is to widen the ends (Figure B), or in some cases, the center section (Figure C). This creates a captive piece that cannot be pulled apart. Simply adding a 45-degree chamfer at the bottom and top will usually do the job.

## MATERIALS

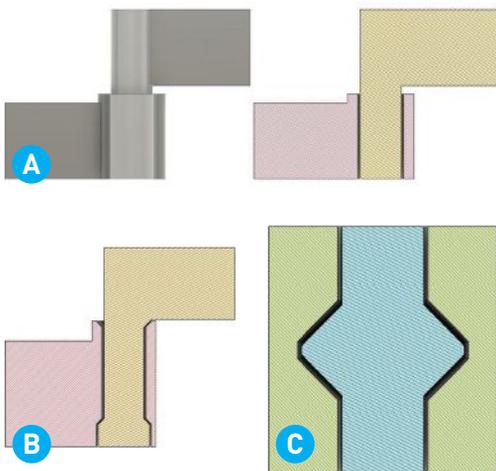
» 3D printer filament of your choice

## TOOLS

» Computer with CAD software and slicer software  
» 3D printer



**CALEB KRAFT** is senior editor of *Make*: and has been 3D printing since 2013.



You can find this single trick at the heart of many cool print-in-place gadgets. We've found five great examples for you to print and try out to see how they work. Remember that for any file you download and slice, you can browse through the layers in the slicer to see exactly how the designer is pulling off these neat effects.

**TIP:** Stand a print-in-place model on its side (Figure **D**) — not how you intend to print it — and do a slice to get a good cross-section of the joints (Figure **E**) and see how they work!

### MECHANICAL IRIS

I designed this one (shown on the opposite page) as a personal challenge to make a mechanical iris that is print-in-place. It was difficult but fulfilling! [printables.com/model/162554](https://printables.com/model/162554)

### **F** CEMENT MIXER

This entire toy prints all at once. You pull it off the print bed and you have a toy truck that has working wheels, a tip-forward cab, and a cement drum that spins. Very elegantly done. [printables.com/model/137746](https://printables.com/model/137746)

### **G** CLAW TOY

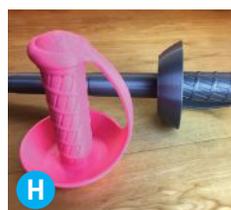
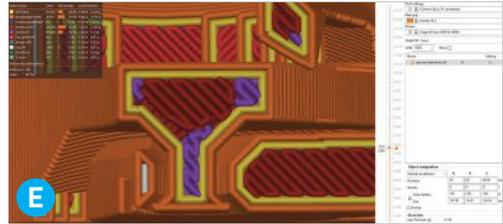
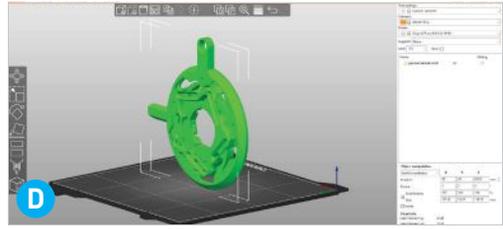
Claw toys are always fun. Here, you can simply print one up and have it work! The Printables page has great cutaway CAD drawings that show how the pivot points are made. [printables.com/model/200963](https://printables.com/model/200963)

### **H** COLLAPSING PIRATE SWORD

A creative application of this basic concept: Print a fully telescoping toy sword. Each segment is captured by the taper in the next largest segment. [printables.com/model/3710](https://printables.com/model/3710)

### **I** TOLERANCE COIN

Print this on your machine to find out how big of a gap you need to leave in your print-in-place designs. Each piece should break free and pivot, though admittedly the smallest gap gets fused even on my well-tuned machines. [printables.com/model/1707](https://printables.com/model/1707) 🚫



# Long-Distance Emoji Lamp

**OMG! Build an interactive lamp that sends emojis to its partner — miles away — using LoRa radio modules** Written and photographed by Bryan Serinese



**TIME REQUIRED:** 1–2 Hours

**DIFFICULTY:** Moderate

**COST:** \$55 Per Lamp

## MATERIALS

### TO MAKE ONE LAMP:

- » **Adafruit Feather 32u4 microcontroller with RFM9x LoRa radio, 915MHz** aka RadioFruit, Adafruit 3078, \$35
- » **LED matrix, 16×16, WS2812B** aka NeoPixel, AliExpress, \$10
- » **Mini rotary switch, 8-way selector** single pole, 8 throw (SP8T), Adafruit 2925
- » **Potentiometer, 10kΩ**
- » **Momentary pushbutton**
- » **Capacitor, 0.1μF**
- » **Switched power supply, 5V 3A** AliExpress, \$4
- » **Breadboard, 400 tie point** AliExpress, \$2
- » **Plywood, 3mm, 18"×9" piece** I use Baltic birch.
- » **Hookup wire** I use Cat5 cable.

## TOOLS

- » **Soldering iron**
- » **Wire cutters**
- » **Laser cutter (optional)** You could build a different enclosure if you don't have access to a laser cutter.



**BRYAN SERINESE** is a self-diagnosed hack who hobbies and cobbles bits and bytes to make new and occasionally useful thingies. He works in his basement on LED projects

to brighten the long Vermont nights when he is not putting the final touches on his forthcoming zombie musical. [serinese.com](http://serinese.com)

## FIRST, THERE WAS LED

In 2013, I was invited to Tom Igoe's wedding. Yes, *the* Tom Igoe, co-inventor of the Arduino and bestselling author of *Making Things Talk*. He married a good friend of mine from college, and she invited me with the stipulation that I didn't ask Tom to sign my first edition of *Physical Computing*. Apparently, she was sick of middle-aged fanboys stopping him on the street for autographs and advice.

In lieu of floral centerpieces, Tom made LED lanterns for each table. If a guest touched one, it not only changed colors but changed the color of the lanterns in its proximity like a rainbow ripple. When I told my daughter, she wanted me to make a lantern for her. I never got around to it, until recently when she threatened to buy one of those internet-connected lamps for her and her friend. Fortunately, my neighbor Pete had just given me a LoRa module and I was looking for a project to use it with. My daughter's friend lives a mile and a half away, so I planned to build two lamps and connect them directly without having to use a LoRa gateway. I call it the LOL-Ra Lamp.

You can find the source code and the SVG files to laser cut the enclosure at [github.com/bserinese/LOL-Ra](https://github.com/bserinese/LOL-Ra). I also posted some tips on how to create your own emojis for the lamp, and additional images and video, at [serinese.com/lol-ra-lamp](http://serinese.com/lol-ra-lamp).

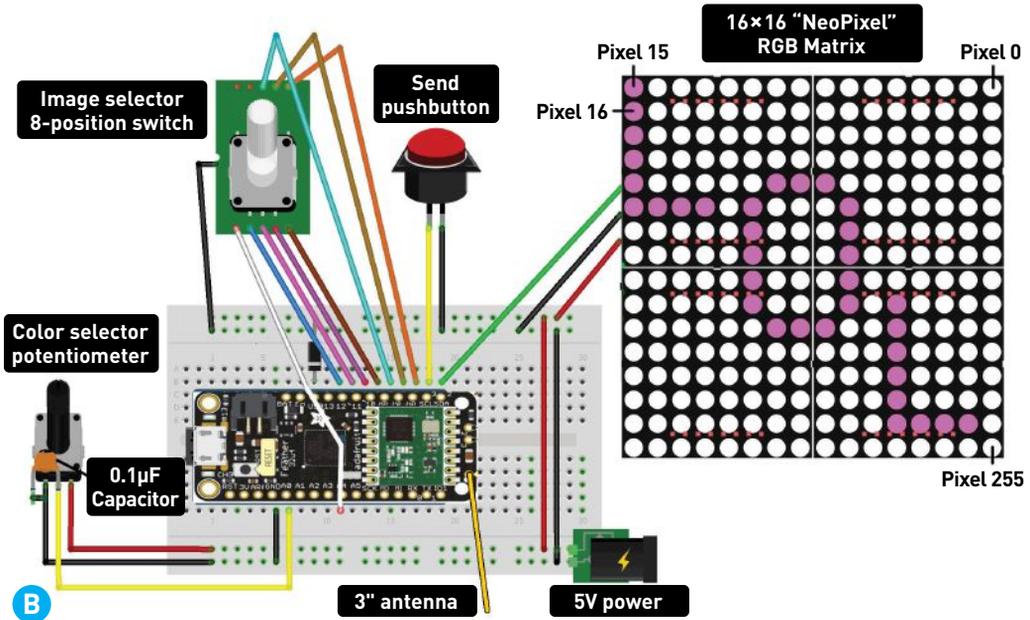
## 1. HARDWARE ASSEMBLE!

By using the Adafruit Feather 32u4 LoRa module, I got a microcontroller and LoRa radio in one. It might have been cheaper to use an Arduino and separate LoRa module, but I opted for simplicity. The lamp uses five components, so assembly is not too difficult using a breadboard.

I started by soldering an 18" wire to each pin of the 8-way rotary switch, to the three legs of the potentiometer, and to each end of the pushbutton. I use solid-core Cat5 since it has eight color-coded wires neatly packaged. That's great for the rotary switch (Figure A), since I just needed one more 18" piece for the ground center pin. To reduce the noise on the potentiometer, I also soldered a 0.1μF capacitor across the ground and center pins; otherwise the LEDs flickered like a bad disco.

The LED matrix comes with two 3-wire





Fritzing

connectors, a male and a female. The male should be connected to the data-out, so it won't be used. I cut the data-out cable, plugged it into the data-in, and soldered jumper wires to it to be connected to the breadboard. This makes an easy way to attach the matrix to the breadboard later.

Next, the Feather is inserted onto the breadboard. If you have to solder the header pins on first, insert the long side of the pins into the breadboard, lay the Feather on top, and then solder the pins. I created many microcontrollers with crooked header pins before I learned that simple trick.

Strip off about 5mm of insulation from the wires and plug them into the breadboard following the wiring diagram (Figure B). For the 8-way switch, it doesn't matter which wire goes to which pin at this point; it's easier to rewrite the code to match the proper pin to each image after you install the switch in the enclosure. For the ground wires, plug them into the ground rail on the side of the breadboard. Additionally, I used a jumper wire to connect the Feather's GND pin to the ground rail and soldered a 3" piece of wire on the antenna pin to act as a simple quarter-wave whip antenna.

You'll need one more jumper or a diode to connect the USB pin to the 5V rail, but don't connect that yet.

## 2. YOU'VE GOT THE POWER

Originally, my emoji designs used all 256 LEDs on the matrix, and I had the breadboard powered by one of those convenient breadboard power supplies that deliver 5V or 3V to the side rails of the breadboard. When I was showing off an early prototype to my neighbor Pete, he pointed out running all 256 LEDs was going to draw over 5 amps of power ( $256 \times 0.02A = 5.1A$ ) and my supply only handles a max of 0.7 amps. He recommended scaling down the designs and using at least a 3-amp power supply. Pete knows math. Pete knows electronics. I listen to Pete. So, I cut off the end of a 3A 5V USB power supply and soldered leads to connect to the breadboard's power rail.

## 3. UPLOAD THE CODE

The Arduino program, *LOL-RaLamp.ino*, has three main components: 1) displaying the images, 2) sending and receiving the image information, and 3) displaying the partner images.

To create the images, I first designed them using a simple 16x16 cell box in Excel. I filled each cell with values 0 through 255; 0-15 on the first row, 16-31 on the second, and so on. Next, I drew the image by filling in each cell with the color I wanted. Since the "first" pixel of the matrix starts in the top right corner and zigzags

down the rows to the final pixel in the bottom right corner, I needed to reverse the order of every other row. Then I filled the colored cell with its hex RGB value and put a 0 in the pixel that won't light up and 1 in the pixel that will be variable based on the color selector. After saving the Excel as a comma-delimited .csv file, I opened it using Notepad to extract the values in the format I need for my code (Figure C).

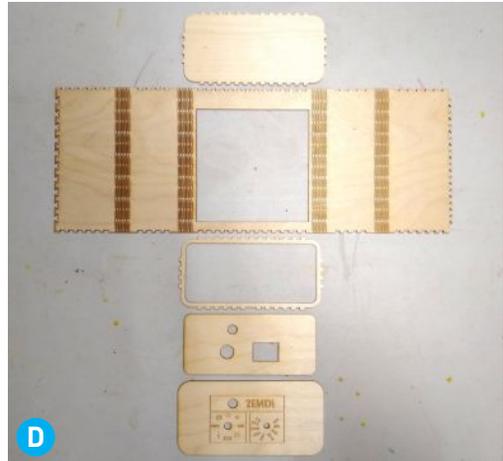
I'm using the **PROGMEM** string modifier to save the string into program memory, as these images will quickly fill up the flash memory without that command. An alternate method to create the image string is to use a tool like LCD Image Converter and copy the C source code it generates. You will still need to reverse the order of every other row; I posted a tutorial fully explaining both methods at [serinese.com/lol-ra-lamp](http://serinese.com/lol-ra-lamp) (you can get the complete code there too).

A series of **IF** statements determine which image function to call based on the position of the 8-way rotary switch. The potentiometer value maps to a string containing the nine colors, so you could add many more if you choose.

The next section contains the send and receive functions. I copied most of it from Tom Igoe's sample LoRa program that loads with the libraries. (My wife wouldn't let me steal one of his centerpieces, but I have no problem taking his code.) If the send button is pressed, the position of the image and color is sent to the partner lamp as the LoRa message. On the receiving end, the lamp will continually listen for a message from its partner and then parse the two-character message into the new image and color values.

I left out the callback confirmation because I wanted to replicate the uncertainty of pre-electronic communications. Think about how much great literature wouldn't exist without all those missed connections! (Juliet: "Not dead IRL. I H8 Paris." Romeo: "K.")

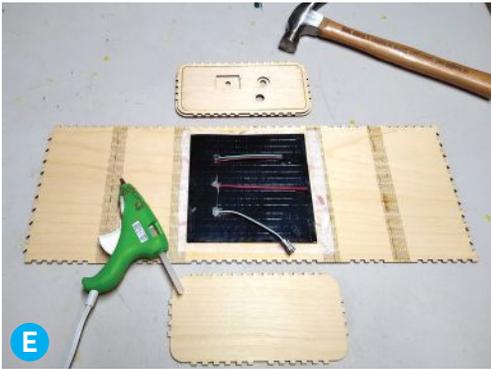
The final component is the partner image display. This is a **CASE** statement that reads the received color value and displays the corresponding image locally. The partner lamp will continue to display this image regardless of what is selected locally. Only by sending an image back can the image change, forcing both lamps to always display the last image in the conversation.



## 4. BUILD THE LAMP ENCLOSURE

After I was confident the lamp would display and transmit properly, it was time to package it up in an enclosure. My local makerspace has an Epilog Fusion laser cutter with a 20"×30" bed so I was able to cut and engrave my enclosure using a single 18"×9" sheet of 3mm Baltic birch. Yes, I switch back and forth between metric and Imperial, so what? It's not like I'm building a Mars lander.

You can download the SVG file at [serinese.com/lol-ra-lamp](http://serinese.com/lol-ra-lamp). I made these plans using the Rounded Box generator (with lid) at Boxes.py ([festi.info/boxes.py/RoundedBox](http://festi.info/boxes.py/RoundedBox)). I like this site because the tabs fit really tight, so I rarely need glue. Once I had downloaded a template for the outer enclosure, I used Inkscape to add a 117mm×117mm window for the LED matrix as well as holes for the various controls. Next, I added the images and text around their respective controls and sent it to the laser to engrave and cut (Figure D).



I glued a sheet of pressed paper over the LEDs as a light diffuser. Hot glue is my go-to for the paper and for setting the LED matrix in place (Figure E). Before you do that though, figure out which side is up. My matrix has no clear marking, so the best thing to do is mark the top *after* you run the code and see the orientation of the image.

Glue the lid bottom to the lid top, using the tabbed top rim piece to help center them. Don't glue this rim, as it needs to be attached to the enclosure. Now fold the enclosure around and use a hammer to gently tap its tabs together. I used a 2x4 over a pair of sawhorses to act as my anvil (Figure F). Take your time and you'll be rewarded with a satisfying *whap!* each time they snap in place. Tap the enclosure bottom piece in, and then the top rim that holds the lid.

Once the lid is dry, screw in the switches and potentiometer and reinsert the wires into the breadboard. Snake the power supply through the back hole and plug the 5V and ground wires into their respective rails. Now that your code is uploaded, you can add that jumper wire from the 5V rail to the USB pin. If you plan to upload new

code, be sure to remove that jumper because Adafruit says so! Finally, place the breadboard in the enclosure and add the lid (Figure G).

## SMILES FOR MILES

Plug in your LOL-Ra Lamp and select the emoji and color that suits your mood (Figure H). Be sure to share that emoji with your BFF — no phone or internet required!

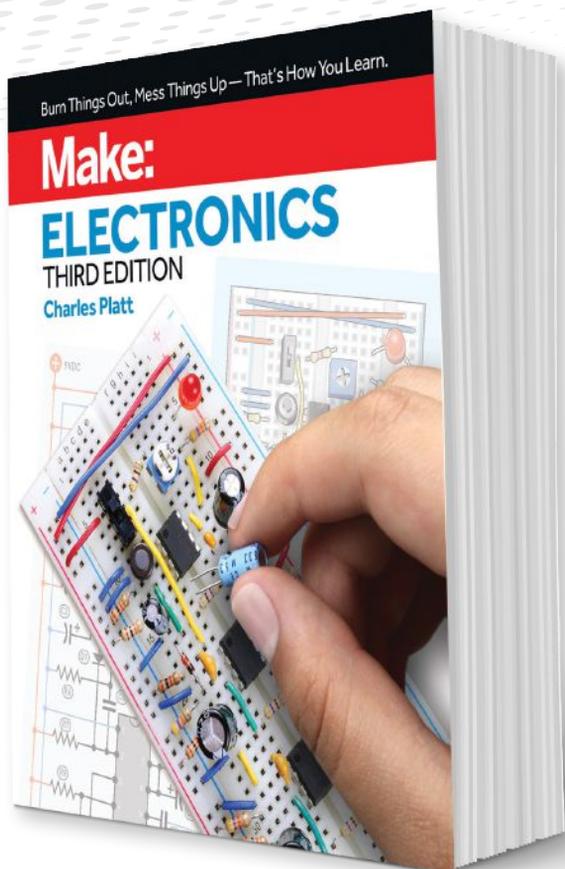
## MAYBE NEXT TIME

Technically, this project was a success — I could broadcast an image to the partner lamp over a mile away! These LoRa modules easily transmit 2km (1.2mi) with quarter-wave antennas, up to 20km (12.4mi) with directional antennas.

Unfortunately, I made the classic project management mistake and didn't confirm the requirements with the client before beginning. According to my daughter, the emojis I created are "too retro" and only a Boomer would use them. To keep up with the times, I might have to build an animated clock option to replicate this TikTok thing she keeps going on about. 🕒

# Make: Books

LET CURIOSITY LEAD YOUR  
LEARNING JOURNEY



## **Make: Electronics, Third Edition**

by Charles Platt \$34.99

**NEW COLOR ILLUSTRATIONS!**

Starting with basic concepts, this friendly and comprehensible guide takes the reader step-by-step toward circuits of increasing complexity. The principle of Learning by Discovery, pioneered by Charles Platt, uses hands-on experiments to create a lasting and entertaining learning experience.

### **OTHER GOODIES TO GRAB:**

#### **Make: Calculus**

by Joan Horvath and Rich Cameron  
\$29.99

#### **Robot Magic**

by Mario Marchese  
\$24.99

#### **Making Simple Robots, Second Edition**

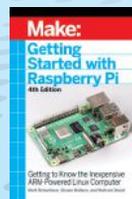
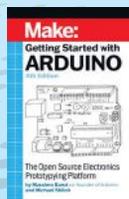
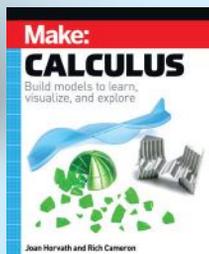
by Kathy Ceceri  
\$24.99

#### **Getting Started with Arduino, Fourth Edition**

by Massimo Banzi and Michael Shiloh  
\$19.99

#### **Getting Started with Raspberry Pi, Fourth Edition**

by Shawn Wallace, Matt Richardson,  
and Wolfram Donat  
\$24.99



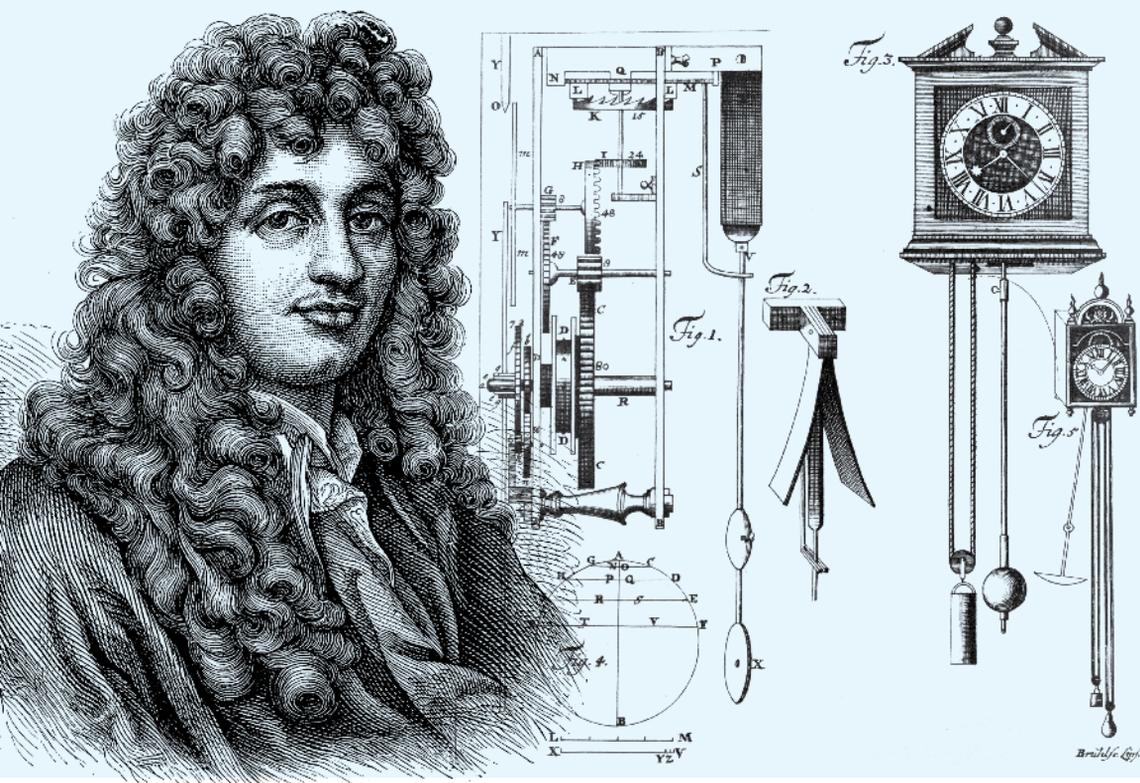
Make: Books are also available on [amazon.com](https://www.amazon.com)

Save \$5 off these titles with promo code **DOESCOMPUTE** today!

[makershed.com](https://makershed.com)

# Christiaan Huygens and the Pendulum Clock

Written and photographed by William Gurstelle



## Build the ingenious gravity-driven mechanism that kept the world's time for 270 years

**The Scientific Revolution was a momentous time.** Most historians of science agree that during this era — 1500 to 1700 CE — people started thinking differently, more scientifically, about the way the world worked. Many began to think of the world as being an orderly machine, one in which laws and rules controlled the way nature worked. Although scientists, or natural philosophers as they were then called, did not necessarily know exactly what these laws were, they were certain

they were there, working behind the scenes.

The superstars of the Scientific Revolution were Galileo, Copernicus, and Newton, but there were a number of other important thinkers as well: Robert Boyle, Johannes Kepler, and René Descartes, to name but a few. Often overlooked in the crowd of luminaries is the great Dutch scientist Christiaan Huygens. Huygens first made sense of Saturn's rings; discovered Titan, the largest moon in the solar system; laid the

**TIME REQUIRED:** 1 Day**DIFFICULTY:** Easy**COST:** \$30-\$40**MATERIALS**

- » **Wood boards, 1' long: 2×4 (1) and 2×2 (1)** for the frame
- » **Wood scraps, 3/4" thick: 2"×2" (1) and 3 1/2"×3 1/2" (1)** for the pendulum crutch and escapement block, respectively
- » **Hardwood dowels, 3/8" diameter: 24" (1) and 6" (1)** for lower pendulum and upper pendulum
- » **Steel bolts, 3/16"×5" (2) and nuts (8)** for pendulum shaft and crank shaft
- » **Plywood, 1/4" thick: 2"×2" square (1) and 1 3/4"×1 3/4" squares (2)** for the crank and the escapement wings
- » **Pan head bolt, #10×1 1/2", with 2 nuts** for the drive pin
- » **Ball bearings, 608 size (4)** from a skateboard shop, or search online for "608 bearing"
- » **Twine, 8' length** such as mason twine
- » **Washers, 3/16", a box**
- » **Weight, 8oz** for the pendulum. I used a 3/16" eyebolt with a nut and some washers
- » **Deck screws, 2 1/2" (3)**
- » **C-clamp**
- » **Tape** the less slippery, the better

**TOOLS**

- » **Jigsaw**
- » **Electric drill and drill bit set**
- » **Screwdriver bit** for deck screws
- » **Forstner bit, 22mm or 7/8"**
- » **Handsaw**
- » **Glue**



**WILLIAM GURSTELLE's** book series *Remaking History*, based on his *Make*: column of the same name, is available in the Maker Shed, [makershed.com](http://makershed.com).

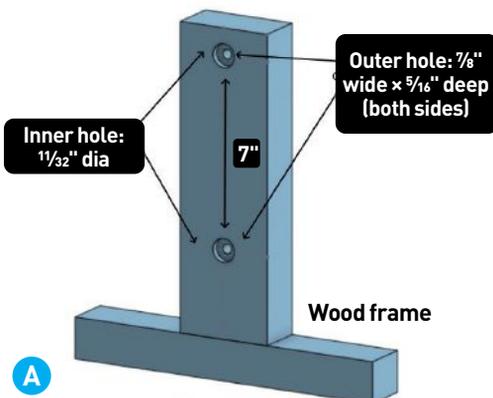
groundwork for the physics of Isaac Newton; and designed and built one of the first vacuum pumps (see my article "Robert Boyle and the Air Pump" in *Make*: Volume 78.) But arguably, his most significant and practical invention was the pendulum clock.

At the time Huygens lived, from 1629 to 1695, clocks were becoming increasingly important in daily life, but they were very inaccurate. The best of them could do no better than gain or lose more than 15 minutes in time every day. While a clock like that could tell you when it was time to eat lunch, it was not suitable for navigation or astronomical observations.

Inspired by Galileo Galilei's investigations of pendulums about 75 years earlier, Huygens began thinking about how he could use pendulums to keep better time. Huygens knew that Galileo had found that pendulums had a fascinating property: they were isochronous, meaning that the time it takes for any single pendulum to swing back and forth is the same no matter how high the point at which the swing begins. This made them useful for keeping time because for a given pendulum length, the time interval for the swing back and forth is always the same. When Huygens substituted the constant-swing-rate pendulum for the imprecise balance wheel used in clocks up to that point, the error rate of a Huygens pendulum clock shrank from 15 minutes to 15 seconds a day! This was indeed a scientific revolution in timekeeping.

In this edition of *Remaking History*, we recreate a Huygens-style pendulum clock *escapement*. An escapement is the heart of any non-electronic clock. It's the mechanism that goes "tick-tock" and actually keeps time. Because Huygens' original clock requires accurately cut gears which are difficult to make, we will instead build a simpler descendant called a *single-pin escapement* clock. The single-pin escapement isn't widely used by clockmakers, but it certainly keeps time, is easy to build, and makes a loud and satisfying tick-tock when you get it to work.

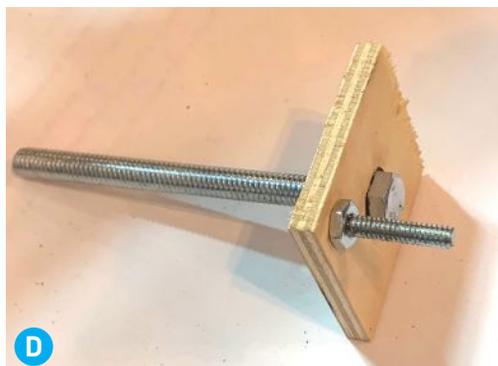
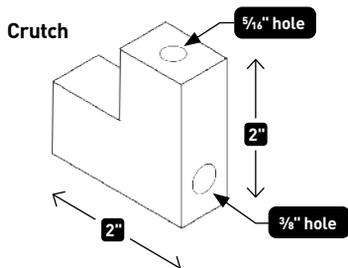
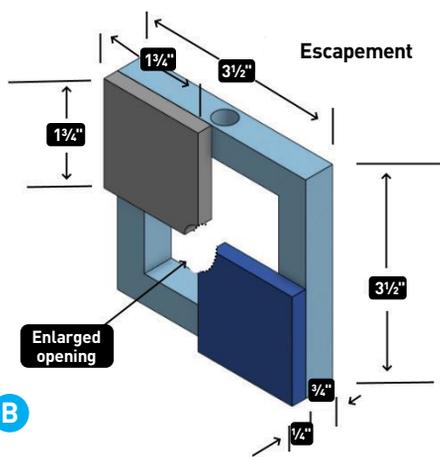
1. Build the frame for your escapement by joining the 2×4 and 2×2 boards as shown in Figure A, using 3 deck screws through the bottom.



- Use the jigsaw to make a  $2\frac{1}{2}'' \times 2\frac{1}{2}''$  square cutout centered in the  $3\frac{1}{2}'' \times 3\frac{1}{2}'' \times \frac{3}{4}''$  escapement block. Next, drill  $\frac{3}{8}''$  holes into the center of the top and bottom of the  $\frac{3}{4}''$ -thick escapement block. Glue the  $1\frac{3}{4}''$  squares to the escapement block as shown.

When the glue is dry, slightly enlarge the opening where the two  $1\frac{3}{4}''$  squares meet, by cutting about  $\frac{1}{4}''$  off each corner (Figure B).

- Cut the crutch as shown in Figure C, then drill the  $\frac{3}{8}''$  hole and  $\frac{5}{16}''$  hole as shown.
- Drill a  $\frac{5}{16}''$  hole in the center of the  $2'' \times 2'' \times \frac{1}{4}''$  plywood square. Then drill a single  $\frac{1}{4}''$  hole,



1" away from the center hole in any direction. This piece is the crank.

- Insert the  $\frac{5}{16}''$  crank shaft bolt all the way into the center hole in the  $2'' \times 2'' \times \frac{1}{4}''$  crank, so that it's flush against the underside of the bolt head. Glue into place.

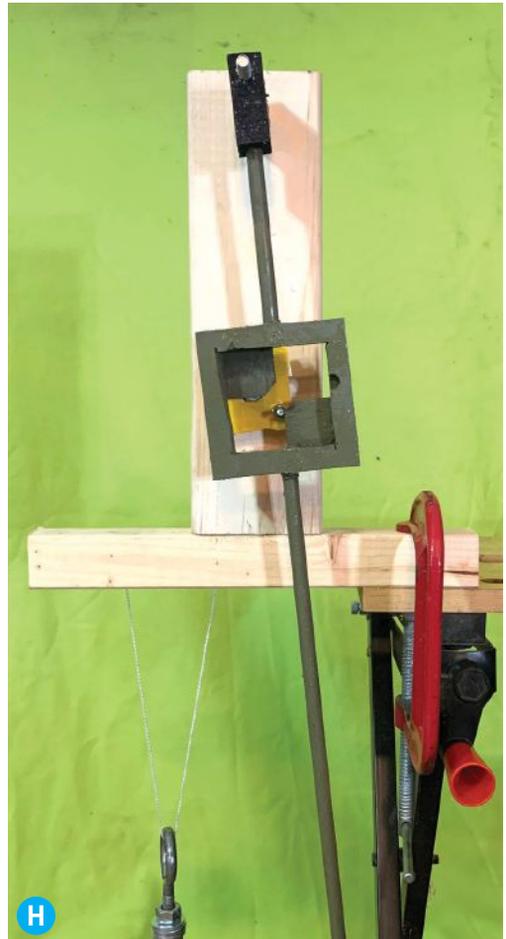
Insert the #10 bolt through the  $\frac{1}{4}''$  hole as shown in Figure D, and glue it in place too. This is the escapement drive pin.

- Drill two  $\frac{1}{32}''$  holes through the  $2 \times 4$  frame piece as shown in Figure A. Countersink both ends of each hole  $\frac{5}{16}''$  inches deep with the 22mm (or  $\frac{7}{8}''$ ) Forstner bit. Insert a skateboard bearing in each 22mm hole, front and back. The top hole is for the pendulum shaft bolt and the bottom is for the crank shaft bolt.

- Use the C-clamp to attach the frame to a workbench. Insert the pendulum shaft bolt through the upper hole in the frame (Figure E).
- Similarly, insert the crank shaft bolt through the lower hole in the frame.
- Push the  $\frac{5}{16}''$  hole on the crutch onto the pendulum shaft bolt. Insert one end of the upper pendulum into the  $\frac{3}{8}''$  hole in the crutch, and the opposite end into the hole on the top of the escapement block. Slide the escapement up or down on the upper pendulum until the opening in the center of the escapement aligns with the axis of the horizontal crank shaft bolt. Then cut the pendulum to length.

Insert the lower pendulum into the bottom hole on the escapement block. Use washers as spacers to align the crank bolt so the midpoint of the drive pin pushes against the escapement wings when the crank turns. The idea is that when the falling weight turns the crank shaft, the drive pin alternately engages the wings, which in turn pushes the pendulum back and forth. See Figures F and G.

- When you're satisfied with the alignment, fix all the wood-on-wood connections with glue to prevent the parts from spinning. The completed single pin escapement is shown in Figure H.
- Wrap the protruding crank bolt with some tape to make a little drum. Then wrap a few feet of twine around the drum and attach 8 ounces of weight to the free end.



### TICK TOCK NOW

Gently release the weight, and the crank will spin around within the escapement at regular, even intervals. Your clock is now ticking! If not, check the alignment or add more weight. Visit the project page at [makezine.com/go/huygens-](https://makezine.com/go/huygens-)

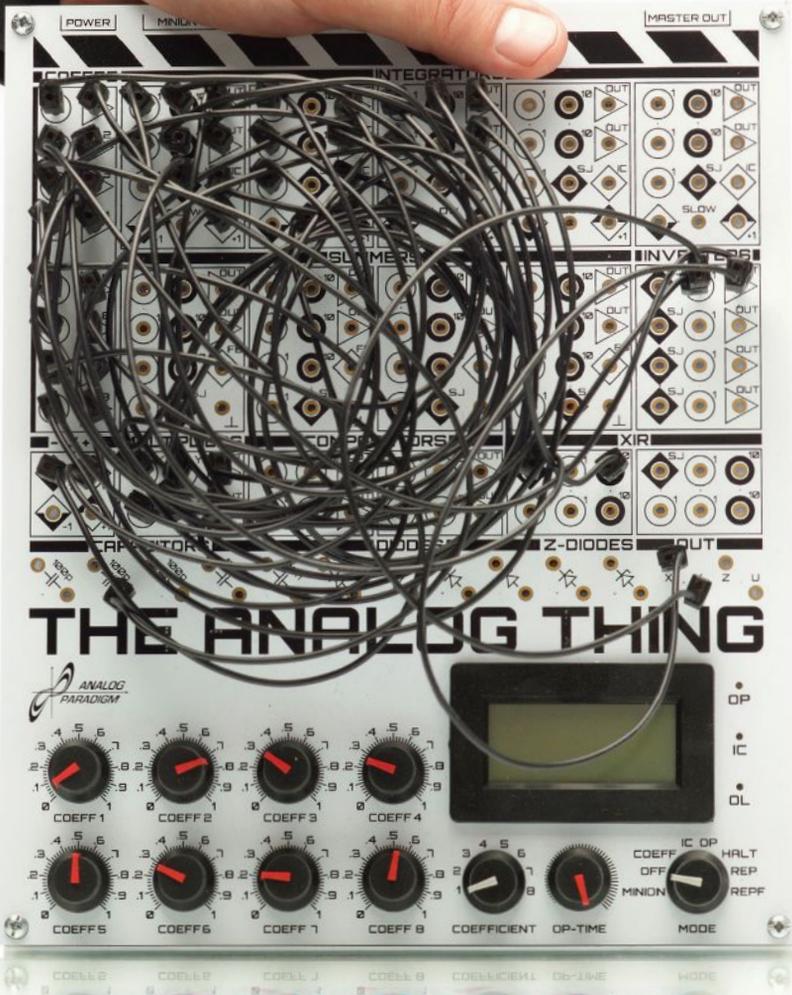
[pendulum](#) to see a video of how the single pin escapement looks when it's ticking.

You can experiment by adding gears and hands and a clock face. If you want to make it tick slower, add weight to the bottom of the pendulum. 🚫

# Analog Reborn

Before the digital revolution there was analog computing — and it's making a comeback

Written and illustrated by Charles Platt



**CHARLES PLATT** is the author of the bestselling *Make: Electronics*, its sequel *Make: More Electronics*, the *Encyclopedia of Electronic Components Volumes 1–3*, *Make: Tools*, and *Make: Easy Electronics*. [makershed.com/platt](http://makershed.com/platt)

In the early 1950s, aircraft development created an urgent need for simulations, and analog computers were ideally suited to run flight simulators.

Digital computers barely existed in those days, but aircraft dynamics could be modeled by the flow of electric current through potentiometers, amplifiers, and capacitors. Such circuits were analogous to the real world, and so they became known as *analog computers*.

You could describe almost all aspects of aircraft performance with differential equations, and the terms in these equations were represented by chaining together analog modules using patch cords on a plug board. This created a formidable tangle of wires, but once you got it right, the output on an oscilloscope screen was immediate, with no processing or programming necessary. Figure A shows an installation at the NASA Glenn Research Center in 1956. (Note the plug boards.)

As late as 1963, analog computation was still being used to create a cockpit simulation of a space vehicle docking in orbit. But microchips promised greater accuracy with much less maintenance, and they became less expensive. By the mid-1970s, analog was obsolete.

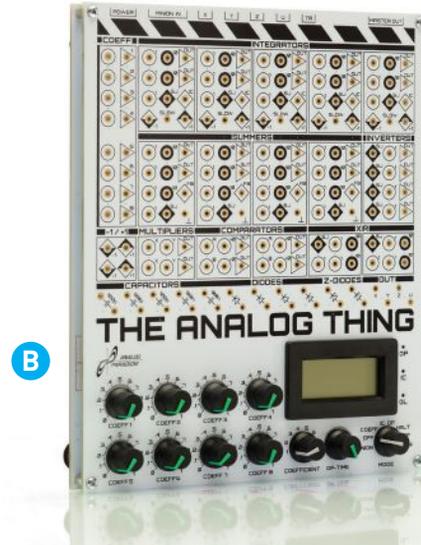
Or was it?

In 2020, Professor Doctor Bernd Ulmann cofounded Anabrid GmbH in Frankfurt, Germany, to develop chip-based analog computers. The company also introduced an educational product named The Analog Thing ([the-analog-thing.org](http://the-analog-thing.org)), abbreviated as THAT. Digital computers running complex simulations had become increasingly power-hungry and plagued with problems such as heat dissipation, prompting Ulmann and his colleagues to foresee a new role for analog.

THAT is now available to buyers in the United States for less than \$350, including shipping (see Figure B). "Our goal is to bring the idea of analog computing back to the world," Ulmann states. "We have hobbyists, musicians (controlling analog synthesizers), many students, sometimes large companies. There is a THAT group at Facebook where users ask questions and post their computer setups. They have simulated events such as a bungee jump with a quite realistic rubber rope, and gravitational waves,



Two EAI PACE analog computers back in the day, 1956.



The Analog Thing, also known as THAT, 2022.

neurons, many more."

Ulmann claims that seriously challenging tasks, such as creating an epidemiological model, "can be done easily" with THAT. And if a problem gets too big, one THAT can be daisy-chained with another. The key is to express a phenomenon as a series of mathematical terms, each represented by a computing element. Connect them together, and the input flows directly to the output.

THAT is open-source, because Ulmann wants to encourage innovation and analog literacy, although he sees THAT being integrated with digital systems. "We have a little application example showing how an Arduino can be used as the digital part of a hybrid computer setup," he says. "The community has already come up with a data logger and an Arduino-based oscilloscope." And to spread the gospel, he has written a book titled *Analog and Hybrid Computer Programming*.

Meanwhile, you can build your own basic analog computing device for around \$5 (see the following project) to understand how they work.

# HANDS-ON ANALOG

## Build the world's easiest analog computer

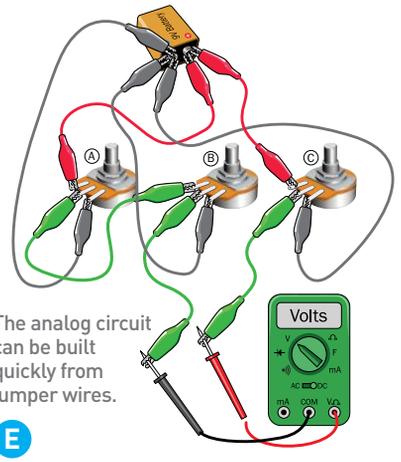
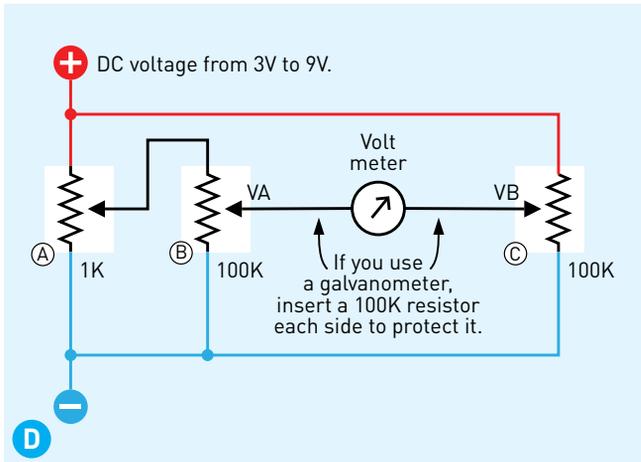
**TIME REQUIRED:** 1–2 Hours

**DIFFICULTY:** Easy

**COST:** \$5–\$15

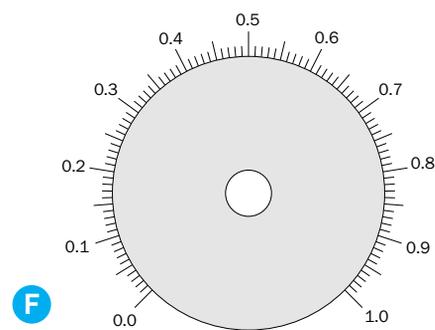
### MATERIALS

- » Potentiometers (3)
- » Galvanometer (optional)
- » Alligator jumper wires
- » 9V battery
- » Multimeter



The analog circuit can be built quickly from jumper wires.

A minimal analog circuit for multiplying two numbers.



A dial for a potentiometer. Your dial design should be customized to match the actual rotation range of each potentiometer.



Deluxe version of EASY analog computing device, with solid wood case, textured ABS faceplate, and optional galvanometer. Shown multiplying 0.3 by 0.7.

**The best way to understand an analog computer is to build one**, which you can do on a very limited scale using only three potentiometers and a multimeter. This little project is derived from the EF-140 kit marketed in 1961 by General Electric (see Figure C). Because analog computers always seem to have names consisting of acronyms, I'm calling my version EASY, for Elementary Analog SYstem.

The schematic for EASY is shown in Figure D, and indeed it's so easy, you can assemble it with alligator jumper wires, as in Figure E. You do need to put in a bit of extra work, though, by mounting pointers on the potentiometers and adding dials like the one in Figure F. Put a blank circle of paper on the front of each potentiometer and use your meter to measure the resistance between the wiper and the left terminal, while you turn the shaft and mark 10 equal intervals between the ends of the range.

How do you use EASY? Well, it's — simple. Here's a quick demo for doing multiplication: First move the wiper of Pot A to a position such as 0.3 on the dial. If letter  $V$  represents the voltage of your supply, the wiper of Pot A will tap a voltage of  $0.3 * V$ . This is passed along to the top of Pot B, so if you set its wiper to a position labeled 0.7, the voltage there will be  $0.7 * 0.3 * V$  — right?

Turn Pot C till there is no difference in voltage between points VA and VB in the schematic, measured with your meter (although I used a \$9 galvanometer to get that authentic analog look, as in Figure G). Pot C now points to 0.21 on its

scale. Yes, you just proved that  $0.7 * 0.3 = 0.21$  (approximately). The answer won't be precise, because Pot B steals a bit of current from Pot A. To minimize this, I used a 100K value for Pot B while Pot A is only 1K.

You can learn some general lessons from your EASY demo. Analog components must be accurately made, inputs are prone to error, results are always approximate, and if you have more than a couple of modules in your computer, voltage amplifiers will be necessary.

Still, the output is immediate and your three-pot computer is versatile. You can use it for division: Set a number on Pot C, divide it by a number on Pot B, and read the result off Pot A when you zero the voltage between VA and VB. You can even do a square root: Set a number on Pot C, then turn pots A and B till their values are the same while the voltage between VA and VB is zero. This is another feature of analog computation: Changing the sequence of operations enables you to process different formulae.

If you want to know how to build serious analog modules, you can find basic circuits in "A Practical Approach to Analog Computers," archived by The Analog Museum at [analogmuseum.org/library/eaiaapproach.pdf](http://analogmuseum.org/library/eaiaapproach.pdf).

Or, you can hunt for vintage kits such as the Heathkit ES-400 (shown in Figure H), which used 73 vacuum tubes for its op-amps. Supposedly, it weighed 168lbs, so if you find one on eBay, don't expect free shipping. 🚚

This Heathkit ES-400 analog computer was beautifully restored by David Goodsell. Photo reprinted by permission of *Nuts & Volts* magazine, [nutsvolts.com](http://nutsvolts.com).



H

# Bubble Plane!



Written and photographed by Bob Knetzger

## Rock crushes scissors, scissors cut paper — but bubbles rule them all

**TIME REQUIRED:** 1-2 Hours

**DIFFICULTY:** Easy

**COST:** \$5-\$10

### MATERIALS

- » Acrylic sheet, 1/8" thick
- » Paper
- » Bubble solution

### TOOLS

- » Laser cutter

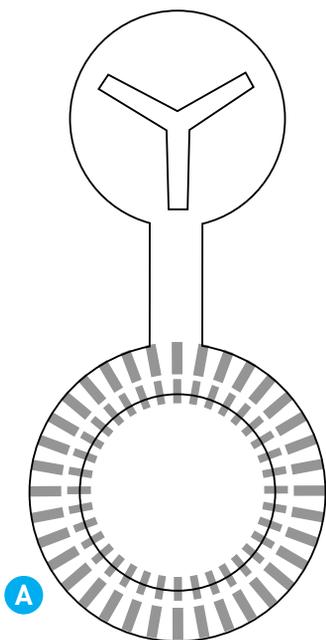


**BOB KNETZGER** is a designer/inventor/musician whose award-winning toys have been featured on *The Tonight Show*, *Nightline*, and *Good Morning America*. He is the author of *Make: Fun!*, available at [makershed.com](http://makershed.com) and fine bookstores.

Here's a super simple mini-toy: a plastic bubble tip for a paper airplane. Dip it in bubble solution and give it a toss. It leaves a puff of bubbles as the plane darts away. Reminds me of the miniature plastic prizes that used to be in boxes of Cracker Jack or vending machines: tiny toy, big fun!

Go to [makezine.com/go/bubble-plane](http://makezine.com/go/bubble-plane) to get the .svg vector graphic file to cut the bubble tip on your laser cutter (Figure A).

You could make this small part by manually cutting and drilling, but the ability to laser-engage the radial ridges is really super neat and easy. The ridges increase the surface area of the bubble ring to hold more soap solution.



**TIP:** Adjust the speed and power settings for the engraving to go a little less than halfway through the  $\frac{1}{8}$ " acrylic. Use a strong magnet or tape to hold the material firmly to the laser cutter bed.

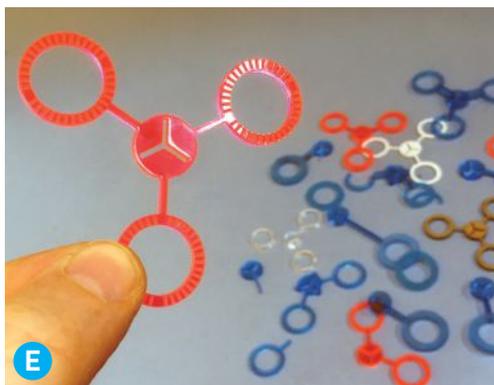
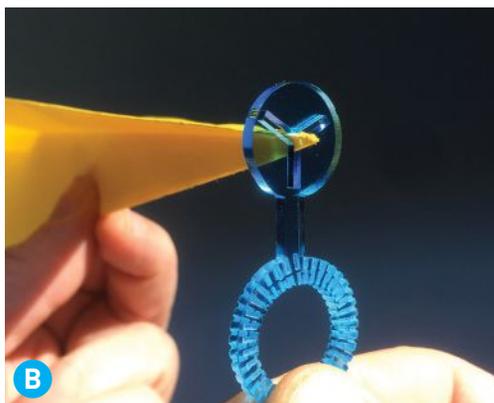
After the first engrave and cut, carefully pick out the part, flip it over, and re-insert it right back into the material blank in the same position. Laser it again, but this time only do the engraving. (The Glowforge I used has a handy Ignore setting to skip the re-cutting.) This will engrave the ridges in the backside of the bubble ring as well. Double the ridges, double the bubbles!

Fold a conventional paper airplane and carefully insert the tip into the triangular T-slot (Figure B). Slide it on until snug.

Dip the plastic part in soap solution and toss for a bubbly flight (Figure C). Try throwing at different speeds and arcs for best effect.

### THANK YOU FOR FLYING AIR BUBBLE

I thought this would be a simple project, but it required many revisions of sizes (Figure D), rib layouts, and number of bubble loops (Figure E). This "final" design is really just a starting point. The symmetrical design can be modified or simply scaled up or down: see what works best for you. Good luck! 🍀



Your accessories

-  **Backpack**  
10.03.21, 20:13
-  **Bag**  
10.03.21, 11:44
-  **Car**  
11.03.21, 07:30
-  **Keys**  
10.03.21, 19:00
-  **Items**  
11.03.21, 05:18



# FIND MY: **DIY AIRTAG TRACKER**

**Understanding, (ab)using, and extending  
Apple's offline finding protocol** *Written by Fabian Bräunlein*

## MATERIALS

- » **ESP32-based development board** such as Huzzah32 Feather, Adafruit 3405
- » **USB power bank**
- » **Cable, USB to Micro-USB**
- OR, FOR A SMALLER FORM FACTOR:**
- » **nRF51822-based tracker/"iBeacon"**  
Search "nRF51822 beacon" on AliExpress.
- » **Coin cell battery** for the tracker, e.g. CR2032 (if not included)
- » **SWD programmer** e.g. J-Link EDU Mini, Adafruit 3571

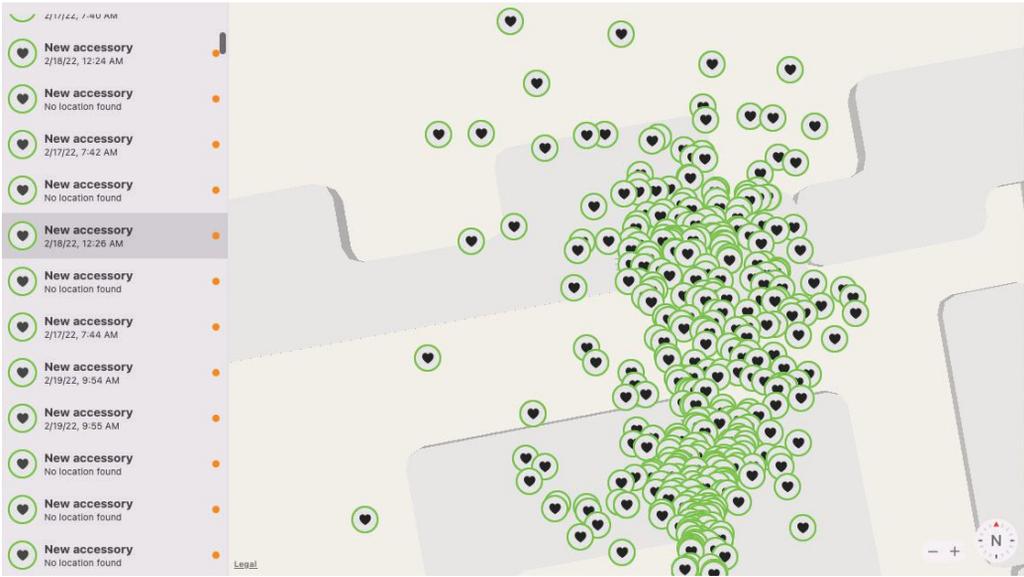
## TOOLS

- » **Computer with Mac OS X / macOS** You can also rent a cloud Mac from AWS or cheaper hosting companies, or run a virtual machine with macOS such as [github.com/sickcodes/Docker-OSX](https://github.com/sickcodes/Docker-OSX).

**Y**ou've probably heard of Apple's Find My network protocol for *offline finding*. Now implemented in over 1 billion devices, Find My has enabled Apple to introduce the AirTag, a low-cost, low-power location tracking beacon with worldwide coverage, but without the need for a GPS or cellular modem.

But did you know you can piggyback on the Find My network with your own tracker? And even transmit arbitrary data over the network? Or operate in a special "stealth" mode to track your belongings without alerting potential thieves?

This article will explain how the protocol works and explore how this ubiquitous network can be



piggybacked on — and even extended — with self-built AirTag clones.

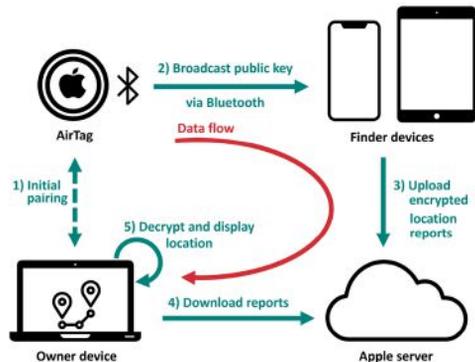
### How Does Find My Work?

When AirTags are not in proximity of their paired device, they constantly emit *Bluetooth Low Energy beacon* messages. Nearby Apple devices that receive those beacon signals recognize them as Find My broadcasts and upload their own location to Apple. The location reports are associated with the received broadcast and encrypted in a way that allows only the AirTag owner to decrypt the location, not even Apple themselves.

#### In more detail, the AirTag pairing and finding process works like this:

1. When pairing an AirTag with an Apple device, a *key pair* and a *shared secret* are generated. The shared secret and the public key are stored on the AirTag, but only the Apple device knows the corresponding private key.
2. Every 2 seconds, the AirTag sends a Bluetooth Low Energy broadcast with a public key as content, which changes periodically and is generated using the previously shared secret.
3. Nearby Apple devices recognize the Find My broadcast, retrieve their current location, encrypt the location with the broadcasted public key, and then upload the encrypted *location report*.

4. When searching for the AirTag, the paired Apple device generates a list of the rolling public keys that the AirTag has used in the last days and queries an Apple service for their *hashes*. The Apple backend returns the encrypted location reports for the requested public key hashes.
5. The Owner Device decrypts the location reports and shows an approximate location.



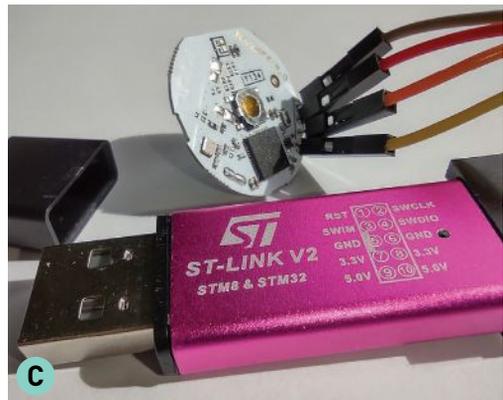
Luckily for hackers and makers, this design does not allow differentiating the broadcasts of legitimate Apple devices (or licensed third parties) from those of homemade clones. Furthermore, the Apple location retrieval backend does not (and cannot) check whether the user actually owns the AirTag they are requesting location reports for. It's a free ride, ripe for DIY experimentation.



A



B



C

Fabian Bräunlein, OpenHaystack

However, the encryption still guarantees that no user can extract the location from downloaded reports for devices that they themselves did not set up, as they would be missing the correct private key.

### Building an AirTag Clone Using OpenHaystack

All of this implies that an open-source implementation of the protocol is possible, which allows clone devices to piggyback on the Find My network — meaning that their location is forwarded by nearby Apple devices and can later be retrieved from Apple’s server and decrypted.

**OpenHaystack**, developed by the Secure Mobile Networking Lab of TU Darmstadt in Germany, is precisely such an open-source implementation and the result of extensive reverse engineering and analysis.

“OpenHaystack is an application that allows you to create your own accessories that are tracked by Apple’s Find My network. All you need is a Mac and a BBC micro:bit or any other Bluetooth-capable device,” the developers claim.

At the moment, you still need either a macOS computer or a virtual machine (and an Apple ID) to retrieve location reports. This is because the Apple backend requires extensive authentication data (based on an Apple ID), whose generation has not been reverse-engineered and reimplemented yet. For this reason, the retrieval app also includes a custom Apple Mail plugin that is used to fetch the required credentials.

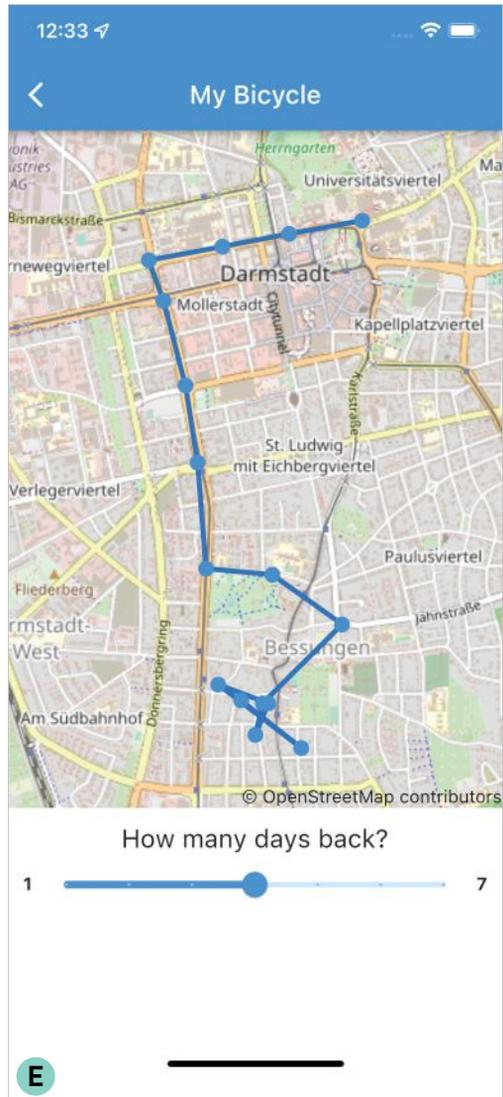
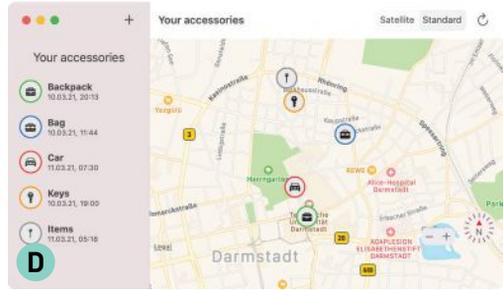
The project page at [github.com/seemoo-lab/openhaystack](https://github.com/seemoo-lab/openhaystack) includes detailed installation instructions and contains firmware compatible with ESP32 and nRF51822 microcontroller boards (it’s currently not possible to track actual AirTags using OpenHaystack).

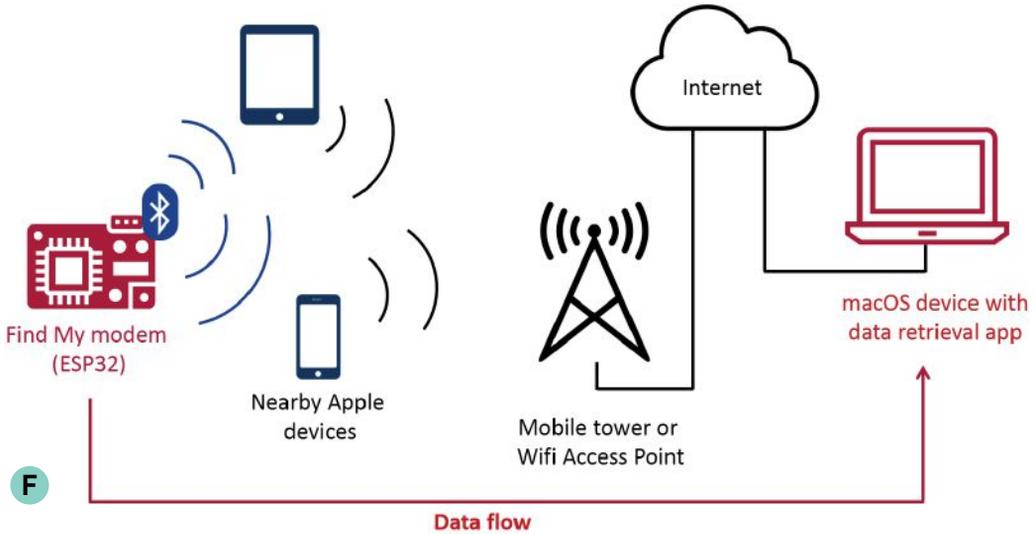
### This provides two appealing DIY tracker deployment options:

1. Using an ESP32, a power bank, and USB cable — all of which you might already have at home (Figure **A**).
2. A sleeker version using an nRF51822-based beacon and a small coin cell in case the battery is not included (Figure **B**). This version also requires the use of an SWD programmer to flash the firmware (Figure **C**).

After following the OpenHaystack installation instructions, flashing a tracker, and waiting a bit for the first location reports to arrive, the AirTag clone’s last location can be seen on a map in the OpenHaystack macOS app (Figure **D**).

The team has recently also released a mobile version of OpenHaystack for iOS and Android (Figure **E**), however it requires the user to build the app themselves and host an API backend running on a Mac.





Fabian Bräunlein

**Adding Features: Arbitrary Data Transmission**

I was curious whether Find My’s offline finding network could be (ab)used to upload arbitrary data to the internet, from devices that are not connected to Wi-Fi or mobile internet (Figure F). Such a technique could be employed by small sensors in uncontrolled environments to avoid the cost and power consumption of mobile internet. It could also be interesting for exfiltrating data from Faraday-shielded sites that are occasionally visited by iPhone users.

I found two options to accomplish this: The first relies on a 1-byte “status” field that is part of Find My broadcasts and forwarded as-is to the Apple backend where it can be retrieved again. This method has been implemented by Daniel Dakhno in the **FakeTag** project ([github.com/dakhnod/FakeTag](https://github.com/dakhnod/FakeTag)) to continuously transmit the state of a 6-bit counter (and 2 bits of battery level information).

The second option is more generic and would still work if Apple were to restrict the usage of the status byte (e.g. via an iOS update). The idea is that we can treat the Apple backend as something like a dead drop, or more precisely as a public *key-value store* with public key hashes as key, and encrypted location reports as value, with basic operations:

- We can probe whether location reports for a specific public key hash exist or not
- We can add location reports for a specific public key hash by broadcasting the corresponding public key

I guess you can already see where this is going: We can set arbitrary bits in the shared key-value store and query them again. If both the sender and receiver agree on an encoding scheme, we can transfer arbitrary data.

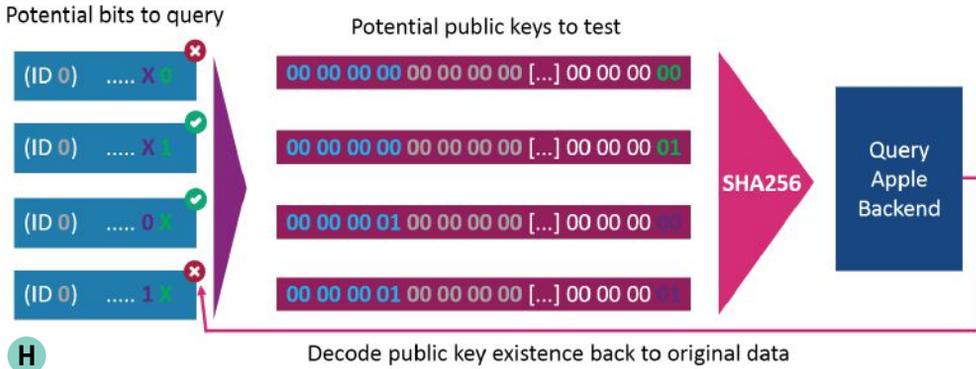
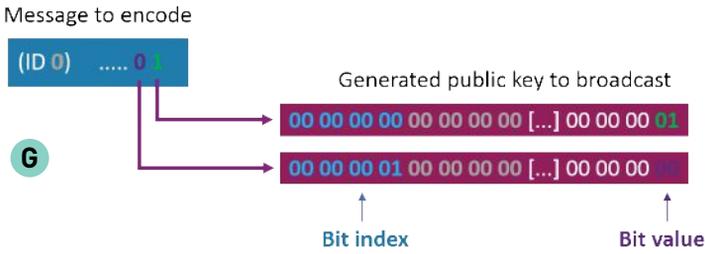
Because there’s no guarantee as to when or whether specific broadcasts are uploaded to the Apple backend as location reports, our data encoding must be independent of the ordering in which location reports are received, and able to recover partial data streams.

To achieve this, I decided to encode a single bit of data per broadcast together with an index value indicating which bit of the message is being set. Additional message and modem ID fields allow the system to be reused for multiple messages and by multiple users.

For sending a specific bit, we create a 28-byte array of the form:

```
[4b bit index] [4b message ID] [4b modem ID] [padding 0s...] [bit value]
```

and treat this as the public key in order to send BLE advertisements to broadcast, for example,



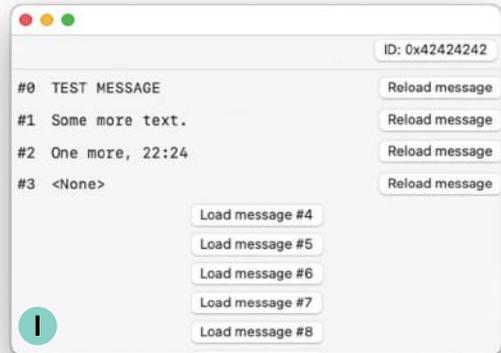
the information “bit 0 of message 0 is 1.”

To send a full message, the program simply loops over its bits and sends out one advertisement per bit with the public key that encodes its index and value (Figure G).

When fetching data, the receiving application will generate the same 28-byte arrays (two per bit, for the possible bit values 0 and 1) and query the Apple service with the hashes of those “public keys.” There should be location reports for only one of the two key IDs, which can then be interpreted; for example, “bit at index 0 equals 1” (Figure H).

Instead of only transferring one bit per message, we could also send a full byte by setting the last 8 bits of the public key. While this increases the sending bandwidth, we now need to request 255 different key IDs to fetch/“brute force” one byte (compared to 16 when encoded bit-by-bit).

I implemented this technique in **Send My** ([github.com/positive-security/send-my](https://github.com/positive-security/send-my)), a modified version of OpenHaystack that turns an ESP32 into a serial (upload only) modem and includes a DataFetcher application (Figure I) to retrieve sent messages for different modems.



## Stealth Mode

Apple added a feature to **iOS** to detect unknown AirTags traveling with the user and warn of them, with the goal of preventing the tracking of other people or their belongings. After numerous news reports of AirTags being used to stalk women or being placed in expensive cars to be stolen when they arrived at the owner’s home, Apple also issued an update for AirTags and iOS to reduce the time until those warnings are triggered to play a sound on the AirTag.

For Android, Apple released the **TrackerDetect** app, though it does not run in the background and

requires the user to perform an “active scan” via the app while being tracked. **AirGuard**, released by the OpenHaystack team, is an alternative app that also supports continuous scanning in the background.

As a side effect of trying to prevent this misuse, AirTags also lose their appeal for recovering stolen items, as thieves will also be notified of an AirTag and can even trigger sounds to locate and remove it.

Both Apple’s and AirGuard’s detection methods rely on the fact that an unmodified AirTag is only changing its public key once per day and can therefore be “tracked” by nearby devices for a limited time. We can bypass their detection by rotating over many public keys and only sending one broadcast per public key (or waiting long enough until repeating one key). This basically emulates thousands of different AirTags and, to a detection app, makes carrying this tracker almost indistinguishable from going through a busy area with many different AirTags quickly passing by, which should not trigger an alert.

I created such a “stealth” AirTag clone and confirmed it working in a real-world experiment, where I tracked an iPhone user (with their consent of course) for over 5 days without them receiving any notification. The stealth tracker is also not detected by an active scan with Apple’s Tracker Detect app for Android (Figure J).

The modified firmware and a macOS retrieval application optimized to handle thousands of virtual trackers can be found in the **Find You** repository on our GitHub, [github.com/positive-security/find-you](https://github.com/positive-security/find-you).

### Potential Use Cases

The possibility to piggyback on the Find My offline finding network with AirTag clones enables many use cases that were infeasible or much more expensive before:

- **Adding loss/theft prevention to anything:** AirTags inform thieves of their presence by playing a sound and triggering tracking alerts. Once found, an AirTag can simply be removed and deactivated.

An AirTag clone would not have this problem and could stay hidden for longer. Devices that

### No Item Trackers Found



You can find an item tracker if it's outside of Bluetooth range from its owner's device. It may take up to 15 minutes after an item tracker is separated before you can find it.

Scan Again

J

**NOTE:** The data transmission and stealth mode modifications can in theory also be implemented in an actual Apple AirTag, by updating its firmware. Check out the paper “AirTag of the Clones: Shenanigans with Liberated Item Finders”: [github.com/seemoo-lab/airtag/blob/main/woot22-paper.pdf](https://github.com/seemoo-lab/airtag/blob/main/woot22-paper.pdf).

already have Bluetooth onboard (e.g. Bluetooth speakers or some 2FA devices) could simply also send out Find My broadcasts (in stealth mode) to make them locatable. For others, a small Bluetooth beacon could be embedded in the product itself (e.g. in a suitcase, purse, or e-bike battery).

After publishing the Find You research, we were also contacted by a security engineer who wanted to use such stealth trackers to fight the rising number of kidnappings of children in their area.

- **Industrial/large-scale usage:** The Find My app limits the number of AirTags to 16 per account, and the raw location reports are not exposed to the user. When using OpenHaystack, no such limits exist and it’s possible to have a fleet of thousands of low-cost trackers whose location reports can be further processed in an automated way. This could make it attractive, for example, for rental car companies to fit their cars with

such trackers, or for logistics companies to put one such tracker in every shipment for live-tracking of all their goods across various delivery subcontractors.

When considering the possibility of arbitrary data transmission, even more possibilities emerge:

- **Low-cost, low-power distributed sensors:**

Using one of the approaches outlined above, it's possible to upload sensor readings or any data from IoT devices without a broadband modem, SIM card, data plan, or Wi-Fi connectivity. Considering the fact that Amazon is running a network called Sidewalk, connecting Echo devices to achieve exactly this, there seems to be some demand for it.

One such implementation is the previously mentioned FakeTag mailbox sensor, which uses a vibration sensor glued to the flap to detect new mail and continuously transmits the current mail count via the Find My network (Figure **K**).

I heard from one person who considered using the technique for gathering sensor readings from a boat out in a harbor (e.g., for its bilge pump) and got contacted by a nonprofit organization that sees a “use for it in environmental, air quality, and microclimate modeling, collecting data from remote sensors.”

Since the Finding devices cache received broadcasts until they have an internet connection, the sensors can even transmit data from areas without mobile coverage as long as iPhone users, even just briefly, pass through Bluetooth range — about 50 meters depending on the environment, hardware, and transmission power.

- **Data exfiltration:** In the world of high-security networks, visitors' Apple devices might become feasible intermediaries to exfiltrate data from certain air-gapped systems or Faraday-caged rooms. Even where another connection to the outside exists, Find My can act as a covert channel that is less likely monitored than for instance a normal IP connection. Also note that newer iPhones at least remain findable — with Bluetooth, NFC, and UWB radios still running — even when the device is powered off.



## Conclusion

We've shown how to create AirTag clones that are compatible with Apple's Find My network and how those clones' capabilities can even surpass the original AirTag's. In particular, it's possible to send arbitrary data over the network and to bypass all of Apple's “anti-stalking features,” making the technology also appealing again in anti-theft (or anti-kidnapping) scenarios.

Both the possibility to use the Find My network with “unauthorized trackers” as well as the described weaknesses are inherent to the privacy-focused design of the system. One interesting trade-off lies in Apple wanting AirTags to be untrackable via Bluetooth (to prevent a network of distributed Bluetooth receivers from tracking devices over a longer period) while relying exactly on this trackability for triggering reliable stalking warnings.

As in the current Find My design, Apple can't limit its usage to only genuine AirTags (and official partners' devices), they need to take into account threats of custom-made beacons (or AirTags with modified firmware) that might implement the Find My protocol in weird or malicious ways.

For hackers and makers, however, unless there's a major redesign of the Find My offline finding protocol, this ecosystem will likely stay open to be explored and tinkered with. 🛠️



Learn more about Find My and AirTags: [arxiv.org/pdf/2103.02282.pdf](https://arxiv.org/pdf/2103.02282.pdf) and [adamcatley.com/AirTag.html](https://adamcatley.com/AirTag.html)



**FABIAN BRÄUNLEIN** is an IT security researcher and co-founder of Positive Security ([positive.security](https://positive.security)). He has a thing for exploiting protocols and has uncovered weaknesses in systems from payment and booking to IP cameras and smart speakers.

# RAZBERRY RESCUE



Take control of smart home gadgets — and revive those zombie bulbs — with a **Z-Wave hat for Raspberry Pi**

Written by Sean Nolan

## TIME REQUIRED

1-2 Days

## DIFFICULTY

Moderate

## COST

\$75 for Pi + RaZberry; plus any Z-Wave devices

## MATERIALS

- » **Raspberry Pi mini computer** any model with header block
- » **RaZberry Z-Wave shield** for Raspberry Pi, from [z-wave.me/products/razberry](http://z-wave.me/products/razberry)
- » **Z-Wave smart lights** eBay is a great resource; search for “z-wave light bulb.”

It's hard not to get excited about the pace of “smart home” innovation. Things have advanced enough to support actually useful smart lights, vacuums, thermostats, cameras, washers, dryers, and a ton of other devices. But that same early energy means that companies enter and exit the market constantly, leaving a lot of our homes full of zombie gadgets. And even the working ones all seem to require a new account with a new password, maybe another hardware gateway, and probably a recurring subscription fee. Underneath a thin veneer of consistency offered by Google or Alexa, things are pretty crazy.

Mostly that's just part of the process. But it bugs me when companies bait-and-switch, adding recurring charges for functionality they sold as a one-time purchase. That happened with my Wink gateway last year, leaving me with a house full of no-longer-smart lights. Annoying — but fixing it gave me a good reason to dive into the underlying tech and generate some fun code.

## Z-What?

Most of the chaos around smart devices comes down to the network — how do they connect to the world? Increasingly this is just Wi-Fi, and that's probably where things will end up. But



**SEAN NOLAN** is a longtime software guy who lives on Whidbey Island in Washington State and spends his days building stuff with code and driftwood. Stop by and say hello at his blog, [shutdownhook.com](http://shutdownhook.com).

Wi-Fi components can be expensive and power-hungry, creating space for a bunch of different protocols. With the X10 protocol fading into history, there are really two that stand out: **Zigbee** and **Z-Wave**. This is very much a Coke vs. Pepsi kind of thing — there are a few differences, but for the most part both Z\* systems are the same thing solving the same problems:

- Both rely on a **gateway** to coordinate communication on the network. This is typically a dedicated hardware unit that keeps an inventory of devices, sends them commands, and receives status updates. The gateway is also the face of the Z\* network to the outside world — usually over Wi-Fi, using an embedded webserver, a cloud service, or both.
- Both are **mesh networks**, which is quite handy in the home environment. Z\* signals can only travel about 20–30 meters (65–100 feet), but each node acts as a relay to pass messages along to devices further away (Figure **A**). Because of the way they're evenly spread throughout a house, lighting devices are particularly well-suited to this kind of network.

The differences aren't super important. Zigbee enables more hops between nodes and more devices on the network; Z-Wave has a longer reach between nodes and is cheaper. Amazon's Echo Plus has a built-in Zigbee gateway, so that's nice. But both are perfectly serviceable; just make sure you're buying devices that match the technology in your gateway! There are a few that contain chips for both protocols if you really want to run both.

### Make a RaZberry Gateway

Of course I didn't know much of this at first. My first step to recover my Wink-abandoned lights was to get out the ladder, climb up, and take a look. It turns out that most of my lights were manufactured by GoControl — pretty neat units that slide right into an existing 6-inch ceiling can (Figure **B**). Most importantly, that little Z-Wave logo told me where to start!

A few hours of googling didn't really point at an "obvious choice" for a new Z-Wave gateway. The closest is probably Aeotec SmartThings hardware with open source Home Assistant control



software. That'd be a fine choice, but I wanted to get a little closer to the metal and control my destiny by building something myself. It turns out that the **RaZberry** daughter board for Raspberry Pi delivers excellent Z-Wave capability and is fully programmable. Perfect!

The RaZberry (Figure **C**) also comes with a license for Z-Way, a software stack that includes everything from a low-level C API up through a user-facing home control cloud service and user interface. Getting it all built out was my first order of business, and was pretty straightforward. You can find more details on the following steps in the Resources links at the end of the article.

### 1. SET UP THE RASPBERRY PI

I used a Raspberry Pi 3 Model B, but the RaZberry is compatible with all models that have the header block. As always, I used Pi Imager ([raspberrypi.com/software](https://raspberrypi.com/software)) to set up the SD card, but picked up two new tricks for a headless setup:

First, adding an empty file named `ssh` in the root of the SD card tells the OS to start the SSH daemon by default.

Second, the text file `wpa_supplicant.conf` in the root pre-configures the network. So much better than having to hook up a monitor and keyboard!

### 2. ATTACH THE RAZBERRY HAT

It covers a few pins at the top of the block but leaves plenty of room for other connections, like the GPIO pin I use to connect passive infrared sensors for motion detection.

Figure **D** shows my RaZberry 2, produced from 2018–2021, with fifth-gen ZM5101 Z-Wave module. The latest versions are the RaZberry 7 and the tiny 7 Pro with long-range external antenna; both use the new ZGM130S module.

### 3. INSTALL THE Z-WAY SOFTWARE

Install Z-Way by downloading and running the install script:

```
wget -q -O - https://storage.z-wave.me/RaspbianInstall | sudo bash
```

### 4. CONNECT IT TO THE INTERNET

Find the Pi's IP address (I use `ifconfig` | `grep inet`) and browse on over to <http://IPADDRESS:8083> on your local network to set up passwords and such. That's it! Your Pi is now running as the gateway node of a brand new Z-Wave network.

**NOTE:** RaZberry is manufactured by Z-Wave.Me, a Swiss-Russian company. The Russian branch received support from Skolkovo Foundation — a Russian government entity now under U.S. sanctions due to the Ukraine war. (Z-Wave.Me tells *Make*: that they 100% oppose the war and have cut ties with Skolkovo, relocating operations to Geneva this year.) By default RaZberry connects to Z-Wave.Me's cloud service to enable configuration backup and device access from outside your home network. The user manual describes how to turn all this off; it's a good idea to do your own research before installing anything inside your home network!



Sean Nolan

### Set Up Devices with Z-Way Interface

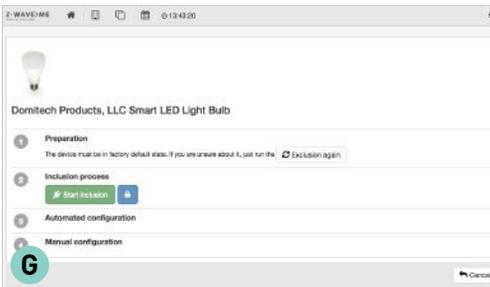
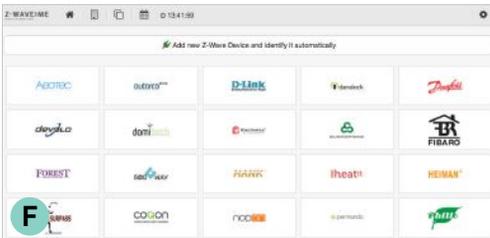
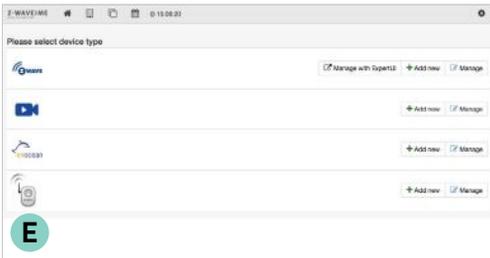
The web-based Z-Way Smart Home Interface provides a ton of solid functionality: device management, diagnostics, device control, automation of complex scenarios, integration with Alexa, and a lot more. You could 100% stop reading this article now and just use Z-Way as your smart home UX and be pretty happy. But the point of this whole exercise was to build it myself, so I'm only using the Z-Way app locally for network administration and troubleshooting. Most importantly I use it to "include" devices in the network.

This part is kind of like Bluetooth pairing. You tell the gateway to enter Inclusion mode, and then tell your device to do the same. When they each notice each other, an association is formed, and the device is registered as part of the network.

**1.** Make sure the device isn't already joined to another network. If you just bought it, all good. If not, dig out the manual and figure out how to reset it. For my GoControl lights, you do this by turning the power on and off four times — the lights blink twice to show they've been reset.

**2.** Put the gateway into Inclusion mode. From the Z-Way top-right menu, choose Devices. Next to Z-Wave in the top row (Figure **E**), choose "Add new" → "Add new Z-Wave Device and identify it automatically" (Figure **F**) → "Start inclusion" (Figure **G**).

**3.** Now put the device into Inclusion mode, according to the manufacturer's instructions. For my lights that just meant turning them off and on again. If all goes well, the gateway and the device



will see each other and live happily ever after. My lights acknowledge this by blinking twice. Newer devices may require a PIN to connect, which should be physically printed somewhere on the device itself.

Two little gotchas: First, each time you do this, only one device will be included. If you have a single switch that controls multiple smart bulbs, that's a little messy. Step 1 will reset all the bulbs at once, but you'll have to repeat steps 2 and 3 multiple times until all bulbs in the group have been included. Don't worry, we'll add some code later so you can easily turn the whole group on or off with one command. If you're starting from scratch, though, using a Z-Wave switch and regular bulbs may be simpler.

Second, inclusion (and exclusion too) can be quite sensitive to proximity; the gateway needs to be pretty close to the device you want to include. When I'm adding devices, I sometimes plug the gateway into a portable power station (such as

Amazon #B07T8XSWFN) so I can easily carry it around the house.

## Talking to the Gateway

OK, first let me say that there is a lot going on in the Z-Wave protocol, and about a million ways to dig in with code, using everything from C++ to JavaScript. I've added some links to various options at the end of the article, but because my development scenario is quite constrained — manage a bunch of dimmable lights — I've chosen the Z-Way VDev (virtual device) API.

VDev simplifies things by providing a normalized and simplified view of the device set; and a consistent set of REST commands across device types. These commands are exposed on port 8083:

**/devices** — returns a list of all devices on the network, including a mapping of friendly names to internal IDs used in control commands.

**/devices/ID** — returns the device status in a consistent JSON format.

**/devices/ID/update** — tells the device to report its current status back to the gateway.

**/devices/ID/on** — turns the device on (for some device types this may mean “do your thing” e.g., as the command to press a toggle button).

**/devices/ID/off** — turns the device off.

**/devices/ID/exact?level=#** — asks the device to apply an integer value from a range, if the device supports it. For example, this might be a 0–100 percentage value for a dimmer switch, or a temperature for a thermostat.

With the RaZberry installed on our Pi, devices added and an API selected, we are finally ready to write some code. I've used Java, but of course pretty much any environment can call those REST interfaces. You can download and build my code ([github.com/seanno/shutdownhook/tree/main/zwave](https://github.com/seanno/shutdownhook/tree/main/zwave)) on your Pi or anywhere that has Maven, Git, and Java installed:

```
sudo apt install git maven default-jdk
# if needed
git clone https://github.com/seanno/shutdownhook.git
cd shutdownhook/toolbox
mvn clean package install
cd ../zwave
mvn clean package
```

Next you'll need a configuration file in JSON format that looks like this:

```
{ "Login": "LOGIN", "Password":  
"PASSWORD", "BaseUrl": "http://  
localhost:8083" }
```

Assuming you'll run this code on the same Pi that is running Z-Way, just replace **LOGIN** and **PASSWORD** with the credentials you used when setting up Z-Way. If you want to run from another machine on your local network, also replace **localhost** with the IP address of your Z-Way Pi. (The version of this article on my blog describes how to run from outside your local network, if that's a requirement; a link is in the Resources section.)

Next, verify your build and configuration by running:

```
java -cp target/zwave-1.0-SNAPSHOT-jar-  
with-dependencies.jar \  
-Dconfig=PATHTOCONFIG \  
com.shutdownhook.zwave.App \  
devices
```

Remember to replace **PATHTOCONFIG** with a reference to your own file. If all goes well, you'll see a list of all the virtual devices attached to your Z-Wave network (name, type, and ID). Woo hoo!

The code that communicates with the gateway lives in *ZWay.java*, ready to use standalone in your own projects. It's pretty simple: provide a **ZWay.Config** and instantiate the object; use **getDevices** to enumerate the network; get status with **getLevel**; send commands with **turnOn / turnoff / setLevel**; remember to call when you're finished.

Of course, there are always some fun details under the covers. The class is a little lazy about authorization timeouts — tokens expire in a week, so we re-fetch them after six days, or each time a new object is created. This interval isn't guaranteed, so it's conceivable the strategy could fail at some point, but that seems unlikely. And do remember to call **close** on the object when you're shutting down — Z-Way remembers these tokens persistently, so if you forget you'll end up with orphan tokens.

Stale device data presents another wrinkle. Sending a Z-Wave command is basically fire-and-forget; some devices send back updated status, but many do not. And the ones that do, do

so asynchronously. So if you set a device value and then immediately query the gateway, almost certainly the data you get back will be stale. I tried to balance performance and hassle by addressing this in two ways:

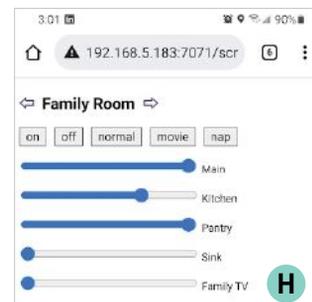
1. The configuration **UpdateOnCommand** (default **true**) causes every command to be followed by an explicit update to the affected device. If you don't need to reflect command changes immediately, this tends to keep the gateway values up-to-date pretty well with minimal chatter. Setting this to **false** makes the set operation a little more performant, but at the cost of more uncertainty about gateway values.
2. The status methods all take a **refresh(boolean)** parameter. If this value is **true**, you will (almost) always receive up-to-date values, but the call will be a little slower and result in a minimum of three network requests. In refresh mode we ask for the value (noting the update timestamp), request an explicit update, and then re-fetch the value until the update timestamp changes or we give up.

In most home scenarios, none of this is going to matter that much; you can refresh at will, and the default values will probably work A-OK. But it's always helpful to know what's going on below the waterline, so there you go.

### A Handy Web UX

Command-line access is great, but for everyday use an actual user interface is non-negotiable. Mine is just a simple web interface that works well on mobile screens (Figure H), pinned to my phone's home screen. Since it's running entirely on my local network, I've ignored login and wire encryption, although neither would be super complicated to add.

*Server.java* implements this web interface using three core concepts:



- **Screen** — a logical collection of devices, and could have reasonably been called a “Room” or “Location.” Each screen is displayed on its own web page and is associated with VLights and Settings.
- **VLight** — a collection of Z-Wave devices that are addressed together. For example, there are four smart ceiling lights in the family room that are controlled by a single switch and should always be on/off/dimmed together — these are collected into a single VLight.
- **Setting** — a list of VLights and values that together put lights into a useful configuration. E.g., the “Movie” setting dims the lights in the family room and turns off all of the lights on the periphery, while “Cooking” turns all the lights in the kitchen to their brightest levels.

All of this is described in a JSON configuration file defined by the **Server.Config** class. You can see a sample configuration in *example-server.json* that exposes the service on port 7071 (Figure 1).

Using the same binaries you built earlier, fire up the server with the following command, which starts it up in the background and saves any log output to **PATHOLOG**:

```
nohup java \
  -cp target/zwave-1.0-SNAPSHOT-jar-with-dependencies.jar \
  com.shutdownhook.zwave.Server
PATHTOCONFIG >> PATHTOLOG &
```

Each screen displays its Settings (plus “on” and “off” which do the obvious) as pushbuttons, and each VLight as a slider. Pressing a Settings button sets all of its VLights to the appropriate levels; sliding a slider sets the brightness of all the devices within that VLight (including 0 which turns the device off). No muss, no fuss, but extremely usable for my purposes. Voilà!

## What Next?

Remember that in my scenario all the devices are dimmable lights, so the UX will need some updates as I include other device types. And while using the phone to control the lights is great, it’s just a start. Check out the links below to learn how to voice-enable your gateway with Alexa and add motion detection to turn the lights on or off based on presence. That’s one of the best

```

"lights": {
  "Screens": [
    {
      "id": "family",
      "name": "family room",
      "Settings": [
        {
          "id": "normal",
          "name": "normal",
          "values": [
            { "VLightid": "family_main", "Level": 100 },
            { "VLightid": "kitchen_main", "Level": 70 },
            { "VLightid": "pantry", "Level": 100 },
            { "VLightid": "family_accent", "Level": 0 },
            { "VLightid": "kitchen_sink", "Level": 0 }
          ]
        },
        {
          "id": "movie",
          "name": "movie",
          "values": [
            { "VLightid": "family_main", "Level": 20 },
            { "VLightid": "kitchen_main", "Level": 1 },
            { "VLightid": "pantry", "Level": 20 },
            { "VLightid": "family_accent", "Level": 0 },
            { "VLightid": "kitchen_sink", "Level": 0 }
          ]
        },
        {
          "id": "hap",
          "name": "hap",
          "values": [
            { "VLightid": "family_main", "Level": 5 },
            { "VLightid": "kitchen_main", "Level": 0 },
            { "VLightid": "pantry", "Level": 5 },
            { "VLightid": "family_accent", "Level": 0 },
            { "VLightid": "kitchen_sink", "Level": 0 }
          ]
        }
      ]
    }
  ]
}

```

## RESOURCES:

**Project code from this article (and more):** [github.com/seanno/shutdownhook/tree/main/zwave](https://github.com/seanno/shutdownhook/tree/main/zwave)

### Article series on my blog, ShutdownHook:

- “Z-What? Rescuing my Z-Wave smart lights” [shutdownhook.com/2021/10/30](https://shutdownhook.com/2021/10/30)
- “Alexa, ask Bellevue House...” [shutdownhook.com/2021/11/04](https://shutdownhook.com/2021/11/04)
- “Can you see me now?” [shutdownhook.com/2021/11/11](https://shutdownhook.com/2021/11/11)

### RaZberry and Z-Way:

- RaZberry product information: [z-wave.me/products/razberry](https://z-wave.me/products/razberry)
- RaZberry installation and Z-Way manual: [z-wave.me/z-way/download-z-way](https://z-wave.me/z-way/download-z-way)

### Pre-configuring Wi-Fi on a headless Raspberry

**Pi:** [raspberrypi.com/documentation/computers/configuration.html](https://raspberrypi.com/documentation/computers/configuration.html)

### Z-Wave protocol and alternative development options:

- Z-Wave official site: [z-wave.com](https://z-wave.com)
- Silicon Labs SDK: [silabs.com/wireless/z-wave?tab=software](https://silabs.com/wireless/z-wave?tab=software)
- OpenZWave: [openzwave.com](https://openzwave.com)
- Z-Wave JS: [github.com/zwave-js](https://github.com/zwave-js)

things about building for yourself; once you know how things work under the covers, tweaking them to fit your unique needs becomes pretty straightforward. Making is magic! 🪄



### Flipper Zero \$169 [shop.flipperzero.one](http://shop.flipperzero.one)

The year is 1996. You idly tend to your Tamagotchi while watching the cult movie *Hackers* on VHS. What you don't realize is that your little Kuchipatchi is watching too, and taking it all in. And all that time since that it was sitting in your junk drawer, it was evolving, generation after generation. 2022 rolls around, and implausibly, you hear strange electronic dolphin noises emanating from the drawer; you open it to reveal: Flipper Zero — the open-source multitool for hackers!

The Flipper Zero is basically a collection of radios and other tools for interfacing with our electromagnetic environment, wrapped up with hacker-oriented firmware in a delightfully designed package. It's a consumer-ready device rather than a kit to assemble, and accordingly provides a premium unboxing experience. Once it's powered on, you are greeted by your own personal Cyber Dolphin, which even has a unique name (mine is "Linpoin"), and cute animations

that really enhance the usage of what might otherwise be a rather dry technical tool.

Uses for the Flipper Zero range from innocuous manipulation of televisions via infrared and cloning of (your own!) RFID and NFC access and payment cards (or say, Amiibo!) to more advanced pursuits such as opening (your own!) Tesla's charging port or executing arbitrary commands over USB HID when physical access to (your own!) target machine is possible.

While the Zero's firmware is still somewhat nascent, updates are frequent and easily applied via desktop or mobile apps. And since the software is open source, and the device exposes GPIO pins along its top edge, the possibilities for expansion and customization are almost endless. If you're looking to get into RF hacking, or just want to annoy your family by turning off TVs and opening garage doors, the Flipper Zero is a delightful, approachable device with practically limitless potential. —David Groom



## Portable RGB Video/ Photography Lights

**\$13-\$29** [amazon.com](https://www.amazon.com)

Buying lighting for YouTube videos and product photography can be daunting: There are so many options, and prices typically are rather high. These little RGB panel lights really caught my attention with their sub-\$15 price tag. I bought Opiheyoo LVL R73, but similar units are sold under the Ulanzi, Simorr, and Lewinner brands.

When I dug into them, I found many features to really like. You can switch between RGB and cool/warm white settings, which is nice so you don't lose your usual white in between sessions. There's a display that shows brightness, color temp, and battery life. They fit in a hot shoe, but also have a shoe notch on top so they can be stacked, or you can mount a microphone on top. And finally, they have a strong magnet in the back so you can just slap them onto a magnetic surface and not worry about mounting them.

The battery life is decent at over 2 hours, and the color range is fantastic. They're not quite bright enough to be considered studio lighting but for a vlogger, or someone filming their DIY projects, they're kind of perfect. At these prices, you can go ahead and slap one or two of these onto your table saw to get a cool lighting effect for a shot. —Caleb Kraft

## GPX PJ308W 1080p Mini Projector

**\$89** [gpx.com](https://www.gpx.com), [walmart.com](https://www.walmart.com)

Projectors have always kind of lived in the world of fancy expensive things that you just usually can't justify purchasing. With LED light sources getting cheaper and more powerful all the time, that's changing. Now you can find affordable small projectors with long-lasting LED light sources for pretty much any job you can imagine.

The GPX PJ308 is very cheap as far as projectors go, at just under \$90 from Walmart. It has plenty of the features you'd expect, such as multiple HDMI inputs, a plain old VGA input, the ability to play some media directly off an SD card or USB, and a remote for easy adjustments. The projection size is advertised as 150" diagonal and in my use, the image was crisp at that size. Sound isn't particularly powerful, but again, at this price I wasn't expecting a whole lot. On the plus side, it has Bluetooth connectivity if you want to pair an external Bluetooth speaker.

This projector only pushes about 2,000 lumens, so you'll need to have a nicely reflective screen and a well darkened room if you want a great viewing experience. However, at this price, this seems like the perfect projector to embed in your projects that can operate at night or in a controlled environment. This is the kind of thing you won't mind sacrificing to a Pepper's ghost illusion in your front yard. —Caleb Kraft





### Keychron Q3 QMK Custom Mechanical Keyboard

**\$194 assembled with knob** [keychron.com](http://keychron.com)

Mechanical keyboards are an interesting topic. You'll find people out there who are fanatical in their selection and build of a keyboard, and you'll find even more people who are lackadaisical, simply typing on whatever happens to be available. If you work on a computer, I highly suggest you give a mechanical keyboard a try. The experience of using the device you actually touch more than anything else can be improved, and improved a lot.

The Keychron Q3 is a kit keyboard; you can buy a barebones version and complete it with your own keys, or choose all the build features you want and they'll ship it to you fully assembled. I picked a machined aluminum body, no number pad, and a nice dial for volume. My selection of keys was the Gateron G Pro Brown, allowing for a fantastic pressure and click under my finger, but not a huge obnoxious *clack* for every key press.

This thing feels rock solid on my desk. It doesn't shift under my palms, and I can't feel the internals flex at all. Frankly, I really enjoy typing on this thing and I've never been a mechanical keyboard fanatic. The RGB lighting is cute but has never been a huge selling point for me — but that volume knob is very nice. —Caleb Kraft

### Comgrow ROBO 3018 CNC Router Milling Machine with Optional 10W Laser

**\$399** [comgrow.com](http://comgrow.com)

For those wanting to play with a bit of CNC at home on small projects such as PCB making, engraving, and light milling in soft materials, the market for tiny CNC machines supplies many options. Most, however, come as kits with little-to-no additional frills. The Robo CNC from Comgrow brings frills aplenty to the table. It came mostly assembled, with included items like cute little dust shields to mount around it in order to keep your work surface clean, and a pendant controller that lets you to move the gantry and spindle around and even run jobs without having the computer there. This unit came with an optional 10W diode laser as well (the 48W spec in the online product listing comes from the full draw of the laser, not the output), which opens up even more possibilities.

In use, I found it to be acceptably strong for what it is. To put it simply, PCB, rubber, wood, and plastic will be fine for milling. Don't expect to be chewing through even soft metals like aluminum though.

—Caleb Kraft



Inventive  
cardboard  
building kits



**DAZZLINKS**

Shop at  
[www.DazzLinks.com](http://www.DazzLinks.com)



**Make:**  
**HOLIDAY  
GIFT GUIDE**

OUR PICKS FOR  
GREAT MAKER GIFTS

[make.co/gifts](http://make.co/gifts)



**Make:cast**

**EXPLORING  
IDEAS, TOOLS &  
PEOPLE BEHIND  
THE MAKER  
MOVEMENT**

Available on Spotify, Apple Podcasts,  
[makezine.com](http://makezine.com), and more

EVERY CONNECTION COUNTS



**THE ONLY LIMIT IS  
YOUR IMAGINATION  
WITH TE'S LEMBAS  
LTE/GNSS USB MODEM**

TE Connectivity's (TE) LEMBAS LTE/GNSS USB modem for LTE CAT4 network and GPS tracking is designed for use on a wide array of single board computers (SBCs) utilizing ARM processors (32-bit and 64-bit) and running Debian or Ubuntu based Linux OS. Plug-and-play features and 1-command initial set-up enables instant access to cellular connectivity and GPS tracking in an easy-to-use package on the Pod Group's Luner platform.



© 2022 TE Connectivity Corporation. All Rights Reserved. TE Connectivity, TE, TE connectivity logo and EVERY CONNECTION COUNTS are trademarks.

# MAKER FAIRE HANNOVER RETURNS

After a two-year hiatus because of Covid, Maker Faire returned in September to the Hannover Congress Centrum in northwestern Germany. For me this was the first Maker Faire outside the U.S. that I've attended since Covid began. Being back in-person, elbow to elbow, is just better for meeting makers and seeing what they do. Around 12,500 people attended Maker Faire Hannover, which included R2-D2s and cosplayers, robots of all kinds, and assorted families. —Dale Dougherty



FabLab Chemnitz constructed a geodesic dome around its exhibition space.



The mythical bird *Garuda*, by Matthias Vijvermann of Belgium, was a popular attraction outside. 🔥



Clemens Gross organizes "Mach Club" in Fulda. He wants kids to get a feel for what making is.



Inspired by *Lego Jeep*, the *Lego Volvo* was fun for kids of all ages (an old Heise company car put to good use).



In the Dark Gallery, *Pixels* by Jonas Vorwerk of the Netherlands were stackable blocks that changed color based on interaction.



This kinetic sculpture by Volker Wessendorf used a bicycle to power a fan that generated enough wind to lift this red face and bring it to life.



# YOU MAKE WE

## AudioMoth

v1 & v1.2  
14 Rounds  
23,000 devices backed

## ATMegaZero

32U4 & ESP32-S2  
2 Rounds  
258 devices backed

## OrangeCrab

ECP5  
1 Round  
364 devices backed

## LoFive

E310 & RISC-V MCU  
4 Rounds  
384 devices backed



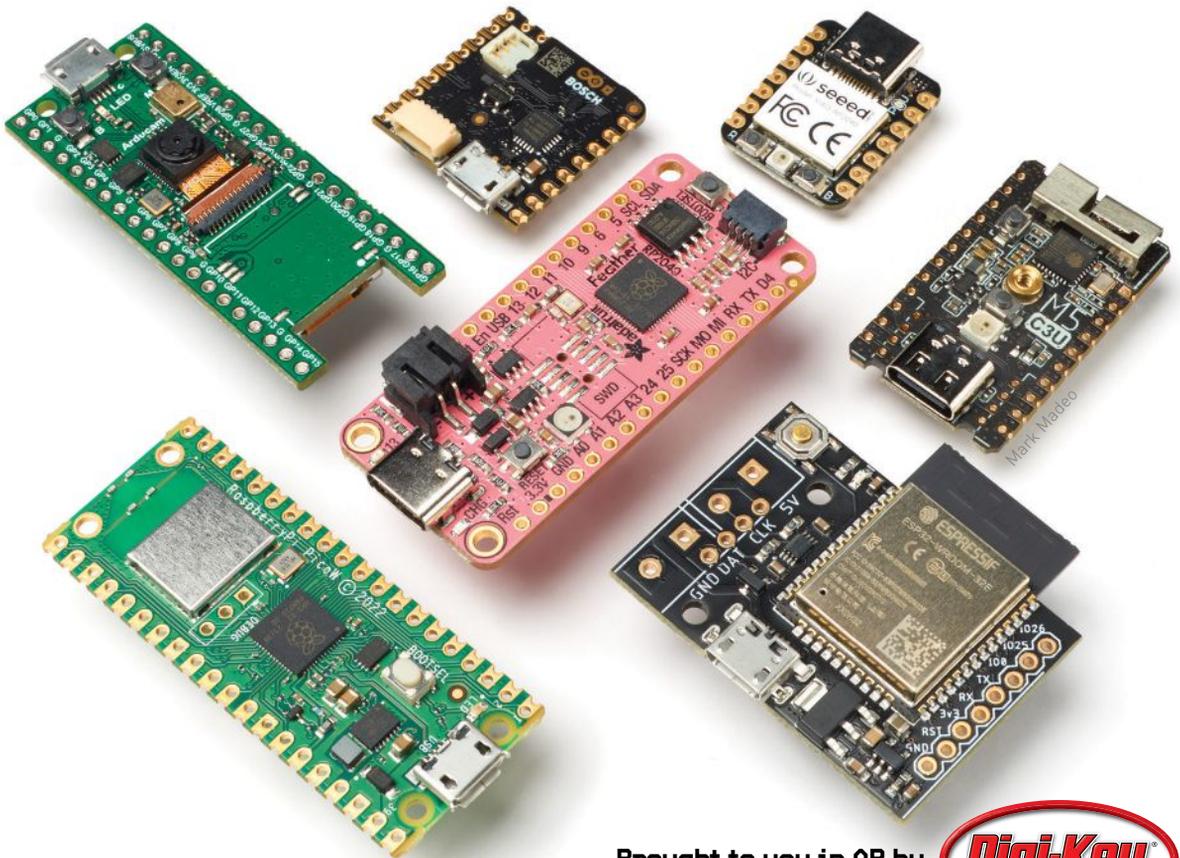
# THEY FUND

# Make:

THE ORIGINAL

# GUIDE TO BOARDS

2022



Mark Madeo

Brought to you in AR by



## History Repeating

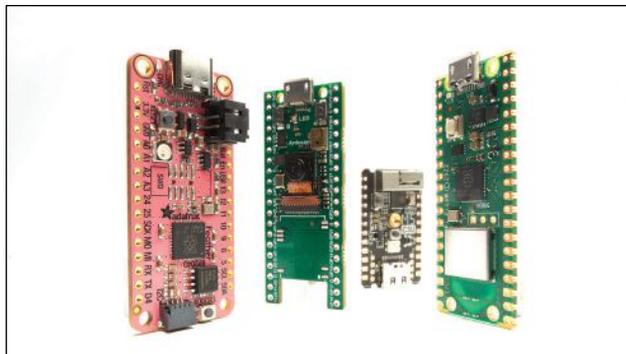
How many more times must we open with “It’s been another crazy year for the industry” before Harold Ramis’ estate comes after us for ripping off his 1993 hit *Groundhog Day*? Well, this will not be the year to buck that trend, as supply chain issues continue to force companies to get creative with sourcing in-demand chips, or designing boards for the components they can actually get, or both. This shift to supply-based design, and the widespread adoption of the RP2040 and ESP32 for maker projects, are just a few of the seismic changes that shaped this year’s *Make: Guide to Boards*. Read on for our comparison of 79 reliable stalwarts, hot newcomers, and smart picks that are actually in stock!



Scan the QR code to get the Digi-Key AR App and see the guide come to life in dynamic AUGMENTED REALITY!

## BOARD IN THE HOUSE

We had a blast this year going hands-on with boards, poring over specs, and talking to the community in order to come up with our list of 12 standout New & Notable boards for 2022. Find out more about how we made our picks — and what makes each board a top candidate for your next project!



Point your Digi-Key AR app at the image to the right to play the video

### MICROCONTROLLERS (MCU)

Board Name	Price	Dimensions	Software	Clock Speed	Processor	Memory
<b>Adafruit Circuit Playground Bluefruit</b>	\$25	2.0" dia.	Arduino IDE, CircuitPython	64MHz	32-bit Nordic nRF52840 (single-core Cortex-M4F)	1MB flash, 256kB RAM, 2MB QSPI Flash
<b>Adafruit CLUE</b>	\$45	2.0"x1.6"	Arduino IDE, CircuitPython	64MHz	32-bit Nordic nRF52840 (single-core Cortex-M4F)	1MB flash, 256kB RAM, 2MB QSPI flash
<b>Adafruit Feather RP2040</b> ★NEW & NOTABLE★	\$12	2.0"x0.9"x0.3"	CircuitPython, MicroPython, Arduino IDE, C/C++	125MHz	32-bit RP2040 (dual-core Cortex-M0+)	264kB SRAM, 8MB SPI flash
<b>Adafruit Flora</b>	\$15	1.75" dia.	Arduino IDE	8MHz	8-bit ATmega32u4	32kB flash, 2.5kB SRAM
<b>Adafruit FunHouse</b>	\$35	3.35"x2.2" x0.43"	CircuitPython, Arduino IDE	240MHz	32-bit ESP32-S2 (single-core Xtensa LX7)	4MB flash, 2MB PSRAM
<b>Adafruit KB2040</b>	\$9	1.4"x0.7"x0.2"	CircuitPython, MicroPython, Arduino IDE, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	264kB RAM, 8MB SPI flash
<b>Adafruit Metro Mini 328 U2</b>	\$15	1.7"x0.7"	Arduino IDE	16MHz	8-bit ATmega328 (single-core AVR)	2kB RAM, 32kB flash

## ...AND THEN THERE WERE TWO

### RP2040 AND ESP32 RISE ABOVE THE SHORTAGES

That the last few years have been difficult for the electronics industry is no secret. Shortages in a range of components, from processors to simple power-management ICs, have necessitated some rapid rethinking when it comes to designing development boards.

Last year we celebrated the launch of Raspberry Pi's RP2040, the company's first in-house silicon and first microcontroller. Little did we know how popular it would become: With Raspberry Pi working hard to ensure a steady supply in both reel and single-chip quantities and offering aggressive pricing, the RP2040 has become the go-to chip for those who've found stock of their usual microcontrollers wanting.

"We'd expected to pick up RP2040 customers in 2022, but that they'd still be at the prototyping and early production stage," Raspberry Pi CEO Eben Upton tells us. "There's certainly a lot of this going on — JLCPCB report that they've seen over 1,100 distinct RP2040-based designs in their first year of availability — but we're also seeing full-scale OEM orders, probably driven by people rapidly

substituting RP2040 into existing designs where the incumbent microcontroller is unavailable. We've shipped about five million units so far, and are shipping about 500,000 units a month at present."

Raspberry Pi isn't the only company riding the supply chain storm: Espressif Systems, long a staple of the maker market with its low-cost, radio-equipped ESP8266 and ESP32 families of systems-on-chips, has worked effectively to keep its parts in stock too — even launching variants specifically designed to boost yields and reduce per-chip costs, while simultaneously making the move to a whole new architecture (see "A RISC-V Revolution" on page 42 of the magazine).

We're not at the point of a Raspberry Pi-Espressif duopoly, but the shift in focus for many has been sudden and swift — and for those who've lost customers as a result of shortages, winning them back could prove a challenge.

—Gareth Halfacree



Digital Pins	Analog Pins	Radio	Video	Input Voltage	Battery Connection	Operating Voltage
8	8 PWM, 6 ADC	Bluetooth Low Energy	—	3.7V-5V	✓	3.3V
18	18 PWM, 8 ADC	Bluetooth Low Energy	1.3" 240x240 color IPS TFT LCD display	3V-6V	✓	3.3V
21	16 PWM, 4 ADC	—	—	3.3V-5V	✓	3.3V
8	4 PWM, 4 ADC	—	—	3.5V-16V	✓	3.3V
5	5 PWM, 3 ADC	Wi-Fi	1.54" 240x240 color TFT display	5V	—	3.3V
20	16 PWM, 4 ADC	—	—	3.3V-5V	—	3.3V
20	6 PWM, 6 ADC	—	—	5V-16V	—	5V

Adobe Stock-arthead

# Make: GUIDE TO BOARDS 2022

## MICROCONTROLLERS (MCU)

Board Name	Price	Dimensions	Software	Clock Speed	Processor	Memory
<b>Adafruit Neo Trinky</b>	\$7	1.2"×0.5"×0.1"	CircuitPython, Arduino IDE	48MHz	32-bit ATSAM21E18 (single-core Cortex-M0+)	256kB flash, 32kB RAM
<b>Adafruit QT Py ESP32-C3</b>	\$10	0.9"×0.7"	Arduino IDE, MicroPython, ESP-IDF	160MHz	32-bit ESP32-C3 (single-core RISC-V)	4MB flash, 400kB SRAM, 8kB SRAM (RTC)
<b>Adafruit QT Py RP2040</b>	\$10	0.9"×0.7"×0.2"	CircuitPython, MicroPython, C/C++	125MHz	32-bit RP2040 (dual-core Cortex-M0+)	264kB RAM, 8MB SPI flash
<b>Adafruit Trinky QT2040</b>	\$8	1.5"×0.7"×0.2"	CircuitPython, MicroPython, Arduino IDE, C/C++ SDK	125MHz	32-bit RP2040 (dual-core Cortex-M0+)	264kB RAM, 8MB SPI flash
<b>Arducam Pico4ML</b> ★NEW & NOTABLE★	\$26	0.9"×2.0"	MicroPython	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	2MB flash, 264kB RAM
<b>Arducam Pico4ML-BLE</b>	\$31	0.9"×2.0"	MicroPython	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	2MB flash, 264kB RAM
<b>Arduino MKR GSM 1400</b>	\$84	2.6"×1.0"	Arduino IDE	48MHz	32-bit ATSAM21 Cortex-M0+	256 kB flash, 32kB SRAM
<b>Arduino MKR WiFi 1010</b>	\$39	2.4"×1.0"	Arduino IDE	48MHz	32-bit ATSAM21 (single-core Cortex-M0+)	256kB flash, 32kB RAM
<b>Arduino Nano 33 BLE Sense</b>	\$41	1.8"×0.7"	Arduino IDE	64MHz	32-bit Nordic nRF52840 (single-core Cortex-M4F)	MB flash, 256kB RAM
<b>Arduino Nano Every</b>	\$12	1.8"×0.7"	Arduino IDE	20MHz	8-bit ATmega4809	48kB flash, 6kB RAM, 256B EEPROM
<b>Arduino Nano RP2040 Connect</b>	\$29	0.7"×1.8"	Arduino IDE, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	16MB flash, 264 kB RAM
<b>Arduino Nicla Sense ME</b> ★NEW & NOTABLE★	\$83	0.9"×0.9"	Arduino IDE	64MHz	32-bit nRF52832 (single-core Cortex-M4)	512kB flash, 2MB SPI flash, 64kB RAM
<b>Arduino Portenta H7</b>	\$114	2.6"×1.0"	Arduino IDE, MicroPython, JavaScript, TensorFlow Lite, Mbed OS	480MHz Cortex-M7, 240MHz Cortex-M4	32-bit STMicro STM32H747X1 (dual-core Cortex-M7, M4 coprocessor)	2MB/16MB Int/Ext flash, 1MB/8MB Int/Ext RAM
<b>Arduino Uno/Uno WiFi Rev2</b>	\$28 / \$54	2.7"×2.1"	Arduino IDE	16MHz	8-bit ATmega328PU/ ATmega4809	32kB flash, 2kB RAM, 1kB EEPROM/48kB flash, 6kB RAM, 256B EEPROM
<b>BBC micro:bit U2</b>	\$18	2"×1.6"	Javascript, MicroPython, CircuitPython, C++	64MHz	32-bit Nordic nRF52833 (single-core Cortex-M4F)	512kB flash, 128kB RAM
<b>DFRobot Beetle ESP32-C3</b>	\$8	0.98"×0.81"	Arduino IDE, ESP-IDF, MicroPython, C, Python	160MHz	32-bit ESP32-C3 (single-core RISC-V)	4MB flash, 400kB SRAM, 8kB SRAM (RTC)
<b>Espressif ESP32-S2 Saola-1</b>	\$9	2.22"×1.1"	Arduino IDE, CircuitPython, ESP IDF	240MHz	32-bit ESP32-S2 (single-core Xtensa LX7)	128kB flash, 320kB SRAM, 16kB SRAM (RTC)
<b>Espruino Pico</b>	\$31	1.3"×0.6"	Espruino JavaScript Interpreter	84MHz	32-bit STMicro STM32F401CDU6 (single-core Cortex-M4)	384kB flash, 96kB RAM
<b>M5Stamp C3U Mate</b> ★NEW & NOTABLE★	\$6	1.34"×0.79"	Arduino IDE, ESP-IDF, MicroPython, C, Python	160MHz	32-bit ESP32-C3 (single-core RISC-V)	4MB flash, 400kB SRAM, 8kB SRAM (RTC)

Digital Pins	Analog Pins	Radio	Video	Input Voltage	Battery Connection	Operating Voltage
0	0	—	—	5V	—	3.3V
13	6 PWM, 5 ADC	Wi-Fi, Bluetooth Low Energy	—	5V	—	3.3V
13	13 PWM, 4 ADC	—	—	3.3V-5V	—	3.3V
0	0	—	—	5V	—	3.3V
26	16 PWM, 3 ADC	—	0.96" 160x80 color LCD	5V-5.5V	—	3.3V
26	16 PWM, 3 ADC	Bluetooth Low Energy	0.96" 160x80 color LCD	5V-5.5V	—	3.3V
22	13 PWM, 7 ADC, 1 DA	GSM 1400	—	3.7V-5V	✓	3.3V
22	3 PWM, 7 ADC, 1 DAC	Wi-Fi, Bluetooth	—	3.7V-5V	✓	3.3V
22	5 PWM, 8 ADC	Bluetooth Low Energy	—	5V-21V	—	3.3V
22	5 PWM, 8 ADC	—	—	7V-21V	—	5V
22	20 PWM, 8 ADC	Wi-Fi, Bluetooth	—	5V-21V	—	3.3V
10	12 PWM, 2 ADC	Bluetooth Low Energy	—	5V	✓	1.8V-3.3V
22	10 PWM, 8 ADC, 2 DAC	Wi-Fi, Bluetooth	MIPI DSI host & MIPI D-PHY	3.7V-5V	✓	3.3V
14	6 PWM, 6 ADC/5 PWM, 6 ADC	Uno WiFi: Wi-Fi, Bluetooth Low Energy	—	6V-20V/7V-12V	—	5V
19	3 PWM, 6 ADC	Bluetooth	—	3V-5V	✓	3V-3.3V
13	6 PWM, 6 ADC	Wi-Fi, Bluetooth	—	5V	✓	3.3V
43	8 PWM, 20 ADC, 2 DAC	Wi-Fi	serial LCD, parallel LCD	3.7V-5V	—	3.3V
22	21 PWM, 9 ADC	—	—	3.5V-16V	✓	3.3V
14	6 PWM, 6 ADC	Wi-Fi	—	5V	—	3.3V

## New & Notable



### M5STACK M5STAMP C3U MATE ESP32-C3 and tiny!

The Lilliputian M5Stamp C3U features Espressif's latest 32-bit ESP32-C3 RISK-V MCU, running at 160kHz with 400kB of RAM and 4MB of flash, and includes Wi-Fi 5 and secure boot, all in a 34x20x4.6mm package!



### DFROBOT LATTEPANDA 3 DELTA Beast mode!

Give a docile panda a latte or two and you might activate beast mode, which is what DFRobot have done with the latest iteration of their Windows/Linux-capable SBC, featuring an Intel N5105, 8GB LPDDR4 RAM, 64GB eMMC, an ATmega32U4 coprocessor and a whopping 42 expandable interfaces!



### RASPBERRY PI PICO W Popular Pico goes Wi-Fi!

Last year Raspberry Pi rocked the industry with their \$4 (and plentiful) Pico. Two million units and a small \$2 price increase later, the Pico W adds 802.11n wireless networking, complete with a free commercial-use license for the lwIP stack.

# Make: GUIDE TO BOARDS 2022

## MICROCONTROLLERS (MCU)

Board Name	Price	Dimensions	Software	Clock Speed	Processor	Memory
<b>M5Stamp Pico Mate</b>	\$5	0.71"×0.94"×0.17"	Arduino IDE, MicroPython, UIFlow	240MHz	2-bit ESP32-PICO-D4 (dual-core Xtensa LX6)	4MB flash, 520kB SRAM, 8kB SRAM (RTC)
<b>M5StickC PLUS</b>	\$20	1.89"×0.95"×0.71"	Arduino IDE, MicroPython, UIFlow	240MHz	2-bit ESP32-PICO-D4 (dual-core Xtensa LX6)	4MB flash, 520kB SRAM, 8kB SRAM (RTC)
<b>Meadow F7v2</b>	\$50	1.9"×0.9"	Meadow.OS	216MHz Cortex-M7, 240MHz ESP32	32-bit STM32F7 (single-core Cortex-M7), ESP32 coprocessor	64MB flash, 32MB RAM
<b>Nordic Thingy:91</b>	\$126	2.4"×2.4"	Nordic Thingy, Zephyr OS	64MHz	32-bit Nordic nRF9160 (single-core Cortex-M33)	1MB flash, 256kB RAM
<b>OpenMU Cam H7 R2</b>	\$85	1.77"×1.41"	MicroPython	480MHz	32-bit STMicro STM32H743V1 (single-core Cortex-M7)	2MB flash, 1MB RAM, microSD Card Slot
<b>Pimoroni PGA2040</b>	\$7	0.8"×0.8"	Arduino IDE, MicroPython, CircuitPython, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	8MB QSPI flash, 264kB RAM
<b>Pimoroni Tiny 2040</b>	\$7 (2MB), \$12 (8MB)	0.90"×0.72"	Arduino IDE, MicroPython, CircuitPython, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	8MB QSPI flash, 264kB RAM
<b>Pixelblaze U3 Standard</b> ★NEW & NOTABLE★	\$35	1.35"×1.56"	Arduino IDE, ESP-IDF, Pixelblaze Pattern Language	240MHz	32-bit ESP32 (dual-core Xtensa LX6)	4MB SPI flash, 520kB RAM (8kB in RTC)
<b>PJRC Teensy 4.0</b>	\$23	1.4"×0.7"	Arduino IDE with Teensyduino extension, CircuitPython	600MHz	32-bit NXP iMX RT1062 (single-core Cortex-M7)	2MB flash, 1MB RAM, 1kB EEPROM (Emulated)
<b>PJRC Teensy 4.1</b>	\$32	2.4"×0.7"	Arduino IDE with Teensyduino extension, CircuitPython	600MHz	32-bit NXP iMX RT1062 (single-core Cortex-M7)	8MB flash, 1MB RAM, 4kB EEPROM (Emulated)
<b>Pycom FiPy</b>	\$59	2.1"×0.7"	MicroPython	160MHz	32-bit ESP32 (dual-core Xtensa LX6)	8MB flash, 4MB RAM
<b>Raspberry Pi Pico</b>	\$4	2"×0.827"	Arduino IDE, MicroPython, CircuitPython, FreeRTOS, RT-Thread, Rust, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	2MB flash, 264kB RAM
<b>Raspberry Pi Pico W</b>	\$6	2"×0.827"	Arduino IDE, MicroPython, CircuitPython, FreeRTOS, RT-Thread, Rust, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	2MB flash, 264kB RAM
<b>Seeed Wio RP2040 Module</b>	\$7	0.71"×1.11"	Arduino IDE, MicroPython, CircuitPython, FreeRTOS, RT-Thread, Rust, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	2MB flash, 264kB RAM
<b>Seeed Wio Terminal</b>	\$37	2.83"×2.24"×0.47"	Arduino IDE, MicroPython, CircuitPython, ArduPy	120MHz	32-bit ATSAMD51 (single-core Cortex-M4F)	4MB SPI flash, 192kB RAM
<b>Seeed Xiao nRF52840</b>	\$10	0.8"×0.7"	Arduino IDE, MicroPython	64MHz	32-bit nRF52840 (single-core Cortex-M4F)	1MB flash, 2MB QSPI flash, 256kB RAM
<b>Seeed Xiao RP2040</b> ★NEW & NOTABLE★	\$5	0.8"×0.7"	Arduino IDE, MicroPython, CircuitPython	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	2MB flash, 264kB RAM
<b>Sipeed Maixduino</b>	\$28	2.7"×2.1"	MaixPy IDE, Arduino IDE, MicroPython, OpenMV IDE, PlatformIO IDE, FreeRTOS	400MHz	64-bit Sipeed M1 (dual-core RISC-V), KPU co-processor	16MB flash, 8MB SRAM
<b>Solder Party RP2040 Stamp</b>	\$12 (\$18.50 with carrier)	1"×1"	Arduino IDE, MicroPython, CircuitPython, FreeRTOS, RT-Thread, Rust, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	8MB flash, 264kB RAM

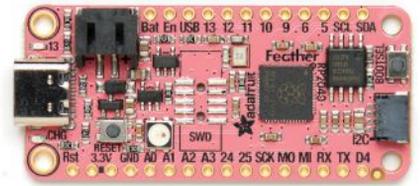
Digital Pins	Analog Pins	Radio	Video	Input Voltage	Battery Connection	Operating Voltage
12	2 PWM, 3 ADC, 2 DAC	Wi-Fi	—	5V	—	5V
3 (2 more via Grove)	1 ADC	Wi-Fi	0.96" 80x160 RGB LCD	5V	✓	3.3V
24	12 PWM, 6 ADC, 2 DAC	Wi-Fi, Bluetooth Low Energy	SPI	3.3V-12V	✓	3.3V (5V-tolerant digital IO)
10	4 PWM, 3 ADC	LTE-M, NB-IoT, Bluetooth Low Energy, NFC	—	5V	✓	3.3V
10	0 PWM, 1 ADC, 1 DAC	—	680x480 RGB camera	5V	✓	3.3V
30	16 PWM, 4 ADC	—	—	3V-5.5V	—	3.3V
12	12 PWM, 4 ADC	—	—	3V-5.5V	—	3.3V
12	5 ADC	Wi-Fi	—	5V	—	3.3V
40	31 PWM, 14 ADC	—	—	3.6V-5.5V	—	3.3V
55	35 PWM, 18 ADC	—	—	3.6V-5.5V	—	3.3V
22	18 PWM, 8 ADC, 2 DAC	Wi-Fi, Bluetooth Low Energy, CAT-M1/NB-IoT, LoRa, Sigfox	—	3.3V-5V	—	3.3V
26	16 PWM, 3 ADC	—	—	1.8V-5.5V	—	3.3V
26	16 PWM, 3 ADC	Wi-Fi, Bluetooth (Awaiting Firmware Support)	—	1.8V-5.5V	—	3.3V
20	16 PWM, 4 ADC	Wi-Fi	—	3.6V-5V	—	3.3V
26	5 PWM, 9 ADC	Wi-Fi, Bluetooth Low Energy	2.4" 320x240 color LCD	5V	—	3.3V
11	11 PWM, 6 ADC	Bluetooth Low Energy, NFC, Zigbee	—	5V	—	3.3V
11	11 PWM, 4 ADC	—	—	5V	—	3.3V
48	6 ADC	Wi-Fi, Bluetooth Low Energy	8-bit LCD interface	5V-12V	—	3.3V
30	16 PWM, 4 ADC	—	—	1.8V-5.5V	✓	3.3V

## New & Notable



### ARDUINO NICLA SENSE ME Tiny, accessible machine learning lab!

Arduino's Nicla series are diminutive boards (22.86mm square) with castellated edges for easy integration into other projects, and the Sense ME is packed with high-quality motion and environment sensors from Bosch, plus an edge AI-ready nRF52832.



### ADAFRUIT FEATHER RP2040 USB-C! Feather! RP2040! Pink!!

With four times the flash of the Pico, Adafruit's Feather spin on the RP2040 joins dozens of boards and "wings" with the familiar form factor, including such conveniences as LiPo charging, a Stemma QT/Qwiic connector, RGB NeoPixel, and USB-C for power, programming, and debugging.



### ARDUCAM PICO4ML Machine vision workshop for \$26!

Arducam's RP2040-based Pico4ML offers a complete tinyML lab for \$26, thanks to a built-in HiMax HM01B0 QVGA camera, microphone, 0.96" TFT display, and 9-axis IMU, all while consuming just 60mA during inferencing!

# Make: GUIDE TO BOARDS 2022

## MICROCONTROLLERS (MCU)

Board Name	Price	Dimensions	Software	Clock Speed	Processor	Memory
<b>Sony Spresense</b>	\$55	1.96"×0.81"	NuttX emulating Arduino IDE, CircuitPython	156MHz	32-bit Sony CXD5602 (six-core Cortex-M4F)	8MB flash, 1.5MB SRAM
<b>SparkFun MicroMod RP2040</b>	\$13	0.86"×0.86"	Arduino IDE, MicroPython, CircuitPython, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	16MB flash, 264kB SRAM
<b>SparkFun MicroMod Teensy</b>	\$22	0.86"×0.86"	Arduino IDE, C/C++	600MHz/1GHz Turbo	32-bit NXP iMX RT1062 (single-core Cortex-M7)	16MB flash, 1MB RAM
<b>SparkFun Pro Micro RP2040</b>	\$11	1.3"×0.7"	Arduino IDE, MicroPython, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	16MB flash, 264kB RAM
<b>SparkFun Thing Plus RP2040</b>	\$20	0.9"×2.3"	Arduino IDE, MicroPython, CircuitPython, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	16MB QSPI flash, 264kB RAM
<b>TinyLily Mini</b>	\$10	0.55" dia.	Arduino IDE	8MHz	8-bit ATmega328P	32kB flash, 2kB SRAM, 1kB EEPROM
<b>Unexpected Maker FeatherS2 Neo</b> ★NEW & NOTABLE★	\$20	0.9"×2.0"	Arduino IDE, MicroPython, CircuitPython, ESP-IDF	240MHz	32-bit ESP32-S2 (single-core Xtensa LX7)	4MB SPI flash, 320kB SRAM, 2MB PSRAM
<b>Wemos LOLIN C3 Mini</b>	\$4	1.35"×1"	Arduino IDE, MicroPython, ESP-IDF	160MHz	2-bit ESP32-C3 (single-core RISC-V)	4MB flash, 400kB SRAM, 8kB SRAM (RTC)
<b>WIZnet WizFi360-EUB-PICO</b>	\$7	0.84"×2.95"	Arduino IDE, MicroPython, CircuitPython, FreeRTOS, RT-Thread, Rust, C/C++	133MHz	32-bit RP2040 (dual-core Cortex-M0+)	2MB flash, 264kB RAM

## SINGLE-BOARD COMPUTERS (SBC)

Board Name	Price	Dimensions	Software	Clock Speed	Processor	Memory
<b>Arduino Portenta X8</b>	\$239 (\$574 with Max Carrier)	1"×2.6"	Yocto Linux	1.8GHz (Cortex-A53), 480MHz (Cortex-M7), 400MHz (Cortex-M4), 240MHz (Cortex-M4)	NXP i.MX8M Mini (quad-core Cortex-A53, Cortex-M4 coprocessor), STM32H747Xi (Cortex-M7, Cortex-M4)	2GB LPDDR4 RAM, 16GB eMMC
<b>Asus Tinker Edge T</b>	\$240	3.37"×2.125"	Debian 9, Android 10	1.5GHz	NXP i.MX 8M (quad-core Cortex-A53, Cortex-M4), Google Edge TPU coprocessor	1GB LPDDR4, 8GB eMMC
<b>BeagleBone AI-64</b>	\$214	3.86"×3.09"	Debian 11, Debian 10, Cloud 9 IDE	2GHz	64-bit TI Jacinto TDA4VM (dual-core Cortex-A72), PowerVR Rogue 8XE GPU, 6 ARM Cortex-R5, 12 PRU, 2 C66x coprocessors	4GB LPDDR4 RAM, 16GB eMMC
<b>DFRobot LattePanda 3 Delta</b> ★NEW & NOTABLE★	\$279	4.91"×3.07"	Windows 10, Windows 11, Linux	2GHz (2.9GHz boost)	64-bit Intel Celeron N5105 (quad-core x86-64), ATmega32U4 coprocessor	8GB LPDDR4 RAM, 64GB eMMC
<b>Hackboard 2</b>	\$199 (4GB), \$249 (8GB)	4.72"×3.15"	Microsoft Windows 10 Pro, Debian 9 (subtract \$24)	2.8GHz	64-bit Intel Celeron N4020 (dual-core x86-64)	4GB/8GB LPDDR4, 64GB eMMC
<b>Khadas UIM4</b> ★NEW & NOTABLE★	\$220	3.32"×2.28"	Ubuntu 22.04, Android 11	2.2GHz (Cortex-A73), 2GHz (Cortex-A53)	64-bit Amlogic A31D2 (quad-core Cortex-A73, quad-core Cortex-A53), STM32G031K6 coprocessor	8GB LPDDR4X, 32GB eMMC, 32MB SPI flash
<b>Mypi MYS-BMMX</b>	\$119 (commercial grade), \$139 (industrial grade)	3.74"×2.72"	Ubuntu 18.04, Yocto 3.0	1.8GHz/1.6GHz	64-bit NXP i.MX 8M Mini (quad-core Cortex-A53), Vivante GC320, Cortex-M4F coprocessors	2GB DDR4 RAM, 8GB eMMC, 32MB QSPI flash
<b>Nvidia Jetson AGX Orin Developer Kit</b>	\$1,999	4.3"×4.3"	Ubuntu-based JetPack SDK	2.2GHz	64-bit Nvidia CPU (12-core Cortex-A78AE), 2,048 CUDA Core 64 Tensor Core Ampere GPU, 2x NVDLA v2, Programmable Vision Accelerator V2 coprocessors	32GB LPDDR5, 64GB eMMC

Digital Pins	Analog Pins	Radio	Video	Input Voltage	Battery Connection	Operating Voltage
17; extension:14	2 ADC; extension: 6 PWM, 6 ADC	—	—	5V	—	1.8V
30	16 PWM, 3 ADC	—	—	3.3V	—	3.3V
12; 35 via shared MicroMod pins	6 PWM, 2 ADC (14 with signal sharing)	—	—	3.3V	—	3.3V
20-18 on the board edge and 2 through the Qwiic connector	10 PWM, 3 ADC	—	—	5V	—	3.3V
18	16 PWM, 3 ADC	—	—	3V-5.5V	✓	3.3V
8	1 PWM, 4 ADC	—	—	2.7V-5.5V	—	3V
22	22 PWM, 13 ADC, 2 DAC	Wi-Fi	—	3.7V-5V	✓	3.3V
12	6 PWM, 6 ADC	Wi-Fi, Bluetooth Low Energy	—	5V	—	3.3V
26	16 PWM, 3 ADC	Wi-Fi	—	1.8V-5.5V	—	3.3V

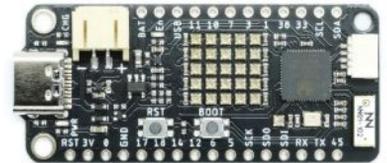
Digital Pins	Analog Pins	Radio	Video	Ethernet On Board	Input Voltage	Operating Voltage
22	4 PWM, 8 ADC	Wi-Fi, Bluetooth	—	—	5V	3.3V
28	3 PWM	Wi-Fi, Bluetooth	HDMI, MIPI DSI	✓	12V-19V	5V
72	7 PWM, 7 ADC	—	Mini-DisplayPort, MIPI DSI	✓	5V	3.3V
23	7 PWM, 12 ADC	Wi-Fi, Bluetooth	HDMI, DisplayPort, Embedded DisplayPort	✓	12V	3.3V
28	2 PWM	Wi-Fi, Bluetooth	HDMI, Embedded DisplayPort (eDP)	—	12V	GPIO 5V; 3.3V
5	1 PWM, 2 ADC	Wi-Fi, Bluetooth	HDMI, MIPI DSI, Embedded DisplayPort	✓	9V-20V	3.3V
30	3 PWM	Wi-Fi, Bluetooth	HDMI, LVDS LCD	✓	5V	5V
28	2 PWM	Wi-Fi, Bluetooth	DisplayPort	✓	19V (USB PD)	3.3V

## New & Notable



### ROCK PI 4 MODEL C+ Pi-compatible contender!

What may appear at first blush as just another Raspberry Pi is in fact Rock Pi's 64-bit Rockchip RK3399-powered SBC in a Pi-compatible form factor. The dual-core Cortex-A72 and quad-A53 provide a blend of power and efficiency for running Debian, Ubuntu, or Android, and onboard M.2 SSD and eMMC module support trump the Pi's mSD storage.



### UNEXPECTED MAKER FEATHERS2 NEO 5x5 RGB matrix onboard!

An RGB LED on an ESP32-S2-based Feather isn't cool. You know what's cool? 26 RGB LEDs on an ESP32-S2-based Feather!



### PIXELBLAZE U3 STANDARD Epicenter of RGB LED hype!

Command complicated RGB LED contraptions without coding! The Pixelblaze provides plentiful patterns, or write your own in JavaScript with the built-in web-based editor to drive up to 5,000 pixels!

# Make: GUIDE TO BOARDS 2022

## SINGLE-BOARD COMPUTERS (SBC)

Board Name	Price	Dimensions	Software	Clock Speed	Processor	Memory
<b>Nvidia Jetson Nano Dev Kit</b>	\$54 (2GB), \$99 (4GB)	3.95"×3.15"×1.14"	Ubuntu-based JetPack SDK	1.43GHz CPU, 921MHz GPU	64-bit Nvidia CPU (quad-core Cortex-A57), 128-CUDA-core Maxwell GPU coprocessor	2GB/4GB LPDDR4 RAM
<b>PINE64 Quartz64 Model B</b>	\$60 (4GB), \$80 (8GB)	3.35"×2.2"	Debian, Ubuntu, Manjaro, Arch, NetBSD	1.8GHz	64-bit Rockchip RK3566 (quad-core Cortex-A55), Mali-G52 2EE Bifrost GPU, Neural Processing Unit (NPU) and 32-bit RISC-V coprocessors	4GB/8GB LPDDR4, 64MB flash
<b>Raspberry Pi 4, Model B</b>	\$35 (1GB), \$45 (2GB), \$55 (4GB), \$75 (8GB)	3.4"×2.2"	Raspberry Pi OS, Raspbian, Ubuntu 21.04/22.04, RISC OS, Windows 10 IoT, more	1.5GHz CPU, 500MHz GPU	64-bit Broadcom BCM2711 (quad-core Cortex-A72), VideoCore V1 GPU	1GB/2GB/4GB/8GB LPDDR4 RAM
<b>Raspberry Pi Zero 2 W</b>	\$15	2.56"×1.18"	Raspberry Pi OS, Raspbian, Ubuntu 21.04/22.04, RISC OS, Windows 10 IoT, more	1GHz CPU, 400MHz GPU	64-bit Broadcom BCM2837 (quad-core Cortex-A53), VideoCore IV GPU	512MB LPDDR2 RAM
<b>Rock Pi 4 Model C+ ★NEW &amp; NOTABLE★</b>	\$69	3.35"×2.13"	Debian 10, Ubuntu 20.04, Android 7/9/10/11, OpenSuSE, more	2GHz Cortex-A72, 1.6GHz Cortex-A53	64-bit Rockchip RK3399 (dual-core Cortex-A72, quad-core Cortex-A53) CPU, Mali T860MP4 GPU	4GB LPDDR4
<b>Seeed reComputer J1010 ★NEW &amp; NOTABLE★</b>	\$199	5.12"×4.72"	Ubuntu-based JetPack SDK	1.43GHz	64-bit Carmel ARM CPU (quad-core Cortex-A57), 128-CUDA-core Maxwell GPU	4GB LPDDR4 RAM, 16GB eMMC
<b>StarFive VisionFive</b>	\$149 (\$179 as Starter Kit bundle)	3.94"×2.83"	Debian, Fedora, Ubuntu	1GHz	64-bit StarFive JH7100 (dual-core RV64GC), Tensilica VP6 DSP, NVDLA, Neural Network coprocessors	8GB LPDDR4
<b>StarFive VisionFive 2</b>	\$55 (2GB), \$65 (4GB), \$85 (8GB)	3.94"×2.83"	Debian, Fedora, Ubuntu	1.5GHz	64-bit StarFive JH7110 (quad-core RV64GC), Imagination BXE-4-32 GPU, Tensilica VP6 DSP, NVDLA, Neural Network coprocessors	2GB/4GB/8GB LPDDR4
<b>Udoo Bolt V3</b>	\$437	4.72"×4.72"	Windows 8.1/10, any Linux distribution	2GHz–3.2GHz	AMD Ryzen Embedded V1202b (dual-core quad-thread x86-64) CPU, Radeon Vega 3 GPU, ATmega32u4 coprocessor	None included, up to 32GB DDR4 RAM supported, 32GB eMMC

## FIELD-PROGRAMMABLE GATE ARRAY BOARDS (FPGA)

Board Name	Price	Dimensions	Software	Clock Speed	Processor	Memory
<b>Alorium Snō</b>	\$49	1.7"×0.7"	Arduino IDE	16/32MHz	Intel MAX 10 10M16SAU169C8G with single-core ATmega328 compatible microcontroller	32kB flash, 2kB SRAM, FPGA 68kB RAM, 16,000 logic cells
<b>Diligent Arty Z7-10</b>	\$199	3.46"×4.3"	Vivado, Xilinx SDK, Vitis	650MHz Cortex-A9, 125MHz external clock	Xilinx Zynq-7000 XC7Z010-1CLG400C with dual-core Cortex-A9 CPU	16MB QSPI flash, 512 MB DDR3 RAM, FPGA 270kB block RAM, 28,000 logic cells
<b>Fomu</b>	\$50	0.38"×0.5"	Yosys/Nextpnr	12MHz VexRISC-V, 48MHz external oscillator	Lattice iCE40UP5K with VexRISC-V core	2MB flash, 128kB SRAM, FPGA 5,280 logic cells
<b>MuseLab iCESugar-nano</b>	\$19	1.54"×0.71"	Yosys/Nextpnr	12MHz MCO (up to 72MHz)	Lattice iCE40LPIK-CM36	2MB flash, 8kB SRAM, FPGA 64kB RAM, 1,280 logic cells
<b>QWERTY ICE-V Wireless</b>	\$75	1.21"×3.17"	Integrated Loader Firmware, ESP-IDF (ESP32 only)	160MHz ESP32	Lattice ICE40UP5K, Espressif ESP32-C3 coprocessor	400kB SRAM, 4MB flash, FPGA 8MB PSRAM, 5,280 logic cells
<b>Seeed Spartan Edge Accelerator</b>	\$40	2.09"×2.7"	Vivado, Arduino IDE (ESP32 only)	100MHz	Xilinx Spartan-7 XC7S15-1FTGB196C FPGA, Espressif ESP32 coprocessor	4MB flash, FPGA 360kB block RAM, 12,800 logic cells
<b>Sipeed Lichee Tang Nano 4K</b>	\$17	2.36"×0.9"	Gowin Yunyuan IDE	100MHz	Gowin GWINSR-LV4C with single-core Cortex-A3 CPU	256kB flash, FPGA 180kB block RAM, 4,608 logic cells
<b>WebFPGA</b>	\$38	2.2"×0.9"	WebFPGA, IceStorm	10kHz and 48MHz internal oscillator, 16MHz external oscillator, 48MHz (STM32F04)	Lattice iCE40UP5k, STMicro STM32F04 coprocessor	32kB flash, 6kB RAM, 16MB external flash, FPGA 128kB SRAM, 15kB block RAM, 5,280 logic cells

Digital Pins	Analog Pins	Radio	Video	Ethernet On Board	Input Voltage	Operating Voltage
28	2 PWM	—	HDMI, DP (4GB only)	✓	5V	5V
28	—	Wi-Fi, Bluetooth	HDMI, MIPI DSI	✓	5V	3.3V
26	4 PWM	Wi-Fi, Bluetooth	2 Micro-HDMI, composite, MIPI DSI	✓	5V	3.3V
26	4 PWM	Wi-Fi, Bluetooth	Mini-HDMI	—	5V	3.3V
27	1 PWM, 1 ADC	Wi-Fi, Bluetooth	2 Micro-HDMI, MIPI DSI	✓	5V	3.3V
28	2 PWM	—	HDMI	✓	5V	3.3V
28	2 PWM	Wi-Fi, Bluetooth	HDMI	✓	5V	3.3V
28	2 PWM	Wi-Fi, Bluetooth	—	✓	5V	3.3V
10, 23 via ATmega32u4, 2 via Grove	7 PWM, 12 ADC, 1 ADC via Grove	—	2 HDMI, 2 USB-C DP	✓	19V	3.3V

## New & Notable



### SEED XIAO RP2040

**Fantastic form factor for little lure!**

Seeed's XIAO series — now totaling five boards — is quickly become a favorite form factor around here; this RP2040-based, ML-capable featherweight packs more features than Pi's own Pico, yet measures just 21×17.5mm, and can be yours for just over \$5 shipped!



### KHADAS VIM4

**Release the Khadas!**

This powerhouse features an Amlogic 2.2GHz quad-core Cortex-A73 and 2.0GHz quad-core Cortex-A53 CPU, plus Mali-G52 MP8(8EE) GPU and a 32-bit STM32G031K6 microprocessor. With 8GB RAM, 32GB eMMC, 40-pin GPIO and HDMI in, this absolute unit is ready for anything!



### SEED RECOMPUTER JETSON

**Meet Seeed's Jetson!**

When it comes to machine learning, CUDA cores are king, but with the Jetson Nano dev kit unobtainable, where do budding data scientists turn? Seeed to the rescue with this Jetson Nano 4GB module-based that serves 128 CUDA cores to deliver 0.5 TFLOPS for all your edge AI applications!

Digital Pins	Analog Pins	Radio	Video	Ethernet On Board	Input Voltage	Operating Voltage
32	6	—	—	—	4V-16V	3.3V
65	6 XADC, 4 Differential XADC	—	HDMI	✓	7V-15V	3.3V
4	—	—	—	—	5V	3.3V
14	—	—	—	—	5V	3.3V
32	2 ADC	Wi-Fi, Bluetooth Low Energy	—	—	5V	3.3V
20, 10 in Shield Mode	20 PWM, 1 ADC, 10 PWM, 1 ADC in Shield Mode	Wi-Fi, Bluetooth Low Energy	Mini-HDMI	—	5V-17V	5V
38	—	—	HDMI	—	5V	3.3V
32	—	—	—	—	5V	3.3V

# MAKING STARTS HERE



Whether recreating Tony Stark's Arc Reactor, or building the perfect companion robot, Digi-Key has the components, resources, and creative inspiration to help you take your next project from concept to reality.\*

Start making today at [digikey.com/maker](https://www.digikey.com/maker)



\*You will need to provide the multiple Gigajoules per second to power any devices created. Digi-Key is an authorized distributor for all supplier partners. New products added daily. Digi-Key and Digi-Key Electronics are registered trademarks of Digi-Key Electronics in the U.S. and other countries. © 2022 Digi-Key Electronics, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA

ECIA MEMBER  
Supporting The Authorized Channel