

**FREE MECHANICAL AUTOMATON INSIDE**

# HackSpace

TECHNOLOGY IN YOUR HANDS

hsmag.cc

May 2018

Issue #06

## AMAZON ECHO

Hack your own voice-controlled hardware

## SILICON SUMO

Build your first robot and let bot-battle commence

# PAPER ENGINEERING

Cut, fold, and stick it together to build anything

**BUILD  
THIS DOG FROM  
PARTS INSIDE**

**PRINT  
EDITION  
ONLY**



## MAKING GIN

Summer's coming. Cool off with a home-made cocktail

May 2018  
Issue #06 £6

CONDUCTIVE INK SMART LIGHTING ARDUINO MATCHSTICKS

## CanaKit Raspberry Pi 3 Ultimate Starter Kit

Model B | 1 GB RAM | 1.2 GHz | Quad-Core CPU



- > Learn to Code
- > Explore Computing
- > Get started with Electronics

KIT INCLUDES RASPBERRY PI 3 AND ...

PREMIUM CASE &  
HEAT SINKS



2.5A  
POWER ADAPTER



32 GB CLASS 10  
MICROSD CARD



USB MICROSD  
CARD READER



PREMIUM  
HDMI CABLE



QUICK-START GUIDE



GPIO TO BREADBOARD  
INTERFACE BOARD



RIBBON CABLE



FULL-SIZE  
BREADBOARD



JUMPERS



MALE TO MALE &  
MALE TO FEMALE

LEDs



RESISTORS &  
PUSH-BUTTONS



Available for worldwide shipping at:

**WWW.CANAKIT.COM**



**Raspberry Pi**

APPROVED RESELLER

**Raspberry Pi Zero W**

Available at CanaKit!



ELECTRONIC KITS • ELECTRONIC PARTS • RASPBERRY PI • ARDUINO



# Welcome to HackSpace magazine

Put down the saw, step away from the soldering iron, and strip your making back to the very minimum. This month we're going back to basics and looking at paper engineering. In an age when almost any material is available to us at the click of a button, and hack- and makerspaces around the world

Hack- and makerspaces around the world are stocked with 3D printers and laser cutters – we have an **overwhelming choice** of what to use

are stocked with 3D printers and laser cutters, we have an overwhelming choice of what to use.

Cutting everything back forces us to think about the materials we have – what are their properties and how can we use them? This forces us to solve problems in different ways and, ultimately, leads us to become better makers. Join us in cutting back to basics and fold your next build together – you can start with the automaton in the back of this magazine.

## BEN EVERARD

**Editor** [ben.everard@raspberrypi.org](mailto:ben.everard@raspberrypi.org)

## GET IN TOUCH

- [hackspace@raspberrypi.org](mailto:hackspace@raspberrypi.org)
- <https://www.facebook.com/hackspacemag>
- <https://www.instagram.com/hackspacemag>

## ONLINE

- [hsmag.cc](http://hsmag.cc)



## EDITORIAL

### Editor

Ben Everard

[ben.everard@raspberrypi.org](mailto:ben.everard@raspberrypi.org)

### Features Editor

Andrew Gregory

[andrew.gregory@raspberrypi.org](mailto:andrew.gregory@raspberrypi.org)

### Sub Editors

Nicola King, Jem Roberts

## DESIGN

### Critical Media

[criticalmedia.co.uk](http://criticalmedia.co.uk)

### Head of Design

Dougal Matthews

### Designer

Lee Allen

### Photography

Brian O'Halloran,  
James Neale

## CONTRIBUTORS

Lucy Rogers, Andrew Huang, Rob Ives, Andrew Lewis, Cameron Norris, Jenny List, Mayank Sharma, Graham Morrison, Sophie Wong, Pepi Valderrama, Dylan Murphy, Natthakit Kangsadansenanon, Rob Wood, Andrew Moxon, David Pribus, Marc de Vinck, Les Pounder, Richard Smedley

## PUBLISHING

### Publishing Director:

Russell Barnes

[russell@raspberrypi.org](mailto:russell@raspberrypi.org)

## DISTRIBUTION

Seymour Distribution Ltd  
2 East Poultry Ave,  
London EC1A 9PT  
[+44 \(0\)207 429 4000](tel:+44(0)2074294000)

## SUBSCRIPTIONS

Select Publisher Services  
Ltd, PO Box 6337, BH1 9EH  
[+44 \(0\)1202 586 848](tel:+44(0)1202586848)

Mann Enterprises Ltd,  
Unit E, Brocks Business  
Centre, CB9 8QP  
[hsmag.cc/subscribe](http://hsmag.cc/subscribe)



This magazine is printed on paper sourced from sustainable forests. The printer operates an environmental management system which has been assessed as conforming to ISO 14001.

HackSpace magazine is published by Raspberry Pi (Trading) Ltd., Station Road, Cambridge, CB1 2JH. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products or services referred to or advertised. Except where otherwise noted, content in this magazine is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0). ISSN: 2515-5148.

# Contents

06

## SPARK

- 06 Top Projects**  
The Uffizi of DIY projects
- 16 Objet 3d'art**  
Summer is on its way. Celebrate with hot PLA
- 18 Cool Build: The Man Engine**  
A huge mechanical puppet stalks the land!
- 22 Meet the Maker: Ken Olsen**  
The joy of masochistic soldering challenges
- 24 Columns**  
Much egg-related silliness
- 26 Letters**  
There's a fight brewing about computer history!
- 28 Hackspace The Making Rooms**  
Building in Blackburn, Lancashire

33

## LENS

- 34 Paper engineering**  
Because we can all do better than paper aeroplanes
- 46 How I Made: Reading glasses**  
Like Google Glass, but homemade, better, and not creepy
- 50 Global Village**  
How to rebuild society after machines become self-aware
- 56 Radio radio**  
Explore the humble transistor radio
- 72 Improviser's Toolbox Matchsticks**  
Don't play with matches (unless you're building things)

### Cover Feature

# PAPER ENGINEERING

Fold, cut and glue your way to miraculous mini models

34



124

### Tutorial

School of making:  
Wearables



82

Put your Adafruit CPX  
to good use

**Interview****Alec Steele****62**

YouTuber, blacksmith, philosopher: meet the man of Steele

**77****FORGE****78****SoM Arduino**

Multiplexing, operators, and digital display

**82****SoM Capacitive switches**

Sew a reactive circuit onto a fabric bag

**88****SoM Découpage**

Put your unique style on project enclosures

**92****Tutorial Smart lights**

Control IKEA lights with Google Home Assistant

**96****Tutorial LED panel**

Program a wall of light for maximum colour

**100****Tutorial Make gin**

DIY booze for the refined, discerning palate

**102****Tutorial Robot sumo**

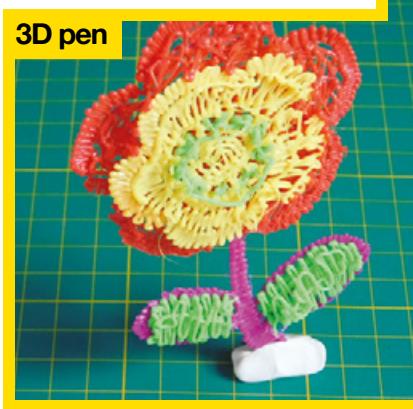
Build a program for your own fighting robot

**106****Tutorial 3D-printed car**

Add micro:bit controls to a toy car

**110****Tutorial Echo**

Control a Raspberry Pi with voice commands

**Direct from Shenzhen****3D pen****116**

3D print freehand with a molten plastic pen

**50****18****40****115****FIELD TEST****116****Direct from Shenzhen 3D pen**

Repair 3D prints (or unleash your artistic side) with a 3D pen

**118****Best of Breed**

Camera modules: give your build eyes!

**124****Can I Hack It?**

Look flash, with LED trainers

**126****Review Bare Conductive Touch Board Starter Kit**

Join your circuits up with conductive paint

**128****Review Arduino Cloud**

Networked programming for this little development board

**129****Book Review The Maker Revolution**

Building a future of creativity and innovation

Some of the tools and techniques shown in HackSpace Magazine are dangerous unless used with skill, experience and appropriate personal protection equipment. While we attempt to guide the reader, ultimately you are responsible for your own safety and understanding the limits of yourself and your equipment. HackSpace Magazine is intended for an adult audience and some projects may be dangerous for children. Raspberry Pi (Trading) Ltd does not accept responsibility for any injuries, damage to equipment, or costs incurred from projects, tutorials or suggestions in HackSpace Magazine. Laws and regulations covering many of the topics in HackSpace Magazine are different between countries, and are always subject to change. You are responsible for understanding the requirements in your jurisdiction and ensuring that you comply with them. Some manufacturers place limits on the use of their hardware which some projects or suggestions in HackSpace Magazine may go beyond. It is your responsibility to understand the manufacturer's limits.

# 3D-printed hummingbird

By Greg Zumwalt

 [hsmag.cc/amlPJF](http://hsmag.cc/amlPJF)

**M**y wife truly enjoys feeding and watching hummingbirds as they pass through our corner of the world, so I decided to create a hummingbird for her to watch during the hummingbird 'off season'.

I started the hummingbird design using the numerous photos we've taken of the hummingbirds. I then imported a few select photos into Autodesk Fusion 360, using the canvas function, then outlined the various components (body, wings, and tail) using the Autodesk Fusion 360 spline function.

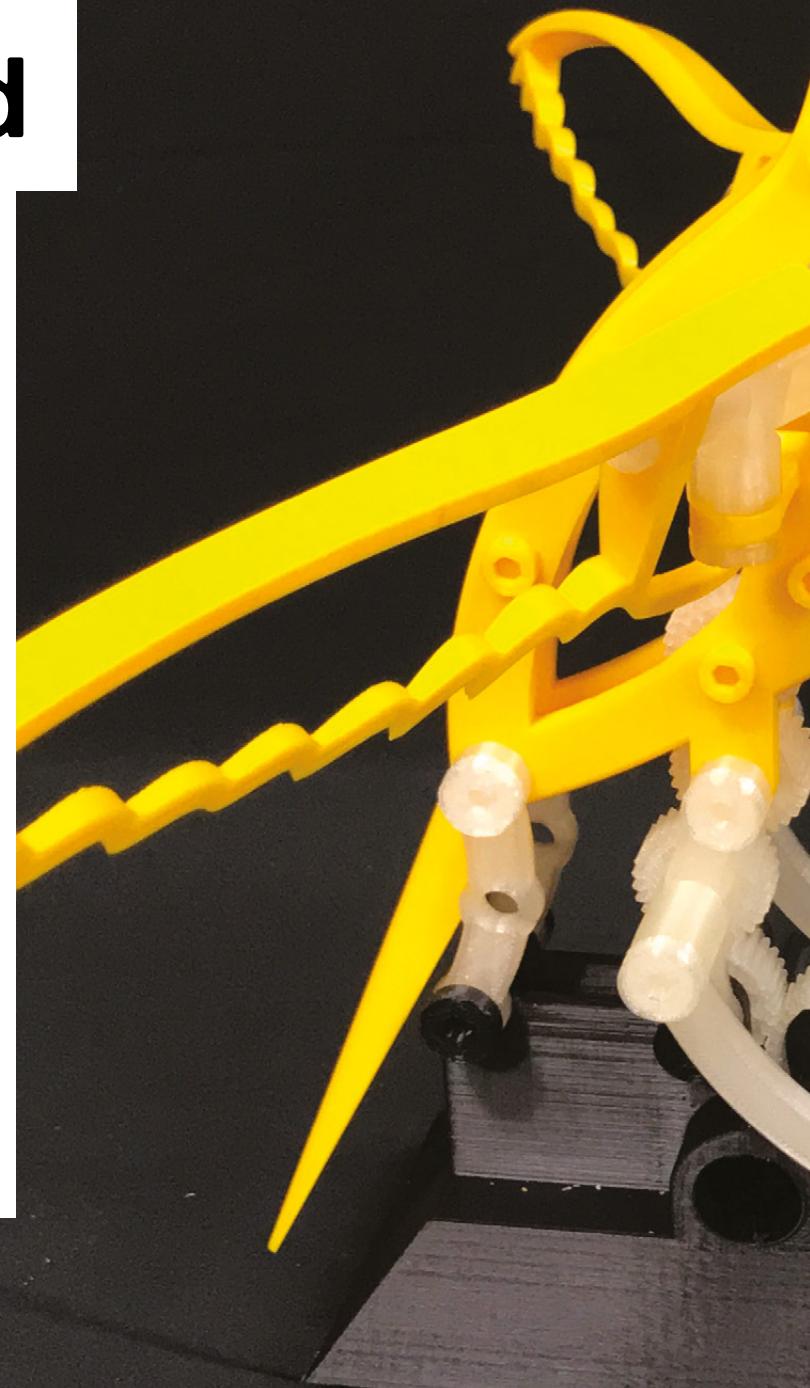
Next, after studying the wing motion of a 'typical' hummingbird, I decided to use a captured ball joint (somewhat similar to a helicopter swash plate), for the pivot and rotate mechanism, and since I try to make my designs 100% 3D-printed, I designed a test fixture to determine the minimal size ball joint I could reliably 3D-print, which ended up being a 6mm ball and a 6.3mm hole. With these two dimensions, I was then able to scale the body, wings, and tail outlines I had traced earlier to a size suitable for the wing motion mechanism. With the scale (size) determined, I next designed the gear train for the hummingbird body using the Autodesk Fusion 360 Gear add-in.

Next was the base design. In order to move the hummingbird fore and aft, I needed to change the size of the gears, which also provides the increased torque required for the fore and aft body motion. I designed the base mechanism using a parallelogram to connect the body to the base which facilitates the fore and aft pivot motion, while maintaining body level.

Finally, the flower is my interpretation of the plastic flowers on the hummingbird feeders my wife uses, and the stem was designed to position the flower at the correct point. □

**Right** 

Greg decided on a ratio of six wing flaps to one complete fore/aft cycle





# Infinity heels

By Angela Sheehan

 @the\_gella

I found this wild pairs of heart-shaped heels and augmented them with custom, programmable infinity mirrors. I used a laser cutter to precisely cut mirrored acrylic to size, to fit in the heart shapes on each shoe. In between a layer of one-way and standard mirrored acrylic, are 5mm NeoPixel LED strips, each controlled by a Qduino Mini microcontroller and programmed with Arduino, utilizing Adafruit's NeoPixel library.

The black wedges use six custom modules that are installed into the heart cutouts. The Qduino and battery are hidden in the ankle strap.

The white wedges have one large infinity mirror installed in the heart shape, and the Qduino and battery embedded in the space cut out of the platform near the toe.

Read about how I built the project and watch a video of them in action on the SparkFun blog – [hsmag.cc/TkWdbs](http://hsmag.cc/TkWdbs). □



**Right** □

NeoPixels make any outfit at least 47.8% better (but not if you're a ninja)





**T - MOVIE QUIZ BOOK EVER!**

LEEKWAY EDITH HAMILTON



CARLTON



NORTON

# Secret bookshelf switch

By Marcus Dunn

 [hsmag.cc/bshQQP](http://hsmag.cc/bshQQP)

I was inspired to make the secret book switch after recently watching *Indiana Jones and the Last Crusade* again. I love the idea that a common object like a book, or in Indy's case a chair, can have a different function than it was made for.

Being able to activate a light, or even a hidden compartment (next project!), with the tilt of a book is a heap of fun. It's worth building your own secret book switch just to watch the surprise and enjoyment that people get when activating it for the first time.

The good news is that building one is relatively simple and cheap. The main part is a ceiling fan switch, which turns off and on by pulling a small chain.

If you want to build your own, check out my Instructables page – [hsmag.cc/CKsByI](http://hsmag.cc/CKsByI). □

**Reader challenge**  
make a secret door switch activated by the first three notes of 'Black Sabbath' by Black Sabbath (as seen in *Batman Begins*)

# Spiral lamp

By Marco Luis Teixeira

 [hsmag.cc/Bgrrbp](https://hsmag.cc/Bgrrbp)

The Spiral Lamp was inspired by a similar piece called the Loxodrome Sconce, by Paul Nylander ([hsmag.cc/AcxLGQ](https://hsmag.cc/AcxLGQ)). Paul's piece consisted of a small 3D-printed loxodrome attached to a solid panel of stained wood. The spiral cast sweeping shadows on the natural wood grain and it was gorgeous!

I set off to reproduce Paul's work using my 3D printer. I printed a small loxodrome, affixed it to a piece of plywood, and illuminated it with a small SMD LED. Aside from it being a comically rough prototype, I learned that an SMD LED wasn't particularly bright, certainly nowhere as bright as the sconce appeared to be in the photos of Paul's work. Perhaps it was an effect of exposure, and I supposed I would have to use a much bigger LED to get the same effect in an illuminated room.

Hence, the idea of a free-standing desk lamp was born. □

**Right** 

There are many ways you can play with shadows – start here for a simple example – [hsmag.cc/Tdfnwg](https://hsmag.cc/Tdfnwg)





**Right ▶**

Loek has made other instruments out of trash, including a gamelan made out of old bike frames – [hsmag.cc/sKMUEh](http://hsmag.cc/sKMUEh)

# Trash instruments

By Loek Vellekoop

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

Admit it; is a lot of fun to make music. Even more fun, however, is to make your own musical instrument from trash, impressing your friends and complete strangers. A fair warning though, making your own musical instruments is highly addictive!

I made this kalimba (a part of the lamellophone family), from some leaf-rake tines and hacksaw blades. Alternatively, I have used car dipsticks, road-sweeper bristles, bicycle spokes, windshield wipers, and sewer springs before. You can make anything create sound!

An added bonus when you invent a new instrument: you are instantly the best-'name instrument here'-player in the world. □



# Objet 3d'art

3D printed artwork to bring more beauty into your life

We're big fans of sitting huddled by the fire, pretending it's about to get warm. Ben's gin making on page 100 is a great example of this – we're freezing cold, but we can still dream of sunny days ahead. On that note, here's a little bit of 3D-printed summer, brought to you by Thingiverse user faberdasher. It's not an iced lolly, but a box cunningly fashioned to look like one. It's made of three parts that friction-fit together, and it promises to keep your keys and loose change in one place while you cool off at the beach. □

↗ [hsmag.cc/xPWAwJ](http://hsmag.cc/xPWAwJ)



Head to [3dhubs.com/book](http://3dhubs.com/book)  
to check out the  
**#1 3D printing book on Amazon**

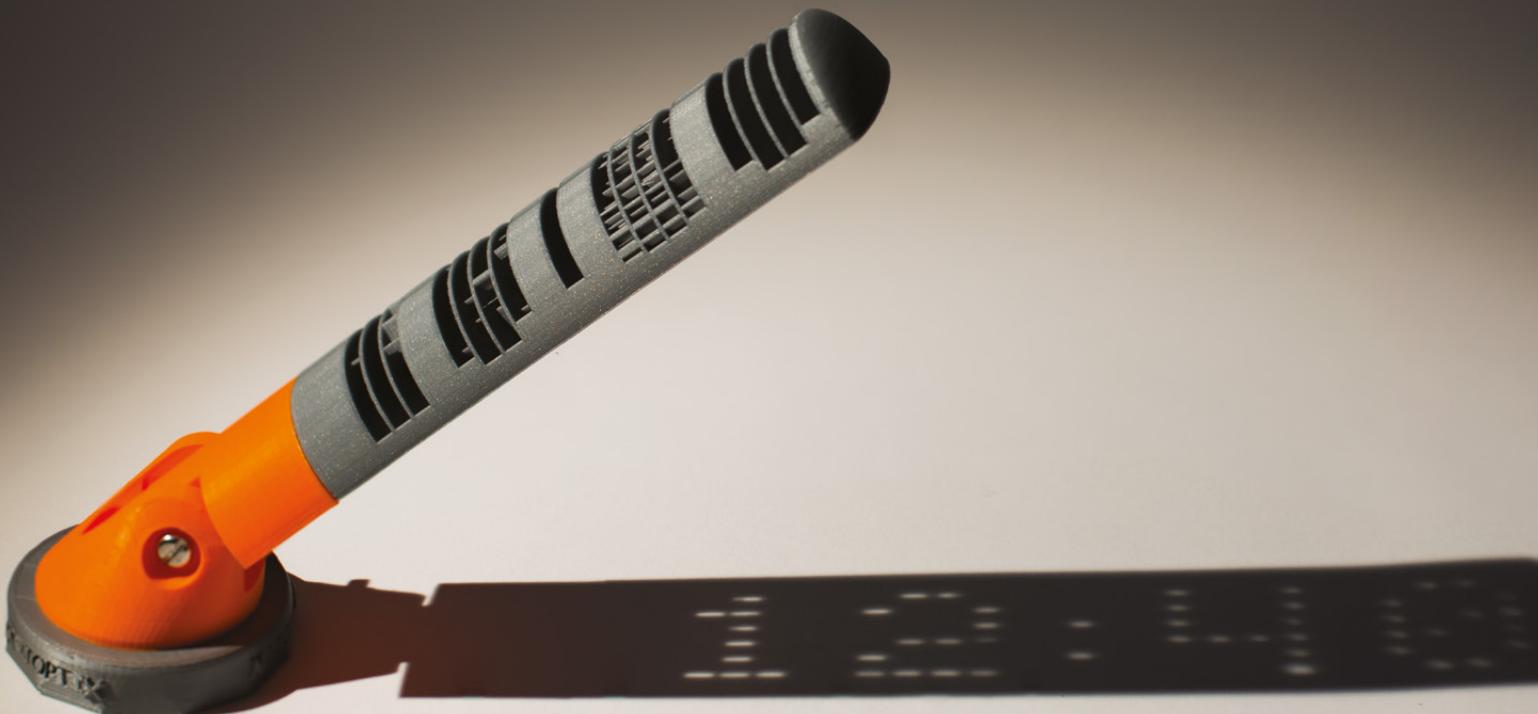


**N**ow this is magic. If and when we ever get sunny days, we strongly recommend you print Mojoptix's digital sundial, because it will blow your mind.

It displays the time accurately between 10am and 4pm, updating every 20 minutes.

To put the clocks back or forward, simply rotate the gnomon (that's what the long thing's called, that the light passes through to make the shadows) – and don't forget to print the correct gnomon for your hemisphere – time goes backwards Down Under. □

↗ [hsmag.cc/QdgzyZ](http://hsmag.cc/QdgzyZ)



# The Man Engine

A piece of Britain's industrial history brought dramatically to life

By Andrew Gregory



@AndrewGregory83

If you've been out and about in the wet West Britain recently, you might have bumped into this amazing piece of engineering – the Man Engine. Intended as a tribute to the lost mining heritage of Cornwall, the 11.2 metre-high puppet has been on a tour of Cornwall and South Wales, with a bonus appearance at Ironbridge in Shropshire.

Building a big project comes with big challenges, as we found out when we spoke to Hal Silvester, the Man Engine's lead designer.

"I wanted maximum size transformation from crouched to standing. He actually folds up really small which is also good for transportation (e.g. if he was 2cm wider in the stowed position he would be classed as a wide load, which would make him more troublesome in transit). He will also go in a shipping container for when he eventually goes overseas to follow his mining ancestors. This level of 3D jigsaw engineering was a challenge!"

Whatever the size of the build, it's important to start by getting the basics right.

"The actual build took five months, with 90 percent of the build taking place in the last two. As there are not many giant puppet regulations, getting suitable paperwork for such an unorthodox project required imagination. In the end, we reclassified the wheel loader as a piece of lifting machinery, and the puppet as a lifting attachment. We could then apply standard lifting tests [on both]."

## POWERED PUPPETRY

The name 'Man Engine' comes from the apparatus that took miners underground, down to the seam and back. These were originally powered by water, and later by steam engines. The modern namesake uses rather different propulsion: →

**Right ↗**  
What better way to celebrate over 4000 years of Cornish mining?





**Right ♦**

By powering the miners' journey to and from the face, man engines boosted productivity and allowed workers more time to dig – essential at a time when workers were not paid for travel time within the mine site





**“The actual build took five months with 90% of the build taking place in the last two**

**Below ↗**  
The Man Engine even has an Arduino on the puppet, running code for the effects



“We essentially had ten months for the whole project. The first few were focused on trying to find a manufacturer of plant machinery who agreed that supporting the puppet was within their machine’s designed parameters. Step forward Volvo.

“The main puppeteer is a 31-tonne Volvo L220H wheel loader. Wheel loaders are used in mines and quarries, so it’s a very suitable machine for this purpose. Ten other puppeteers, dressed as miners, pull on ropes and levers to bring the giant to life. I wanted people to see the miners’ endeavour and effort in all this.

“There are electrical components on board such as winches that operate the shoulders, and the steam effects are made by an on-board smoke machine combined with piped-in compressed air from a compressor.”

If you’ve missed it, never fear: the Man Engine will also be appearing at Hillhead Quarry, Derbyshire, and Elsecar Village and Wentworth Woodhouse, both in South Yorkshire – and there’s a European tour planned for 2019, and a world tour coming up in 2020. Visit [hsmag.cc/LEEvAD](http://hsmag.cc/LEEvAD) for details. □



**Above ↗**  
In imperial measurements, 11.2 metres equals 2.5 double-decker buses

# Meet The Maker: Ken Olsen

The man behind Maker's Box and masochistic soldering challenges

# M

**Y**our first real success with designing PCBs was at a family reunion. I designed this little tiny circuit board that had an ATtiny on it, and it had two LEDs and it could detect light. The kids would turn it into a craft project where it was a bug with foam and felt and pipe cleaners. For that one I used toner transfer, so you'd print out the circuit layout on a laser printer and use an iron to transfer it onto a PCB, then etch it with really nasty chemicals in your garage. That's another fun one to go back to look at. The layout was pretty primitive and the firmware was kind of fun.

I did some workshops at a local group called **chicktech.org** who run events for high school-aged girls to expose them to STEM. We did some work where they could sew a circuit board into a bracelet or plush toy and get it to react to light or sound. It was a lot of fun to design those. The last project I did with them was a robot that would draw with a pen using turtle graphics. That was probably my most advanced project to date, as it uses 3D design, a little bit of circuit design, and the firmware and the girls come into the class and we break them into three groups – the mechanical engineers who have to assemble the chassis and the wheels and stuff, the electrical engineers who have to solder the board together, and the firmware engineers who have to get it to move and follow their instructions.

**GOING PRO**

The first project I sold on Tindie was a programmer for ATtiny chips – it was an Arduino shield. We had a group at work who would meet at lunchtimes and we'd all become interested in the ATtiny. People were just starting to port Arduino stuff to it, which I really liked because it's so simplified. We wanted to

design something where you could just drop the ATtiny chip in, and be able to program it without wiring it every time.

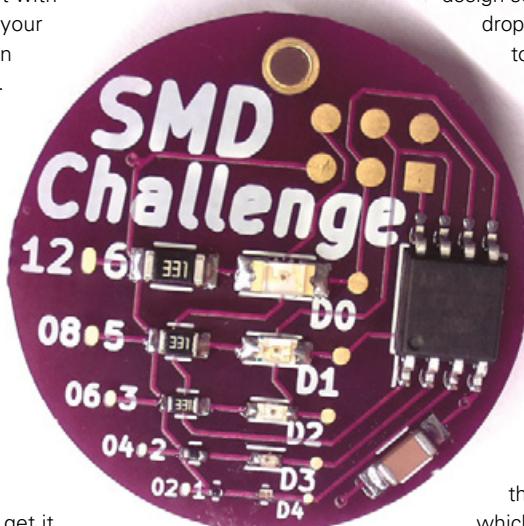
A group of three of us sat down and laid out some schematics using a zip socket. We looked at what the price of the build was, and the price of the unit, and we thought that people would buy kits. We were really on the fence about whether to try and kit it out or not.

Two things happened that made it: we found Tindie, which is kind of an Etsy-like retailer where you can post your materials for

sale, and we also got a grant from a company – I can't remember their name – but we put their silkscreen on the back of the board. They gave us \$500, which basically covered the initial purchases.

People were buying them and we weren't getting any real negative feedback, but we were only selling a few. It was clear to me that if you really want to make any money at this, you need to have a number of kits, not just one.

There have been a couple that have been pretty good hits. It's always funny to see – the orders start



**Above** Try your hand at the SMD challenge available from [hsmag.cc/jccSXV](http://hsmag.cc/jccSXV)

pouring in and you try to figure out what happened. One of my projects got featured on Lifehacker, so these kits sell out, they're just gone in a couple of weeks, so you order more and get them kitted up, then these ones sell a little less fast, and then you kit them up again ... I haven't sold one in the last year.

This latest one that I did, that really kind of surprised me. I was doing a surface mount soldering kit for beginners, to teach people just how easy it is to do surface mount. The surface mount stuff goes very, very tiny and I wanted to see how tiny you could do that stuff by hand. I whipped out a little board that went all the way down to 0201 LED, which are very hard to work with. I posted it on Hackaday – I wasn't going to sell a kit because it was just so abusive. I didn't actually think you could do the smallest ones by hand, so I just posted the instructions, firmware, and board designs on Hackaday and people just started going nuts about it. I did eventually do a kit for it, and it's sold hundreds. People are actually able to do it by hand! SparkFun did a thing where their four senior electrical engineers had a little contest, and out of the four, only one successfully completed the thing in an hour. It was really fun to see the pros struggle with it.

## TRIALS AND TRIBULATIONS

The biggest issue with the kits selling is that you're competing with the likes of Banggood, where you can buy a clock kit with a full circuit board, with dozens of LEDs, for \$5 or \$6 dollars – I can't even get a circuit board for that price. [Some people say] 'I can get an Arduino from China for \$4 or \$5 dollars, why would I go to Adafruit and pay \$20 or \$25?' The thing I've been thinking about is what the value proposition is. Adafruit publishes all their design files. They have tutorials. All their boards are made in the USA at their factory in New York. You can get this stuff really cheaply from overseas manufacturers, but you're not getting the value of the ecosystem that goes with it. You can email people and get information. The plans are really well documented. Some of these Chinese kits are not even in English. You get a schematic and a bom, and you have to take it from there. To me, buying something from a reputable manufacturer, such as Adafruit or SparkFun, who have put time into creating it, and building an ecosystem around it and will support you – there's a lot of value in that, and I think that's why they're being successful.



**You can get this stuff really cheaply from overseas manufacturers, but you're not getting the value of the ecosystem that goes with it**

I've been giving a lot of thought to open source, and there was an incident where someone was using my robot design without attributing it to me, and they were selling a kit. It made me upset at first, but I got thinking about how I learned from Adafruit and how I'm trying to teach, so I have to be willing to let people copy me and use my things. As a maker, my goal is to help people learn. For me, looking back at how I learned from Adafruit – they freely give away all their information and they're still able to make a business out of it. That is what I'm trying to emulate, even at a small level. □

**Above** Program an ATTiny from an Arduino, without messing about with wires

**Below** Spin your boredom away with Maker's Box Persistence of Vision Fidget Spinner



# Yolking around

Finding interesting solutions to non-existing problems



**Lucy Rogers**

 @DrLucyRogers

Lucy is a maker, an engineer, and a problem solver. She is adept at bringing ideas to life. She is one of the cheerleaders for the maker industry and is Maker-In-Chief for the Guild of Makers: [guildofmakers.org](http://guildofmakers.org)

If you boil a penguin egg, the albumen doesn't go white, it remains translucent. This factoid was shared with me by @wren154.

It was in response to me tweeting: "Why is it perfectly acceptable to have a runny yellow in a boiled egg, but a runny white is a no-no?"

Eggs in the UK, with the British Lion mark, are safe to be consumed runny or even raw. Although I knew this, and my egg had the Lion mark, it took my brain a bit of re-adjustment before I could eat it.

@GregorJClark issued a challenge: "Find a way of cooking a hard-boiled egg which leaves the white runny, but the yolk solid."

Now this is just the sort of challenge I love. And it seemed to eggcite Twitter, judging by the number of replies it got.

"Custom-build a microwave with a frequency that excites fat molecules instead of the traditional water-targeted ones?", @RobOnABike. "Some form of radiotherapy? I mean, that's the principle of treatment, kill the middle but not the surrounding cells... so if it generated enough heat..." – @archieroques.

"Selective Laser sintering? ... guess you'd need some sort of scan of the egg to locate the yolk" – @SueArcher6.

"Looks like it won't show up on X-ray. I reckon we need an ultrasound machine..." – @archieroques.

"Are you allowed to inject anything into the egg first? If so, and you can find a way

to bind something specifically to the yolk, then magnetic hyperthermia may be the way to go" – @MillicentOak.

"I'm thinking sous-vide to slowly bring the full egg to just below cooking temp. Then rapidly cool the exterior so white is cooler than yolk. Then back in sous-vide to higher temp" – @GregorJClark.

"Infra-red light from a high power LED or laser focused at the yolk through a really big, high NA lens? Thin shell and transparent white shouldn't absorb much energy" – @martinjones78.

"Maybe multiple IR sources with phasing adjusted so that the peaks & troughs cancel out in the white zone, but amplify the heat in the centre" – @GregorJClark.

"Seems like you could 'cook' it by injecting, uh, acid" – @rynbtmn.

"Yolk has 27 times more iron than white, so ferromagnetic cooking might work? Try putting an egg directly on an induction hob? Or in a solenoid with high frequency AC?" – @martinjones78.

But @SueArcher6 summed it up with: "It would certainly be a waste of an egg, by serving both parts in their less good states (solid yolk is okay, but runny is better!)" ([hsmag.cc/JIHYHu](http://hsmag.cc/JIHYHu)).

Sometimes it's good to ponder out loud. You never know where the responses may take you.

You can view the whole conversation on Twitter at [hsmag.cc/xgQSHc](http://hsmag.cc/xgQSHc). Thanks to everyone who contributed, retweeted, and/or enjoyed the conversation. □

**Now this is just the sort of challenge I love. And it seemed to eggcite twitter, judging by the number of replies it got**

# Compliance made easy

My favourite feature of the Raspberry Pi 3 Model B+



**Bunnie Huang**

 @bunniestudios

Andrew 'Bunnie' Huang is a hacker by night, entrepreneur by day, and writer by procrastination. He's a co-founder of Chibitronics, troublemaker-at-large for the MIT Media Lab, and a mentor for HAX in Shenzhen.

The Raspberry Pi 3B+ was announced last month. There's a lot to be happy about – faster CPU, faster connectivity, PoE support, all for the same price tag.

However none of those things got me as excited as the fact that the 3B+ has been compliance tested as a radio module.

I know what you're probably thinking – 'compliance testing' sounds incredibly mundane, how could anyone get worked up over that?

Most governments publish a set of strict wireless emissions standards that every consumer product must comply to in order to be sold. The problem is these standards were written decades ago, when AM radio was common. AM is very 'fragile'; minuscule amounts of stray noise will affect

it. And so, laws were passed decades ago that codified tests designed to protect these ancient devices. The tests themselves can be gruelling; testing a non-modular WiFi 'active transmitter' requires a custom firmware blob that forces the transmitter to send continuous signals in every possible modulation over select frequency bands. You'll be required to solder a connector in place of the antenna on the PCB for some of the tests. These tests can cost thousands of dollars

to execute at an approved facility, and delay projects by weeks or even months.

Unfortunately, no amount of regulation can protect something as fragile as AM; so, technology progressed to more interference-robust standards like FM or, more recently, spread-spectrum techniques such as OFDM. Yet the emissions standards – hard-coded into law – were never revised to reflect these ground shifts in technology. So here in 2018, we're expending a disproportionate amount of time and effort designing counter-measures to protect these now-defunct radio applications.

By certifying the Raspberry Pi 3B+ as a 'radio module', Raspberry Pi has already passed the most onerous tests on our behalf. So long as we don't modify the Pi's radio function or design, products incorporating the

Raspberry Pi 3B+ as a radio module are now subject to a greatly abbreviated series of tests. The metal shield over the radio circuitry, along with other more subtle counter-measures, also means there's more headroom for the noise that will be generated by our application-specific circuits and peripherals. Overall, this will save numerous hackers, engineers, and startups thousands of dollars, countless sleepless nights, and weeks on already too-tight delivery schedules. □

# Letters

## CREDIT WHERE IT'S DUE

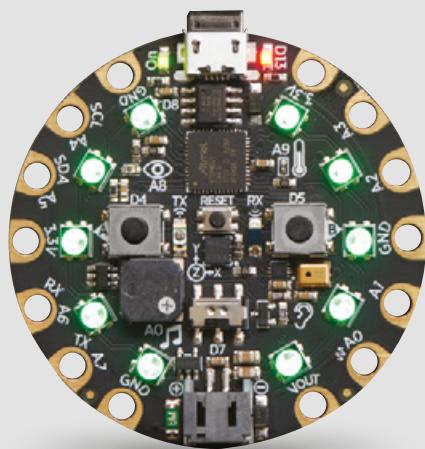
Lucy Rogers is absolutely bang on the money about sharing credit where you've used someone else's work. There's no shame in admitting that you don't know everything yourself – that would be impossible anyway, so to see someone give credit to the authors of guides that they've used only makes me respect their work more.

To paraphrase a great man, if I have made a better 3D-printed gadget, it's only because I've been standing on the shoulders of giants.

**Charles MacMillan**

Seattle

**Ben says** Yep – open hardware, open software, science, literature, music all depend on shared knowledge. Blessed be the information sharers.

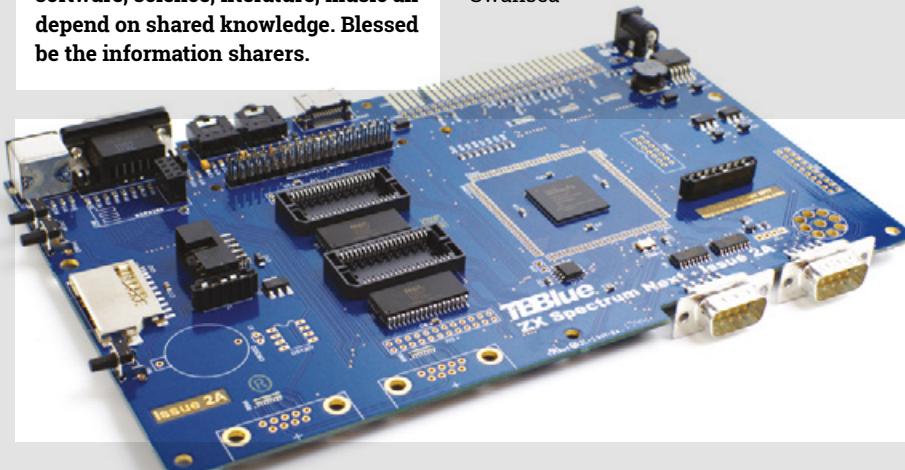


## ADAFRUIT

Thanks for the look inside Adafruit last issue. Obviously components don't land on PCBs by accident, so it was an eye-opener to read about the decisions that go into making a device like the CPX.

**David Maher**

Swansea



## ATTENTION ALL MAKERS!

If you have something you'd like to get off your chest (or even throw a word of praise in our direction) let us know at [hsmag.cc/hello](http://hsmag.cc/hello)

## CHEERLIGHTS

I remember SETI@home [it's still going – [setiathome.berkeley.edu](http://setiathome.berkeley.edu)].

The plan was to link computers together with spare processing capacity to analyse radio waves from space, in the hope that we'd be able to identify alien transmissions.

Fast forward to today, and we've got Cheerlights [as seen in issue 5]. We've gone from searching for new life to changing the colour of LEDs!



**Adam Church**

Manchester

**Ben says** SETI@home has its priorities all wrong. We should be searching for intelligent life on earth first.

## BEST OF THE BEST

Reading the newsletter, I disagree strongly that the Spectrum was the best. We got a BBC Acorn in 92/3 and had it till recently. To us that was the best one.

**Patricia Brogan**

# CROWDFUNDING NOW

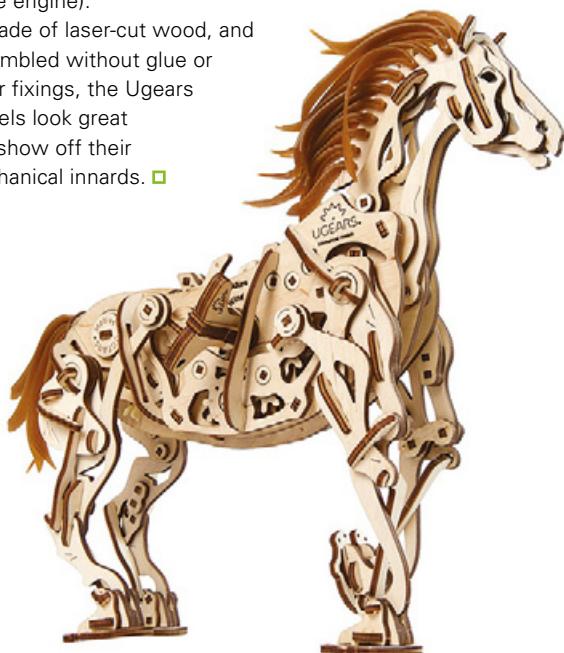
## Ugears

Motion like it's 1599

From \$39 | [hsmag.cc/lavHEo](http://hsmag.cc/lavHEo) | Delivery: July 2018

**t's easy to get caught up in electronics.** An Arduino or Raspberry Pi can transform something ordinary into a flashing, blinking, internet-connected device of the future. However, electronics is really only a tiny part of a long story of technology that goes back almost as far as humanity. For most of history, technology was about mechanics, and generations of makers and hackers developed ingenious devices using physical motion. Ugears celebrate this with their mechanical models. They already make a wide range and are crowdfunding an expansion to their existing line up, including a horse with a walking mechanism powered by an elastic band, and an archballista (a medieval siege engine).

Made of laser-cut wood, and assembled without glue or other fixings, the Ugears models look great and show off their mechanical innards. □



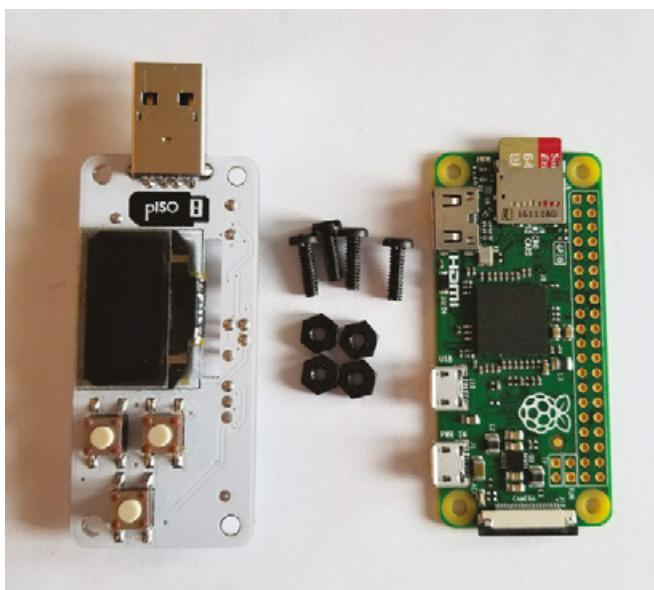
## Piso

The easy way to transfer files

From \$27 | [hsmag.cc/jaxGdm](http://hsmag.cc/jaxGdm) | Delivery: July 2018

**M**emory sticks are one of the easiest ways of moving files around. They can store huge files, don't require an internet connection, and work with almost any computer. However, they can feel a little primitive. After all, aside from their size and form factor, are they really much different from the floppy disks we had twenty years ago? Piso takes all the advantages of a USB memory stick and uses a Raspberry Pi to give it a few more features.

You can create virtual drives so that your USB stick can be reformatted, yet still retain the original data, you can snapshot the drive, and even access it wirelessly. It's half USB memory stick, and half file-server all tied up in one device. It's sure to make life much easier for heavy users of memory sticks. □



# Space of the month: The Making Rooms



The Making Rooms

[makingrooms.org](http://makingrooms.org)

@TheMakingRooms

**T**he Making Rooms is a makerspace where creativity, technology, and advanced manufacturing come together in a community facility for use by businesses, artists, inventors, students, children, and just about everyone else to design and make anything from high-tech products and gadgets to toys, artworks, home decorations, and T-shirts.

At the heart of The Making Rooms is a Fab Lab containing an impressive range of modern fabrication equipment combining advanced manufacturing with traditional craft processes including 3D printing, laser cutting, electronics, CNC machining, pottery, and screen printing, to name but a few. During the week the lab operates commercial prototyping and machine hire services, while at the weekend the lab is open to the public free of charge.

The Making Rooms is a Community Interest Company, so any profits are reinvested into better serving the local community. Their two main goals are to turn people into makers and turn makers into business owners.

The space is located right in the heart of Blackburn to make it as easy as possible for people to visit.

## TENANTS

The Making Rooms also house two floors of flexible studio and office space to help locals get their businesses off the ground. The team are creating an environment that encourages collaboration and helping tenants to explore new processes.



**Above** The Making Rooms is home to a coffee shop, so people can go in 'incognito mode', peering up from a cup of coffee to see what folks in the lab are up to, before taking the plunge into their first taste of making



## Kate Illustrate ♦

Hello, I'm Kate, a 24-year-old freelance illustrator and graphic designer based in Lancashire. From a young age, I've always been interested in art and creating. I went through education determined to make a career for myself within Art and Design. Therefore, alongside whatever it was that I was doing, I'd find myself picking up a pencil or a pen and drawing something new.

I developed my own style after graduating from the University of Leeds in 2015 with a degree in Art and Design. Kate Illustrates officially launched in September 2017, offering calligraphy services, illustration, and design for events, businesses, branding, and more.

[kate-illustrate.co.uk](http://kate-illustrate.co.uk)  
[kate.illustrate](http://kate.illustrate)  
[kate@kateillustrate.co.uk](mailto:kate@kateillustrate.co.uk)



## The Bakery - David Boulbee

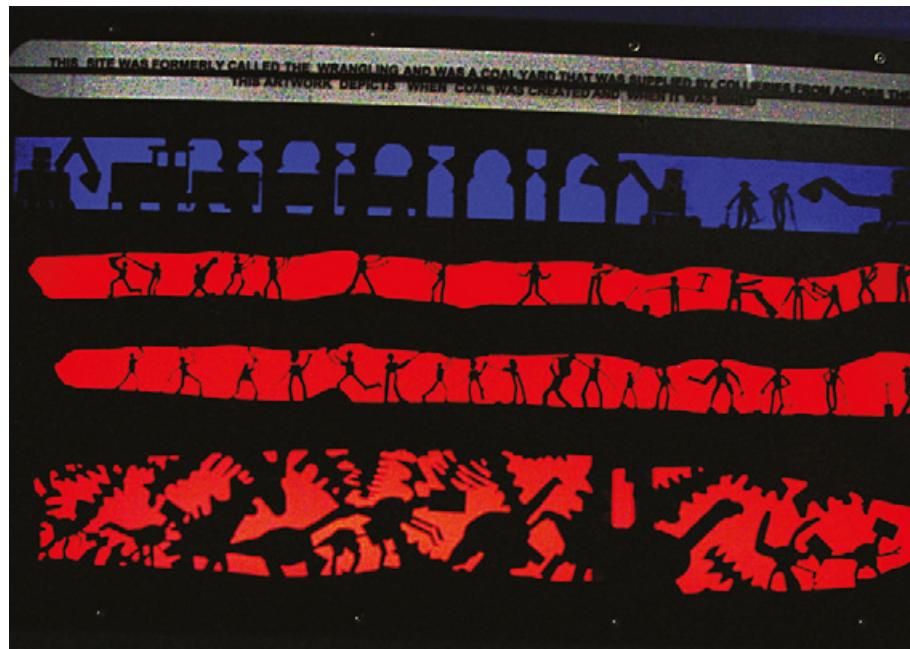
In 2009 David decided that he would begin to explore his own creative direction and set up BREAD art Ltd. His work comprises three interwoven strands:

- An engaged practice that works with people to elicit stories and develop connections between them, the artists, and the places they inhabit, occupy, and discover
- Artworks that use light, sound, electronics, and digital manufacturing to create engaging, responsive, or interactive physical objects
- Development of interactive exhibits for museums, galleries, and exhibitions

David is currently artist in residence at The Making Rooms and is developing a new sculpture for installation in Huddersfield town centre. Through his engaged practice, he observed how effective digital techniques are in engaging young people and encouraging them to explore their own creativity. His background means that he is always keen to highlight crossover between STEM and creativity, especially to young people making decisions about their futures.

David aims to found a design/build/make studio staffed by young people from Blackburn. Named The Bakery, the project is in its early stages but showing great promise: "In 2016/17 we worked with the first cohort to create a clock for The Making Rooms. This year we have more young people joining us, and are delighted that some participants from our first project have returned to help train them. At the moment we're working project-to-project while we learn about approaching this." Their long-term aim is for the studio to self-fund by executing real-world projects for commercial clients with a team of highly skilled, motivated apprentices, and they anticipate that the project will directly provide employment opportunities for some, and equip others with skills and experience to forge their own path in the creative, manufacturing, and technology sectors.

↗ [breadartcollective.co.uk](http://breadartcollective.co.uk)  
 ↗ [bakeryblackburn.com](http://bakeryblackburn.com)  
 ↗ [makingrooms.org/the-bakery](http://makingrooms.org/the-bakery)



### Above ♦

The lab's maker shop sells only products that have been made in the space. It's primarily to encourage the casual weekend maker to become a creative mastermind



## Alastair Nicholson ♦

I work in participation with community groups including schools on a variety of projects often educational or addressing a particular issue, such as memorials, safety campaigns etc.

I have been working freelance for ten years. In the last two years since the Fab centre opened, I have been able to extend my range of materials to include transferring community artwork into laser-cut materials, wood, acrylic 3D printing etc.

This has enabled my clients to bring more public participation into the project that they hire me to help with, extending the usual mosaic mural community pieces to be laser- or CNC-cut material.

I have been able to trial large pieces at the Fab centre in wood and cardboard, and then send them to a local engineering firm, WEC of Darwen, to be cut in more durable material such as stainless steel. Trialling cutting at Fab has helped a lot to prevent mistakes at the steel-cutting stage.

## THE AMBITION OF PROJECTS

My latest project is a lit sign at a new locomotive maintenance works in Blackburn. I worked with a local school on the history of the site, producing a two-sided stainless steel back-lit sign with acrylic inserts; the steel was cut by WEC, and the acrylic inserts at the Fab centre. The sign is 7ft x 4ft, mounted on a concrete and brick plinth, and it welcomes everyone to the site as they enter.

# The Making Rooms team



## Dr Eddie Kirkby, director ♦

Eddie is a multidisciplinary engineer who graduated from Manchester Metropolitan University with a first-class honours degree in Mechatronics, during which he spent a year as a manufacturing engineer at Vauxhall Motors.

Eddie went on to work for The Manufacturing Institute within the New Product Development department, developing many of The Manufacturing Institute's training and educational programmes. Eddie progressed through the Institute to become Charity Operations Manager, supporting charitable projects such as the Make It campaign whilst developing the Institute's ISO business management systems. He led the introduction of the UK's first Fab Lab in Manchester and The Manufacturing Institute's two subsequent Fab Labs at Ellesmere Port and Altrincham. Eddie has supported many other organisations around the UK to set up new Fab Labs and helped grow the Fab UK network to its current size of around 20 Fab Labs.

Eddie took on the role of Director of The Making Rooms, Lancashire's first Fab Lab, in early 2016.

## Julia Jacob, technician ♦

My job as technician at The Making Rooms is so varied that I often get asked 'How in the world did you get into this job?' This happened most recently when I was 3D-scanning a client's bare broken foot! My answer involves being proactive about creating your own positions and getting involved in helping out at fun institutions, whether or not I could see any direct benefit at the time.

When I could see no feasible route to a design job, I created my own. I started up a product design studio to sell my work from and a blog to gush about other designers' work. These have both helped me show people, including employers, what kind of maker I am now, and aspire to be in the future.

I met lots of my fellow makers – without whom I wouldn't have known this job was available – at events and art/design/tech centres I've volunteered at, most notably at Fab Lab Manchester, where I met the future Making Rooms team.

## Tom Macpherson-Pope, lab manager ♦

Tom is a habitual maker. He started his career as part of the small startup team that developed the PiFace range for Raspberry Pi and the pocket-size programmable badge CodeBug. The team also featured on *Blue Peter* and *The Royal Institution Christmas Lectures* with their Frozen Pi bullet-time-style camera rig.

While studying Computer Science at the University of Manchester, Tom discovered Fab Lab Manchester and his fate was sealed – he would spend the next seven years making and tinkering. "When I first went to a Fab Lab, I couldn't understand how such an amazing place could exist!"

Tom now runs his own electronics product development company in his spare time. Tom and his co-founder James Fletcher listen to their clients' ideas and turn them into reality. The pair met years ago at Fab Lab Manchester.

"Our vision for The Making Rooms is to show people that they can have a successful career doing what they love. It's not a job – it's a paid hobby!"

"I love to visit as many makerspaces as possible to gain their insights and ideas for how to make the best makerspace possible. Creating a sustainable lab that gives so much of its time for free is a challenge, but with what we have learnt over the past few years, we are well on our way."



# Projects

## Festival of Making ♦

Last year the lab opened its doors to the first annual National Festival of Making. The festival was a huge success, with people coming from all over the country to see what was going on in Blackburn. The whole town took part in all sorts of crafts and creations. The Making Rooms engaged with over 2000 people over the course of the weekend.

The town will be hosting the second Festival of Making from 12–13 May this year, and it's planned to be bigger and better than ever before. The Making Rooms will be hosting other makerspaces and creators from around the UK to show off what they've been working on.



## Blue Dot ♦

Last July the team packed up their laser cutter and 3D printers and set up shop in a field. This may sound a little nuts, but it was for the Blue Dot festival to show festival-goers the joys of making.

They ran workshops, sewing conductive thread using a kit the team produced, where attendees would sew around the outline of Jodrell Bank's Lovell Telescope to connect a circuit made up of an LED, battery, and switch button.

People laser-cut their own version of a personalised golden record from the Voyager probes, with their names added to the original record's etched instructions.

They also made a huge golden record that would speak some of the original messages of peace from around the world that were recorded onto the Voyagers' golden records. This used a 3D-printed satellite model, some conductive vinyl, and a Raspberry Pi. The team made a homage to Carl Sagan's pale blue dot quote using some NeoPixels and laser-engraved Perspex.



## CONTACT US

We'd love you to get in touch to showcase your makerspace and the things you're making. Drop us a line on Twitter @HackSpaceMag or email us at [hackspace@raspberrypi.org](mailto:hackspace@raspberrypi.org) with an outline of what makes your hackspace special and we'll take it from there.

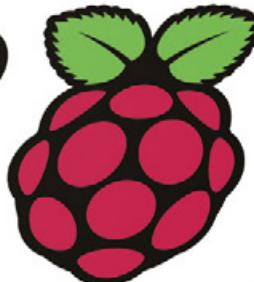
DON'T MISS THE BRAND NEW ISSUE!

YOUR OFFICIAL RASPBERRY PI MAGAZINE

# The MagPi

The official Raspberry Pi magazine

Issue 69 | May 2018 | [raspberrypi.org/magpi](http://raspberrypi.org/magpi)



AFFORDABLE  
3D PRINTING

Buy your first 3D printer  Use OctoPrint for easy control  All the greatest 3D projects

SUBSCRIBE FROM JUST £4

- FREE! £5 ModMyPi Voucher
- FREE! Delivery to your door
- NO OBLIGATION! Leave any time

NEW  
AIY KITS

Now with Pi Zero WH  
includes



FROM  
£99

TRANSFORM A  
RETRO CAM

Kodak Brownie adapted  
into a modern digicam

PIZERO

FREE PI ZERO W

With your 12-month subscription to the print magazine

[magpi.cc/Subs1](http://magpi.cc/Subs1)

PLUS! FREE  
CASE, THREE  
COVERS &  
CABLES

Buy online: [store.rpipress.cc](http://store.rpipress.cc)

# LENS

HACK | MAKE | BUILD | CREATE

Uncover the technology that's powering the future

PG  
**46**

## GLASSES THAT READ TO YOU

How a pair of spy glasses and a Raspberry Pi can help someone see the world through sound

PG  
**50**

## MAKING A VILLAGE

Meet the people trying to create an open-source blueprint for restarting civilisation

PG  
**56**



## BUILD A RADIO

From crystal sets to Raspberry Pis, there's a radio project for everyone

PG  
**34**

## PAPER ENGINEERING

A sheet of paper: it's cheap, ubiquitous, and almost endlessly hackable.  
Let's get making!

PG  
**62**



## INTERVIEW ALEC STEELE

Discovering the philosophy of making through Damascus steel

PG  
**72**

## MATCH STICKS

Launch, build, and puzzle with incendiary fire sticks

# PAPER ENGINEERING

Fold, bend, curl, and stick your next build together



**P**aper is probably one of the first materials you learned to hack with. While Lego, Meccano, and other engineering mediums are designed for children to build with, paper isn't. The transformation of a sheet of A4 paper, designed for writing, into a plane that sails through the air much further than it seems like it should, is almost the epitome of hacking. It's slightly subversive, it's using something in a way it's not meant to be used, and its simplicity belies the engineering principles that underpin its flight.

You can leave the plane wide to increase the lift and watch it float like a butterfly, or fold it in tight to reduce drag so it flies fast and stings like a bee (and can be pointy enough to injure someone's eye). You can even attempt to 'program' in an acrobatic trick with the careful bending of ailerons. While it's a hobby of school children the world over, it's also a serious (well, semi-serious) engineering challenge, with people competing for the longest distance and the longest time in paper flight (that's 69.14 m and 29.2 seconds respectively).

Paper engineering isn't just about planes – that's just most people's route into it. From this humble beginning, you can develop your skills in many ways. Purists may prefer the fold-only approach in the famous Japanese art of origami, where practitioners start with a square sheet of paper and transform it into a three-dimensional structure using only folds.

There is, of course, no need for the budding paper engineer to limit themselves in this way if they don't want to. Cutting and gluing open up whole new worlds of paper-based modelling. You can even incorporate other materials – paint, tape, weights, or

even electrical components (see our review of Bare Conductive paint on page 126 for details).

Your builds can be as beautiful or as functional as you like. Paper can bring colour, it can bring structure, and it can do both. Paper planes, for example, have to fly through the air, while origami cranes are popular for their appearance.

Paper is cheap enough to experiment at will – there aren't many materials where you can iterate through 500 designs for the price of a cup of coffee – yet still interesting enough to be worth experimenting with. It's

## PAPER ENGINEERING ISN'T JUST ABOUT PLANES – THAT'S JUST MOST PEOPLE'S ROUTE INTO IT. FROM THIS HUMBLE BEGINNING, YOU CAN DEVELOP YOUR SKILLS IN MANY WAYS...

deceptively strong (at least in tension) and incredibly lightweight. It requires very little to get started, just a pair of scissors, and perhaps some glue or tape. If it all goes wrong, well at least you've got some scrap paper to make a note of what you've learned in this failure.

Let's dive in and take a closer look at the world of paper engineering and what you can make with this most flexible of materials. →



# TINYSAURS

## QUETZALCOATLUS

**W**hen I took over TINYSAURS [writes Herb Hoover], Kelly McGuire, the founder, had miniaturized five models using NYC Resistor's new

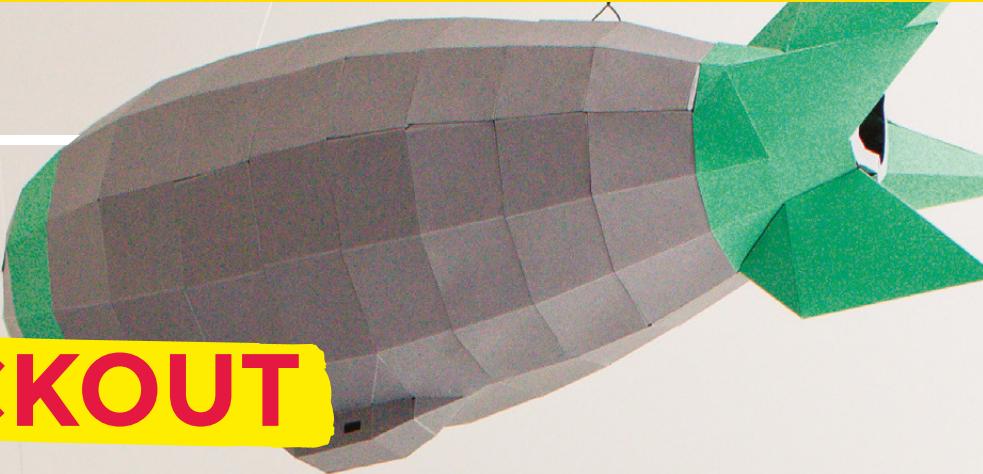
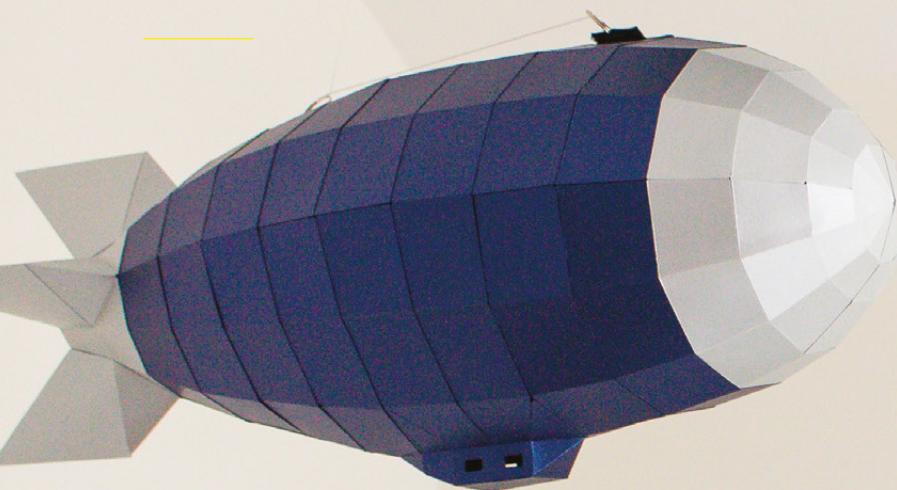
**laser cutter.** I have expanded the line to 21 models – some of them humanoid. I particularly love making the fantastical humanoids and figuring out their skeletal anatomy (e.g. would a mermaid have a pelvis?). I have also expanded the line of prehistoric creatures to both add more customer favourites, and to make more anatomically-correct versions (e.g. the True Rex isn't built on a tail-dragging kangaroo concept). I am always getting suggestions from our customers, whether it be from palaeontologists (who explained that T-Rex would have had a tail print in the fossil record with the fossilised footprints if she had dragged it) to six-year-olds who explain how to identify a proto-mammal based on the number of holes in the skull. My job is a fun mixture of design, anatomy, palaeontology, and shooting a laser beam! RAWR! □



**Right** □  
TINYSAURS began life at NYC Resistor – a makerspace in Brooklyn, New York City

# KABLACKOUT

## AIRSHIP



**B**

ased in Aghalee, Northern Ireland, KaBlackout is a duo of papercraft designers who have created a range of digital papercraft templates [writes Colin Johnston].

Our aim is to enable anyone to make their own beautiful piece of home décor from paper. Paper is such a wonderful and diverse medium to work with. It's also cheap and available everywhere! We use everything from wallpaper to cereal boxes for our creations.

We start our work process by designing an original 3D computer model. From this we can create a 2D digital template, which we use to build a prototype from paper. This is usually the first of many test builds, as we have to refine our design and repeat the testing phase until we're happy with the template. Our aim is to create a paper sculpture, that not only looks great, but is also easy to craft. Building the sculpture should be a smooth, hassle-free experience as we have already done all the engineering for the user.

One of the most rewarding aspects of our work is when crafters send us photographs showing their creations made from our templates. We especially like to see how our templates have been embellished or used in ways we never expected when they were designed, such as 3D cake-making, window displays, or photo-shoot props.

The amazing creativity that people bring to our templates is wonderful to see! □

**Above □**

For more affordable and devilishly clever paper designs, visit [kablackout.com](http://kablackout.com)



# DIANA BELTRAN HERRERA

## BIRDS

Diana Beltran Herrera is a Colombian designer and artist who works with ordinary materials such as wire, cardboard, plastic, and paper, the primary medium in the production of her work. One of her most extensive series is her work with birds, insects, fish, and plants.

She lives in Bristol, where she recently graduated with distinction from an MA in Fine Arts at the UWE, (University of the West of England). Recent commissions include a piece for the Festival of Ideas (programme developed by Bristol City Council), and her first solo show in Asia (Jakarta, Indonesia) supported by the Embassy of Colombia in Indonesia, and Cancillería de Colombia en el Exterior.

[dianabeltranherrera.com](http://dianabeltranherrera.com) □



**Right** □  
In contrast to the low-poly look of most papercraft, Diana's work is incredibly intricate

# STUDIO66DESIGNS

## DARTH VADER AND FIRST ORDER STORMTROOPER BUSTS

**A**lexey Odincov of Studio66Designs has been working as a low-poly 3D artist since 2012, when he came across a post on Pinterest about papercraft. Being a fan of the *Star Wars* universe, he decided to implement a long-standing plan – to create a paper sculpture of the First Order Stormtrooper, and then Darth Vader. If the current IndieGogo funding goes well ([hsmag.cc/KEsATF](http://hsmag.cc/KEsATF)), he plans to create a whole line of sculptures such as Boba Fett, Imperial Stormtrooper, C-3PO, etc. □



**Right ↗**  
For this Sith Lord, Alex recommends you use at least 160 gsm paper (preferably 220gsm)

# MAKING A PAPER AUTOMATON

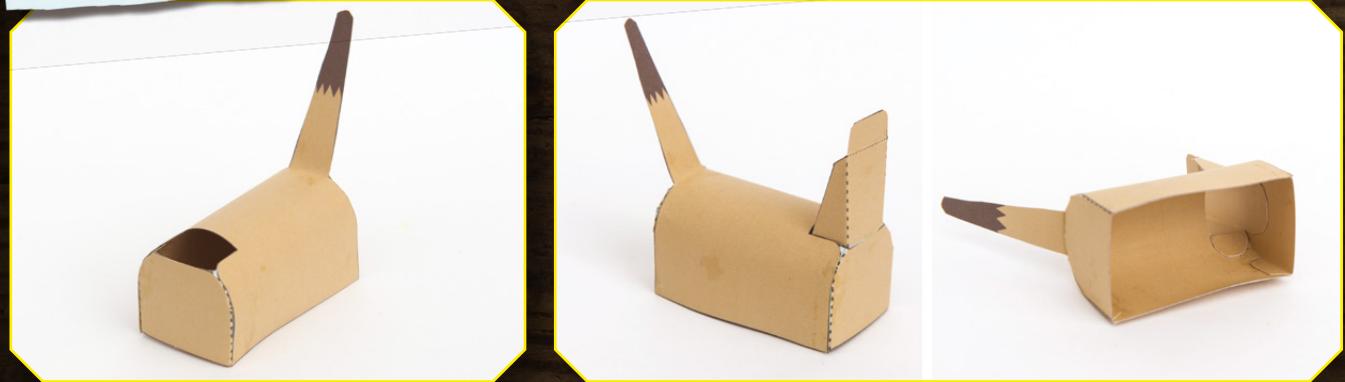
Add motion to your paper engineering with a handle and a cam

PRINT  
EDITION  
ONLY

In the back of this magazine, you'll find the parts to make your very own paper automaton. Turn the handle and the cute dog barks, powered by the cam in the box.

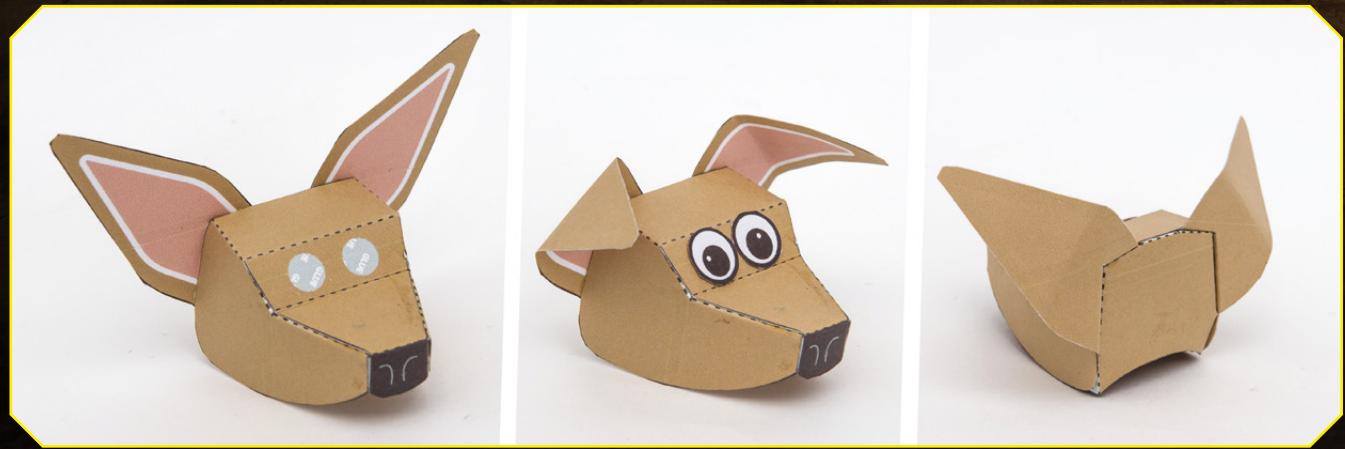
Score the dotted and dashed lines and cut out the holes, before carefully cutting out the parts. Try to be as neat and accurate as possible for best results! Dotted lines are valley folds, dashed lines are hill folds.

By Rob Ives

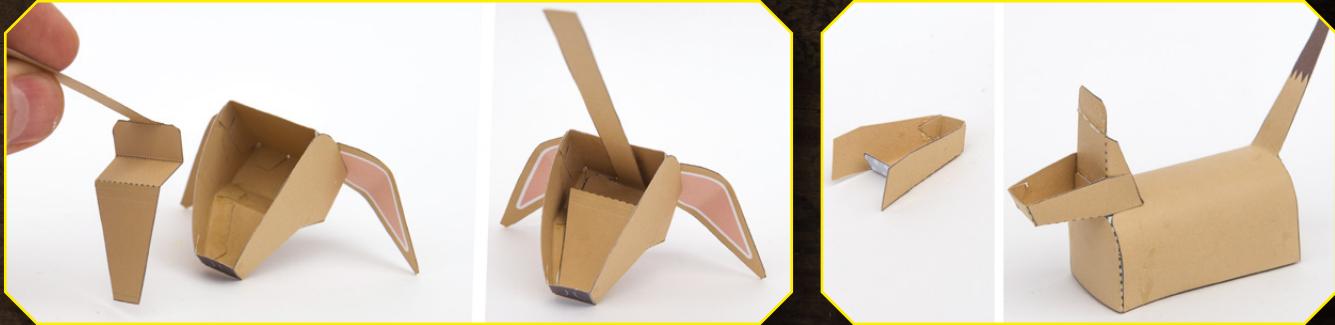


**01** Gently curve the Body round and glue the Body End and Body Back into position.

**02** Fit the Neck into position, gluing the centre tab to the inside of the Body Front and the side tabs to the underside of the Body.

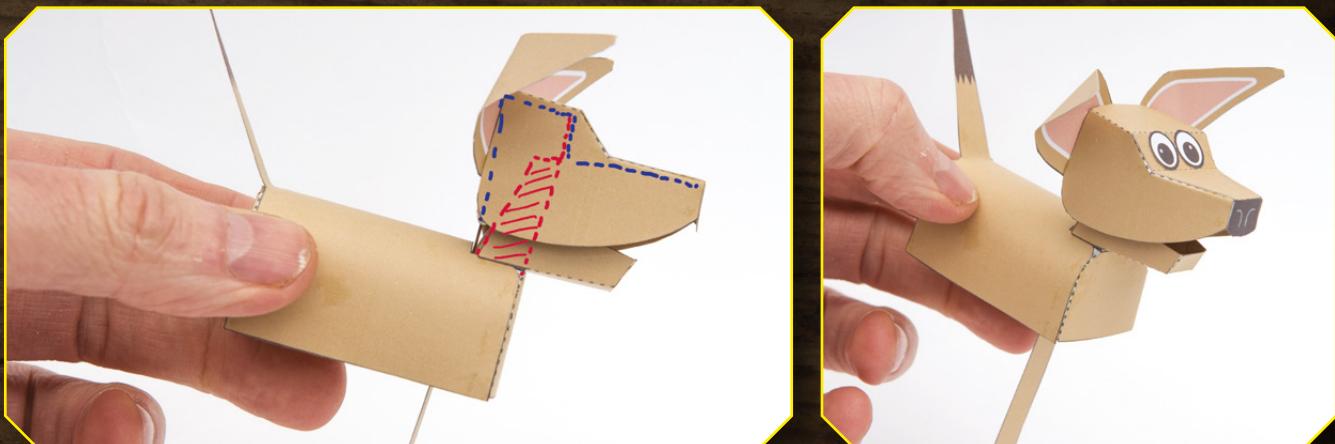


**03** Fold up and assemble the Head and glue the Ears into place. Curve the Ears round as shown, then glue on the Eyes. Glue the Head Back into place.



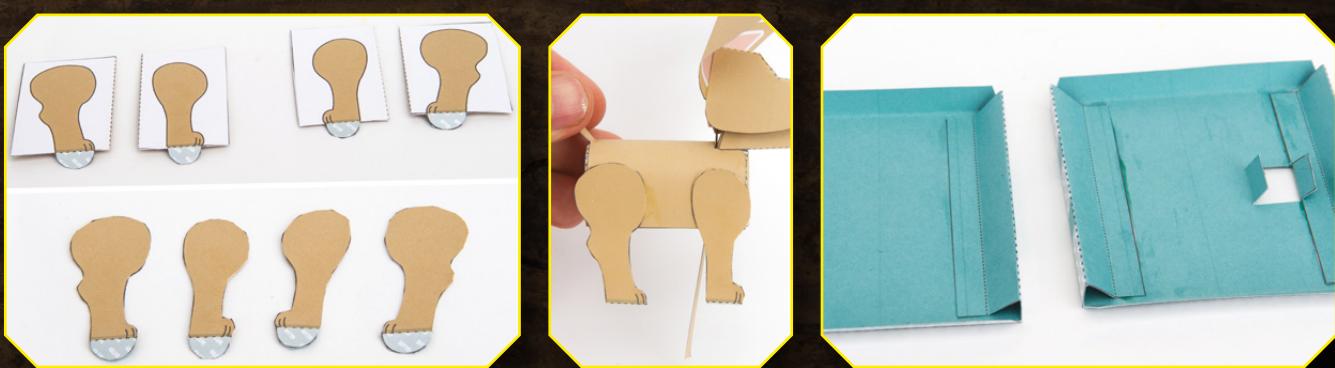
**04** Fold up the Head Inner & Push Rod as shown, and glue it to the underside of the snout and the underside of the top of the head.

**05** Assemble the Lower Jaw and glue it to the Neck.



**06** Exercise the tab on the top of the Neck so that it moves easily. Glue the tab to the vertical section of the Head Inner & Push Rod piece. Avoid getting glue on the hinge joint.

**07** Check that the Head moves freely by pulling the Push Rod.



**08** The Legs are made from double thickness card. Fold them over and glue them down before carefully cutting them out.

**09** Glue the Legs to the side of the Body.

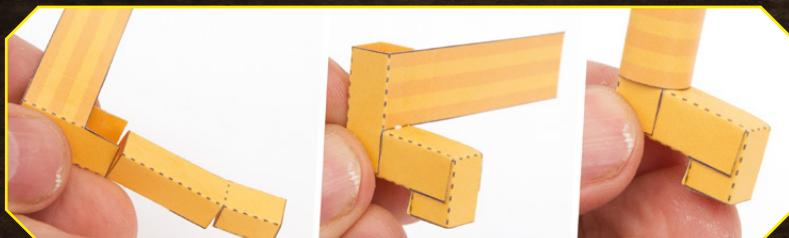
**10** Fold up and glue down the Box Base and Box Top to make right-angled triangle tubes.



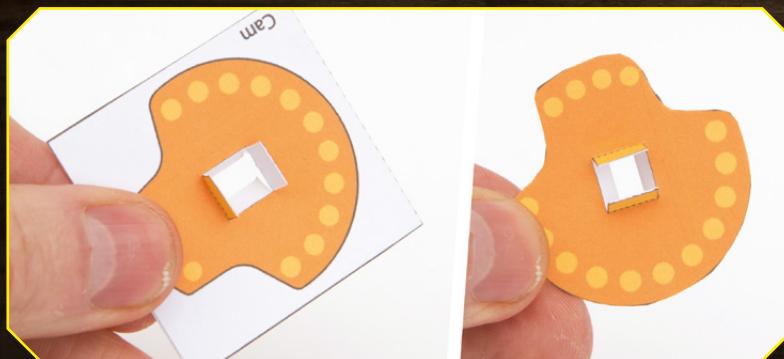
- 11** Fold up and glue the Box Sides to make equilateral triangle tubes.



- 13** Fold round and glue together the base. Glue down the tabs at the back, but leave the front tabs unglued.



- 14** Assemble the Handle in three steps as shown.



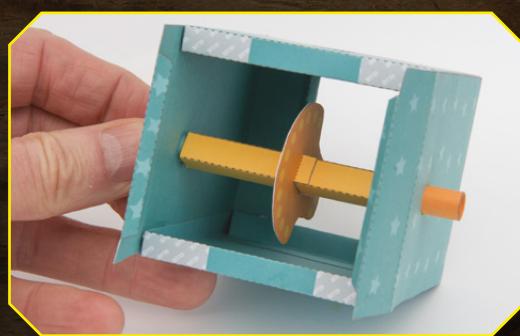
- 15** Fold over and glue together the Cam to make double thickness card. Don't glue down the tabs.



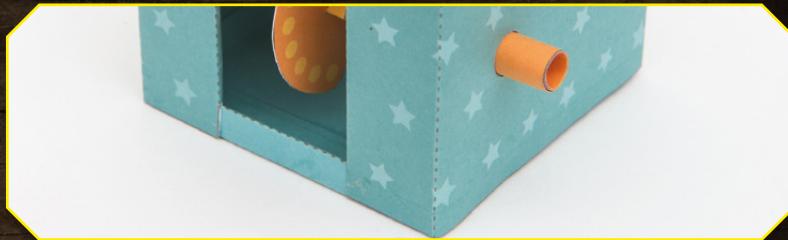
- 16** Roll up and glue the Axle Ends and Cam Follower End, lining up the edges as accurately as possible.



- 17** Thread the Cam onto the Axle, lining it up with the grey line. Glue down the tabs. Glue one of the Axle Ends into place.



- 18** Fit the Axle into the box, then thread the other Axle End into position through the hole in the box side, securing it with a dot of glue. Make sure that the Axle is free to turn.



**19** Glue closed the remaining flaps on the base.



**20** Assemble the Cam Follower.  
Fit the Cam Follower End into place as shown.



**21** Find a small coin, a UK penny is perfect, but any coin of roughly 20 mm diameter and 4 grams weight will do nicely. Wrap the coin tightly in an off-cut of paper. Glue it to the underside of the Cam Follower as shown.



**22** Fit the Cam Follower to the marked area on the base.



**23** Glue the Handle to the Axle tube.



**24** Stand the Dog on the top of the box without gluing it down. Thread the Push Rod down through the hole in the Box Top. Turn the Handle so that the Cam Follower is at its highest point of travel. With the Dog's mouth closed, glue the Push Rod to the Cam Follower. Glue the dog to the Box Top so that the Push Rod doesn't snag on the hole.

**25** Turn the handle to make the dog bark!

Share your build with us

@HackSpaceMag

HackSpaceMag



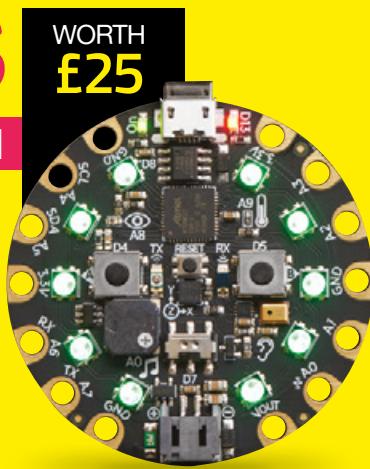
FREE

# CIRCUIT PLAYGROUND EXPRESS

WITH 12-MONTH PRINT SUBSCRIPTION



FROM JUST  
**£55**



12-month  
subscription  
from £55:

- UK: £55 per year
- EU: £80 per year
- US: £90 per year
- RoW: £95 per year

Visit: [hsmag.cc/subscribe](http://hsmag.cc/subscribe)

# SUBSCRIBER BENEFITS ↓

**SAVE UP TO 35% ON THE PRICE  
FREE DELIVERY TO YOUR DOOR  
EXCLUSIVE OFFERS AND GIFTS  
GET YOUR COPY BEFORE STORES**

## OTHER WAYS TO SUBSCRIBE

### **Rolling subscription from £4 a month:**

- Quick and easy to set up
- Cancel any time
- No long-term commitment
- No large up-front cost

### **Digital subscription from £2.29 a month:**

- Direct to your mobile
- For both Android & iPhone
- No delivery fees
- Back issues available



**Visit: [hsmag.cc/subscribe](http://hsmag.cc/subscribe)**

## How I Made

# RASPBERRY PI READING GLASSES

Using a Raspberry Pi and a camera to read text out loud

By Andrew Lewis

**T**here are approximately 370 000 people registered as blind or partially sighted in the UK. It's estimated that around half of those people suffer from age-related macular degeneration (AMD). While AMD doesn't cause total blindness, it affects the centre of a person's field of vision and makes reading text very difficult. Conventional magnifying glasses, Fresnel lenses, and other assistive devices can only help a person with AMD up to a certain point.

A Raspberry Pi, with a camera connected, can help people with AMD read text, using the Python programming language, computer vision, and a text-to-speech library. The flexibility and portability of the Raspberry Pi means that the system can take advantage of a pair of camera glasses, or use a fixed camera on a stand to recognise text.

This is a simple project to assemble, since it uses off-the-shelf components and requires minimal soldering. Although the project is presented here with a pair of camera glasses, it can be modified to work with a camera clipped to the frames of an existing pair of glasses or as a standalone



**Above** The Raspberry Pi Reading Glasses can be adapted to do much more than reading text out loud

desktop reader on a photographic copy stand. The most important part of the project is the choice of camera. A better quality camera will give you a better image, and that will give you better recognition results.

### JUST PLUG EVERYTHING IN

Begin by connecting the switch to the Raspberry Pi. The code in this article uses GPIO21, which relates to physical pin 40 on the GPIO header. Connect one side of the switch to pin 40, and the other side to pin 39 (GND). The cable should be long enough to reach the outside of your choice of enclosure, without having too much excess.

Fit the Raspberry Pi into its case and connect the battery pack to the Raspberry Pi with a USB cable. Try to use the shortest USB cable you can find, as this will of course look neater.

Plug the camera into the Raspberry Pi and then position the Raspberry Pi on top of the battery pack, so that the shape of the case provides some protection from damage. This position will vary depending on the choice of USB power supply you have and the case you are using. Once you are happy with the positioning of the

One morning, when Gregor Samsa awoke from troubled dreams, he found himself transformed in his bed into a horrible insect. He lay in his arms, his head down on his belly, and his head was surrounded by an enormous red and black mark, like the one a weaver蜜蜂 leaves on his loom. His eyes were filled with a kind of despair; he was hardy able to see anything.

components, use double-sided foam tape to hold them in place. You may also want to give extra protection to the USB and button cables by adding a small dab of hot glue, to stop them from working loose.

With the exception of positioning the switch in the device's carry case and plugging the headphones into the audio socket, this completes the hardware portion of the project assembly. The next step is to start working on a Python application to make everything work.

The Reading Glasses application doesn't have a screen, so the first thing that the application needs is a way to communicate with the user. The application will already be using a text-to-speech engine, so a combination of spoken word and sounds (earcons) can be used to provide suitable feedback to the user. Earcons are just short, uniquely recognisable sound files that get played to indicate something has happened. Playing these using Python is easy, as the following code demonstrates.

## What I used

- Raspberry Pi 3 and case
- USB battery pack and power cable
- Camera glasses, or a Raspberry Pi camera
- Push-button switch (normally open)
- Earphones
- Double-sided foam tape
- A camera case

**Left** Macular degeneration affects the centre of a person's vision, but the end result is not always as simple as a blank space. Sometimes, a jumbled or distorted image (or even a visual hallucination) may appear in the affected part of the eye

```
#use pygame mixer to play boot sounds
#talkey takes a long time to initialize,
#so do this now
import pygame
pygame.mixer.init(frequency=16000)
pygame.init()
pygame.mixer.music.load("startup.wav")
pygame.mixer.music.play()
```

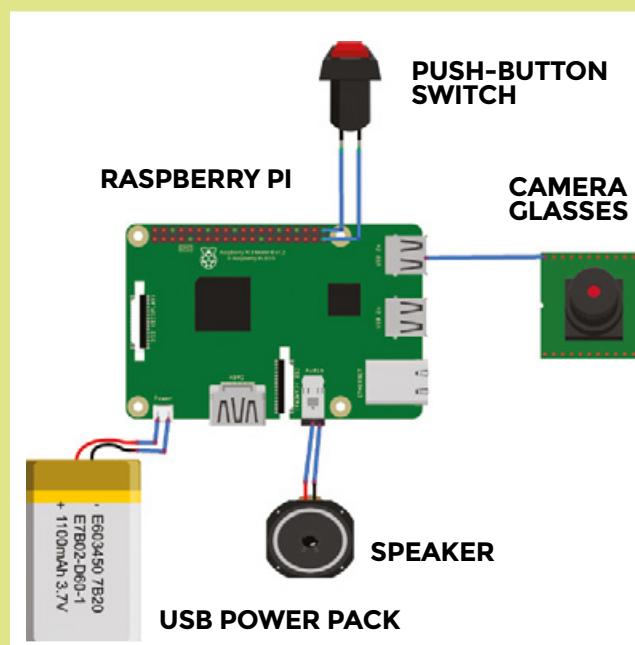
This code uses the Pygame library to load a file called **startup.wav** and play it through the default sound device. In this example, the sound file is a recording of a computer-generated voice, saying that the system is booting up, and that it might take a minute to initialise everything. A recording is used here because the Python text-to-speech engine takes several seconds to initialise, and the message is

played as soon as the application starts. This early feedback is important because it tells the user that the device is working, and that there will be a short delay before the next thing happens. Other messages generated after the text-to-speech engine has initialised will use the engine directly, but this early message needs to be pre-recorded.

Python has several different libraries available that deal with text-to-speech, and we are going to use the easiest of them, Talkey. Talkey acts as a front end to many different speech engines, and uses a simple syntax to read text strings out loud. While it's tempting to use something like Google's own voice synthesis to generate near-human-sounding voices, the Reading Glasses application would require an active network connection to do this, and so the self-contained Pico engine is used instead.

## GIVING FEEDBACK

There are several different states that the Reading Glasses can be in, and each of these states can be represented with a sound. Starting up the app has already been dealt with, but other messages will come directly from the text-to-speech engine, so this needs to be initialised →



**Left** Assembly of the parts is straightforward, with minimal soldering

## QUICK TIP

This project has several dependencies, all of which must be installed before you start: Python 2.7 with Pygame, Talkey, PIL, OpenCV, NumPy, RPi.GPIO, pytesseract, Pico text-to-speech engine, Tesseract.

# How I Made: Raspberry Pi reading glasses

## FEATURE



### Left

The physical effect of AMD on the retina is easy to see when viewed up close

device will vary, depending on the camera you use. In this example, the device has a frame width of 2304 pixels, and a height of 1536 pixels.

```
import cv2
import time
import pytesseract
import numpy
from PIL import Image

def getImage():
    # Try to open the camera, resolve if fails
    # returns an image from the camera
    cam = cv2.VideoCapture(1)
    cam.set(cv2.cv.CV_CAP_PROP_FRAME_WIDTH, 2304)
    cam.set(cv2.cv.CV_CAP_PROP_FRAME_HEIGHT, 1536)
    while not cam.isOpened():
        say(msg['camfail'])
        cam = cv2.VideoCapture(0)
        cv2.waitKey(10000)
    ret_val, image = cam.read()
    return image
```

It's likely that the camera will get accidentally disconnected from the Raspberry Pi at some point, particularly if you are using camera glasses with cables that can be snagged on clothing during use. You can see from this section of code how the application provides feedback to the user if the camera is not found.

The next part of the Reading Glasses application is the `main` function, which ties the other functions together. The `main`

next. All of the messages that the app uses are held in a single dictionary called `msg`. The following code sets this up, imports the TTS engine, and creates a function called `say()` that generates audible output.

```
msg={
    'camfail':'I\'m having trouble connecting to the camera. Check the cable is connected, or try holding down the camera button for 5 seconds.',
    'init':'Almost ready..',
    'ready':'Hello. I\'m ready to work now.',
    'notext':'Sorry, I couldn\'t find the text.'
}

import talkey
#setup the TTS engine and notifications - happens before everything else
tts=talkey.Talkey(engine_preference=['pico'])

def say(txt):
    tts.say(txt)
    print txt
```

`say(msg['init'])`

The app now has a method of output, but it doesn't have any way to get input from the user. This means activating the camera, and configuring the action of the button plugged into the Raspberry Pi. The camera is accessed through the OpenCV library, and the button uses the Raspberry Pi GPIO library. First, let's deal with the button by setting up the GPIO pins and activating a pull-up resistor to stop the input from switching randomly when not being pressed. In this example, the switch is connected to the GPIO21 pin.

```
# import the necessary packages
import RPi.GPIO as GPIO

# activate the pullup on the GPIO pins
GPIO.setmode(GPIO.BCM)
GPIO.setup(21, GPIO.IN, pull_up_down=GPIO.PUD_UP)
```

Now that the button is active, the camera needs to be configured. Properties like the frame height, width, and camera

## Failure

These reading glasses can be put through pretty tough usage. As they're important to the user, it's good to make sure they're made as solidly as possible. It's also good to keep a spare SD card loaded with the software that can be swapped in if there are any problems.

function checks the camera is working, and then tells the user that the system is ready. When the button is pressed, the application calls the `look()` function that takes a photo, detects the text, and reads it out loud.

```
def main():
    cycle_init=time.time() # this is for
    # the button-press debounce
    ret_val = getImage() #check the
    camera is working as soon as we start
    say(msg['ready'])
    while True:
        btn = GPIO.input(21)
        #if the button
        is pressed and not same press as last
        time
        if btn == False and time.time()-cycle_
        init > 1:
            look()
            cycle_init=time.time()

if __name__ == "__main__":
    main()
```

The `look()` function is too long to be reproduced completely in a magazine article, but the full code is available for download from [hsmag.cc/issue6](http://hsmag.cc/issue6). The

basic outline of what the function does is shown below, and the code itself is heavily commented.

1. The `look()` function takes an image from the camera and plays an earcon to let the user know what's happening.
2. The image is converted to greyscale, and areas of high contrast are detected. These high-contrast areas typically happen at the edges of letters on a page of text.
3. The contrasting areas are clustered together into 'blobs', and if the blobs are the right size they get added to a list of paragraphs. If you are having trouble getting the camera to recognise text, then try adjusting the values at the beginning of the `look()` function.
4. The blobs in the list are manipulated to remove any skew and rotation in the chunks of text, and then they get scaled and passed into Tesseract, which deals with the text recognition part of the function.

## Interaction

Getting feedback from an application or device is very important. When you touch a button, your action is usually acknowledged with a click or a beep. Lights, sounds, and text displays will tell you where an elevator is and where it will be calling next. Even virtual buttons on a smartphone screen will click or vibrate as a user pushes keys on the keyboard. Not providing adequate feedback for an action can make the user feel less confident about using an application or device, and may even mean that they stop using it altogether.

5. The Tesseract engine takes the images and converts them into strings of text.
6. The strings of text get passed to the text-to-speech engine, which reads them to the user.

To get the Reading Glasses up and running, you will need to get the Raspberry Pi to automatically execute the script when it boots. First, make sure that your reading glasses python script is called `glasses.py`, is located in the default user's home directory (`pi`), and has `#!/usr/bin/python` at the top of the file.

Now use nano to add a line to the bottom of `/etc/profile`:

```
sudo python /home/pi/glasses.py
```

These instructions should be all you need to create a basic set of Raspberry Pi Reading Glasses that can read text out loud to the user. There are lots of potential upgrades that you could make, especially if you are involved with computer vision. It's possible that gestural interfaces could be used to control the Reading Glasses, or you could add extra functions like facial recognition, object detection, or colour recognition. □



**Left** The assembled reading glasses, ready for use

## FEATURE

# OPEN SOURCE GLOBAL VILLAGE CONSTRUCTION SET

Everything you need to build a small-scale civilisation with modern comforts



Cameron Norris

[@cameronsnorris](#)

Cameron is a technology and communications specialist, passionate about the use of open source hardware for social innovation.



In a farm in rural Missouri, a group of engineers, agriculturalists, and self-sufficiency enthusiasts – led by a former physicist – are busy developing a library of open-source blueprints known collectively as

**the Global Village Construction Set.** These low-cost, easy-to-fabricate open-source machines include everything needed to build or rebuild a small, sustainable civilisation with modern comforts.

It all started when Marcin Jakubowski was unable to source affordable agricultural machinery for his 30-acre off-grid farm. Having earned a PhD in Fusion Physics, he'd begun to wonder how much knowledge was being lost in academia due to the fear of idea theft. So he founded the 'Factor e Farm' to see what it would take to build and support a small-scale civilisation.

Within the first year of moving to his Missouri farm, however, Marcin's commercial tractor broke down twice, and he soon realised that the low-cost tools required to support a sustainable off-grid lifestyle just didn't exist. "I needed tools that were robust, modular, highly efficient, optimised, low-cost, made from local and recycled materials that would last a lifetime, not designed for obsolescence. I found that I would have to build them myself," he explains.

With almost zero experience, Marcin built a new tractor from scratch using basic tools. He then shared his designs on a website he named the 'Open Source Ecology Wiki'. Many fellow visionaries soon joined him on the Factor e Farm to prototype modular, affordable, open-source hardware, which became known as the Global Village Construction Set (GVCS). Machines in the GVCS are designed to be integrated with one

**Right ♦**  
Housing in harmony  
with nature

**Credit**  
Open Source Ecology



another to improve the overall system's efficiency. For example, the LifeTrac tractor can be used to till the soil with removable blades via rotary action, before attaching its removable power source to the Compressed Earth Block Press to make bricks. As the name suggests, the LifeTrac tractor is designed to last a lifetime, requires little maintenance, and costs \$12 000 to build, compared to at least \$25 000 for one of the cheapest store-bought machines.

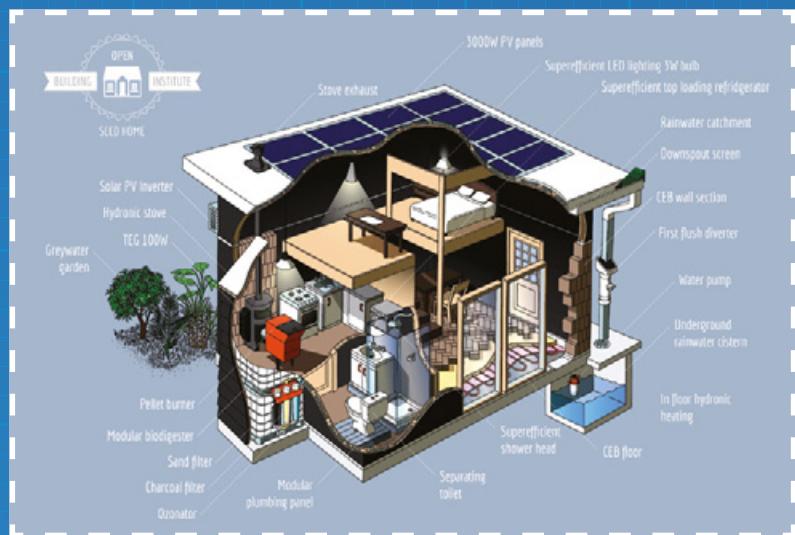
Andrew Spina from Baltimore, Maryland, USA joined Open Source Ecology specifically to build the LifeTrac tractor, "I want to understand it and be able to maintain it. It's also a project that will teach me many valuable skills. I've already learned a ton about machining and CAD," he reveals.

The Machine Index of the OSE Wiki describes 50 machines the Factor e Farm residents feel are necessary for a small off-grid community to sustain themselves. So far, twelve have been designed, documented, and prototyped on the farm. According to Marcin, each costs approximately one-third the purchase price of a comparable commercial model, including labour. What's more, the parts and power source for every machine are interchangeable. "It's designed for efficiency and cheap maintenance," says Marcin. "We have proven that the economic sense is there: the brick press costs \$5000 in materials; the nearest competitor costs you \$52 000."

## WE'RE TRYING TO REINVENT CIVILISATION

However, the original hope – that people would take the ideas and run with them, replicating the open-source technology and improving on it until it became commonplace – didn't happen.

"It's hard to replicate what we do," admits Marcin. "We have to train people. There's a lot of cultural clash with how people do things normally and how we do them, and so we decided to start this training programme."



**"We aim to tame the power of mechanical technology, by putting it in the hands of average people to use and modify according to their needs. Moreover, we will be developing PowerCubes running on steam power in addition to standard fuels, using pelletised biomass so you can make your own fuel as well"**

**Marcin Jakubowski**

With that, Catarina Mota, an accomplished researcher and open-source advocate who served on the board of directors of the Open Source Hardware Association (and who also happens to be Jakubowski's wife), spearheaded the creation of the Open Building Institute (OBI). OBI is an immersive training programme that teaches builders and entrepreneurs how to construct anything from a single-room studio to a large family home – all in a sustainable way that utilises aspects of the GVCS.

The OBI is Catarina's attempt to spread the open source construction model and raise public awareness of the cost savings potential the GVCS can provide to those seeking to build their own home.

Part of the OBI training programme demonstrates how a 700sqft (65m<sup>2</sup>) home can be built using locally-sourced materials for 10 percent of the cost of an average new home – furthermore, the construction takes just five days.

Marcin says he hopes aspiring entrepreneurs will be able to use what they learn to create new businesses: "We envision that DIY builders and startup entrepreneurs will be our early adopters, and we believe that some of our designs will be adopted by mainstream builders, once the open-source cost-to-performance advantage becomes clear." →

**Above** ▶  
Looking to build your own home? Start here

**Credit**  
Open Source Ecology

**Left** □  
Staying warm in winter is key to power efficiency

**Credit**  
Open Source Ecology

## FEATURE

# THE WORLD'S FIRST OPEN SOURCE TRACTOR

**Right □**  
The LifeTrac needs to handle muddy farm conditions

**Credit**  
Open Source Ecology

## DIY

Inspired by Marcin's TED Talk, Hayden Betts and Daniel Leon, two students from Polytechnic School in California, raised \$8200 by selling t-shirts, bumper stickers, and hats to pay for the materials required to construct their own LifeTrac. Hayden and Daniel then donated their Life Trac to the South Central Farmers Cooperative for use on their 120-acre farm that grows produce for underprivileged inner-city families.

**L**ifeTrac is a low-cost, multi-purpose open source tractor. The four-wheel drive, hydraulically driven, skid-steered tractor is intended to be a minimalist, but high-performance, workhorse and power unit for any off-grid operation.

The project aims to provide a life-size 'Lego Set' for constructing a wide range of heavy machinery, from small, two-wheeled garden tractors, such as the MicroTrac (a scaled-down, walk-behind version of the

full-sized LifeTrac) to 20 000 lb bulldozers, and everything in between. The machine uses most of the same components as LifeTrac to retain part interchangeability.

Marcin built the first LifeTrac in three months for \$6000 – about \$30 000 less than a comparable mass-produced model. Seeing room for improvement, he assembled a second prototype in just six days and, since starting Factor e Farm, Marcin has built a total of six tractors.



## I WAS AMAZED TO FIND THIS ACTUALLY WORKS

"A tractor is basically a solid box with wheels, each with a hydraulic motor," he said. "So, conceptually, it's actually very simple. And when I first did it, it was like, 'Wow, a tractor' ... I was amazed to find this actually works."

The latest iteration of the LifeTrac is powered by two modular motors, each a single-cylinder diesel engine of 27 hp that drives a hydraulic pump. Hydraulic motors drive the wheels, which operate in pairs to move the 4000 lb tractor on tank-like treads. When fully powered the machine is capable of moving up to two tonnes.

Standardized beams of 4"×4" rectangular tubing serve as the backbone of the LifeTrac, along with an XYZ space frame, bolt-together construction, and stock bonding and pivot plates that allow for scalable and pivoting joints. While long-term, reliable performance will tell if the LifeTrac is worthy of its name, the results after one year of operation are promising as the hydraulic motors and cylinders remain leak free.

"The highlight of our design is that it is modular and allows for a high degree of modification and flexibility – such as scalable frames, interchangeable power units, quick connect wheel drive, and plug-and-play hydraulic power," explains Marcin. He also notes that a Gasifier Burner may also be used for propulsion in a piston engine, and biodiesel from oil crops or the destructive distillation of wood (known as Fischer-Tropsch Synthesis) can also provide diesel substitutes. Furthermore, the inclusion of a 'Universal Rotor', designed to allow for different size and speed motors and shafts, means the tractor can be converted into a truck with little more than the addition of suspension or shocks. In addition to this, a 'Soil Pulveriser' has been designed to attach to the LifeTrac to be used to smash soil into a fine consistency suitable for producing Compressed Earth Bricks (CEB) using The Liberator, OSE's automatic, high throughput CEB Press. It can be used to produce up to ten blocks per minute, and the bill of materials cost of the machine is \$3000 – \$6500, depending on part availability and the build process used.

## QUITE LITERALLY, DIRT-CHEAP!

The CEB Press takes pulverised soil and uses it to form compressed earth bricks for building. Compressed earth bricks can be made directly from soil at the proposed building site, eliminating the need to transport bricks from elsewhere, theoretically reducing the cost and environmental impact to build.

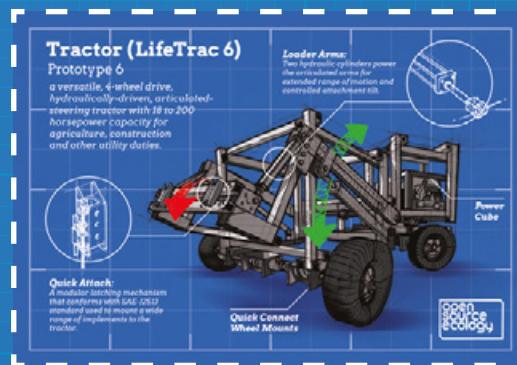


Compressed earth blocks are surprisingly strong and perform as excellent insulators against both heat and sound, making for a very energy-efficient building.

"I'd like to be able to show that a full modern standard of living can be created from any parcel of land using only the local resources on site in a small fraction of the time," adds Marcin.

The LifeTrac can be used to provide the Liberator with approximately two cubic yards of soil per hour, providing a six-block per minute pressing rate. A tractor loader is needed to keep up with the machine throughput as approximately 30 people would be required to load the machine manually with shovels and buckets without the LifeTrac. However, using these machines in conjunction, three people can build a six foot-high round wall, 20 feet in diameter, one foot thick, in an eight-hour day, though construction time will vary somewhat depending on preparation time and the quality of the soil.

"I've pressed 5000 bricks in one day from the dirt beneath my feet and built a tractor in six days. From what I've seen, this is only the beginning," explained Marcin. →



**Above** Testing the stability of the LifeTrac

**Credit**  
Sean Church

**Left** Up to 200 horse power to utilise for all your farming needs

**Credit**  
Open Source Ecology

# DITCH YOUR MORTGAGE FOR AN EARTH BRICK HOME

**B**riana Kufa, 20, of San Diego, is a certified welder who studied architectural engineering in college. Kufa came to the Factor e Farm to learn how to build the Liberator, the world's first automated, open source compressed earth brick press. "It's a lot of cutting, a lot of welding, a lot of torching, and that's basically all the skills you would need to make one of these," Briana explained. "I could probably teach someone how to do those things in a day."

Commercially manufactured earth brick presses sell for upwards of \$10000, but Open Source Ecology's Liberator can be constructed for around \$4000. In

Cedar Creek, Texas, a compressed earth brick press was built from scratch by the non-profit organisation, Creation Flame. With the help of Jason Smith, Tiffany Kerr, and many others, Creation Flame became the world's first replicator of the Liberator Compressed Earth Block Press, and the first independent party to successfully replicate any GVCS machine.

"I come from the software world. I didn't know how to weld. I hadn't had much machining experience," said James Slade, co-founder of Creation Flame. "It took some learning. We made some errors, but if I can do it, just about anybody can do it." Today, Jason works closely with Marcin and other OSE contributors on the design, fabrication, testing, and documentation of GVCS equipment.

## I COME FROM THE SOFTWARE WORLD. I DIDN'T KNOW HOW TO WELD

The Liberator CEB Press is the only GVCS machine to reach product release status, and bricks from the press have been used to build nine studios, a kitchen, bathroom, and 3000 square foot workshop on the Factor e Farm. These earth brick buildings are constructed with a dual brick wall design that creates a space between the two walls. This space forms an air transfer block with natural insulation and sound-proofing qualities. Furthermore, the thermal mass of compressed earth brick is effective in keeping the house cool in summer, while retaining heat longer in winter. Throughout several experiments, Marcin and Catarina learned that it is possible for a group of 35 people to build the shell of a 24x16 ft structure in just two days. Their latest goal is to develop all the modules necessary to build a Living Building Challenge

### Seed Home

A 700 sq ft 'seed home' packed with ecological features, including rainwater catchment, an ozone water filtration system, 3000 watt PV solar panels, super-efficient LED lighting, passive solar heating, a modular biodigester, and top loading refrigerator, will cost you less than \$25000 in materials.





**Left ♦**  
Make sure you're  
wearing eye  
protection when  
welding  
**Credit**  
Sean Church

compliant home, at a reasonable cost by July 2018. It's worth noting that the Living Building Challenge is the highest standard for eco-construction in the world.

"To make the house hackable, we focused on keeping all systems accessible," Catarina explains. "Rather than pouring concrete over the hydronic-heated floor water lines, we buried them in sand. If there is a leak in the system, we can lift the floorboards to repair it. The electric lines are not embedded in the walls, they run along the ceiling edge, inside an easily accessible channel. And the water lines run along the edge of the rooms, in a channel under the floor. If there is a leak or if we wish to add another valve, we can simply remove the boards to make the necessary repairs or changes."

Marcin and Catarina are now offering a library of modular home designs that anyone can download and import into Sweet Home 3D, an open-source interior design software. The library of modules is also open source, and anyone can contribute their own designs to it, provided they meet the Open Building Institute requirements.

"The design itself will dictate the way that you build, but the number of options you can build is infinite," says Marcin. Any new designs that meet the OBI's requirements are automatically published to the library of designs, but a team of advisory architects and ecological building experts select the best designs to be featured in the website gallery.

One of the greatest functional benefits of Marcin and Catarina's modular approach to construction is the relative ease with which you can add and remove entire sections of a building. A large expansion with the same functionality as the starter home is expected to cost around \$12 000 for a 225 sq ft addition, with each additional 225 sq ft anticipated to cost \$5000.

**"Can you imagine how much lighter your financial burden would be if your house was made from renewable locally-sourced materials, consumed significantly less resources, and produced most of its own energy, water and food? Eco-houses should cost less – not more – than standard houses. This is what we see as the future of environmentally and economically sustainable building"**

**Catarina Mota**

Thanks to Marcin and Catarina's work, eco-friendly housing represents a more affordable and more accessible option to a wider range of people than ever before. "People are hungry for meaning and authenticity in today's world. Part of such meaning comes only from seizing one's raw productive power," says Marcin. "All of the technology for building an autonomous house exists, it's just a matter of harnessing them all under one roof."

For more information visit [opensourceecology.org](http://opensourceecology.org). □



**Below ♦**  
We don't recommend  
sledge hammering  
your power cube  
**Credit**  
Sean Church

There's life in the old radio yet

FEATURE

# THERE'S LIFE IN THE OLD RADIO YET



A gentle introduction to a whole world of radio technologies

**t seems difficult to believe when sitting at a bench surrounded by microcontroller boards and Internet of Things devices, but there was once a time when to a large extent electronics meant simply radio.**

We take radio communication so much for granted that it has now become merely a component in the form of a little wireless module, but in the earlier part of the 20th century it was the miracle of the age. Amateur radio was at the cutting edge rather than a niche hobby, and to open an electronics book was to see a comprehensive selection of radio receiver designs.

The close association between radio and electronics persisted into the solid state era, to the extent that through the 1950s and 1960s the word 'transistor' was often synonymous with 'small portable radio'. An introduction to hobby electronics would

have started with a crystal set and, where a modern technically inclined teen might have an Arduino or a Raspberry Pi, their 1960s equivalent would have had a transistor regenerative AM receiver. The technological advancements that have given us the huge range of exciting projects we have at our fingertips today have been very beneficial to our community but, in leaving simple radios behind, they have allowed us to ignore a fascinating corner of the world of electronics. It's worth taking another look at these technologies not to wallow in the past, but simply because they are interesting.

There are two halves to every radio system: a transmitter and a receiver. Radio transmission is tightly regulated in all countries, so while we are talking about transmitters here, they are only for illustrative purposes unless you happen to possess a legal authorisation such as an amateur radio licence.

Radio waves are simply electromagnetic radiation, the same as light or radiated heat. They consist of an electric and magnetic field that oscillates from one polarity to the other, and their frequency is the number of times that oscillation occurs per second. For example in the UK, the BBC Radio 5 Live AM transmitter on 909 kHz oscillates between polarities 909 000 times per second.

The simplest radio transmitter produces this oscillation as an electrical waveform, and induces the radiation by passing that waveform to an antenna. →

**Left** The Regency TR-1 from 1954, the world's first commercial transistor radio

**Credit** Joe Haupt [CC BY-SA 2.0], via Wikimedia Commons

## BUILDING A CRYSTAL SET

The simplest possible radio is the crystal set, a tuned circuit coupled to a high-impedance earpiece through a germanium or Schottky signal diode. A quick web search will reveal multiple crystal radio kits, but this is a very easy circuit to build from parts you have found for yourself or retrieved from an older scrap AM radio.

Looking at each part individually, if you can't find them in a scrap AM radio, you can easily buy both the tuning capacitor and ferrite rod for the antenna from multiple suppliers including Rapid or Bitsbox. Your scrap radio may provide you with a ready-wound coil, but if not then you should wind 50 turns of enamelled copper wire on the ferrite rod and secure it with tape.

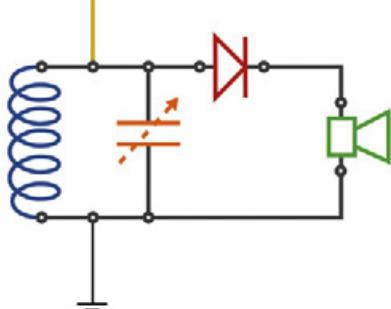
The diode should be a germanium point-contact type, which was once ubiquitous but is now obsolete. You will find them in older scrap radios and TV sets, but Bitsbox carries the 1N34A type, should you need to buy one.

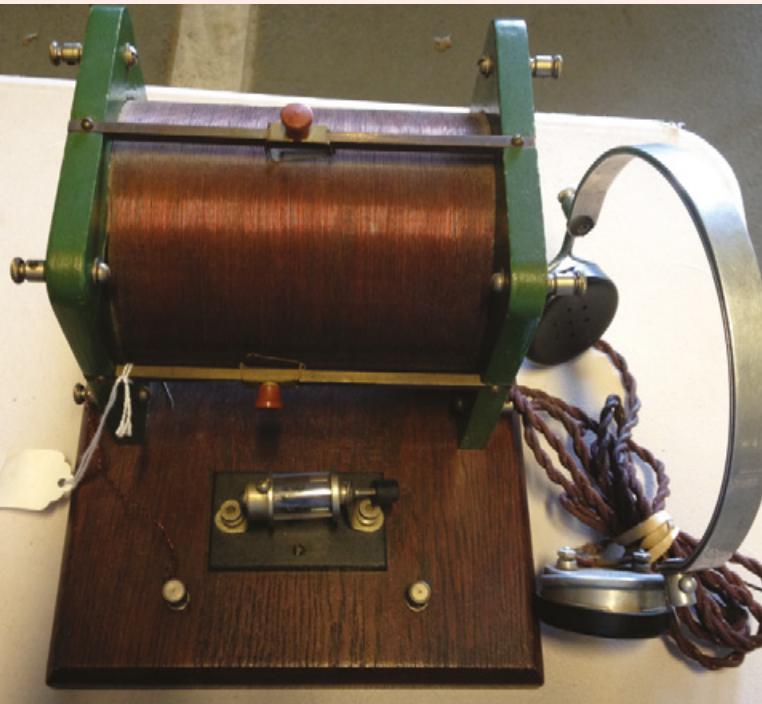
A high-impedance earpiece such as older telephone units, army surplus headphones, or a crystal earpiece completes the component list. The crystal earpiece should be available from multiple suppliers.

### Supplier addresses

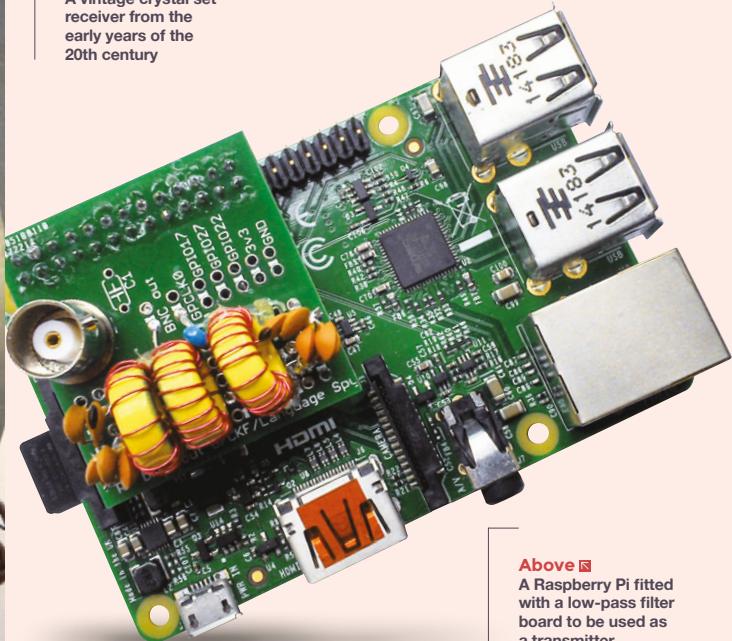
Bitsbox ([bitsbox.co.uk](http://bitsbox.co.uk))  
Rapid ([rapidonline.com](http://rapidonline.com))

**Below** The circuit diagram of the simplest crystal set receiver





**Left** A vintage crystal set receiver from the early years of the 20th century



**Above** A Raspberry Pi fitted with a low-pass filter board to be used as a transmitter

## BOTH TRANSMITTER AND RECEIVER DESIGNS CAN BE ANYTHING FROM THE VERY SIMPLE TO THE EXTREMELY COMPLEX

The radiation travels at the speed of light, so the distance between the start of each oscillation and its end, for a given frequency, will always be the same. This is called the wavelength, and most antenna designs achieve maximum efficiency by matching their dimensions to a fraction of the wavelength at their chosen frequency.

So, if you build an oscillator and connect it to an antenna, you have created a radio transmitter. Were you to listen to it though, you would hear only silence. It has no speech, music, or anything else applied to it. To attach some information to your transmission, you must modulate the information upon it, and there are many different ways in which this can be done depending upon the information in hand.

For analogue broadcast radio, you will probably recognise the two different modulation schemes as AM, for Amplitude Modulation, and FM, for Frequency Modulation. In an AM transmitter, the speech or music is mixed with the radio frequency signal to produce a result with varying amplitude, while in an FM transmitter the frequency of the radio signal changes with the speech or music waveform.

### COMPLETING THE CIRCUIT

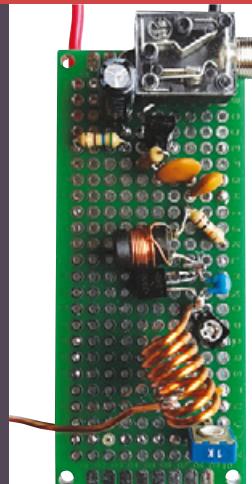
Both forms of transmission can be received with relatively simple circuitry, but an AM transmitter can be constructed with the bare minimum of components. A radio receiver captures the radio waves from

electromagnetic radiation through air and space into an electrical signal which it then amplifies and demodulates to retrieve information such as speech or music. It must possess selectivity, or the ability to narrow down to a single signal on one frequency, among the many others that will be pulled in by its antenna, and in all but the very simplest of receivers it must also have sufficient gain to amplify any signals such that they can be demodulated.

With well over a hundred years of development behind them, both transmitter and receiver designs can be anything from the very simple to the extremely complex, and leave many opportunities for the experimenter and home constructor. Surprisingly this does not mean only the oldest of designs – the latest software-defined radios have provided a new vista for anyone who wishes to tinker with radio, through free software such as GNU Radio. It's worth looking at some of the simplest radios to give a basic introduction to radio technology. We'll show you the simplest of AM receivers, a simple regenerative FM receiver, and – even though it is illegal to operate without a licence in many territories – we will also show you a transmitter to help illustrate some of the technology.

The earliest radio systems were developed long before transistors or even valves or tubes were developed. Their owners didn't have the luxury of amplification, and thus their receivers had to work using only the components available to them. The crystal set, so named because its earliest versions used a rough germanium crystal, is the simplest possible radio receiver and receives AM broadcasts. It uses a minimum of parts and has the handy bonus of not requiring any →

## FM REGENERATIVE RECEIVER



**Above** Our prototype two-transistor regenerative FM radio receiver

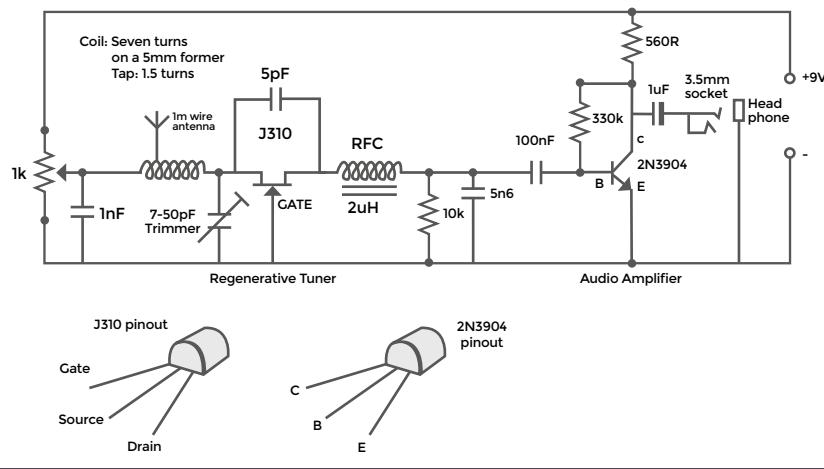
**Below** The circuit diagram of the two-transistor FM receiver

For the more advanced or confident constructor, it is possible to make a simple FM receiver with only two transistors. This is a regenerative receiver using a J310 field-effect transistor and a 2N3904 bipolar transistor as an audio amplifier, that can be built on a piece of prototyping board, as long as it is made with care to keep all component leads as short as possible.

All the parts should be available from most component suppliers – try either Bitsbox or Rapid if you draw a blank. The coil is seven turns of stiff enamelled copper wire wound on a 5 mm former, which is removed to leave a free-standing air-cored coil. You will need to scrape a little of the enamel off at about one-and-a-half turns, to solder on a 1 m antenna wire. If you don't have any enamelled copper wire, follow the example in our prototype and scavenge some from a toroidal mains choke found in a dead PC ATX power supply.

In use, this radio is a little more tricky than those you may be used to because, in addition to the trimmer capacitor which is the tuning control, it has the 1 kΩ variable resistor which is a regeneration control. You should adjust this to the point at which you hear the noise in your headphones, and tune the radio to a station. The regeneration may need adjustment for each station and, if you can find a plastic screwdriver for the tuning, you will find that it does not cut out while you are adjusting it. With some trimmers, you can fashion a tuning tool from a matchstick.

Our prototype was able to receive several strong local FM stations in this way. It may not perform as well as a commercial radio, but for its simplicity it does quite an impressive job.



## FEATURE

batteries, but with the penalty of working best with the strongest of stations. It consists of only four components: a coil of wire and a tuning capacitor that together make a tuned circuit to select a particular frequency, a germanium diode to recover the speech or music signal from the radio signal, and a crystal earpiece to play the sounds to the listener.

A crystal set may be extremely simple but, given a long piece of wire as an antenna, it can deliver surprisingly good

results for very little outlay. With a bit of searching, the parts can be scavenged from scrap electronic devices, making it an extremely cheap first radio project.

If AM and a crystal set is a little tame for you, simple radios don't stop there. The regenerative receiver was one of the earliest improved radio designs, and uses a single transistor or tube as an amplifier adjusted to the point at which it is almost oscillating. At that point its selectivity is hugely enhanced, making it much more

sensitive on the particular frequency it is tuned to. Regenerative radios can be made to work at many frequencies – including the FM broadcast band, where they are probably the simplest way possible to make a receiver.

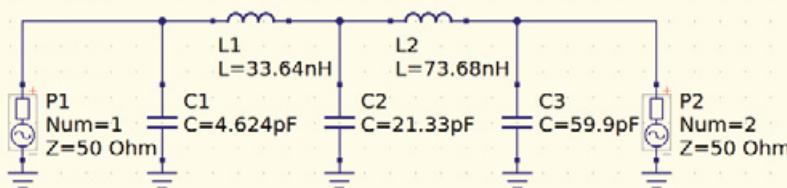
### THIS IS LONDON CALLING

Probably the simplest transmitter possible for most readers will come from a surprising source. The Raspberry Pi contains an oscillator designed to provide a clock signal to peripherals, which some clever hardware hackers realised could be repurposed into a low-power transmitter. The PiFM package can be readily downloaded and, as its name suggests, it turns the Pi into an FM broadcast transmitter. By connecting a piece of wire to a GPIO pin and running it, you can transmit for a short distance, perhaps throughout your house.

As we mentioned earlier, though, running your own transmitter without a licence is illegal in most countries. This is partly for bureaucratic reasons, but also it is this way because part of the licence conditions mandate that any transmitters have good spectral purity (which is to say that they must transmit on only one frequency). If your transmitter also interferes with your local emergency services or air traffic control, then obviously it shouldn't be allowed to continue, and that is why government agencies, such as OFCOM in the UK or the FCC in the USA, have teams enforcing the technical side of the licence conditions. With a tiny transmitter such as the Raspberry Pi you might get away with it, but to ensure that it or any other transmitter does not emit on other frequencies, you will also require a filter on its output.

## HARMONICS AND LOW-PASS FILTERS

A transmitter that is capable of producing a pure sine wave should in theory only emit one single frequency. Unfortunately it is almost impossible to produce such a perfect transmitter, and inevitably any real-world device will produce an element of distortion. This distortion appears as spurious frequencies at multiples of the original, referred to as harmonics. A transmitter producing square waves, such as those from the Raspberry Pi's clock generator, will contain a significant proportion of these harmonics, enough to cause interference to radio users on other frequencies. All transmitters will therefore contain some form of low-pass filter designed to only let through frequencies below a certain point. The low-pass filter is a network of inductors and capacitors calculated for a particular cut-off frequency. The figure shows a design calculated by the Qucs circuit simulation package for a filter with a 120 MHz cut-off. The component values are calculated by the software; a real-world version of this filter would use the closest available off-the-shelf values.



### S parameter simulation

SP1  
Type=log  
Start=12MHz  
Stop=1.2GHz  
Points=201

Equation  
Eqn1  
 $dBS21 = dB(S[2,1])$   
 $dBS11 = dB(S[1,1])$

Bessel low-pass filter  
120MHz cutoff, pi-type,  
impedance matching 50 Ohm

Above  
A low-pass filter designed using  
the Qucs simulation package

# BY CONNECTING A PIECE OF WIRE TO A GPIO PIN AND RUNNING IT. YOU CAN TRANSMIT FOR A SHORT DISTANCE, PERHAPS THROUGHOUT YOUR HOUSE

A low-pass filter (a network of capacitors and inductors calculated to let through only frequencies below a certain point) removes anything above the desired frequency.

Calculating component values for such a filter is beyond the scope of this article, but fortunately you can locate plenty of freely available software such as the Qucs simulation suite ([hsmag.cc/nuYhkD](https://hsmag.cc/nuYhkD)), which can be a great help if you want to design one for yourself.

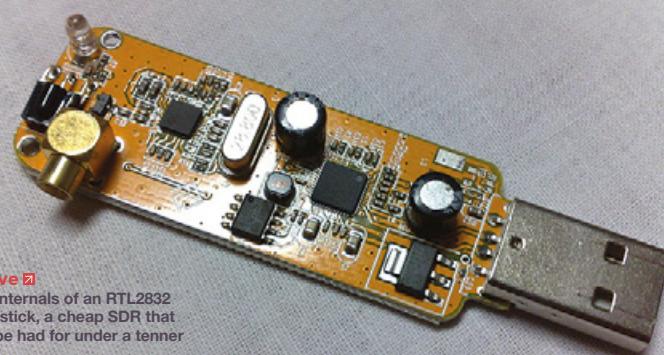
## AMATEUR HOUR

If you are new to radio, then maybe the information here has given you something of a taster for what can be an extremely interesting and multifaceted field of electronics. You may be happy to play with a simple regenerative receiver or a cheap RTL software-defined receiver but, if your interest goes further, the field of amateur

## SOFTWARE-DEFINED RADIO ON THE CHEAP

The radios described so far owe their roots to a much earlier age in radio experimentation. The cutting edge of the radio engineer's art lies in software-defined radio, or SDR, in which the hardware simply digitises a piece of radio spectrum into a computer, and all the signal processing work is performed in software.

Unusually for a cutting-edge technology, SDRs are extremely affordable thanks to a happy accident when a commodity USB TV receiver chipset was found to have an undocumented mode allowing it to be used as an SDR. The Realtek RTL2832-based USB sticks can be bought for under £10, and readily form a software-defined receiver that can have a bandwidth from 30 MHz to 2 GHz. They work with the free GQRX, SDRSharp, and GNU Radio software, delivering a lot of scope for radio experimentation at pocket-money prices.



**Above**

The internals of an RTL2832 USB stick, a cheap SDR that can be had for under a tenner

**Credit**

Dsimic [CC BY-SA 4.0], via Wikimedia Commons

radio is a logical next step. As part of the global regulation of the radio spectrum, there are multiple internationally agreed frequency bands upon which

private experimenters can get a licence to operate using any transmitter that meets the regulations, including ones they have made themselves. The result opens up a hugely varied array of different radio technologies, from satellites through computerised data modes to atmospheric propagation research, television, and much more.

Different countries have their own routes to getting an amateur radio licence, but in most cases one can be yours for passing a technical examination.

In the UK you can find more information from the RSGB ([rsgb.org](http://rsgb.org)), and in the USA from the ARRL ([arrl.org](http://arrl.org)), but then all other countries will in turn have their own similar organisations.

However you experiment with radio, though, have fun and let us know what you build. Tweet us at [@HackSpaceMag](https://twitter.com/HackSpaceMag) or email at [hackspace@raspberrypi.org](mailto:hackspace@raspberrypi.org). □

HackSpace magazine meets...

# ALEC STEELE

Because there's nothing cooler than hot metal

**T**here's something literally awesome about **blacksmithing**. The heat, the noise, the glowing lump of metal at the heart of it all. It's incredible to think that people did this kind of work centuries ago without lathes or power tools or modern knowledge of chemistry.

It's almost as incredible to think that anyone's doing it nowadays – demand for horseshoes, weapons, and agricultural implements has dropped massively since the end of feudalism and the Industrial Revolution. For anyone to make a living as a blacksmith these days, they have to be doing something a bit special – and what Alec Steele's doing is special indeed.

We went to speak to Alec and take pictures of shiny stuff – we weren't disappointed. Read on for an instruction manual of how to make Damascus steel, why perfectionism gets things done, and the best reasons for making stuff that we've ever heard. →





Above ↗

"When I was 11 I saw this blacksmith hammering on a piece of steel and thought it was the craziest thing. Fire! Hammers! Loud noises and sparks! What more does any 11-year-old kid want?"

**HACKSPACE** First of all, tell us about Damascus steel: what is it, and how do you get those wonderful patterns in the metal?

**ALEC STEELE** The way I'm making Damascus steel now is I'm taking alloys of steel with different chemical compositions. The first alloy of steel is one that has a high carbon percentage, 0.8% carbon and not a whole lot else. There's a tiny little bit of manganese and stuff like that, but it's essentially plain 0.8% carbon steel. That holds a good edge, it's good for a knife.

What we then do is we add another steel to it that has 0.8% carbon, and also 2% nickel. That nickel doesn't affect the edge quality at all, it's still great knife steel, but when we make something, we then polish it. We put it in acid and the nickel will resist the etch. It's along the lines of how stainless steel is more resistant to staining because it has the chromium and sometimes also the nickel.

By bumping the nickel percentage up we increase its corrosion resistance, so we're able to put it in acid and have the plain carbon steel etch away, leaving the nickel steel proud on the surface, and we can then polish the nickel steel. It gives us a relief where we're able to polish and show the contrast in the patterns. That's the general principle behind it, and that's how we're getting the patterns in the steel.

**HS** How do you go about making two different styles of steel come together in one object?

**AS** We stack up the bits of steel – they start as individual little coupons of steel and we tack-weld them together, so they don't fall apart as they're going in and out of the fire. Then we go to the forge. We heat it up in the forge until it's a yellow or an orange temperature – not to welding temperature, but just to the degree that we can close it up so air is less likely to get in there. Then we can apply some flux. Flux

is borax, basically – disodium tetraborate. The flux supposedly helps clean your welding surfaces as it welds, and helps seal it from oxygen. You don't want any impurities in there, otherwise they're going to end up in the final piece. If you have a void in the final piece, you're going to have a big problem.

From there you heat it back up, to something like 1300–1400 degrees Celsius. At that point it's a bright, high yellow, almost white. The surface of the steel can sometimes look as though it's bubbling. It's as hot as it gets, and it can be painful to even look at if you're really staring at it. It's beyond red, it's beyond yellow, it's white. At that point

You're creating this interesting mosaic of steel as you continually develop the pattern and move it around to create different patterns

when you hammer it, or when you press it, the surface of the steel, the different layers are to some extent molten and will create a bond, but you still have the clear differentiation between one alloy and another.

Starting off, you have straight layers of steel. But when you draw it out, manipulate the pattern as you go, you cut it, re-stack it, you're creating this interesting mosaic of steel as you continually develop the pattern and move it around to create different patterns.

There are people who will twist the Damascus, there are people who will cut into it to reveal different layers in the steel, there's all sorts of stuff that you can do to make interesting patterns on the finished thing.

From there you forge the blade, and then you grind the blade and get ready for heat treat. The heat treat basically involves taking the steel in its normal state, heating it up, then quenching it in oil at the critical temperature. The critical

temperature is where there's like a phase change in the steel or something like that, where it converts to austenite. Then what you do is you cool it down very quickly so you can form martensite crystals (I think! I'm no scientist, but I'm pretty sure that's what's happening).

The practical reality of that is that you heat it up, you cool it down fast, it gets hard. It gets very brittle though – it can almost be as brittle as glass. So then what you do is you heat it to a lower temperature and hold it at that temperature for a certain amount of time. This is called tempering. That takes what is very, very hard and brittle, and you're allowing it to be tough, flexible, be able to keep an edge, but not have the edge break and be too friable, too brittle. So the tempering is important. These are the things that really make a blade. You're constantly trying to balance between cutting something really well and it not being able to cut after the first time you use it because it breaks or bends or something like that.

It's all about getting this equilibrium of hardness and toughness, and all about getting the right equilibrium of blade geometry that's going to be cutting through well enough, but also well enough supported for the task at hand.

**HS** Is this the same process no matter what you're making?

**AS** You don't put on a splitting axe the same edge that you put on a straight razor, because the splitting axe doesn't want that edge. That edge is going to roll over and chip and get damaged and be no use to the splitting axe. Whereas on the straight razor, while you still need to cut, you need a very thin edge, but you're not going to be subjecting it to a lot of impact, so you can have the thin edge. It's about creating the right balance. In making a blade, that's the start: it's the balance of the different attributes of what the steel will give you and what the geometry of the thing you're making will give you. →



Above "I'm really happy with the things that I've made. But that doesn't mean that I'm not committed to doing a hell of a lot better tomorrow."



Above Alec learned from books, asking experts, from internet forums – but most of all, by doing

So from there, you polish the thing. And it takes a long time and a lot of effort to polish it well enough that you can get a good etch. You have to take it from your belt grinder – it looks like you're leaving a flat finish, but a belt grinder won't leave a flat finish, there'll be a lot of scratches, and they'll be in the wrong direction. The scratches that a belt grinder makes are typically going laterally to the blade, which is problematic because the blade is not going to have a perfectly flat surface.

Grinding will stop at, say, 400 grit, but to be able to switch the scratch pattern around, we then have to go back to 80 or 120 grit with the hand sanding and put the scratches along the length of the blade, so that when it's etched you can have a nice polish. It takes hours and hours and hours to build those grit scratches back up to 400 grit so you can get the clean finish. When it's ready for the etch, we'll then go into a 10% ferric chloride solution.

#### **HS This is the whole blade we're talking about here, not just the cutting edge?**

**AS** Absolutely. We polish the whole blade and, to reveal the Damascus pattern, we then go into the etch, which is ferric chloride. You leave it in there until the carbon steel etches away and the nickel steel will stay where it is, so you're going to get this negative. Then after the etch, and various processes afterwards, you have to take a 2000 grit over the nickel steel to polish up the nickel, shine up the high spots. You get the beautiful contrast between the two, and that's essentially it. Obviously you need to sharpen the blade. Sharpening a blade is complex, but it can be made a lot more complex than it needs to be. What you're essentially trying to do is bring steel down to almost nothing, along a straight line, while making sure you don't have a burr that's going to mess with your cuts.

Getting back to reality, what I will do is take the edge on a secondary level,

usually to a 400-grit finish, then I'll take it to a buffer, which will then take the burr that's created from that, get it nice and clean and finally, get a sharp edge that will make a mess out a lamb chop.

#### **HS Sounds easy!**

**AS** It is. It's just really tiring.

**HS** **We made our own little knife in issue 2, the most basic knife possible, admittedly, and for that we used 1080 steel, so that terminology rings a bell. Just a blank of steel, get a pattern on paper, cut it out with a hacksaw, file, it, heat it, temper it (if I remember rightly, we used engine oil), and that felt really cool, that you can do that with just a blowtorch and a hammer. Nothing at all like this beautiful expanse of power tools that you've got here.**

**AS** I'm working on something similar right now. I'm making a blade with just basic hand tools. This here, it's all going to be finished out with just files and some sanding, just a hand hammer, an anvil with no complex bits of tooling. I'm going

**It takes incredible discipline, it takes incredible understanding of the basics, it takes incredible skill and knowledge, years of practice to be able to get to that level**

to be using a power drill to drill a hole in it, but everybody's got a hand drill right?

It can be made a lot simpler – making stuff with steel, making knives, it can be done with seriously limited tools. It's the fundamentals that are key. I don't consider myself a knife maker, I consider myself a blacksmith who's just getting into knife making. It's such a complex craft: the attention to detail that's required to make great blades that many thousands of extremely talented knife

makers and craftsmen out there have spent decades trying to acquire.

It takes incredible discipline, it takes incredible understanding of the basics, it takes incredible skill and knowledge, years of practice to be able to get to that level. The great thing is that someone who has got to that level would be able to take a bit of steel and basic tools and knock anything I could make out of the water in a similar amount of time, just because they know the basics so well. Because they know the foundational principles.

We think we need crazy amounts of tools to do anything, and it's nice to have all the tools, and if you can have the tools, it's great. But you can do superb work with very little, really. It's pretty cool to see when people are capable of pretty awesome work with minimal tooling.

**HS** **You make it sound like you're just an apprentice to your craft, when anyone looking around here and noting all the stuff you've made would probably think you're already damn good at the blacksmithing game. To me it looks like you've pretty much cracked it.**

**AS** No! I've not cracked it at all, honestly, I'm still very much a beginner. Every single day there's a torrent of information being thrown at my head from the material I'm working with. I'm learning so much. I am at the very beginning of my journey of craftsmanship, of learning stuff, of creating things; I like to think that's because I've got high standards of where I want to end up. I want to be able to make works of art that people are going to appreciate for generations to come.

And that's a high standard, you know. If you want to be making the best stuff that's going to be appreciated by your great-grandkids, you've got to set your standards high. It might look like I know what I'm doing, but not in the details that I want to be really, really focused on. I want to be able to have a very high degree of control over what I do, so that what ↵



### Left ♦

This Union Flag is in the steel itself, an effect achieved through careful beating and cutting

is made is exactly what is in my mind's eye. I don't want to let the material take control, I don't want to let the tools take control. I think that's one of the greatest things about craftsmanship, that people are able to get so incredibly skilled at a craft, to the point where there is nothing between what they imagine and what they create, other than their own sweat and hard work. That's real skill: they don't ever have to say "it turned out like this"; they say "I made it like this".

Every single scratch, every single micron on this blade is exactly how I intended it. There are people who can do that – OK, so maybe not to the micron, maybe every 10 or 50 microns. But where every 100th of an inch is exactly planned for, or exactly wanted. I'm not necessarily saying they drew it out, but they intended it to be exactly like this. There's not a single mistake in there, there's not a single thing that they haven't rectified to make better. There are people out there who can do that. And they do it every single day. That takes some serious discipline and an unbelievable dedication to the craft.

And that's why I describe myself as a beginner: the material is still taking me in certain directions. It's the ownership over the material that's the really exciting thing, and I think one of the biggest roots of craft in all of human pursuits. It's saying to the things around us, "I know

you want to be a mountain right here, but we're going to blow you up and put a road through you". That's what we do: we take ownership of the world and we make sure that we make what we want from it.

If there's a big river and we want to drive across it we say "OK, let's build a bridge". Engineers make sure they build a bridge and they make sure they build in the allowances to account for heat expansion and cooling, and the different seasons and account for exactly how much load's going to be put on it, and be sure that they build a bridge that is exactly what is required, provides what they want, and does the job to the greatest degree of perfection possible. That's what we do. We don't pick up an iPhone and say "Oh golly, my iPhone doesn't work today."

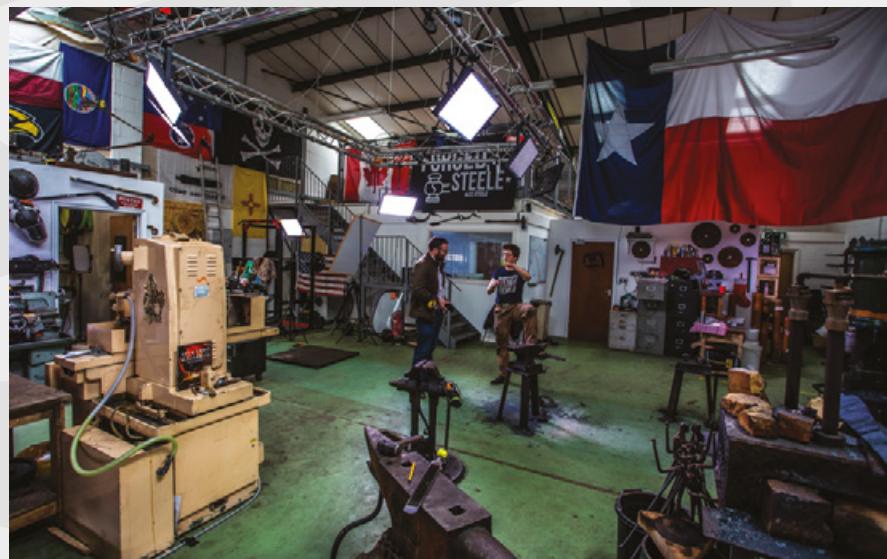
I guess the makers tried". We expect it to work. We expect it to be exactly what was intended. And this is the beautiful thing about craft. Generations behind us, people have made things with unbelievably minimal tools exactly as they intended, because of skill, because of determination, because of the discipline of their craft.

The first steam engine – imagine having to make the first steam engine with the tools that were available to you then! That took some doing. That took some serious thinking, some serious willpower to make right, some serious craftsmanship.

Something that I very often fall prey to every single day in the workshop is saying "that's good enough". We wouldn't have all the nice things we have in the world if people before us said "that's good enough".

**HS Arguably. Maybe we wouldn't have anything at all if nobody said "that's good enough".**

**AS** There's tolerance to it, right? You could say that you want a blade that's bang on 6.253 mm thick, and I end up with 6.2531...



### Right ♦

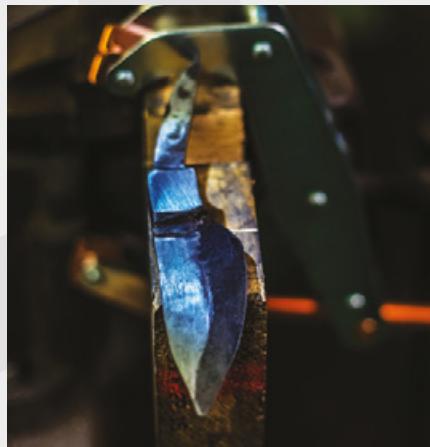
See if you can spot the Baratheon war hammer in this picture in time for series 8 of *Game of Thrones*

**Right ↗**

If you like long shots of sparks coming off hot bits of metal, you'll love Alec's YouTube channel: [hsmag.cc/ShEXZH](http://hsmag.cc/ShEXZH)

you're off, that's not perfect. Perfection doesn't exist; what there is is a degree of tolerance that you've got to work with. Obviously at some point you've got to say that's good enough. But the standard is very high historically, it's high in the cars we drive, of course. Tolerance is important and I don't want to act as if I'm trying to make sure that I can craft things to a micron by hand with a file and a toothbrush, for goodness' sake. What I mean is that the tolerance has to allow the thing to work to the best that it can.

**HS** What's the closest you've got to this mythical state of perfection? Do you have a favourite thing that you've made?

**Above ↗**

Alec's making this knife from a piece of 1080 steel using only hand tools – like we did in issue 2

**AS** I've not got anywhere near to the degree of excellence that I'd like to get in the pieces that I've made – again, I'm a beginner. But the best thing I've made so far is the kris. It's not an authentic kris by any stretch of the imagination, but what it is is a wavy blade. It has this sinusoidal shape, and it's a double bevel, a dagger with an 18-inch-long blade.

You have to have this bevel going up this sinusoidal shape, and you keep the bevel centred, and you make sure that you have an even finish the whole way down. It's made in Damascus steel, which itself requires a lot of thinking to get right (and I didn't get it as right as I wanted). It's got this integral guard, so instead of there being a guard that's placed on after the fact and it's attached on, it's forged out of the material. So the Damascus flows in and out of this guard. The transitions between these points are very difficult, it's very easy to mess it up.

That then transitions into a keyhole handle; keyhole was invented by Rodrigo Sfreddo, who is a Brazilian knife maker. You have this cut-out in the steel that looks like a keyhole, and you have to get the wood to fit in there tightly, you don't even have to pin it. You don't need to use any glue between the wood and the metal. It's so tight that there's no air, there are no gaps, the wood doesn't go anywhere because the pressure of the metal is holding it in there. It took ten tries for me to get the wood to fit right. I spent seven days alone on the handle of

this thing. I ended up using cocobolo; it has these beautiful carved facets where I was paying more attention than I've ever paid to making sure everything's straight, everything's neat and even and equal, so that when it gets picked up you don't look at it and look closely and go, "that doesn't look square, that doesn't look centred".

You're trying to do justice to the thing that you want to make. If you say that you want to make a nice clean piece of work, you've got to make sure that things are square and neat and centred, and your own two hands are to blame for every single fault along the way. Every single mistake in there is your fault. And that's pretty fun, because you can always improve: you're always going to be making mistakes.

I look up to incredible craftsmen who do incredible work and I go, "Oh my goodness, how could you do better than that?" But I tell you what: they're looking at the same things and going, "Man, I messed this up", looking at all the flaws that you or I would never notice. There are tens of millions of craftsmen who have gone before us who have made extremely clean work with only their two hands by having an extreme mastery of the craft. I find that really inspiring, and really exciting. What's to stop me from making better work? Nothing more than my own determination to make it happen. Nothing more than my ability to learn from my mistakes the day before. And hopefully improve and get better. □

# pi-top

Inspiring inventors and creators to seek the skills of tomorrow and create their future, today.

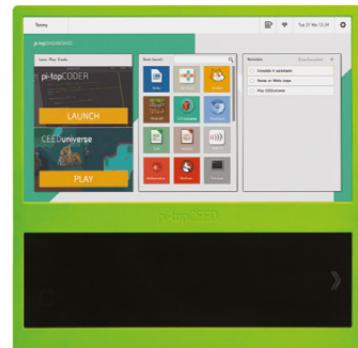


OCR  
Oxford Cambridge and RSA



The modular laptop with sliding keyboard

- 8HR BATTERY LIFE
- 14" FULL HD 1080P SCREEN
- 180° HINGE
- CUSTOM PASSIVE COOLING BRIDGE
- MODULAR RAIL



The modular desktop

- 14" FULL HD 1080P SCREEN
- MODULAR RAIL
- ADJUSTABLE VIEWING ANGLES



pi-topPROTO

pi-topSPEAKER

pi-topPULSE

## pi-top

Colors Raspberry Pi 3 optional

### AWESOME INVENTOR'S KIT INCLUDED

**20+** projects to explore

Explore beyond the screen and keyboard by creating with the all-new **pi-top** modular laptop.

Get started with 20+ inventions in the inventor's guide booklet. There are 3 inventor's journeys - Smart Robot, Music Maker and Space Race.

## pi-topCEED

Colors Raspberry Pi 3 optional

**pi-topCEED** is the plug & play modular desktop. It's the easiest way to use your Raspberry Pi. We've put what you love about our flagship laptop in a slimmer form factor. Join hundreds of code clubs and classrooms using **pi-topCEED** as their solution to Computer Science and STEAM-based learning.

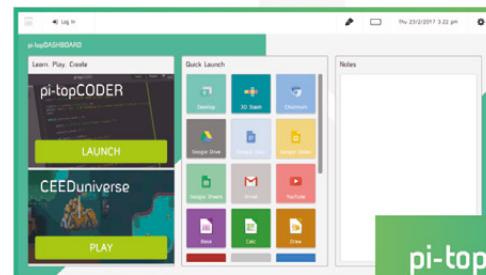
## Modular Accessories

Stay up to date with our latest news by following our social media

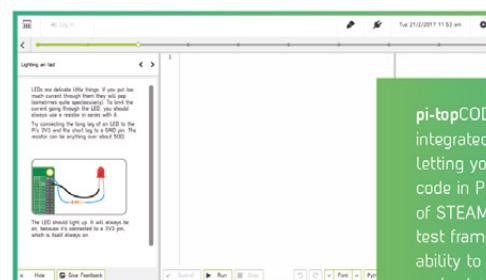


**pi-top** is an award-winning ecosystem designed to make experimenting, coding and building electronics, simple, affordable and fun. **pi-topOS** is here to guide you through the world of making!

The OCR\* endorsed **pi-topOS** (Operating system) platform comes pre-installed on the SD card shipped with every unit. **pi-topOS** software suite lets you - browse the web, - check emails, - create and edit Microsoft Office compatible files. Gain access to dozens of hands-on learning lesson plans with **pi-topCODER** and have fun learning to code with **CEEDuniverse**!



**pi-topOS**



**pi-topCODER** has a fully integrated coding environment letting you program hardware, code in Python and learn lots of STEAM skills! Our integrated test framework gives you the ability to assess your own understanding as you learn.



**CEEDuniverse**  
Learn programming concepts through our minigames, for example, learn problem decomposition by solving visual programming puzzles.

# MATCH STICKS



Mayank Sharma

@geekybodhi

Mayank is a Padawan maker with an irrational fear of drills. He likes to replicate electronic builds, and gets a kick out of hacking everyday objects creatively



For something that's essentially made up of just two components, a matchstick is an extremely dexterous tool

**H**istory would have us believe that the ability to start a fire has a direct correlation with the growth of human civilisation. It was our ancestors' desire to control fire that distinguished them from animals. While we gained control over fire more than a million years ago, matchsticks are a fairly recent invention. Some of the earliest examples of matches are the Chinese fire sticks from 577 AD that were used by the women of a town to ward off invaders.

Over the years intrepid alchemists have generated heat and fire with many chemical reactions. In 1680,

Sir Robert Boyle produced a flame by scratching a splinter of sulphur-coated wood on a piece of paper coated with phosphorus. Jean Chancel, the assistant of famous French chemist Louis Jacques

Thénard, gave us the first modern self-igniting match in 1805. The head of this match was made from a mixture of potassium chlorate, sulphur, sugar, and rubber, and the user

ignited the match by dipping it into a small bottle filled with sulphuric acid. The match didn't get much attention because of the dangerous and expensive mix of ingredients.

The first friction matches came along in 1826, when John Walker combined a paste of sulphur with gum, potassium chlorate, sugar, and antimony trisulphide, and ignited it by drawing the match between a fold of sandpaper. His matches however were considered too dangerous, and were even banned in France and Germany. Charles Sauria then reformulated the match using white phosphorus, which turned out to be dangerous in other ways.

The calls for banning white phosphorus matches led to safety matches that were devised by two Swedes. First, Gustaf Erik Pasch in 1844 proposed placing some of the match's combustion ingredients on a separate striking surface, rather than incorporating them all into the match head, as a precaution against accidental ignition. This idea, along with the discovery of the non-toxic red phosphorus, led J. E. Lundström to introduce safety matches in 1855. Although safety matches posed less of a hazard, many people still preferred the convenience of strike-anywhere matches, and both types are still used today.

Modern matches are manufactured in several stages. In the case of wooden-stick matches, the matchsticks are first cut and soaked in a vat filled with a dilute solution of ammonium phosphate. They are then dried and inserted into holes in a long perforated belt, which dips them into several chemical tanks before they are dried and packaged in boxes.

# MATCHSTICK ROCKETS



**Project Maker**  
**WAREN GONZAGA**

**Project Link**  
[hsmag.cc/PTQtqX](http://hsmag.cc/PTQtqX)

## Did you know?

The hobby of collecting match-related items like matchboxes, matchbox labels, and such is known as phillumeny.

**E**ver since he was in middle school, Waren Gonzaga dreamt of working for NASA and building rockets. Since he couldn't find a hobby class to pursue his interest in his city in the Philippines, he decided to enrol in Mike Warren's Rocket Class on Instructables.com ([hsmag.cc/ZbtTMu](http://hsmag.cc/ZbtTMu)). Waren wanted to build a solid propellant rocket but couldn't find a shop from where he could source the materials. "Also, I am not aware of any law regarding making and launching rockets," he adds. He remembered seeing an article from NASA on matchstick rockets some eight years earlier. Since that article lacked specifications, Waren decided to construct his rocket

**Above** Waren is a true rocketeer, and says that he even enjoys the "smoke from the failed rocket launch attempts"

by experimentation. His Instructables page has details on the science involved, along with lots of illustrated build information to help anyone replicate the project. The construction involves wrapping a piece of aluminium foil around the strikeable tip of the matchstick. Pinch the foil at the top to make sure the exhaust is directed towards the bottom to achieve lift-off. Waren experimented with several designs and shares details about the failed ones, along with the successful ones. □

## ALERT!

Matches are flammable, and some of these projects shoot flaming sticks. Make sure there's nothing flammable nearby and bear in mind that they may not end up where you expect.

# MATCHSTICK CRAFTS

Project Maker

**CORDELL VON  
MALEGOWSKI**Project Link  
[hsmag.cc/vLrLnH](http://hsmag.cc/vLrLnH)**A**

**A**few years after a classmate bought a cube made of matches for one of the projects in Cordell's high school design class, he decided to make one of his own.

"After some trial and error I was

able to make it and then I made a video on how I made it. As that video became popular, I made a few more matchstick creation videos, with one of them becoming my most viewed video," shares Cordell. All his videos detail each and every step, with clear and detailed annotations. Unlike the majority of matchbox crafts, most of Cordell's are put together without

glue and can be easily duplicated with a little bit of patience and lots of matchsticks: "The motivation behind my matchstick creations is to make something impressive out of what is a seemingly simple object. It is also to inspire others by showing what can be achieved with some time and patience."

Like the Buddhist monks who destroy a mandala on completion, Cordell lights up his creations once they're done: "The reason I burn them after is mostly for the entertainment value. As they are made from matches, it's what most people would expect and want me to do." □



**Right** □  
The matchstick  
creations serve no  
practical purpose but,  
as Cordell suggests,  
they do make  
excellent gifts

# MATCHSTICK GUN

**I**f you've ever had the desire to rain flaming arrows on your enemies, follow Grant's project to unleash your medieval warrior. He converts a regular clothes peg into a piece of desktop weaponry that can shoot matchstick arrows over 20 feet. "When I made that project, I had a small son who instantly fell in love with it, and he had me make a few more so we could run around the house having matchstick wars. It was a great way to bond, father and son", shares Grant.

To make the gun, simply remove the spring from the clothes peg and place them back to back. Then carve out the channels for the arrows with a utility knife, and cut a small notch to hold the spring. Then, glue the pieces back together and attach the spring. Grant's Instructable has detailed illustrations that'll help you easily repurpose the clothes peg. □



**Project Maker**

**GRANT THOMPSON**

**Project Link**  
[hsmag.cc/YVMpqD](http://hsmag.cc/YVMpqD)

**Above** □

Let's be clear, shooting pointy or flaming projectiles is dangerous. Be sensible and safe if you decide to recreate this project.

# MATCHSTICK PUZZLES

**U**narguably, the most popular non-incendiary use of matchsticks is to build puzzles. Matchstick puzzles became popular in the 19th century, and are perhaps the earliest examples of board games. Dawie van Heerden from Cape Town, South Africa has always been intrigued by "these little sticks with their rounded heads". He has been collating matchstick puzzles on his blog, that's ranked among the Top 100 Puzzle Blogs on the web. He has been posting puzzles since 2011, and the blog currently has over 500 puzzles. Each puzzle post has a matchstick illustration, along with a textual

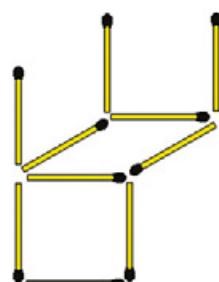
**Project Maker**

**DAWIE VAN HEERDEN**

**Project Link**  
[hsmag.cc/YBcMrK](http://hsmag.cc/YBcMrK)

**85. Chair Fell Over**

Easy   Medium   Hard   Extreme



challenge and is tagged with one of the four difficulty levels: easy, medium, hard and extreme. Thanks to the images, you can replicate the puzzles offline with real matchsticks with little trouble. Each post also has an illustrated answer, and you can browse the blog without accidentally revealing the solutions. □

**Above** □

Dawie is the creator of the Quickstick board game (quickstick.info) which challenges you to solve digital numerical puzzles

# HackSpace

TECHNOLOGY IN YOUR HANDS

## Download the app

Out now for smartphones & tablets



**£2.29**

rolling subscription

or

**£26.99**

subscribe for a year



# FORGE

## HACK | MAKE | BUILD | CREATE

Improve your skills, learn something new, or just have fun tinkering – we hope you enjoy these hand-picked projects

PG  
**92**



### HOME ASSISTANT

Control your IKEA lighting with a Raspberry Pi

PG  
**96**

### BAO BAO LED PANEL

Tessellated triangular lighting inspired by the iconic bags

PG  
**100**

### HOMEMADE GIN

The perfect way to relax after a long day in the workshop

PG  
**102**

### ROBOT SUMO

Can you build the ultimate fighting bot?

PG  
**78**

### SCHOOL OF MAKING

Start your journey to craftsmanship with these essential skills

**78** Arduino

**82** Wearables

**88** Découpage



PG  
**106**

### HACKING AN RC CAR

Upgrade your miniature clunkers with a bit of 3D printing and a micro:bit

PG  
**110**

### AMAZON ECHO

Keyboards are so 2017. It's time to control your builds with your voice



# Arduino programming: multiplexing, operators, and four seven-segments

Use the simple power of operators to multiply your project's capabilities without adding code



Graham Morrison

@degville

Graham is a veteran Linux journalist who is on a life-long quest to find music in the perfect arrangement of silicon

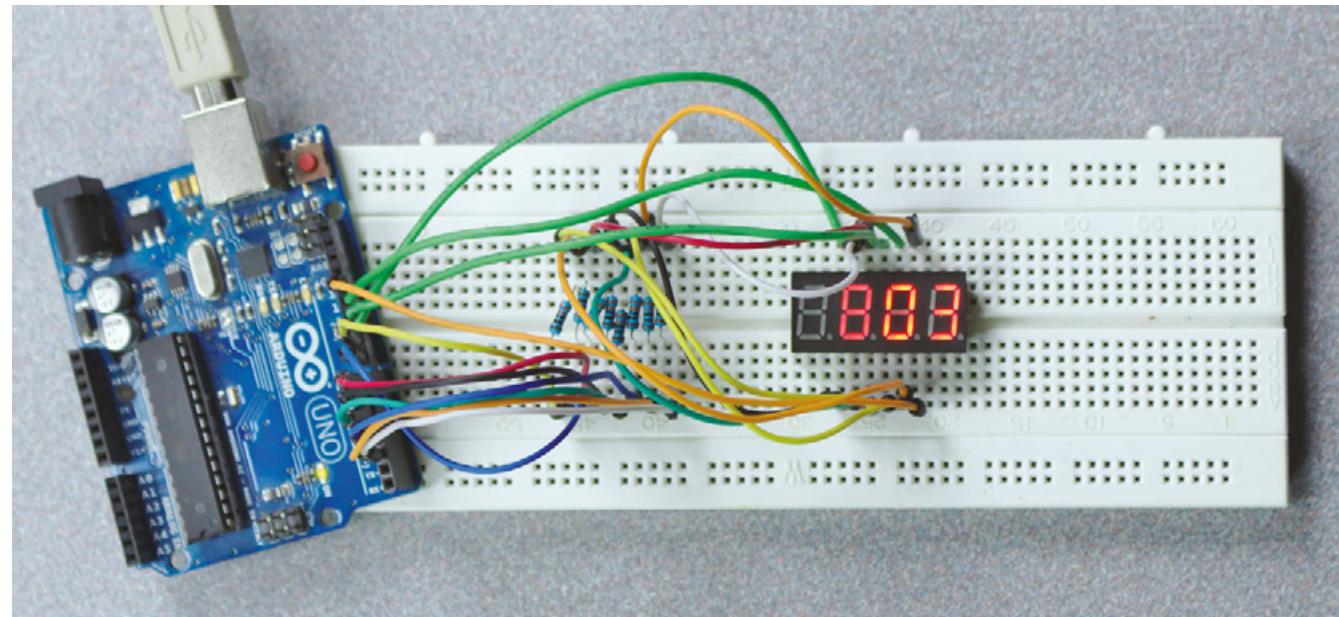
Last month we had fun getting a seven-segment display to work and writing the code to make it show something useful. This month we're going to expand on those foundations to build something four times better. Four times better exactly, in fact, as we're going to upgrade our hardware from a single digit to four, transforming the humble seven-segment into something capable of far more – numbers up to 9999 in base ten, and even a few words.

The first thing that likely crossed your mind with this plan, apart from trying to work out which swear words can be shown, is how this is all going to be wired to a humble Arduino. If you followed our tutorial last time,

you'll know that we needed to use a total of eight pins on the Arduino to control the display, exactly as we would if we were driving seven LEDs separately, which is all a seven-segment display really is. With eight pins taken, there aren't enough remaining on a normal Arduino to handle another seven-segment display, let alone another three. So how is it going to be done? The answer to this was also in the last issue, though not in the seven-segment tutorial. It was part of the LED section listed as 'multiplexing' (see right).

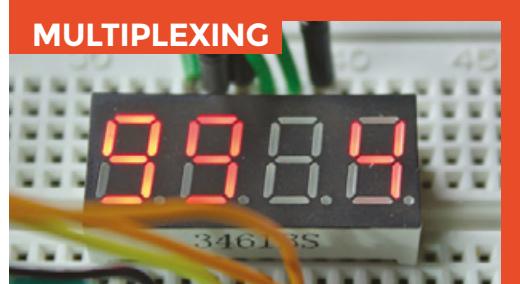
**Below** ♦

With multiplexing, you can light up more LEDs than you've got pins for. Bring on the blinkenlights!



The main problem with multiplexing is that you can only turn on one segment at a time. Turn on any more and other segments on other digits will also light up. The solution to this is to light each LED briefly as part of a cycle through the LEDs that need to be lit. It may seem remarkable in an age where computers take seconds to boot and webpages minutes to load, but the Arduino can do this quickly enough that the persistence of vision effect, where your eyes still see an object for a brief moment after the object is no longer visible, makes them appear solid.

The specific unit we're using is a 3461BS four-digit seven-segment display, although each digit also has a decimal point. This unit has twelve pins, six on the top edge and six on the lower edge, and while other four-digit displays may place these pins in different locations, the physical configuration will be the same after you've identified (from the unit's specification sheet) which pin does what. The specification for our display uses pins 1,2,3,4,5,7,10, and 11 for segments E, D, decimal point, C, G, B, F, and A respectively, and pins 6, 8, 9, and 12 for the common cathode



**MULTIPLEXING**

Multiplexing allows you to drive multiple LEDs, ergo multiple seven-segment displays, by taking advantage of the way LEDs use a potential difference in voltages to activate rather than just simply being 'on'. This dependence on a differential means that if the two pins connected to a segment are set the same, such as both being set to HIGH or LOW, the LED won't light, whereas any difference in the two connections, such as LOW and HIGH or HIGH and LOW, will light the LED. This behaviour can be exploited by wiring multiple LEDs or segments to a grid of crossing connections. As long as each pair of connections is unique, such as (A,B), (A,C), (B,C), the specific LED using those connections can be targeted. This saves your breadboard doubling as a tapestry loom, but it also means you can drive many more LEDs with the Arduino's humble allotment of digital I/O pins. However, there's one significant caveat: only one element or segment can be lit at any one time. Try to turn on more than one and the crosstalk in the wiring matrix will light other segments too.

or anode. Those last four connections are going to be used to multiplex the limited digital connections from the Arduino to the display. See the Wiring box (overleaf) for more details on how to connect these to the pins on your Arduino.

### CODE WORDS

With everything wired up, we can finally start playing with some new code. Rather than start from scratch, we're going to augment the code from last month, both to avoid repetition and to provide some continuity, but the code can be grabbed from [git.io/vAS8Y](http://git.io/vAS8Y).

With the old code loaded into the Arduino IDE, we're going to start at the top of the file with something we should have added initially – code to automatically handle whether your seven-segment display uses a common anode or common cathode configuration, as explained last month. As programmers, we should be making as few assumptions about the people using our code as possible, and that often means making things that could be specific more generic. In this case, we start by setting a global true or false value for whether a common anode display is being used:

```
const bool ANODE = true;
```

This line does nothing on its own but, like the array we used to hold the pin order for the connections, it's used by later logic to change the behaviour of the code. If we were using old-school C, we'd typically use a #DEFINE statement to declare a global constant value like this. The compiler then effectively swaps a defined value whenever it is referenced within the code. But for Arduino's Processing language, `const` is recommended as it better obeys the rules of variable scoping, which means they're far safer when working with multiple files.

### BITWISE OPERATORS

The only part of the code that cares whether the display we're using is common anode or cathode is the part that sets the HIGH or LOW values for the segments. This is because a common anode display requires the opposite signals to the common cathode. The behaviour can be described using something called a 'truth table', which is a very useful tool for understanding your hardware requirements and how they might best be implemented in code. In our case, a truth table can be used to show how we want to reverse the output depending on whether we're using a common anode configuration or not. Using 0 for off and 1 for on, the table would look like the following: →

### YOU'LL NEED

- ◆ 1 × 3461BS display
- ◆ 7 × 330 Ω resistors
- ◆ 20 × patch cables
- ◆ Arduino Uno

## SCHOOL OF MAKING

### QUICK TIP

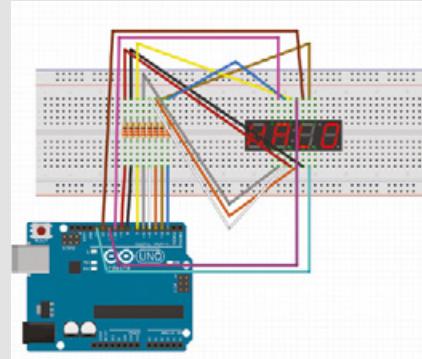
While we definitely recommend the use of resistors to keep your displays and Arduino safe, segments are only turned on for milliseconds, which means you could get away without them.

### WIRING

To wire this up, connect the following Arduino pins to those segments indicated on the display, via a  $330\Omega$  resistor. These aren't required for the common anode/cathode pins 10–13:

2 → A  
3 → B  
4 → C  
5 → D  
6 → E  
7 → F  
8 → G

10 → D1  
11 → D2  
12 → D3  
13 → D4



#### Right ♦

The exact wiring will depend on the specification and pin configuration of your specific display

	A B	Output
1.	0 0	LOW
2.	0 1	HIGH
3.	1 0	HIGH
4.	1 1	LOW

1. If the segment is off ( $A=0$ ) and the display isn't common anode ( $B=0$ ), output is LOW.
2. If the segment is off ( $A=0$ ) and the display is common anode ( $B=1$ ), output is HIGH.
3. If the segment is on ( $A=1$ ) and the display isn't common anode ( $B=0$ ), output is HIGH.
4. If the segment is on ( $A=1$ ) and the display is common anode ( $B=1$ ), output is LOW.

The reason for mapping everything out like this is that the simple behaviour described in truth tables can be mapped to special logical operators in code. You're likely already familiar with the logical operators AND and OR – they turn on output depending on where

input 1 AND input 2 are on, or they turn on output if either input 1 OR input 2 are on, including both inputs. Their truth tables look like the following:

A B AND	A B OR
0 0 0	0 0 0
0 1 0	0 1 1
1 0 0	1 0 1
1 1 1	1 1 1

In Arduino C, the operators that process this simple input are looking at individual bits, the true and false values, and these actually correlate to low-level gates and low-level code. This makes these operators incredibly efficient, which is why it's always worth attempting to refactor your code into these simple computational terms.

Going back to our example and the requirement we have to reverse the input for common anode displays, the first truth table correlates exactly to an operator called XOR, or exclusive 'or'. The X differentiates this operator from the ordinary OR above by not including a positive output when both the inputs values are on (or 1 in the truth table).

We're going to use this operator in a new function that isolates the `digitalWrite` commands:

```
void setSegment(int pin, bool state) {
    if (state ^ ANODE) {
        digitalWrite(pin, HIGH);
    } else {
        digitalWrite(pin, LOW);
    }
}
```

The XOR operator appears on the second line as the circumflex symbol (^). The function itself is called with two arguments: the pin to send the signal to and whether that pin needs to be HIGH or LOW. The efficiency comes because we can question both the requested state and whether the values need inverting with the XOR command, which is going to act exactly like the first truth table.

### FOUR TIMES SEVEN

We now need to augment our original routines to handle both the new digits and what will be our method of rendering them. This starts with a new array to hold the pin numbers for the connections to the common anode or cathode. This array will be called `digPin` and the backwards order we've used – 13, 12, 11, and 10 – is intentional as these are connected from least to most significant digits respectively, which will help when we write the program logic. Also, we're updating the pin values



we used in the `segPin` array as we've reorganised our circuit to use sequential pin ordering rather than the random plug and pray approach used last month:

```
const byte segPin[8] = {2, 3, 4, 5, 6, 7, 8, 9};
const byte digPin[4] = {13, 12, 11, 10};
```

The `setup` function also needs to be updated to initialise the new pins we're using. To do this we just add another `for` loop to handle the pins used to target the separate digits:

```
void setup() {
    for (int i = 0; i < 8; i++) {
        pinMode(segPin[i], OUTPUT);
    }
    for (int i = 0; i < 4; i++) {
        pinMode(digPin[i], OUTPUT);
    }
}
```

The next new functions we're going to add will be used to display a number on one of the four displays, rather than displaying a number on the single display we coded last month. The big difference in this implementation is the multiplexing, and this is accomplished by first making the common pin for the digit 'HIGH', writing the number to the seven-segment display, waiting a period for the number to remain visible, and then setting the common pin to LOW to terminate the drawing process.

Here's the code:

```
void displayDigit(int digit, int number) {
    digitalWrite(digPin[digit], HIGH);
    for (int i = 0; i < 8; i++) {
        setSegment(segPin[i], segNum[number][i]);
    }
    delay(5);
    digitalWrite(digPin[digit], LOW);
}
```

The `delay` function pauses execution of the code, allowing the character on the display to linger for a set number of milliseconds. The 5 milliseconds we're using is virtually imperceptible to the human eye, but if you wanted to see how the multiplexing works, set this to something like 200 (a fifth of a second) and watch each seven-segment display update with each different number.

The final piece of this puzzle is to transform the `displayNum` function we used last month to adapt to the four digits rather than a single one. The main job of the new additions will be to split a four-digit number, such as 2543, into its constituent digits, which can then be sent individually to each display.

To do this, we're going to rely on another incredibly useful operator, the modulo, which uses the percent character (%). Modulo will return the remainder of a division, rather than the number of times one number goes into another. This makes it useful in loops as a zero is often interpreted as false, but it's also perfect for peeling off digits.  $1234 \% 10$ , for example, will return the last digit, 4. If we then divide the number by 10 and run the modulo again, we'll get the next digit. And that's exactly what we do in this function:

```
void displayNum (int number) {
    int tens = 0;
    while (tens < 4) {
        displayDigit(tens++, number % 10);
        number /= 10;
    }
}
```

The above code includes one last new operator, the `/=` operator. This is closely related to the iterative operators we looked at last month, but instead of incrementing a value, here we divide `number` by 10 and assign the result to `number` in a single command.

All that's now left to do is update the main loop to remove the delay and count to an appropriately large number. This is as simple as changing it to the following:

```
void loop() {
    for (int i = 0; i <= 9999; i++) {
        displayNum(i);
    }
}
```

With that done, upload the code to your Arduino and pretend you've got the ultimate Geiger counter. The code can be found here: [git.io/vxMz6](http://git.io/vxMz6). □

**Above** We've used the cheap and readily available 3461BS for this project, but almost any other quad seven-segment display will work

## QUICK TIP

If you connect the segment pins to the same Arduino pins we used last month, you won't need to modify your character or pin order code.

SCHOOL OF MAKING

# Flaunt your skills with a light-up bag

Make an interactive tote bag with a Circuit Playground Express and a little sewing



**Sophy Wong**

@sophywong

Sophy Wong is a designer, maker, and avid creator. Her projects range from period costumes to Arduino-driven wearable tech. She can be found on her YouTube channel and at [sophywong.com](http://sophywong.com), chronicling her adventures in making.

**N**eed a project idea for your brand new Circuit Playground Express? Make it a wearable one! The CPX has oodles of sensors and NeoPixel LEDs built right in, so there's no soldering or breadboarding required to set up a circuit. You'll want to take it with you everywhere so you can tinker whenever inspiration strikes you. In this project, we'll sew it onto a tote for coding on the go!

The CPX is a little powerhouse, and we'll use the on-board accelerometer, capacitive touch capabilities, and ten NeoPixel LEDs to run some fun animations on the front of the tote bag. To make our program interactive, we'll add snaps for simple touch-sensitive buttons. Each button will play a different NeoPixel animation when pressed. While we're at it, we'll use the accelerometer to trigger a flashy animation if you happen to run, skip, or jump while out and about with your bag.

For portable power, we'll use a battery holder and three AAA batteries. Look for a battery holder with a power switch and a male JST connector, which will work perfectly with the JST connector on board the Circuit Playground Express. To be safe, leave the batteries out of the holder until your build is complete.

Your snaps should be metal – bare metal will work best, but these colourful enamelled snaps also worked well. If your snaps are coated, check them with a multimeter to make sure they are conductive. You can use any bag you like for this project, and you may already have the perfect bag to start with. If not, pick up a blank canvas tote at a craft store and decorate it to make it your own. The Circuit Playground Express is a perfect match for this 'hello world' text: the board fits nicely over the 'o' in 'world'. Keeping it front and centre makes it easy to plug the board into your computer any time you feel like changing the code!

**Right** Stencilling, rather than drawing the design, means less chance of slip-ups

## STENCIL YOUR TOTE

If you're starting with a brand new, blank canvas tote bag, be sure to wash, dry, and press it before decorating it. The fabric will take the paint better and make for a nicer finished project.

A vinyl cutter is nice for making stencils, but it's definitely not necessary. To make the stencil by hand, trace the 'hello world' text onto your contact paper, keeping a border of at least 2–3cm around the text. Carefully cut out the text with a craft knife. Apply the stencil to the front of your tote, pressing down firmly along all the edges for good adhesion. Remember to place the hole shapes into the letters with negative spaces.

Using a craft sponge or sponcer, dab fabric paint over your stencil. For nice, crisp text, take your time and aim for an even, opaque application. Carefully remove the stencil and let the paint dry. When it's completely dry, it's a good idea to heat set your fabric paint according to the manufacturer's instructions. Now you're ready to make it light up! →

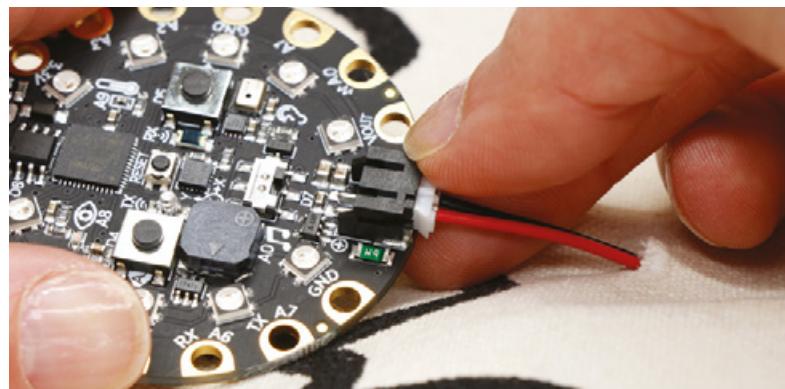




**Above ↗**  
Time to hit the high  
street and let the  
world see your  
making skills

# Make a 'hello world' tote with Circuit Playground Express

## SCHOOL OF MAKING



### YOU'LL NEED

#### For The Bag

- ❖ Blank canvas tote bag
- ❖ Contact paper or adhesive shelf liner
- ❖ Craft knife or vinyl cutter
- ❖ Textile paint
- ❖ Craft sponge or spouncer
- ❖ Hand sewing kit
- ❖ Sewing machine (optional)
- ❖ Heavy duty or upholstery thread

#### For Making The Circuit

- ❖ Circuit Playground Express
- ❖ 3 x AAA battery holder, with male JST connector and power switch
- ❖ Conductive thread
- ❖ Metal-capped prong snaps and setting tool (sometimes it's included)
- ❖ Fusible interfacing
- ❖ Fray Check
- ❖ Superglue
- ❖ Disappearing ink fabric pen

### MAKE A BUTTONHOLE

The Circuit Playground Express will be attached to the front of the bag, but the battery holder will be inside the bag. This means we need a neat way to pass wires through the front of the bag. A small buttonhole is perfect for this, and if you have a modern sewing machine, chances are it has a buttonhole function.

Place the Circuit Playground Express in its spot on the front of the bag, and mark where the buttonhole will go: about 1 cm below the JST battery connector on the CPX. Follow your sewing machine's procedure for making a buttonhole on your line, about 1 cm wide. Apply Fray Check along the centre

#### Below

The large holes on the CPX are easy to sew through, both for making the circuit and holding the board in place

#### Left

The battery connector pushes firmly into place, so should hold even when the bag is in use

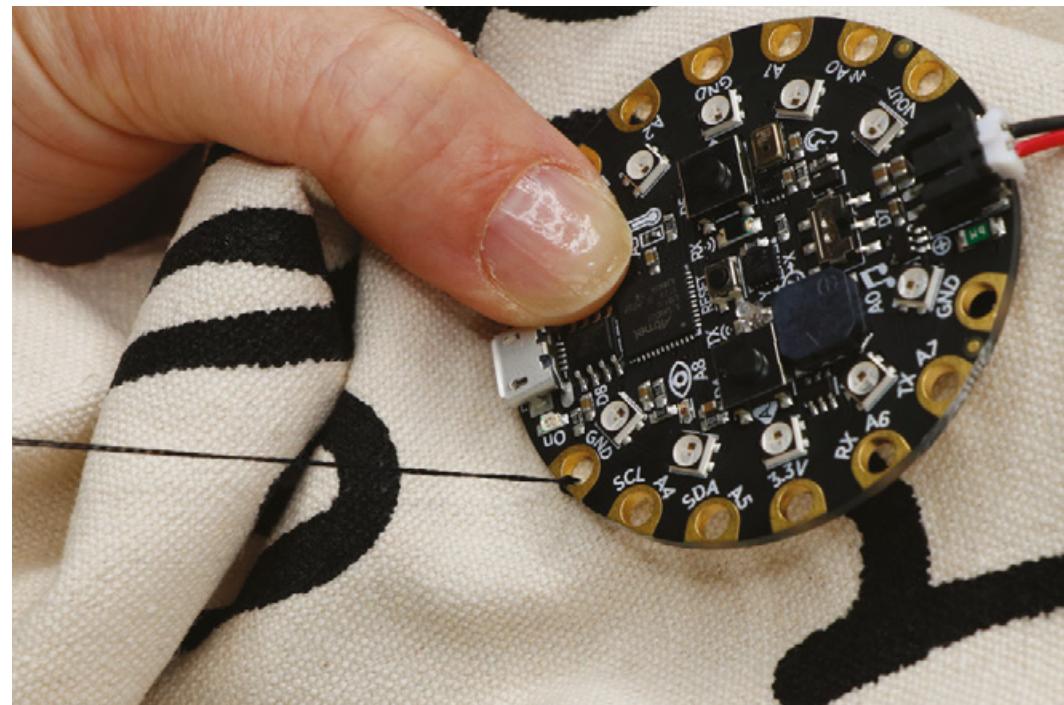
of the buttonhole and let dry. Use a craft knife to cut the buttonhole open, being careful to only cut the fabric in the centre of the buttonhole.

If you don't have a sewing machine, you can cut a small slit in the bag and hand-sew around the edges to bind them. Run a bead of Fray Check along the bound edges to stabilise the hole.

### ATTACH THE CIRCUIT PLAYGROUND EXPRESS

Place the empty battery holder inside the bag, and push its JST connector through the buttonhole to the front of the bag. Plug the JST connector into the black port at the bottom of the Circuit Playground Express, and lay the CPX in place on the front of the bag. You can use a dab of hot glue to hold the board in place while you sew it down.

Since we won't be using the power and ground pins in our circuit, we can use those holes to attach the board to the bag. Thread a needle with heavy-duty thread, or use a double strand of regular hand-sewing thread. Sew through the three GND pins, the two 3.3V pins, and VOUT – that's six holes – for a secure attachment.





**Above** Carefully mark your placements out before making any permanent incisions

### ADD CONDUCTIVE BUTTONS

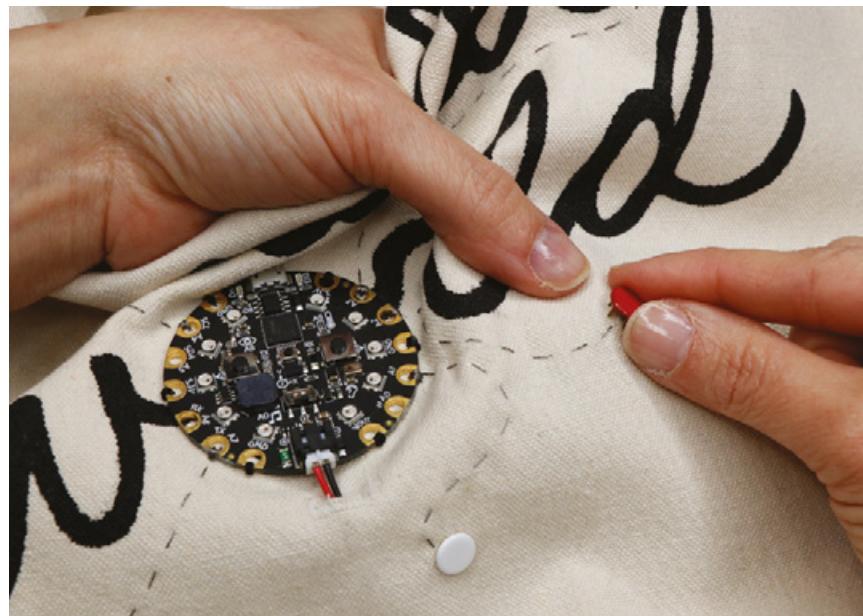
Decide where your metal snaps should go, and mark their locations on the front of the tote. Keep in mind that the A0 pin cannot be used for capacitive touch, but you can use any of the other I/O pins (A1 through A7). Draw the path you will sew from each snap to its I/O pin with disappearing ink. In this ‘hello world’ design, we are connecting to A1, A2, A3, A4, and A7.

Thread a needle with conductive thread and tie a tight knot at the end. For each button, start at the point where your snap will be, and push the needle through the fabric from the inside to the outside of the bag. Take seven or eight small stitches right on that spot to make a big dot – this will be the connection point for the snap to sit on top of. From there, sew with a running stitch along your path to the I/O pin you’ll be connecting to.

When you get to the board, stitch around the pin about five or six times, and pull tight for a good connection. End with your thread on the inside of the bag, and tie a tight knot to finish off your run. Dab a bit of Fray Check or superglue onto the knot to keep it secure, and cut the thread tail short. Repeat this step for all the paths you marked.

Prong snaps come in four pieces, but for this project we’ll only be using the top two pieces: the cap and the socket. To install each snap, place the pronged cap over your sewn connection point, and press the prongs through the fabric. Insert the prongs into the socket part of the snap, sandwiching the fabric and the conductive thread dot between the snap pieces. Set the snap with a setting tool and hammer.

When you’ve set all your snaps, turn the bag inside out so you can take a good look at your circuit. Make sure all the thread tails are cut short and not touching the other legs of the circuit. If everything looks good,



you can remove the temporary markings you made. Now plug it in and see if it works!

### PROGRAM THE CIRCUIT PLAYGROUND EXPRESS

MakeCode is a great way to get started with code. It’s visual and block-based, so programming is a matter of connecting the right blocks together. It’s simple to get started, and perfect for unleashing all the bells and whistles packed into the Circuit Playground Express. So grab your computer, head to [makecode.adafruit.com](https://makecode.adafruit.com), and start a new project.

Click on the Light category, where you’ll find lots of different blocks that can be used for animating the NeoPixels on the Circuit Playground Express. Grab the ‘Show Animation’ block and drag it into the Forever loop on your workspace. The CPX simulator on the left will now show the rainbow animation. →

**Above** You don’t have to stitch in a straight line. Wearables are meant to be creative!



**Left** A gentle whack fixes the snaps in place

# Make a 'hello world' tote with Circuit Playground Express

## SCHOOL OF MAKING



### Right ♦

You can reprogram your CPX in place if you decide you want different animations

You could simply download this code to your CPX and have an awesome flashy bag. But let's go further and use our capacitive touch buttons!

MakeCode has several different animations built in, and for this project we can think of them as modes. So let's set up a variable to make it easy to change from one animation mode to another.

In the Variables category, click 'Make a New Variable' and name it 'mode'. Drag your new 'mode' variable onto your workspace, and drop it on top of the rainbow icon in the 'Show Animation' block. This will replace the rainbow animation with your variable.

### Below ☐

The MakeCode environment makes it easy to link events to touch actions

Now we need to assign each of our capacitive touch buttons to play a different animation. Buttons are inputs, so head over to the Input category and drag the 'On Button A Click' block onto your workspace. We'll start with the button connected to pin A1, so change 'Button A' to 'pin A1'. From the Variables category, grab the 'Set Item to 0' block, and drop it into your 'Pin A1 Click' block. Change the variable drop-down from 'item' to 'mode'.

Head into the Light category and grab the animation drop-down menu block – you'll need to scroll down to 'More'. Drag the animation block on top of the '0' to place it in your 'Set Mode to' block. Now, when you click on the A1 pad on the CPX simulator, the rainbow animation will play!

Repeat the steps above to set up the rest of your capacitive touch buttons on pins A2, A3, A4, and A7, and choose a different animation for each one. To quickly copy the block you just created, right-click it and select 'Duplicate'.

Check your work by clicking on the pins in the simulator. If your animations are playing as expected, move on to setting up the accelerometer! Since you're likely to be moving around while out and about with your bag, we'll tell the CPX to flash the NeoPixels when it detects a big movement.

Just like our buttons, the accelerometer is also an input, so go back into the Input category and grab the 'On Shake' block. Chances are there will be an animation playing when the CPX is shaken, so the first thing we'll need to do is stop whatever is already running. From the Light category, grab the



'Stop All Animations' block and drop it into the 'On Shake' block. Under this, add the 'Show Animation for 500 ms' block and select the running lights animation. Let's give this animation a little more airtime by changing the duration from 500 ms to two seconds. Check your work by clicking 'SHAKE' at the top of the CPX simulator. The lights should flash red for two seconds, and then go back to playing whatever

Since we won't be using the power and ground pins in our circuit, we can use those holes to attach the board to the bag

animation was already running. If your code is working perfectly, it's time to load it onto your actual Circuit Playground Express!

Click 'Download' and follow MakeCode's instructions to load your new code onto your board. Tapping the snaps on your bag should trigger the animations you programmed, and shaking the bag should trigger the flashing runner lights. If something is not working as expected, check for crossed conductive threads and make sure your snaps are anchored well to the thread dots under them.

Check out the finished code for this project here: [hsmag.cc/ORBbKo](https://hsmag.cc/ORBbKo).

Need help? Go to [learn.adafruit.com/makecode](https://learn.adafruit.com/makecode) to learn more about programming your Circuit Playground Express with MakeCode.



## BACK WITH FUSIBLE INTERFACING

Now that we're sure everything is working, let's cover the back of the circuit with fusible interfacing. This will insulate our circuit nicely and keep the hand-sewn stitches from catching on objects inside the bag.

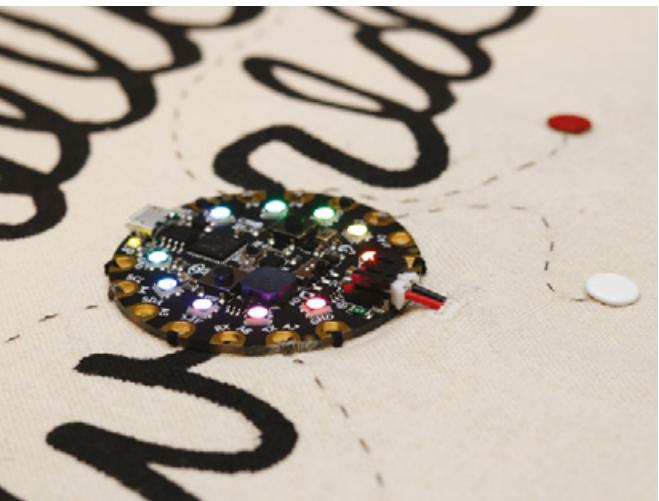
Turn the bag inside out again, remove the battery holder and set it aside. Cut a piece of interfacing that will cover the whole circuit. Follow the instructions for your fusible interfacing to apply it to the fabric. In general, you'll iron the interfacing on with a little bit of steam – you can iron over the conductive thread, but don't iron directly over the CPX or snaps. Just press around them so the fusible interfacing is held down securely. Make sure to avoid covering the buttonhole with interfacing.

If you've used steam to apply your interfacing, let your project dry completely before connecting it back to power. If you'd like to add some weatherproofing to the Circuit Playground Express, paint its surface with clear nail polish.

## GRAB AND GO!

It's time to plug in your battery holder, add batteries, and take your project for a spin! Flip the power switch to on, and shake your bag to see your NeoPixels flash. Try out your conductive buttons – what else can you make them do besides play animations? Whenever you have a new idea your Circuit Playground Express will be right by your side, and we want to see what you make! Show us your projects at [@HackSpaceMag!](https://@HackSpaceMag)

**Above** Keep your circuit insulated from anything jangling around inside



### Left

With the battery hidden inside the bag, only the circular outline of the CPX adorns the bag

# Brighten up your project enclosures

Transform the ordinary into unique personal pieces using découpage



Pepi Valderrama

@depepi

Pepi Valderrama is a writer who loves technology, crafts, and shiny things with LED lights. You can find her on [depepi.com](http://depepi.com)

**Right** Découpage works well with different materials. It's a perfect technique to transform the mundane into unique pieces, gifts, and art



## YOU'LL NEED

- ❖ Scissors
- ❖ A glue brush
- ❖ Wrapping paper
- ❖ Comic books
- ❖ Mod Podge gloss
- ❖ Glitter
- ❖ A craft knife
- ❖ Something that needs decorating

**T**here are loads of great cases for Raspberry Pis, Arduinos, and other electronics, but most of them are, well, a bit dull. Laser-cut acrylic or injection-moulded plastic might be perfectly functional, but it can leave a little to be desired from an aesthetic point of view. Let's take a look at one technique for making these enclosures a little more lively.

Découpage, from the French verb 'to cut', is a simple technique that involves cutting pieces of paper or cloth and gluing it onto objects by applying layers of varnish. It's a simple way to personalise objects that seem dull, look old, or just need a

personal touch. Although découpage might seem simple at first sight, forgetting certain details, like materials and weather conditions, could be a fatal mistake and end up ruining the result.

The best glue for découpage is one that combines varnish and glue. The best options are water-based and should come in matte or gloss types. We'll be using Mod Podge, which is a tried and tested découpage staple. You can also use PVA glue or permanent spray glue, as they work just as well and are quicker to dry, but the final look might not be as flashy. And of course, you can just use varnish. However, the best is the gloss type since it doesn't go all cloudy.



**Left** ♦  
Raspberry Pi cases work better when applying glitter. However, using small pieces of paper to cover them up also works perfectly

The most common décopage pair is gluing paper on paper. Boxes, notebooks, and postcards are the perfect candidates for creating your own artsy look, but décopage works on other materials, like plastic, metal, or wood as well. So, if you find yourself wondering whether to get a simple cover for your Raspberry Pi or Arduino project, or the cool, geeky and shiny one that costs ten times more, with these simple steps you can make your case personal and unique with little extra cost.

However, using décopage on a Raspberry Pi case can be rather tricky. Most cases out there have tiny holes and corners, and are made of plastic. This can be challenging when applying the Mod Podge glue, paper, and glitter. Plastic and steel are slippery, which means that extra care is needed to achieve excellent results. Also, a thicker paper works best since it will be easier to work with. However, if your target is to create textures, or you like that slightly transparent look, tissue paper is the way to go.

## GETTING STARTED

Before starting any project with décopage, you need to clean all surfaces of the object you want to work on and make it dust-free. Glue and varnish amplify

“ Paper is not your only choice, though, as glitter can also be used for decoration. It can be used either as a base or an add-on for your projects ”

imperfections, so even a small amount of dust can spoil the results. If the project lasts for several days, it's advisable to use a clean, dry cloth to get rid of all dust as you go along.

Secondly, you need to consider what kind of papers you're going to use, and prepare them in advance by cutting them to the correct shape and size. The thickness of the paper and the ink used on it have a big impact on the final result, as different papers, inks, and colours react differently with the glue and varnish.

Depending on your goals, you might want to use fine, decorated tissue paper, or thicker papers, like comic book pages or wrapping paper. The first one is good for creating textures, while the second type works best on flat or tricky surfaces. It is important to remember, however, that the average printing paper rarely works well with décopage, so you are better off buying proper paper, rather than printing something off the internet. →

## QUICK TIP

For a final matte look, the best option is to use gloss varnish for all the project, except for the last coating layer.

## SANDY LOOK GLITTER

To have a glittery, sandy look, you need to apply a thin Mod Podge layer on the surface and then throw raw glitter on it until it covers the surface completely. Once the glue is dry, shake the case to get rid of the extra glitter. Apply another thin coating layer of Mod Podge, and repeat the process. When you're satisfied with the thickness and sandy look, you can apply a covering of the final coating layer. It's not advisable to leave the glitter layer raw without a final coating layer, since the glitter might come off with time and mess with the components in the Raspberry Pi.



**Left** ♦  
Thin coating layers of Mod Podge work better for any project. Applying thicker glue and varnish layers to go quicker will have disastrous results, as seen in this picture

## SCHOOL OF MAKING

### WRAPPING PAPER ON A CARD BOX

Boxes are easy to style by using découpage, whether you are giving them a fresh look for keeping your things safe, or creating something festive and personal for a gift.

First, you need to unfold the box until you have it flat on the table. Then, apply a thin layer of Mod Podge and glue the wrapping paper, taking into account the places where the box will fold back – you don't want the paper to break and tear when you reassemble it. Use a sponge to push the paper and to get rid of air bubbles where necessary. Once the glue is dry, turn the flat box over, and cut the

paper carefully along the lines of the box by using a craft knife or a box cutter. Once you have reassembled the box, you can choose to apply a matte coating layer or a glitter one to it. Or, you could leave it as is and enjoy the feel of the paper itself.

Make sure you take care of the folding areas. Applying paper too tightly might look good when the box is flat on the table. However, when folding it back to its original shape you can end up with broken paper areas. A good idea is to fold the corners of the box when gluing the paper, so the little extra you need comes naturally.

Paper is not your only choice, though, as glitter can also be used for decoration. It can be used either as a base or an add-on for your projects. Small objects usually look better when you use glitter as a base, since covering them with paper is tricky, unless you're planning to use smaller pieces or you want to create textures. Glitter also comes in handy if we're going for a sandy look and texture.

#### A GLITTERY RASPBERRY PI CASE

Put some glitter in a glass, along with some Mod Podge glue, and mix it with a stick or the glue brush until the mixture is smooth. Apply a light coat of the mix on the Raspberry Pi case, and wait until it's completely dry. Continue applying light layers



#### Above

Edges can be tricky. We can either fold the paper, or apply smaller pieces of paper on top of one another

with the mixture, but always wait for each layer to dry completely. Otherwise, the end result won't be smooth and even.

Don't use a hair dryer to speed the drying process – it's preferable to wait until it dries naturally, otherwise bubbles can appear on the coating layers and spoil the end results. As an average, a twenty-minute wait between coating layers would be enough. However, it's advisable to wait double when the atmosphere is wet, rainy, or too steamy during summer.

Once the glitter base is done, apply a light layer of Mod Podge, without glitter, on the surface of the Raspberry Pi case. Then, glue the paper. If needed, use a small sponge to pressure the paper a bit to get rid of air bubbles. Wait until the glue is dry and apply a new light coating layer of Mod Podge and paper. Repeat until you're satisfied with the results. You can apply as many layers of paper as you see fit. The end result can be minimalist or gothic, depending on your taste and style.

### COMIC BOOK PAPER

Comic book paper can be a tricky choice, since ink might fall off if you try to glue several layers of paper without thinking about the design first. Re-adjusting the paper once it's wet with Mod Podge isn't advisable, or even possible, without ruining the paper. Plan your design early, or go for a simple cover-up.

Apply a thin coating layer of Mod Podge and then the comic book paper. Wait until it's totally dry before applying another thin coating layer of the glue. Wait for it to be completely dry before applying the next coating layer.

Using other magazines or even posters is possible as well, and, depending on the paper and ink, they often work in the same way as comic book paper. Always allow time for the glue to dry before applying more layers. Also, think carefully about the design, since different kinds of ink can react differently as well.

#### QUICK TIP

Consider adding some glitter on the final layers for a flashy look. Go for a smooth layer of glitter instead of a sandy look, unless you want to create drawings with the glitter on the surface.

#### A PAPER-COVERED RASPBERRY PI CASE

If you want to use only paper to cover your Raspberry Pi case, you will need to be more patient and take into account all the tiny holes, cuts, and corners of the case. First, apply a thin coating layer of Mod Podge, and put the paper carefully in place. Use a small, dry sponge to push the paper gently and get rid of all the air bubbles. Once the glue is dry, cut the paper carefully with a craft knife, around the edges of the case. Then, apply a new thin coating layer and wait until it's dry. You can use as many thin coating layers as you want or need.

Be careful with the round edges of the Raspberry Pi case, since these can be very tricky. The best plan of action is to glue strings or small pieces of paper to cover the area. As long as you apply thin coating layers and wait until they're dry, the end results will look professional.

**QUICK TIP**

Newspaper looks great with découpage, but make sure that each layer of glue is completely dry before starting the next, as the ink can run.

**Left & Below ↗**  
Craft knives and cutters are great tools to cut unwanted paper. To cut without creating marks on the paper, we must wait until the coating layer is completely dry

**If you want to use only paper to cover your Raspberry Pi case, you will need to be more patient and take into account all the tiny holes, cuts, and corners of the case**

**APPLYING TISSUE PAPER ON METAL**

Raspberry Pi cases are usually plastic, but many projects are better suited to metal enclosures such as Mentos tins, or die-cast aluminium project boxes. You can use découpage on these as well.

If your goal is to get a textured paper look, your best choice of paper is tissue paper. To get the feeling, we used an empty coffee tin can to create a pencil holder. First, you apply a thin coating layer of Mod Podge, and then glue the first tissue paper layer on it. The trick is to glue it as smoothly as possible. We need a first layer of paper on the tin so we can work comfortably creating the texture with other paper layers later.

Once the first layer is dry, you can apply another thin coating layer of Mod Podge, and glue another layer of tissue paper. This time you can use your fingers to give some texture to it. The trick is to make the paper slide a bit, creating small wrinkles



and textures. However, using a sponge will give you a different texture result. Repeat as many times as needed to hide the text on the tin.

Next, cut a 2 cm line of tissue paper. This is to cover the edges of the tin can. You can apply as many coating and paper layers as you wish until you get the desired results.

These basic techniques can be used to brighten up almost any project – or even quite a lot of tools! Let us know what you've decorated on Twitter, and how – **@HackSpacemag**. □

# Controlling smart lights with Home Assistant

Making IKEA's smart lights a bit smarter



Dylan Murphy

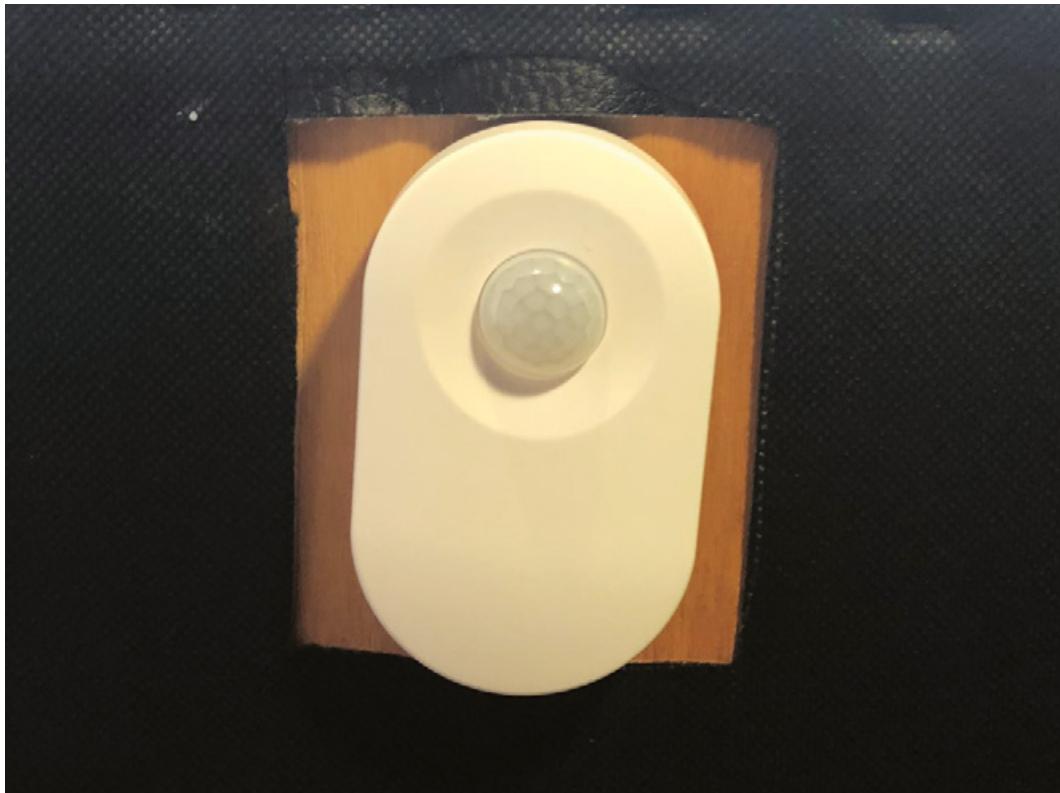
Dylan is an app and website developer. He is a vintage Macintosh collector and lover of all things IKEA.

## YOU'LL NEED

- ❖ Raspberry Pi (preferably Pi 3 for better performance but can work with any model)
- ❖ Trådfri gateway
- ❖ Trådfri bulb
- ❖ Trådfri steering device
- ❖ iOS or Android mobile device

## OPTIONAL

- ❖ Trådfri motion sensor



**H**ome automation is great for a plethora of possibilities, but can be expensive and restricted by the manufacturer. Home Assistant aims to change that and IKEA's Trådfri helps to reduce the cost.

IKEA's Trådfri smart home lighting system is an inexpensive way to introduce yourself into the world of smart home gadgets. The Trådfri system is relatively inexpensive but, unfortunately, is very restricted – that's where Home Assistant comes in.

Home Assistant is an open-source home automation platform that runs on Python 3. It requires very little processing power and therefore is perfect to be run on a Raspberry Pi. Home Assistant combines smart home devices and allows them to all be controlled together from a single app or online UI. At the time of writing, Home Assistant has support for 988 components. A component is an extra piece of software that is installed onto your HA instance (the Home Assistant UI), to allow a smart home device to be controlled with Home Assistant. This

number is increasing all the time. Home Assistant then allows you to create automations that link all of these components together. This opens a vast range of possibilities that were never open before. An example of this is turning on a fan connected with your smart plug when the temperature goes above 20°C. Or, you could do something more extreme such as playing music on your Sonos at home and turning on your Roomba to vacuum your house if your Tesla door is unlocked, but only if it is precisely 03:15 in the

## IKEA's Trådfri smart home lighting system is an inexpensive way to introduce yourself into the world of smart home gadgets

morning. This allows the restricted Trådfri system to unlock its full potential for a fraction of the cost of a more expensive smart lighting system.

To get started with IKEA's Trådfri smart home lighting system, you will first need to set up the gateway on the Trådfri app. This is free from the Google Play Store or the Apple App Store. Firstly, plug in the Trådfri gateway to a wall socket and a network port. Launch the IKEA Trådfri app on your mobile device and connect to the same network as your Trådfri gateway. Follow the instructions on the app to connect your bulb and steering device together (**Figure 1**).

To be able to still use and edit your HA instance, you will use your Raspberry Pi in headless mode (using a Raspberry Pi without a keyboard, mouse, or monitor) – the end goal is to leave your Raspberry Pi alone acting as your home automation hub.

Once you have successfully set up the Trådfri hub you need to set up Home Assistant. There are 14 different ways to install Home Assistant through varying methods and on different systems. Our favourites are hass.io and Hassbian. Despite it requiring a few more command lines, we've found it to be more reliable in the long run and gives you more freedom to do what you.

To install Hassbian, go to [hsmag.cc/xRzRvm](https://hsmag.cc/xRzRvm) and select 'Download the Hassbian image'. Use Etcher to flash the image onto your SD (or microSD) card. Etcher is a free piece of software available on macOS, Windows, and Linux. Once the image has flashed

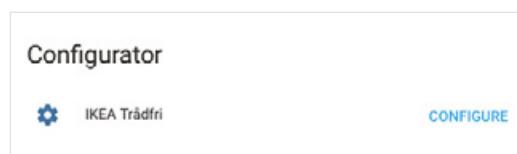


**Figure 1** IKEA also sells a small wireless dimmer which is more minimalist than the steering device. You can set that up here as well

onto an SD card, insert the card into the Raspberry Pi. Connect the Pi to the internet and boot it up. Boot up the Pi and wait approximately five minutes. After that time, go to [hassbian.local:8123](http://hassbian.local:8123). This is your Home Assistance instance.

To be able to use your Raspberry Pi without a keyboard, mouse, or monitor, you need to access it via an SSH connection. Connecting via SSH is when you connect from one computer to another. This can be done on macOS, Windows, Linux, or even a mobile device.

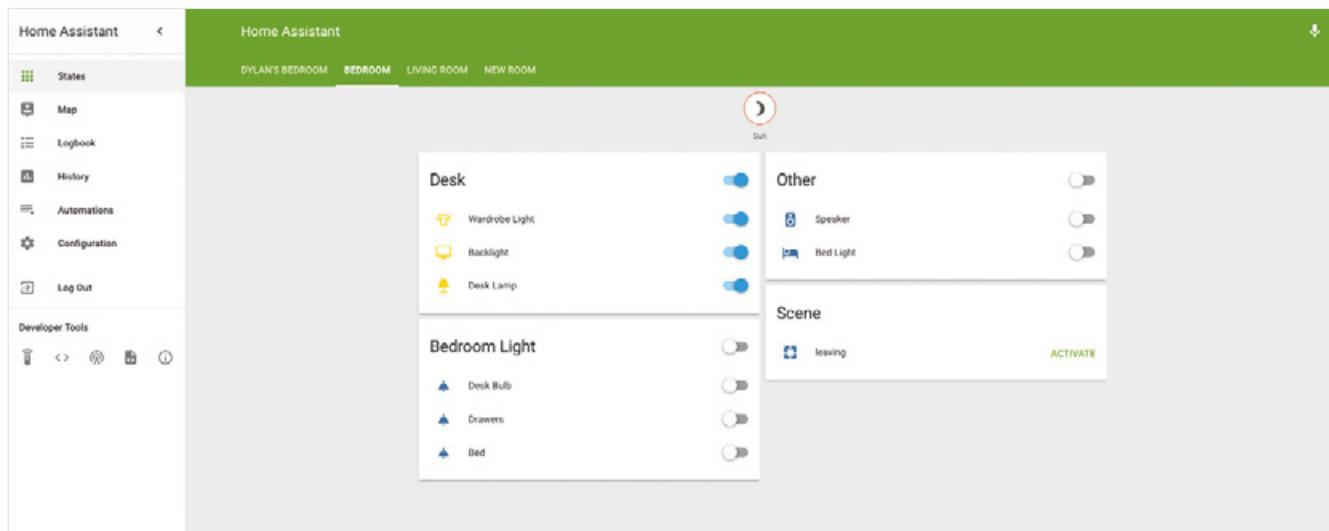
Setting up an SSH connection on a macOS or Linux computer is simple. All you need to do is find out the IP address of your Raspberry Pi. (This can be easily found on your router's online UI, although this will differ with each manufacturer. Your Raspberry Pi will usually be labelled 'hassbian'.) Then open up a Terminal and type.... ➤



**Figure 2** Trådfri can be a bit temperamental sometimes, so you may need to restart the gateway

# Controlling smart lights with Home Assistant

## TUTORIAL



**Above** With 988 Home Assistant components, your Home Assistant Instance could get really messy really quickly

### `ssh Pi@your_IP_address`

Then type 'yes' if prompted and log in with the password – **raspberry** (**Figure 3**). Setting up SSH on Windows is more tedious as you need to use separate software such as PuTTY. Go to [putty.org](http://putty.org) and download the appropriate version for you.

Once you have opened up the application, go to input the IP address into the Host Name field and select the SSH option. Press open and select 'yes'. Enter the login as **pi** and the password as **raspberry**, (**Figure 4**).

You are now successfully into your Pi's command line and you can install some scripts. Firstly, install Samba. This allows your Pi to be accessed as a network drive. This will be useful when you want to create automations but you don't want to log

## GOING FURTHER

These are some useful – and not so useful – ideas for other automations:

- Set off an intruder sequence for a room in your house, and send a notification to your phone when your door sensor is triggered.
- Set your lights to dim and turn blue when you turn on your Chromecast and start watching something.
- Start playing music on your Sonos and flash the lights when your Bitcoin goes above a certain value.
- Use an Amazon Dash button as a trigger for any automation.
- Home Assistant automations lack reliability from time to time, so all automations are created at your own risk...

into your Pi via an SSH connection. To do this, run the command:

```
sudo hassbian-config install samba
```

Once this has been installed, you need the Trådfri script to eventually use your Trådfri lights. To do this type:

```
sudo hassbian-config install tradfri
```

into the terminal. The Home Assistant Instance is now configured and ready to use.

Connecting your Trådfri lighting system to HA is also simple. All you need to do is to go to **hassbian.local:8123** and then press the CONFIGURE button next to IKEA Trådfri (**Figure 2**). Here you need to input the security code written at the bottom of your IKEA Trådfri Gateway. You'll then see all of your Trådfri bulbs show up on a 'Light' card.

You now have all of the ingredients to start creating Home Assistant automations. To view your Home Assistant automations, go to your network folders and find **automations.yaml** in the Home Assistant folder. These are located under the **hassbian** directory. The **automations.yaml** file is where you create your automations. The YAML file is formatted as a list as opposed to being formatted as code. Automations consist of three main parts. The 'alias' is the name for your automation. The name of your automation doesn't



**Figure 3**

SSH gives you a tunnel into your Raspberry Pi and allows you to execute commands as though you were directly connected

matter and is only for your personal reference. The second part of an automation is the ‘trigger’. The trigger is what activates the automation, for example a motion sensor inside your house. The final part of an automation is the ‘action’. This defines what happens when the trigger is activated. For example, the action could be something simple such as turning on a light or it could be something more advanced such as turning on an intruder sequence which consists of sending you a notification whilst also flashing your smart lights and locking all your doors.

Let’s start with a simple automation: turn the lights on when the sun sets. We have installed the Trådfri component and the sun component is pre-installed onto Home Assistant, so all that we need to do is join them together in **automation.yaml**. Firstly, give a name to your automation. The second part of an automaton is the trigger. In our case, the trigger would be the sun. Finally, the action is the turning on of the light. Putting all of this together gives you this:

```
- alias: "Bye Sun, Hello World"
trigger:
  platform: sun
  event: sunset
  #time offset is optional
  offset: '-00:45:00'
action:
  - service: Home Assistant.turn_on
    entity_id: light.bathroom
```

This is an example of a basic Home Assistant automation, but you can get more creative. A more advanced automation is turning on your desk backlight, desk lamp, and speaker charging dock when you sit in your chair. Despite this seeming complicated, it uses the same foundations that the sun rising automation does – it just expands on it by adding more actions and a more advanced trigger.

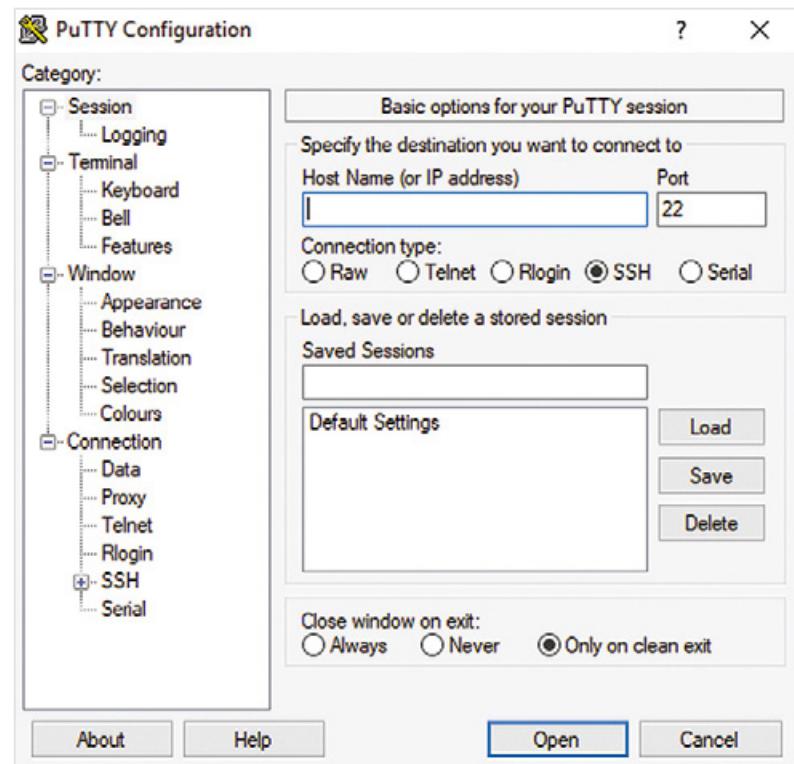
IKEA has a Trådfri motion sensor (£16, €17.99, \$17.99, 179kr.) Unfortunately, IKEA does not currently expose the motion sensor to HA, which means it can’t be used on its own (this is correct at time of writing). Therefore, the only other option is to install a smart bulb in your desk lamp and then connect to the motion sensor. From there you could attach the motion sensor to the bottom of your desk chair. This will give the result of turning on your desk lamp when you sit on your chair, but it doesn’t turn on your desk. To solve this you need to create an automation to turn on your desk when your Trådfri desk light-bulb goes from an ‘OFF’ state to an ‘ON’ state:

## Home Assistant is continuously expanding its components list, so the possibilities of automations subsequently carry on expanding

```
- alias: Trådfri Motion Sensor On
trigger:
  - platform: state
    entity_id: light.desk_lamp
    from: 'off'
    to: 'on'
action:
  - service: Home Assistant.turn_on
    entity_id: group.desk
```

This article is the tip of the iceberg when it comes to the possibilities of Home Assistant and Trådfri. Home Assistant is continuously expanding its components list so the possibilities of automations subsequently carry on expanding. IKEA’s Trådfri system will eventually release more colour bulbs, so the future is bright for the low-cost smart home lighting system. □

**Figure 4** Putty can save your SSH connections to make it easier when connecting next time. Just press Save



# Build a Bao Bao-inspired LED panel

Uncover a new world of geometric style with a triangle-based LED grid



Natthakit  
Kangsadansenanon

Natthakit  
Kangsadansenanon  
is an architect /  
artist from Thailand.  
He is interested in  
incorporating the  
intelligence of the digital  
technologies into the  
existing objects of daily  
life. His goal is to make  
objects that are at once  
simple, artistic, and  
imbued with a greater  
awareness of humanity.

## B

**ao Bao bags are built up out of right-angled triangles where two of these triangles, joined along the hypotenuse, make squares.**

The resulting grid pattern works well for bags because it allows a hard surface to bend in most directions. Here, we've adapted this pattern for an LED panel, where it allows us to light up a wider range of patterns than the more typical square-grids.

### STEP 1

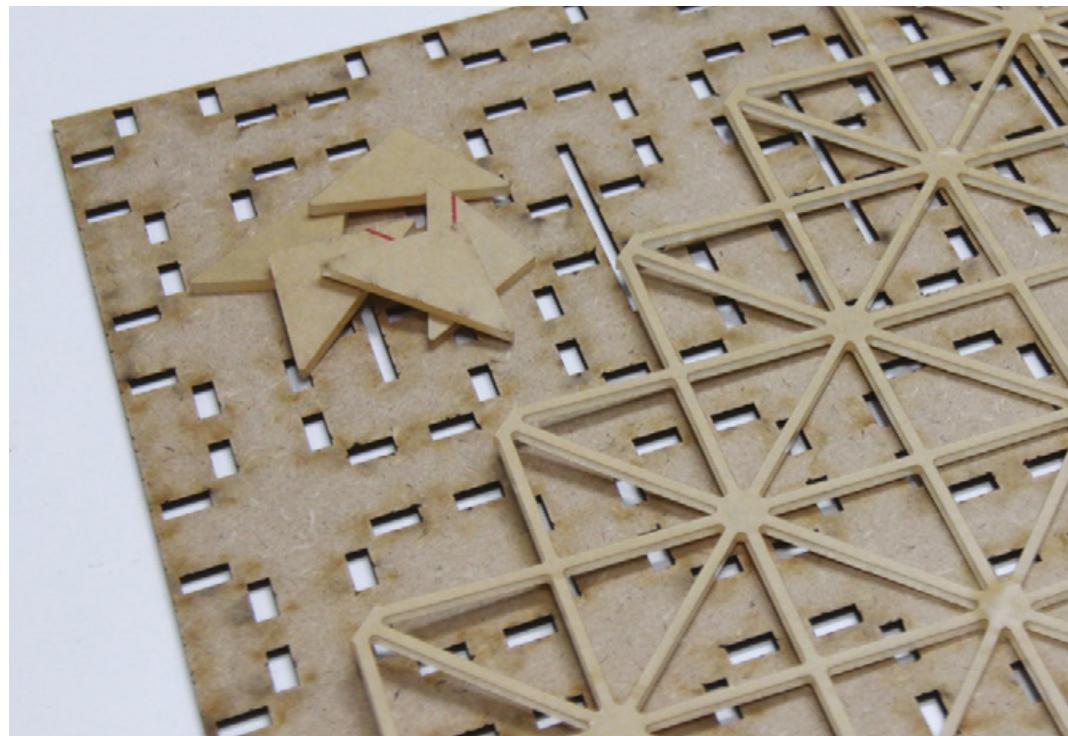
We designed this panel by measuring the actual sizes from a Bao Bao bag and designing an enclosure in

Rhino 3D. You can download the design files from [hsmag.cc/issue6](http://hsmag.cc/issue6). We made ours out of 3mm material, using black acrylic for the pattern frame, opal acrylic for the individual pattern, and MDF for the back and sides. You could use alternative materials if you choose to – provided they're 3mm thick and strong enough.

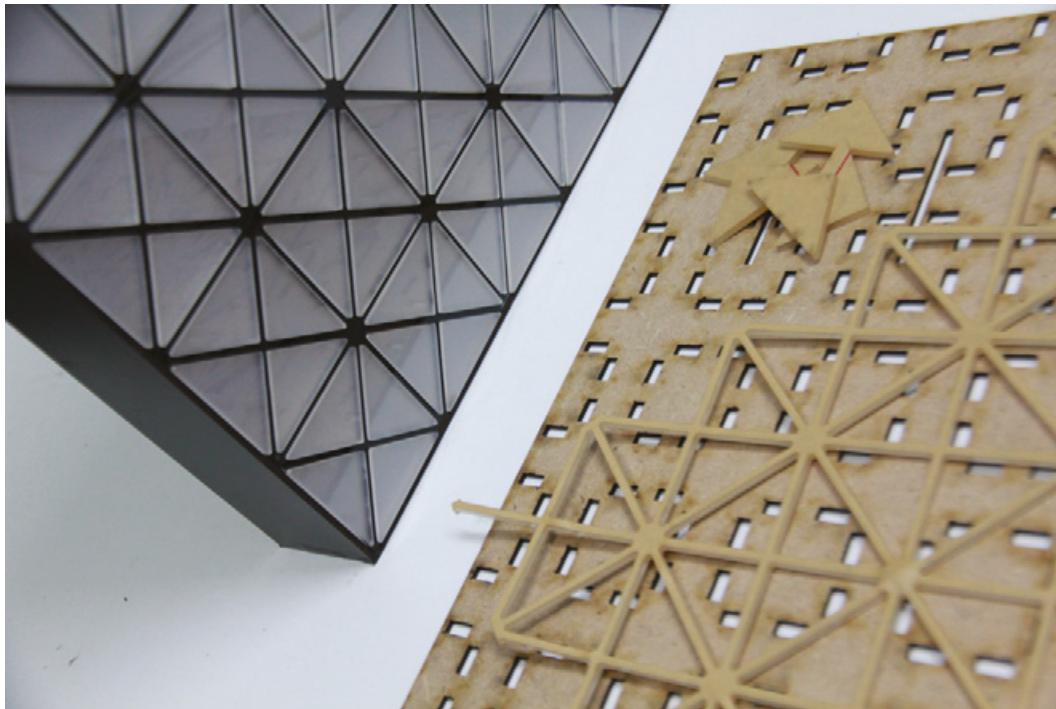
We finished the enclosure with a non-glossy black sticker to give it an anodised aluminium look, but this is optional.

### STEP 2

The pattern sticks out of the frame by 1.5mm to give it more resemblance to the real bags. This is done



**Right ♦**  
Laser-cutting  
allows us to create  
complex patterns  
much easier than  
traditional tools



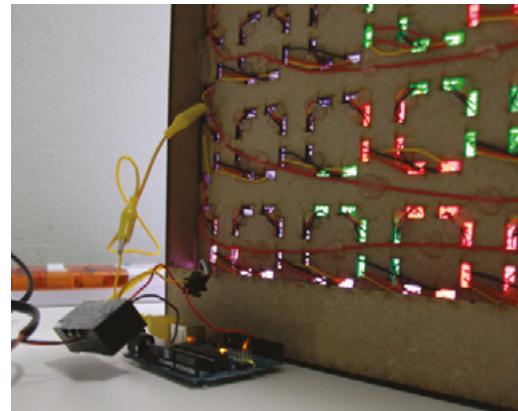
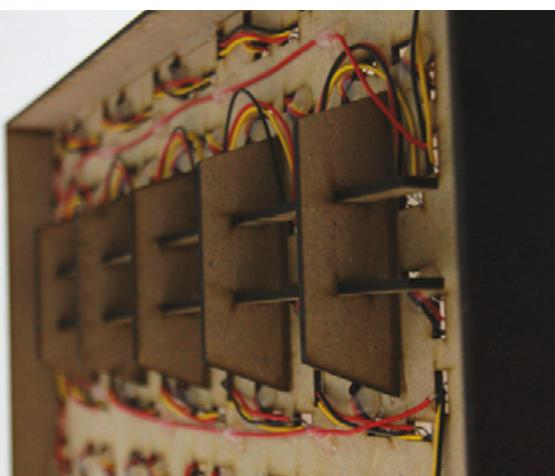
**Left** From initial cutting to final look

**Below** Addressable LEDs allow far simpler wiring than matrices of simple lights

by lifting up the frame 1.5mm from the table and inserting paper between them on both sides of the frame, so it leaves some gap in the middle to put the tiles in before gluing them in place.

### STEP 3

Now it is time to put the LEDs on the board. When it comes to this step, be aware that there are several ways to connect smart 144 LEDs (WS2812B) together, but we're going to detail the way that we found easiest. There are three input pins on each LED (5V, Ground, and Data In), and three output pins (5V, Ground, and Data Out). Each LED is connected to the next by linking the output ones of the previous to the input pins of the next.



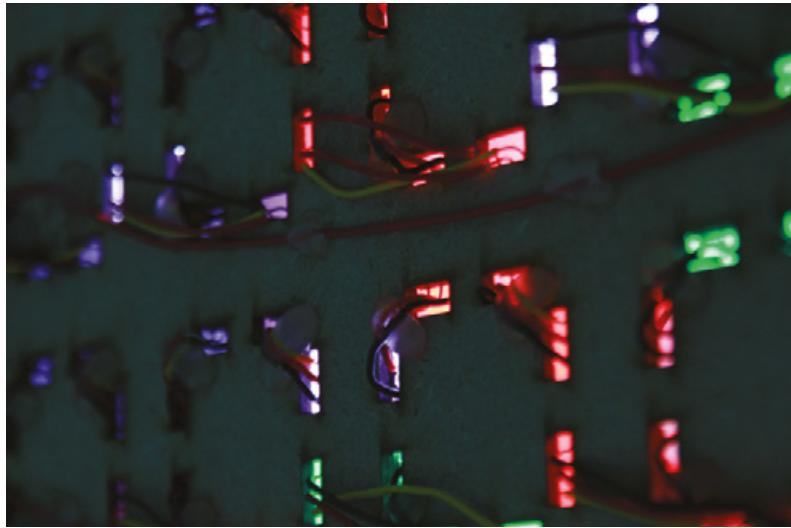
Each LED can consume 0.06A when it's running at its brightest white, which means that our panel could consume up to 8.64A. However, you don't necessarily need this much power. For example, each primary colour (red, green, or blue, obviously) only consumes 0.02A. As long as you're aware of the limitations when programming patterns, you can use a power supply that can supply less current. We used a 5A power supply, but this does mean we can't go full-white if we want.

### STEP 4

Feed the wires through the board and solder each LED in place. This can result in the back of your board looking quite spaghetti-like. We found it easiest to hot-glue all the wires in place, then flip the board over and do all the soldering in one go. →

**Left** An extra laser-cut structure on the back makes it easy to mount our panel

## TUTORIAL



**“The resulting grid pattern works well for bags because it allows a hard surface to bend in most directions”**

### STEP 5

We used a layer of paper to diffuse the LEDs. This gives an even glow to the pattern on the front, rather than showing every individual LED placing.

### STEP 6

With everything wired up, add a support to allow the panel to be hung on a wall.

### Below

Heated acrylic is easy to manipulate into different shapes. It's a great way of adding little details to projects

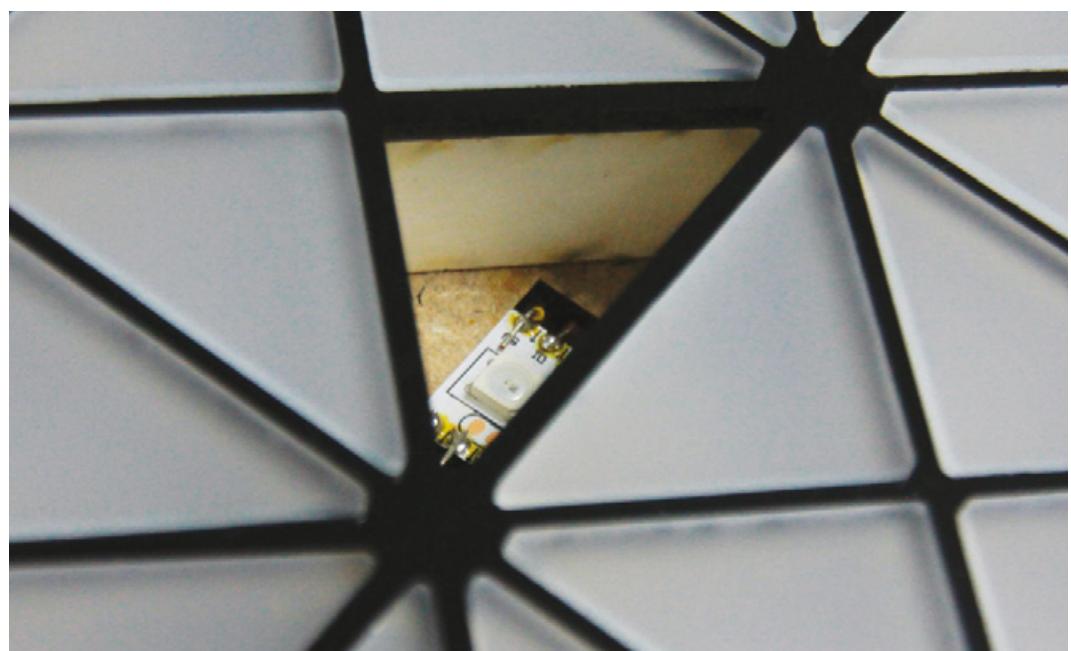


### STEP 7

As a nod to the panels' inspiration, you can add a bag strap. This is made from acrylic that's been softened with a heat gun until it becomes pliable enough to bend. Make sure you don't over-heat it though or you may end up with a puddle.

### STEP 8

Attach the strap. The Bao Bao bags have a keyhole attachment to allow you to change the length of the strap. We simulated this with a cut-out in the acrylic and glued the handle in place.

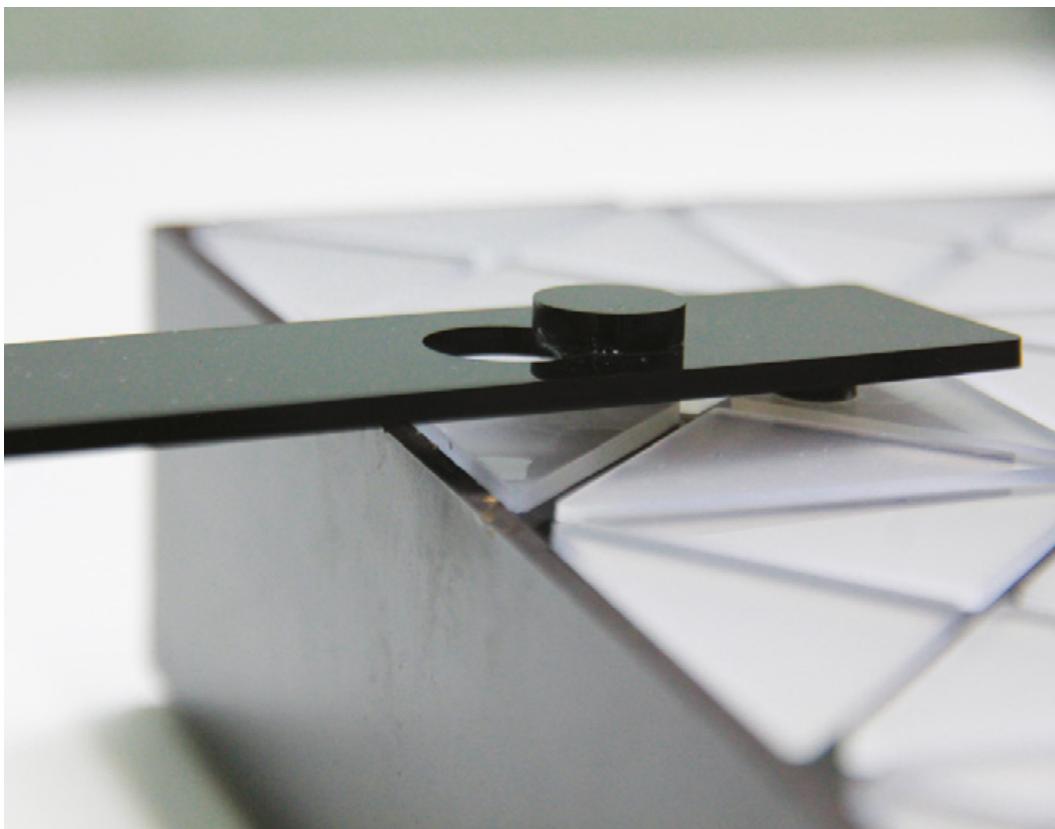


#### Above

When viewed from behind, high fashion turns to cyberpunk

#### Right

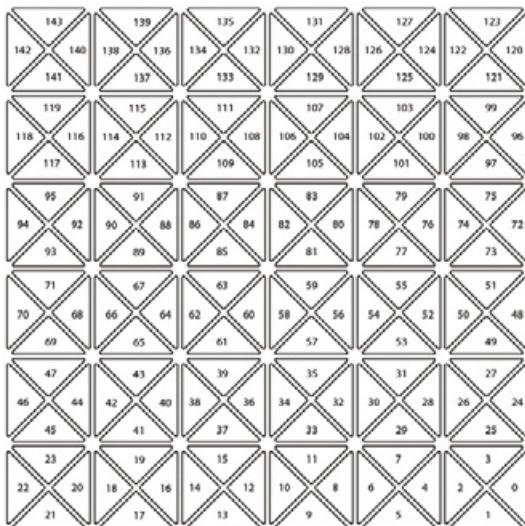
The NeoPixel LEDs come in many form factors, but the wiring is always the same. Pick whichever is easiest for you to wire up

**Left**

A little homage to the original bag designs that inspired the panel

**STEP 9**

Connect the Data In pin of the first LED to a data pin on an Arduino, then connect the 5V and Ground pins to your power supply. You'll need to connect the ground on your power supply to the ground on the Arduino, even if the Arduino is powered separately (such as through the USB connection). This common ground is needed to ensure that the GPIO signals can be properly transmitted.

**STEP 10**

The Adafruit\_NeoPixel library gives you a convenient way to control the LED, and includes some demonstration functions that should put your LED panel through its paces. Once you've got these running, you should be able to create your own designs by modifying the example code. Both animations and static images can look great. Don't forget to let us know what you come up with! □

```
Issey.pallette | Arduino 1.6.5
File Edit Sketch Tools Help
Issey.pallette
Issey.pallette
487
488 void hockeystickPattern() {
489     int hockey1[10] = {100, 98, 90, 78, 60, 50, 68, 64, 40, 38};
490     int hockey2[10] = {101, 79, 81, 59, 61, 39, 65, 43, 69, 67};
491
492     for(int k = 0; k < NUMPIXELS; k++) {
493         strip.setPixelColor(k, 0);
494     }
495     strip.show();
496
497     for(int k = 0; k < 8; k++) {
498         for (int i = 0; i < 10; i++){
499             strip.setPixelColor(hockey1[i], navy);
500             strip.setPixelColor(hockey2[i], yellow);
501         }
502         strip.show();
503         delay(500);
504     }
505     for (int i = 0; i < 10; i++){
506         strip.setPixelColor(hockey1[i], yellow);
507         strip.setPixelColor(hockey2[i], navy);
508     }
509     strip.show();
510     delay(500);
511
512     for(int k = 0; k < NUMPIXELS; k++) {
513         strip.setPixelColor(k, 0);
514     }
515     strip.show();
516
517 }
```

**Left**

You don't have to use this ordering, but you do need to remember the way they're wired

**Below**

There's nothing special about the code for this. Any NeoPixel code will do the business

# Make your own Gin

Relax after a hard day in the workshop with a homemade tipple



**Ben Everard**

@ben\_everard

Ben has tried far more ways of making alcohol than are really sensible from the surprisingly nice mango cider to the frankly horrible strawberry jam wine. If he ever offers you a drink, be wary.

**T**here are basically two ways of making gin – most commercial gin makers use distillates of the various botanical flavours to keep the resulting spirit crystal-clear. DIYers, on the other hand, usually soak the flavourings in the gin to impart their flavour more directly. This second method is known as compound gin (though it's often known as bathtub gin in reference to the containers used during Prohibition in America), and it's what we'll do. The biggest advantage of making your own gin is that it can contain exactly the flavours you want. You don't need to follow our recipe directly – throw in a few more juniper berries if that's your thing; some citrus peel can give it a nice tang. You can go even further off-piste if you like: Twisted Spirits make a Douglas-Fir flavoured gin, while Hendrick's Gin is famously made with cucumber.

## WHISKY

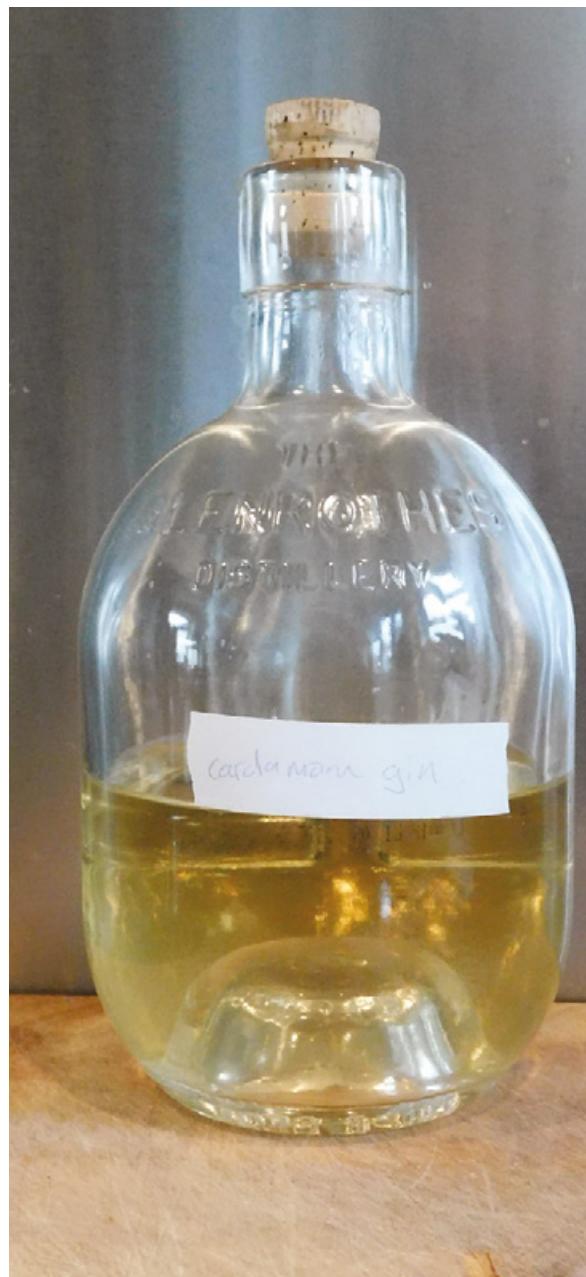
Many spirits are really just a grain alcohol that's flavoured in some way. Whisky, for example, starts life as a clear spirit from grain (or malted grain), but it's then left to age in oak barrels. It's these oak barrels imparting their flavour in the drink that makes it whisky.

So, does this mean that you can take vodka, infuse it with wood, and end up with whisky? We'll let you know. Unlike juniper, oak takes a long time to flavour the drink (hence why whiskies are often aged for several years). Two years ago, this writer started an experiment with oak in vodka, using toasted oak shavings.

It hasn't become whisky yet – or at least, it doesn't taste like it yet. While there's a definite whisky note to it, it's still a subtle background flavour and couldn't really be confused with whisky just yet. Two years though is still quite young, so it may later take on the full whisky flavour. Of course, it may never make it. Whisky is barrelled at a higher alcohol percentage than the vodka we started with, and this may help it dissolve more sugar from the oak than our elixir will manage.

### Right ↗

Your gin will take on a slight colour from the botanicals – don't worry, it's all part of the homemade aesthetic, honestly





**Left** The botanicals we used for our gin – they're more of a starting point than a recipe

## BATHING IN FLAVOUR

The process couldn't be simpler. Start with a neutral-flavoured grain spirit – which is exactly what most commercial vodkas are. To this, you need to add your botanicals and leave them to steep. There are no hard-and-fast rules for how long – essentially it comes down to a combination of how potent your raw ingredients are and your personal tastes. We recommend tasting every day until you get the desired flavour – when tasting, make sure that you test in the same way you'll be enjoying the end product (such as with a mixer), as the raw spirit can have a deceptively strong flavour.

We've given a recipe for 500 ml of gin, but you can adjust the quantities upwards or downwards if you prefer. As there's a wide range of options available, we'd recommend starting with several variations on this recipe in small batches to see what you prefer, then honing in your personal recipe.

Our recipe makes a gin with a cardamom undertone. We'd recommend using cheap vodka for experimenting with; you may or may not decide to try it with mid-range vodka once you've found a flavour combination that you enjoy. We wouldn't recommend using high-end vodka, as the taste difference isn't likely to be noticeable. Botanicals lose their flavour over time, so you may find that you need to increase the quantities if your ingredients have been sitting in your spice cupboard for a long time.

**The biggest advantage of making your own gin is that it can contain exactly the flavours you want**

The only ingredients that are essential to gin are vodka and juniper. Everything else is optional, and good gin is often the result of bold flavour choices.

- 500 ml vodka
- A tablespoon of juniper
- Half a teaspoon of coriander
- 5–6 cardamom pods

If you can't get juniper, you can always start with a basic juniper gin and add other flavours as you wish. This is the method commonly used by people making fruit gins, such as sloe gin. However, in these cases, you have much less control over the botanical flavours in the final product, so we'd recommend starting with a neutral spirit and adding all the botanicals yourself.

If you haven't reached the desired flavour after four days, we'd recommend adding more botanicals.

Although the end result is delicious, please remember to drink your gin in moderation. □

## TOM COLLINS

The classic way of enjoying gin is with tonic, but this can mask the subtle flavours of gin and you may prefer to enjoy your homemade gin as a Tom Collins. To start with, you'll need to make some sugar syrup – dissolve some sugar in an equal quantity of hot water, then let it cool. Then mix your gin with a squeeze of lemon, some sugar syrup, and top up with soda water (adjust the quantities to your taste). Drop in a couple of ice cubes and enjoy, as the sun sets on another happy day of hacking and making stuff.

# Build your own sumo robot

An ancient Japanese art meets robotics on the (not so) big stage



Rob Wood

Rob has been using computers almost all of his life. He is a keen amateur robot maker, and works in the telecoms industry.



**O**n 20 February 1998, a new television show hit the UK airwaves. Based on an idea from the USA, and hosted by Jeremy Clarkson, *Robot Wars* made its global television debut on BBC2 and a generation

of people of all ages became hooked on watching homemade robots destroy each other in a bid to become the champion. Once the dust had settled, Road Block was crowned champion.

For many technically minded people in the UK, this series marked the start of an interest in building things. It gave a rare glimpse into the world of physical robotics and made these machines seem achievable to the hobbyist.

The TV show has had a tricky relationship with the BBC and was cancelled after eight series. It rose again in 2016, but, as we go to press, the series has not been renewed in 2018 (though this may have changed by the time you read this).

**Right** ♦

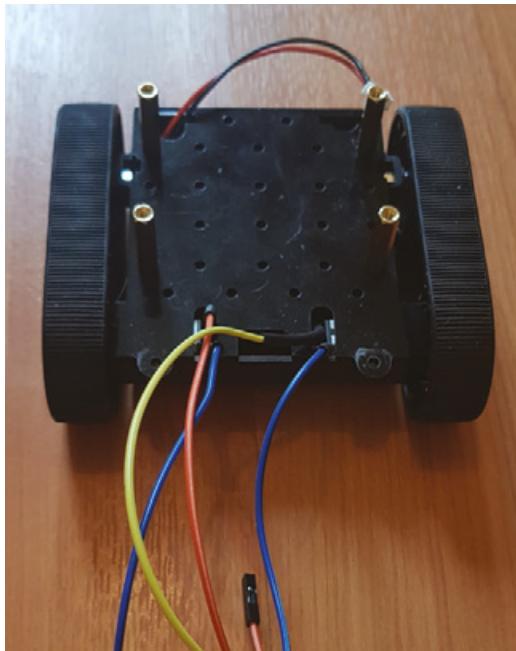
Just like real fighters, sumo robots need a stare to intimidate the opponent

While *Robot Wars* provides an impressive spectacle for TV viewers, there are a few drawbacks to it for makers – the sheer level of damage inflicted on the robots makes it a time-consuming hobby. For each round, there's much building, fixing, and general maintenance. Secondly, it's almost totally hardware-based. The robots are remote-controlled, so there's little scope for people interested in software as well as hardware.

Robot sumo is another robot battle discipline where, instead of trying to destroy the opponent, you need to push the competitor out of the ring. Robots rely on their own programming – rather than a human operator – once in the field of combat, so a good bot needs both good hardware and software to succeed.

There's a range of different weight classes, with mini sumo being one of the most popular. It's small and cheap enough to make it accessible, but still has enough grunt to make it interesting. Let's take a look





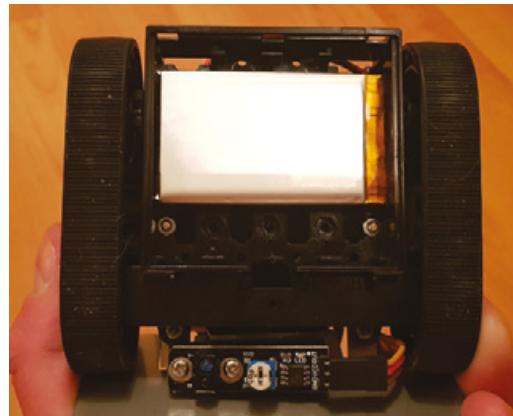
**Above** View of the completed chassis, with the motor wires and the battery wire ready to be connected later in the build

at how to build a simple mini sumo bot using some off-the-shelf hardware and a Raspberry Pi Zero.

The following tutorial will guide you through building a robot capable of being entered into mini robot sumo class. The base for this design is Pololu's Zumo chassis, combined with a 4tronix Picon Zero motor controller board, and a Pi Zero W.

## STEP 1

Assemble the chassis. As we're using an off-the-shelf chassis, all the assembly instructions are provided through the Pololu website. There are plenty of other



options available, and you could even build your own from scratch if you wish.

## STEP 2

On the Zumo chassis there are no mounting points that line up with a Raspberry Pi. The next step is to take all the components that will sit on top of the chassis and also position the sensors. First the base plate for the Pi: the plate measures 75mm x 65mm. Drill holes to mount the plate to the chassis and the Raspberry Pi. You can take the measurements for the mounting holes from the chassis, the Pi, and the LiPo battery circuit. The mounting plate for the sensors is a bit more complex, as it has to be wide enough to fit the IR range finder and the NeoPixel strip, but narrow enough to fit in the gap between the chassis and the sumo blade. The mounting plate has a small piece of plasticard (30mm x 15mm) on the back to attach it to the base plate with two M2.5 screws. This is reinforced with two small offcuts of plasticard to provide structural integrity. Lastly, connect the cables for the range finder and the line follower and solder ground, power, and signal input leads to the NeoPixel. Attach the IR range finder, the NeoPixel strip to the top of the front plate, and the line-following sensor at the bottom and attach with screws to the base plate.

## STEP 3

Now that we have somewhere to mount our components, let's put it all together. Firstly, bolt a 10mm M2.5 brass standoff to each of the Raspberry Pi's mounting points, leaving the screw holes facing the top and the screw thread on the underside of the Pi. Next, mount the Picon Zero to the Pi's →

## YOU'LL NEED

- ◆ Pololu Zumo chassis
- ◆ 2 x Micro metal gear motors
- ◆ Raspberry Pi Zero /Zero W with a 40-pin GPIO header attached
- ◆ 4tronix Picon Zero robotics controller
- ◆ Adafruit PowerBoost 500 5 V LiPo charger / power supply
- ◆ 1200 mAh LiPo battery pack
- ◆ Adafruit NeoPixel strip (RGBW LEDs)
- ◆ 1 x Line follower sensor pack for Robo:Bit Buggy (part of a pack of 2 with included wires)
- ◆ 1 x Sharp GP2Y0A2TYK0F IR range sensor 10cm to 80cm
- ◆ 1 x SIRC-01 Sharp GP2 IR sensor cable
- ◆ Push-button
- ◆ 10 kΩ resistor
- ◆ 1 x sheet A4 plasticard 1mm thick
- ◆ Liquid poly / plastic modelling kit glue
- ◆ Jumper jerky
- ◆ Assorted M2.5 standoffs

## TOURNAMENT RULES

There are a wide range of different rules for robot sumo, which dictate the specifications of robots that can enter. Make sure you know what particular rules are used before committing to an event. A common option is the UK Mini Sumo rules:

The robot must be able to fit in a box with a width and depth of 10 cm. There are no restrictions on height.

Weight (including accessories) must not exceed 500g.

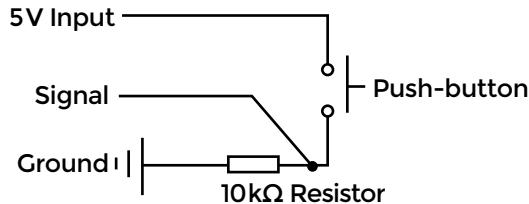
There are no restrictions on the type of control method used with self-operated robots.

A self-operated robot should be designed to begin action five seconds after the contestant presses the robot's start button.

There are no restrictions on the brand or memory volume of the microprocessor used for a robot.

Source: [hsmag.cc/CVuNnJ](http://hsmag.cc/CVuNnJ)

## TUTORIAL



**Right** A button with pull-down resistor is the only interactive element a sumo bot needs

GPIO connector and secure with three screws – the final one we will use later. Next, attach a 20mm-tall brass standoff to the underside of the Pi and a 5mm-tall standoff to each of the mounting holes on the LiPo circuitry. Mount the LiPo circuitry first, then the Pi to complete the construction of the base plate.

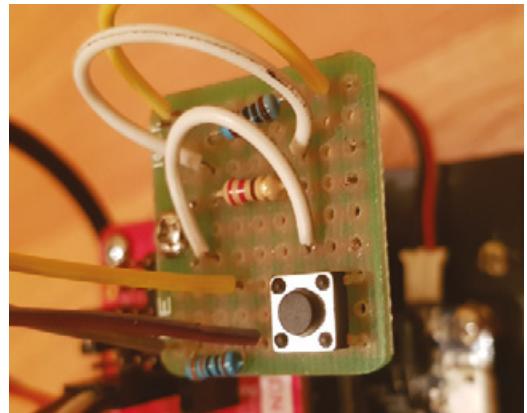
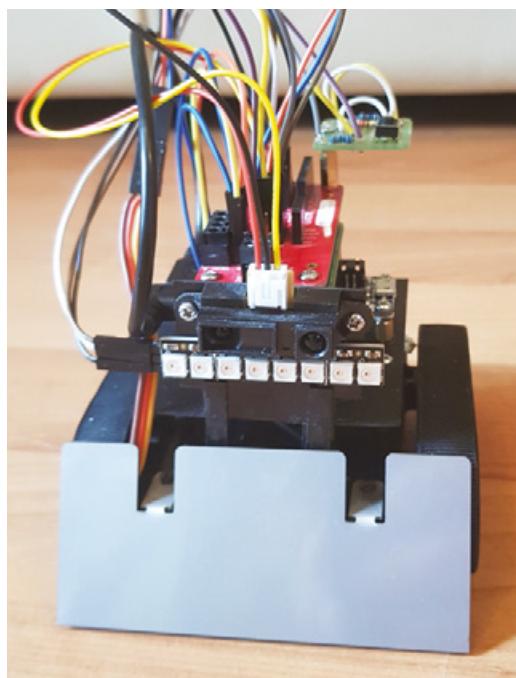
### STEP 4

Now we will connect the sensors to the Pi. Before we connect the sensors to the Picon Zero, it is worth noting that the channels are 0–5 for the outputs and 0–3 for the inputs. These are marked on the board. Connect the range finder to input channel 0 and the line follower to channel 1. Channel 2 is left free for the control button which we will add later. The NeoPixel strip must be connected to output channel 5, as this is the only channel that supports the NeoPixels.

### QUICK TIP

If you are new to robotics, you may want to start with motors that have a high gear ratio like the 298:1 motors we have used in this build. The slower speed is ideal for minimising accidents and they can be easily swapped for lower-ratio, i.e. faster, motors for competition.

**Right** LEDs can provide useful status information. In our build, the purple light denotes that the sumo program is running and waiting to start the bout



### STEP 5

Attach a button. The circuit has a 5V input on one side and an output on the other. A 10kΩ resistor is connected between the signal and ground. Once the circuitry is soldered, connect the ground, power, and signal wires to input channel 2 on the Picon Zero. We used a small piece of protoboard to create the buttons circuitry. This is then mounted using a 20mm brass standoff in the final hole we left earlier when connecting the Pi and the Picon Zero.

### STEP 6

Now that we have all our parts assembled, we need to create some code to get the robot to fight. The first step is to create small test scripts for each component so that you can check everything is wired up correctly. The Picon Zero has a reference library available, and there is also a function guide on its website, [hsmag.cc/xJmcng](http://hsmag.cc/xJmcng).

The button and the line follower are both digital sensors in that they are either 0 for low, or 1 for high. The range finder is an analogue sensor and reports back a value based on the voltage of the IR receiver. The further an object is away from the sensor, the

## RASPBERRY PI OR ARDUINO?

Your sumo robot needs a brain and there's a wide range of options out there. For many makers, that choice comes down to Raspberry Pi or Arduino, as these two have far more help and information available online than any other option.

When considering what platform to base your project on, it's a good idea to choose something that matches your skill set. If you are new to programming you may wish to choose the Raspberry Pi, as there are many tutorials on programming for both Python and Scratch. The Arduino platform also has many examples for those getting started.

Of course, any computer or microcontroller that's both small enough to fit in and low power enough to run off batteries could be used.

lower the value becomes. In the example code, the range finder only triggers a function within the code once the received value hits 175 or higher. This equates to distance of about 25cm.

## STEP 7

Now that we are happy that everything works, it's time to bring it all together and create the main code (see Tactics For Battle box). Simplicity is your friend when starting with sumo bots. The bigger and more complex your code is, the more likely something will go wrong like a mistake or coding error. We've got

the bare necessities of code that you can expand on as you find new and improved tactics.

## STEP 8

Now that the coding is complete, it's time for the ultimate test – battle! Mini robot sumo has large communities all over the globe, with competitions taking place in the USA, Japan, and mainland Europe. There are a few fights in the UK, and the place to find them is at [hsmag.cc/oddHCL](http://hsmag.cc/oddHCL). Take a look at the forums where like-minded individuals can swap stories and even arrange friendly bouts. □

## TACTICS FOR BATTLE

There's no correct set of code for a sumo bot – it's all about the battle and different tactics. Let's start with a simple script that you can get started with (before developing it in your own direction). This code simply spins the robot (which we've named BodgeIT) until it sees a competitor, then tries to push it out of the way.

```
#!/usr/bin/python
import piconzero as pz
from time import sleep

#Initiate Piconzero and set the inputs and outputs
pz.init()
pz.setInputConfig(0, 1)          #IR range finder
pz.setInputConfig(1, 0)          #Line detector
pz.setInputConfig(2, 0)          #starting button
pz.setOutputConfig(5, 3)         #Neopixels

#Main combat routine
def sumo():
    while True:
        #Reads the IR range finder and the line
        detector
        a = pz.readInput(0)
        b = pz.readInput(1)
        #Detects white line at the edge of the arena
        #Orange light when tripped
        if b == 0:
            pz.stop()
            sleep(0.1)
            pz.setAllPixels(255, 128, 0)
            pz.forward(100)
            sleep(1)
            pz.spinLeft(100)
            sleep(1.6)
            pz.stop()
            sleep(0.1)
```

```
#Detects opponents in range, red light for
attack
elif a >= 175:
    pz.stop()
    sleep(0.1)
    pz.setAllPixels(255, 0, 0)
    pz.reverse(100)
    sleep(2)
    pz.stop()
    sleep(0.1)
#Spins BodgeIT to search for opponent, green
light for search
else:
    pz.setAllPixels(0, 255, 0)
    pz.spinRight(100)

#main loop, waits for user input via the button
try:
    while True:
        #Reads the status of the button
        x = pz.readInput(2)
        #Starts the sumo routine
        if x == 1:
            pz.setAllPixels(255, 128, 0)
            sleep(5)
            sumo()
            #Waiting for input, displays purple light
        else:
            pz.stop()
            pz.setAllPixels(255, 0, 255)

#Cleans up
finally:
    pz.cleanup()
```

# Hacking a remote-control car

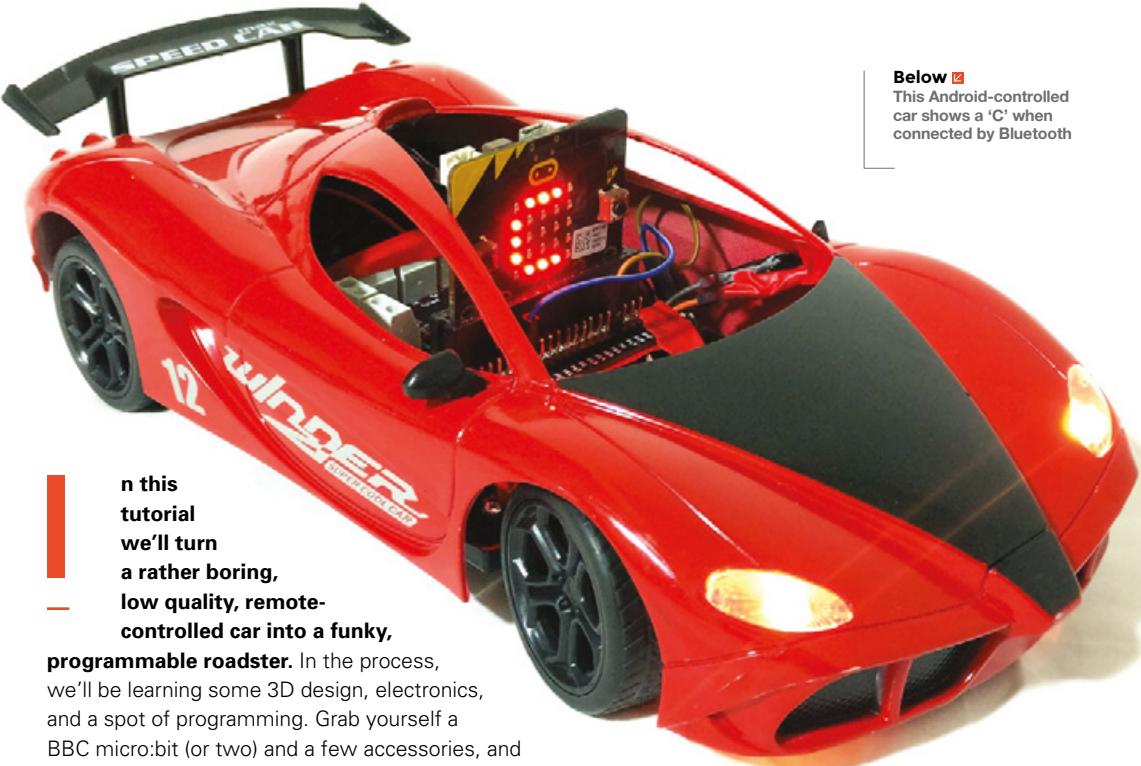
Pop open the bonnet to improve that tired toy



Andy Moxon

↗ [hsmag.cc/XxaAuV](http://hsmag.cc/XxaAuV)

Andy Moxon is a former Rolls-Royce engineer, now working in Kosovo, where he runs a club which enables teenagers to explore science through the Raspberry Pi, BBC micro:bit, and electronics.



In this tutorial we'll turn a rather boring, low quality, remote-controlled car into a funky,

**programmable roadster.** In the process, we'll be learning some 3D design, electronics, and a spot of programming. Grab yourself a BBC micro:bit (or two) and a few accessories, and watch how easily a toy can be used to learn some fundamental engineering principles.

So where did it all start? Some friends gave my two young boys one of these remote-control cars each for Christmas. However, much to their initial amusement (and then annoyance), being on the same frequency, one controller operated both cars! Then Daddy noticed, to his dissatisfaction, that the steering had just three positions (left, centre, and right) and the car slowed significantly on turning. Out came the screwdriver, off came the lid, and things were never the same again!

## DADDY TOOK MY CAR APART! INSIDE THE CAR

We can split the car's workings into three sections: steering, driving motor, and controller.

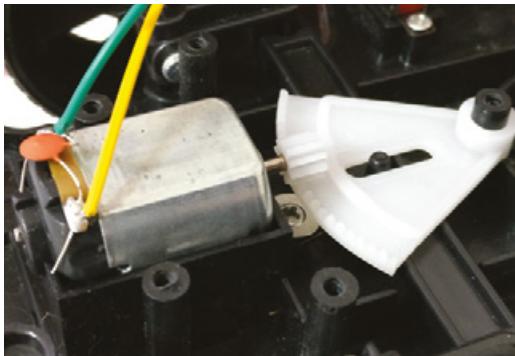
**Below** ↗

This Android-controlled car shows a 'C' when connected by Bluetooth

## QUICK TIP

Measurements for creating 3D-printed parts are done much faster and accurately using Vernier Callipers.

The steering offers the greatest hacking potential and the answer to two of our problems. The front wheels have short 'steering arms', neatly joined to each other by a black 'tie rod'. On top of this, we have a white 'rack', which is pushed left and right by the white 'pinion gear'. Using a DC motor to rotate this gear might be a cheap solution, but it is also the root cause of two of our three problems. Firstly, without a sensor, you can't set the exact position of a DC motor. When powered, the motor pushes the rack to one or other side. When off, the spring returns it to centre. Secondly, after the initial movement, the motor can no longer rotate. It is 'stalled' – a condition which leads to high motor currents. This reduces the

**Above**

The old steering system features a rack-and-pinion, driven by a simple DC motor and returned to centre with a spring

power available to the driving motor and drains the batteries faster.

In the driving motor, we find a second reason for slowing down on cornering. At the back of the car, the driving motor is held inside a gearbox, attached to a solid rear axle. When the car steers, the inside wheel travels a shorter distance than the outside one. The axle in our car forces both wheels to rotate at the same speed, requiring one tyre to slip on the ground. This is normally solved using a mechanism called a 'differential'. All very interesting and important in real life, but we're going to rule this improvement outside the scope of our project!

The controller is a small circuit board which takes power from the four AA batteries (6V), receives signals from the controller, and outputs power to the motors and headlights. This is the cause of our final problem. Both cars operate on the same frequency!

## HACKING THE SOLUTION

The centre of our solution will be a BBC micro:bit and the excellent Kitronik motor driver board. As well as having two high-current outputs (designed for two separate motors), it provides 3V power for the micro:bit itself. The complete board sits neatly on top of the battery compartment of our car.

### Here's how the project's going to work:

1. Connect the Kitronik driver board
2. Design and manufacture the steering solution
3. Program the micro:bit

## CONNECTING THE DRIVER BOARD

Cut the wires connecting the circuit board to all the peripherals and remove it. Strip off 5mm of insulation from these. Insert the motor wires into one of the motor output terminals of the Kitronik board and the battery wires (maintaining the use of the switch) into the terminal block, to provide power.

A BBC micro:bit can power LED lights straight from the breakout pins (up to a maximum current of 5mA

per pin). However, the headlights on this car are not LEDs and would draw too large a current, potentially damaging the micro:bit. The solution is to use the spare 'motor' output on the Kitronik board, designed to draw up to 1.5A. If we wire the lights together in parallel, then we'll maintain their brightness, while being able to operate them both from the same output.

## DESIGNING THE STEERING SOLUTION

To control the steering, we're going to remove the DC motor and replace it with a servo motor. This can be commanded to any position in a 180° arc and can be powered directly from the micro:bit's 3V output. We'll mount it in the same place as the original motor, in order to maintain the 'rack-and-pinion' mechanism.

### However, we're going to need to 3D-print two additional parts to make this work:

- a new pinion gear (connecting the servo to the rack);
- a bracket (to hold everything together).

## PINION GEAR

Firstly, we need to size the pinion gear so we can travel from lock to lock (full left to full right) on the steering, using a servo which only rotates 180° (i.e. half a rotation). To do this, the rack must slide a distance of 36mm. Do the maths and we get a minimum gear radius of  $r=11.5\text{ mm}$  ( $2\pi r=36 \times 2$ ). This turns out to be almost exactly the height of the servo's spindle from the rack, if we simply glue the flat side of the servo to the top of the old steering motor's housing. Sorted! →

**Below**

The MG90S servo motor gives exact control over the steering position of the front wheels



## YOU'LL NEED

### Equipment

- ◆ Cheap remote-control car (and permission to take it apart!)
- ◆ BBC micro:bit (optionally two if you're not going to control it from a mobile device)
- ◆ Kitronik motor driver board for the BBC micro:bit
- ◆ SC90S mini servo motor (The 'S' has metal gears. Without the 'S', the gears are plastic which would be fine for this application.)
- ◆ A few jumper wires
- ◆ Insulating tape

### Tools

- ◆ Access to a 3D printer (or try an online shop)
- ◆ Soldering iron
- ◆ Hot glue gun

## TUTORIAL

**Right** The steering servo is wired directly into the micro:bit's breakout pins



### MATHS IN ACTION

Secondly, we need to get the pitch of the gears right (i.e. the distance between the teeth). Get this wrong and the pinion gear will not mesh with the rack. There are 22 teeth on the rack, so  $36/22 = 1.6\text{mm per tooth}$ .

Finally, we move into FreeCAD. In the 'Part Design' workbench there is a helpful 'Involute Gear' function, which draws the gear for us. Playing around with the various parameters produces a gear which fits the above specification: a 37-toothed gear with modulus of 0.5mm. Now we must remove a hole in the gear's centre which will push-fit to the servo's spindle. To do this, we click the gear's face and then 'Create a new sketch'. We draw a circle at the centre of the gear, of the same diameter as the servo's spindle. This will produce a tight fit. Finally, we exit the sketch and 'Create a pocket' from it, to bore a hole through the gear.

### BRACKET

As we measure the car for the new bracket, it's always best to produce a rough sketch of what it's going to look like, before starting the 3D design. This saves us time in front of the computer.

### ROUGH SKETCHES SAVE YOU TIME LATER

Our design is based on the original bracket. We're just removing the steering motor cover to allow room for our new pinion gear, hence the cut-out from the triangle.

It's important to allow at least 0.2 mm tolerance (extra diameter) on the holes, so that the fit (particularly to moving parts) isn't too tight. This varies according to

### MOTOR DRIVERS

Motor drivers allow low-current microelectronics to control high-current devices, but watch out: not all motor drivers are created equal! Since our project is using low voltages (6 volts from 4 x AA batteries), if we lose a lot of voltage just across an unsuitable chip, then we'll find we've got a clunker, not a roadster! The Kitronik board is based around the DRV8833 integrated circuit, known for its low voltage drop. Stay clear of stone-age chips such as the L293 series!

3D printer settings but, to an extent, can be corrected with a craft knife post printing, if the fit is too tight.

In FreeCAD, the basic shape of the bracket is just one sketch (in the Part Design workbench). In addition we need some small 'Pads' (extruded sketches) and 'Pockets' (cut-outs) to produce additional 3D features.

### MANUFACTURING THE STEERING SOLUTION

Let's move to the 3D printer. Both the above parts are exported from FreeCAD as STL files. Since we're using an Ultimaker 2+, we used Cura, but you may need to use different software depending on your printer.

In general, the bracket is not that challenging for a 3D printer. One face is completely flat (for good adhesion to the build plate) and there are no overhangs.

The gear is challenging because of its fine detail, so change the nozzle to the smallest size (on the Ultimaker, this is 0.2 mm) and slow the printing down to just 10% of the default first-layer speed to enable build plate adhesion and accuracy.

### ACTION!

To operate the car, these are the pins we need to write to:

#### Pins 0 and 16: Main Motor

- 0,0 (pin 0 = 0, pin 16 = 1) to stop
- 0,1 to go forwards
- 1,0 to go backwards

#### Pins 8 and 12: Headlights

- 0,0 to turn on
- 0,1 to turn off

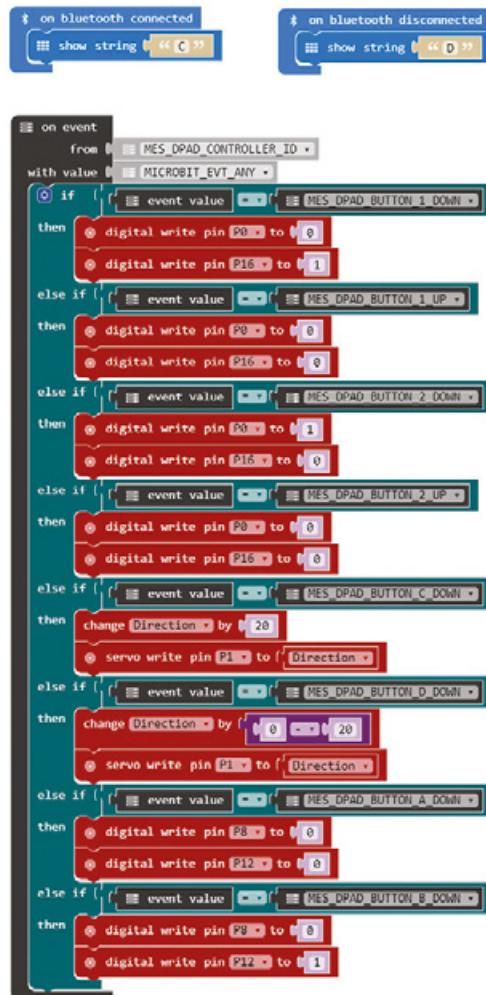
#### Pin 1: Steering

- Angle of servo using 'Servo write'

Once printed, the pinion gear is pushed onto the servo's spindle and the servo can be hot glue-gunned into place. The surface area for gluing is low, so an elastic band is used to ensure good meshing between the rack-and-pinion gears. The new bracket then holds the steering mechanism together, after the addition of the original screws. Finally, the servo's brown wire is attached to the micro:bit's ground, the red wire to the micro:bit's 3V output, and pin 1 to the yellow 'signal' wire.

## PROGRAMMING THE BBC MICRO:BIT

For this tutorial, we'll look at how to control the car from a mobile phone and program the micro:bit using the Microsoft MakeCode platform. However, the possibilities are endless and you could easily control from another micro:bit (try using the in-built accelerometer to control the steering and using the



'radio' function to transmit to the car) or program it in JavaScript, C, or Python.

Firstly, we need to download Bitty Controller on an Android or iOS device. This app communicates with the micro:bit using Bluetooth Low Energy.

Start a new project at [makecode.microbit.org](https://makecode.microbit.org). We need to add the Bluetooth extension (Advanced>Add Package>Bluetooth) which will at the same time remove the radio package (they conflict with each other). The three main blocks we need are shown in **Figure 1**.

- 'On Bluetooth connected/disconnected' – a letter is shown on the micro:bit's LED matrix.
- The 'On event' block receives the button presses from the Bitty Controller app – you can find the button names at [hsmag.cc/IKPmJC](http://hsmag.cc/IKPmJC).
  - When DPAD\_BUTTON\_1 is pressed (\_DOWN), the car drives forwards. When it is released (\_UP), it stops.
  - DPAD\_BUTTON\_C increments the variable 'Direction' by 20 degrees and makes the steering servo motor rotate 20 degrees, turning the front wheels left.
  - DPAD\_BUTTON\_D decrements 'Direction' and makes the front wheels turn right.
- DPAD\_BUTTON\_A and B turn the headlights on or off.

Finally, we save the code and download it to the micro:bit. When that's finished, we just need to pair it with the mobile device.

In the mobile device's settings we turn on Bluetooth, then reset the micro:bit, keeping buttons A and B held down. This takes us into 'pairing mode', so the micro:bit will be found by the mobile device. Having paired (a tick is displayed on the micro:bit's LED matrix), reset the micro:bit again and open Bitty Controller on the device. Scan for the micro:bit, select it from the list, and you'll be away.

## IMPROVED 'COOL FACTOR'

After all that, what have we done? Well, we've solved our three initial design problems: giving better cornering speeds, variable steering angles, and eliminating interference from other cars. However, and more importantly, we've created a cool, smartphone-controlled car and learned some great engineering principles on the journey. Similar techniques can improve a wide range of toys. Go forth and see what you can hack. ☺

## QUICK TIP

### Extensions

**Why don't you make a police car?**  
Just attach a small speaker and some blue LEDs to a spare breakout pin on the motor driver board.

# Control a Raspberry Pi with your voice

Build a voice-activated temperature sensor with a Raspberry Pi



**Above ↗**  
All that's required! Of course, the Amazon Echo can be located anywhere within WiFi range



**David Primus**

[@dmprimus](#)

David Primus lives in the mountains of Colorado tinkering with technology, writing, and doing historical research. Visit [barlazym.net](http://barlazym.net) for more information.

**V**oice-controlled artificial intelligence (Voice AI) is spreading rapidly. In addition to Amazon's Alexa, other popular Voice AI systems are Google's Assistant, Apple's Siri, and Microsoft's Cortana. Voice control will soon be integrated into automobiles, appliances, and other common technology.

In this tutorial, you will create a simple interface between your Raspberry Pi and an Amazon Echo. Using a weather sensor attached to the Pi, you will be able to ask Alexa (Amazon Echo's voice assistant) the temperature, humidity, and barometric pressure inside your house. You will learn how Voice AI works by developing a simple Amazon Echo application, called

an Alexa skill, running in the Amazon Cloud. This skill will control a Python program on your Pi. Once you learn more about developing software for the Echo, perhaps you will have a good idea for an Amazon Alexa skill and will be able to publish your own!

The first step is to get both computers up and running. Install Raspbian on your Pi and connect it to your network. Then, if you haven't done so already, set up the Echo, ensuring it is connected to your WiFi and responding to voice command.

Adafruit's BME280 sensor returns temperature, humidity, and barometric pressure. It can be interfaced to the Pi using either I<sup>2</sup>C or SPI; we'll use I<sup>2</sup>C, as the wiring is a little simpler.

We connected the BME280 to the Pi using a breadboard and a T-Cobbler, but as long as the right pins are connected together, you could use any other connectors. Connect the ribbon cable to the Pi, ensuring Pin 1 (indicated by the white stripe on the cable) is at the end near the microSD card. Connect the other end to the T-Cobbler (it only goes in one way) and insert the T-Cobbler into the breadboard. Now you can easily identify the various GPIO pins.

Next, solder the small header to the BME280 (**Figure 1**). This allows you to insert the sensor into the breadboard.

Shut down the Raspberry Pi and unplug the power. Wire the BME280 to the Pi (**Figures 2 and 3**).

**Red** 3V3 of Pi to VIN of BME280  
(3V DC positive, providing power to the BME280)

**Yellow** SDA of Pi to SDI of BME280  
(I<sup>2</sup>C data pin)

**Blue** SCL of Pi to SCK of BME280  
(I<sup>2</sup>C clock pin)

**Black** GND of Pi to GND of BME280  
(3V DC negative)

## FEELING SENSITIVE

Turn on the Raspberry Pi and open a terminal window. Type the commands to install and configure the I<sup>2</sup>C library.

```
sudo apt-get update
sudo apt-get install build-essential python-pip
python-dev python-smbus git
sudo apt-get install -y i2c-tools
```

Configure the I<sup>2</sup>C interface with raspi-config – select Interfacing Options, I<sup>2</sup>C, then answer Yes to enable ARM I<sup>2</sup>C, and exit.

```
sudo raspi-config
```

Reboot the Pi, then test the I<sup>2</sup>C connection to the BME280. The command `i2cdetect` shows all of the connected devices connected via I<sup>2</sup>C – the BME280 should appear on channel 77 (**Figure 4**).

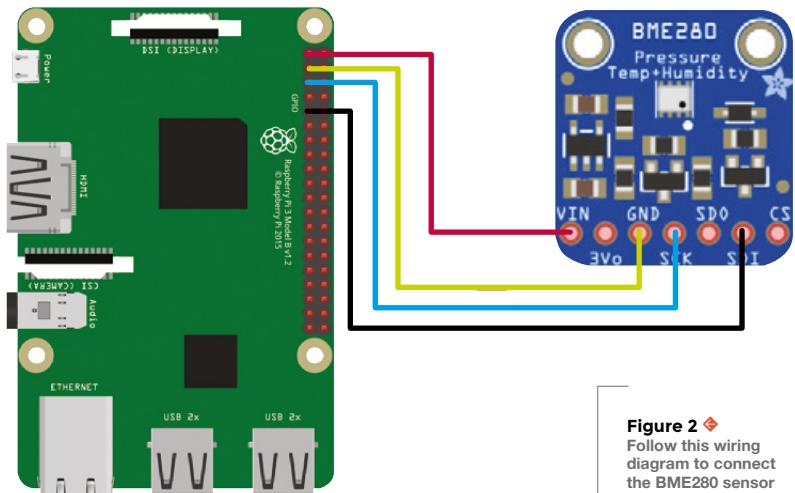
```
sudo i2cdetect -y 1
```

Install the Adafruit GPIO and BME280 libraries.

```
cd ~
git clone https://github.com/adafruit/Adafruit_
Python_GPIO.git
cd Adafruit_Python_GPIO
sudo python setup.py install
cd ~
```

**Figure 1**

Although not as reliable, you can leave the header off and press-fit wires into the holes to avoid soldering



**Figure 2**

Follow this wiring diagram to connect the BME280 sensor to the Raspberry Pi

```
git clone https://github.com/adafruit/Adafruit_-
Python_BME280.git
cd Adafruit_Python_BME280
sudo python setup.py install
```

Now run Adafruit's test program – you should see the temperature, pressure, and humidity displayed.

```
python Adafruit_BME280_Example.py
```

## GETTING ONLINE

The first step is to install and start ngrok. This is a program that allows your skill to connect to the Pi as if it were a server on the internet. Using the browser on your Pi, go to [ngrok.com](https://ngrok.com). Click download, and select Linux (ARM). Do not click on the big Linux button – this is the wrong version for the Pi. A zip file will now be in your downloads directory. Open a terminal window, then install and run ngrok.

```
cd ~
cp /home/pi/Downloads/ngrok-stable-linux-arm.zip ~
unzip ngrok-stable-linux-arm.zip
sudo ./ngrok http 5000
```

The last command starts ngrok, creating a tunnel to the local host (your Pi) on port 5000. This program creates a randomly generated web address that routes to your Pi. Carefully write down the web address starting with 'https:' – in **Figure 5**, it's <https://bf35fdfl.ngrok.io> (your address will be different). Leave ngrok running – you will use this later.

Next, install Flask-Ask. This is a library designed to easily build Amazon Alexa skills using Python.

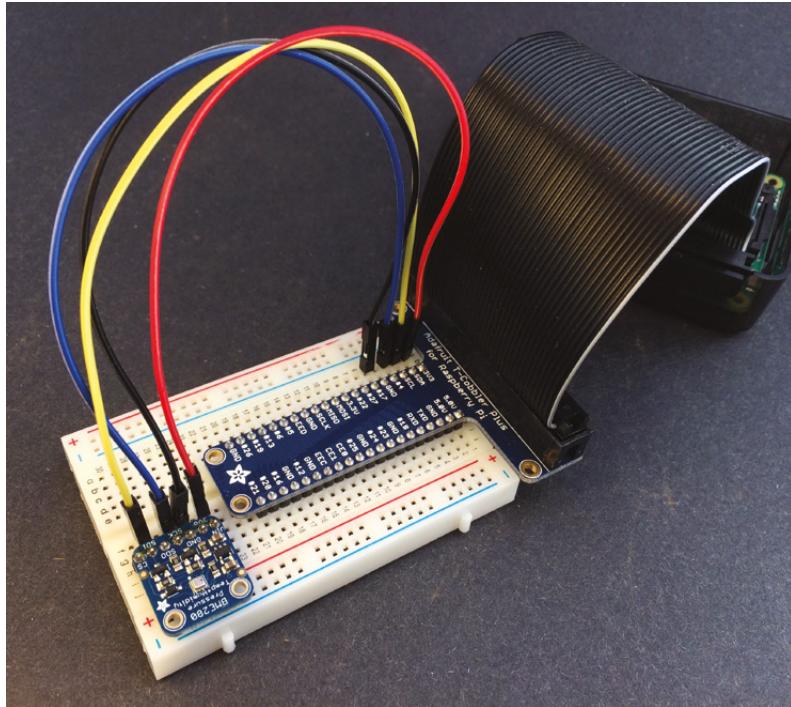
```
sudo pip install flask-ask
```

Now download the Python program, `pi_echo.py`. The git command also downloads text files for the Alexa skill we'll set up later. →

## YOU'LL NEED

- ◆ **Raspberry Pi**  
We used a Raspberry Pi 3 running Raspbian Stretch, but other Pis should work as well
- ◆ **Amazon Echo**  
The least expensive model is the Echo Dot, although any version will work
- ◆ **Adafruit BME280**  
I<sup>2</sup>C or SPI  
Temperature  
Humidity  
Pressure Sensor
- ◆ **Soldering iron and solder**
- ◆ **Breadboard, breadboarding wires, Adafruit T-Cobbler**  
Not required, but makes the wiring easier

## TUTORIAL



**Figure 3** Using a breadboard and T-Cobbler makes it simple to connect the GPIO pins to the sensor

```
cd ~  
git clone https://github.com/davidprimus/pi_echo.  
git  
cp /home/pi/pi_echo/pi_echo.py ~
```

### LEARNING NEW SKILLS

Before creating the skill, let's learn a little about how Amazon's Voice AI technology works. First, you wake up the Echo with its 'wake word', which is 'Alexa'. Then you ask her a question. For example, you might say "Alexa, what is the time?" and she will tell you the local time. You will create a custom skill called Pi Station. To invoke your skill, you will say "Alexa, ask Pi Station."

The Echo records the request and sends the audio file to Amazon's cloud service, called Amazon Voice Services (AVS). This converts the audio file to a command that AVS can understand and interpret.

### MOVING OUTDOORS

If you are interested in local weather, you'll want to report on the outdoor conditions. This is considerably more complicated, especially to achieve accurate readings.

Your first challenge is to move the sensor outside. Due to limitations of I<sup>2</sup>C, the sensor shouldn't be more than a few feet away from the Pi. One solution is to move both the Pi and the sensor outside, as long as you have reliable WiFi. The sensor needs to be far enough away from

the house to not be affected by the warmth of the wall, yet it needs to be in the shade. And don't forget power!

Also, the sensor and the Pi need to be protected from moisture. It is easy to put the Pi in a weather-proof case, but the sensor has to be both exposed and protected. To do this, consider building a Stevenson screen, a shelter for meteorological instruments. See more about my outdoor weather project here: [youtu.be/QVF1E0XY1Z0](https://youtu.be/QVF1E0XY1Z0).

```
pi@raspberrypi:~ $ i2cdetect -y 1  
 0: --  
 1: --  
 2: --  
 3: --  
 4: --  
 5: --  
 6: --  
 7: --  
 8: --  
 9: --  
 a: --  
 b: --  
 c: --  
 d: --  
 e: --  
 f: --
```

**Figure 4**

The BME280 uses channel 77. Multiple sensors can be simultaneously connected via I<sup>2</sup>C, as long as they are assigned their own channel

The skill Pi Station is set up at AVS with code that tells it what to do when it hears "Ask Pi Station." The code includes two significant sections: Intents and Utterances. Intents are actions associated with words. Utterances are words used to interact with the skill. In this project the intents and utterances are very simple, but they can be as complex as needed. Your skill simply launches, reports the sensor information, and stops. Therefore these intents and utterances, although required, are not used by your skill.

When the AVS skill runs, it sends a request to the Python program running on the Pi. The Python program interprets the request and formulates an appropriate response. This is sent back to AVS, converted to an audio file, sent to the Echo, and played. The whole process happens in a few seconds!

To create your skill, you need to have an Amazon Developer account. Sign up for one at [developer.amazon.com](https://developer.amazon.com). Once you have created your account, log in and set up the skill.

Click Developer Console at the top right. Click Alexa Skills Kit, then Create Skill. This will walk you through a series of screens, starting with 'Create a new skill'. Fill out the screens as follows, leaving everything else at default.

**Create A New Skill:** Enter **Pi Station** into the SkillName field. Click Next.

**Choose A Model To Add To Your Skill:** Click Select in the Custom box, then click Create Skill.

**Skill Builder Checklist:** Click Invocation Name, then enter **pi station** into the Skill Invocation Name field. Click Save Model, then click your browser's back arrow to return to the skill builder checklist.

Click Intents, Samples, and Slots, then enter **YesIntent** into 'Enter name for intent field', then click 'Create custom intent'. In the Sample Utterances field, enter **Yes**, then click the + sign. Add a No intent by clicking the Add button next to Intents, then enter **NoIntent** into 'Enter name for intent field', then click 'Create custom intent'. In the Sample Utterances

field, enter **No**, then click the + sign. Click Save Model, then click your browser's back arrow several times to return to the skill builder checklist.

Click Build Model. This will take several minutes. Click Endpoint, then click HTTPS. Be sure ngrok is still running in a terminal window; if not, run it again and use the new address. Enter the web address from ngrok into the Default Region field. In the 'Select SSL certificate type' field below, select 'My development endpoint is a subdomain of a domain that has a wildcard certificate from a certificate authority'. Click Save Endpoints.

At this point, you may need to enable the skill for your Echo device. Using the Alexa app, find the skill Pi Station, and click Enable. The skill will then be accessible from your Echo. Note that it may take a few minutes for the skill to be recognised.

## FIRST TRY

On the Raspberry Pi, open another terminal window (ngrok will still be running in the other terminal window). Start the Python program, **pi\_echo.py**, and leave it running (**Figure 6**).

```
python pi_echo.py
```

You're now ready to test everything out. Walk up to your Amazon Echo and say "Ask Pi Station." It should respond with the temperature, humidity, and barometric pressure. If you have the Amazon Alexa app on a mobile device, you will see the results there also.

## THE PYTHON CODE

The Python program acts as a server waiting for requests from AVS. The first line of code indicates this is a Python 2 program, which is required by Flask-Ask.

```
#!/usr/bin/python
```

The next few lines of code import the modules necessary for the BME280 sensor and Flask-Ask.

```
from Adafruit_BME280 import *
import logging
from flask import Flask, render_template
```

**Figure 6**

Awaiting a request from Amazon AVS. If your program has runtime errors, the skill will fail and the errors displayed here

```
pi@raspberrypi:~ $ python pi_echo.py
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger pin code: 116-614-871
```

```
ngrok by @inconshreveable (Ctrl+C to quit)

Session Status           online
Version                  2.2.8
Region                   United States (us)
Web Interface            http://127.0.0.1:4040
Forwarding               http://bf35fdf1.ngrok.io -> localhost:5000
Forwarding               https://bf35fdf1.ngrok.io -> localhost:5000

Connections             ttl     opn     rtt1    rt5     p50     p90
                        0       0      0.00   0.00   0.00   0.00
```

```
from flask_ask import Ask, statement, question,
session
```

The modules are initialised...

```
app = Flask(__name__)
ask = Ask(app, "/")
logging.getLogger("flask_ask").setLevel(logging.DEBUG)
sensor = BME280(t_mode=BME280_OSAMPLE_8, p_
mode=BME280_OSAMPLE_8, h_mode=BME280_OSAMPLE_8)
```

`@ask.launch` is the Flask-Ask command that is invoked when a message from AVS is received, indicating the skill has started. This runs the function `Intro` which gets the temperature, humidity, and pressure from the BME280 sensor. It converts the pressure to hectopascals, then adjusts the raw pressure to sea level pressure corresponding to my altitude (7810 feet). It then builds the string `speech_text` for the Echo to play. The return statement sends this string to AVS along with text for an 'Alexa card'. The card will be displayed in the Alexa app on mobile devices.

```
@ask.launch
def Intro():
    degreesC = round(sensor.read_temperature(), 1)
    pascals = sensor.read_pressure()
    hectopascals = pascals / 100
    sea_level_hectopascals =
    round(hectopascals+255.1508512, 1)
    humidity = round(sensor.read_humidity(), 1)
    speech_text = "The Raspberry Pi reports
" + str(degreesC) + \
                 " degrees Celsius. The humidity
is " + str(humidity) + \
                 " percent. The pressure is " +
str(sea_level_hectopascals) + " hectopascals."
    return statement(speech_text).simple_card("Raspberry Pi", speech_text)
```

The final two lines of code start the program and the debugger, waiting for a request from AVS.

```
if __name__ == '__main__':
    app.run(debug=True)
```

You can use this same technique to add a wide range of skills to Alexa. The only limit is your imagination, so set forth and create your own skills. ☐

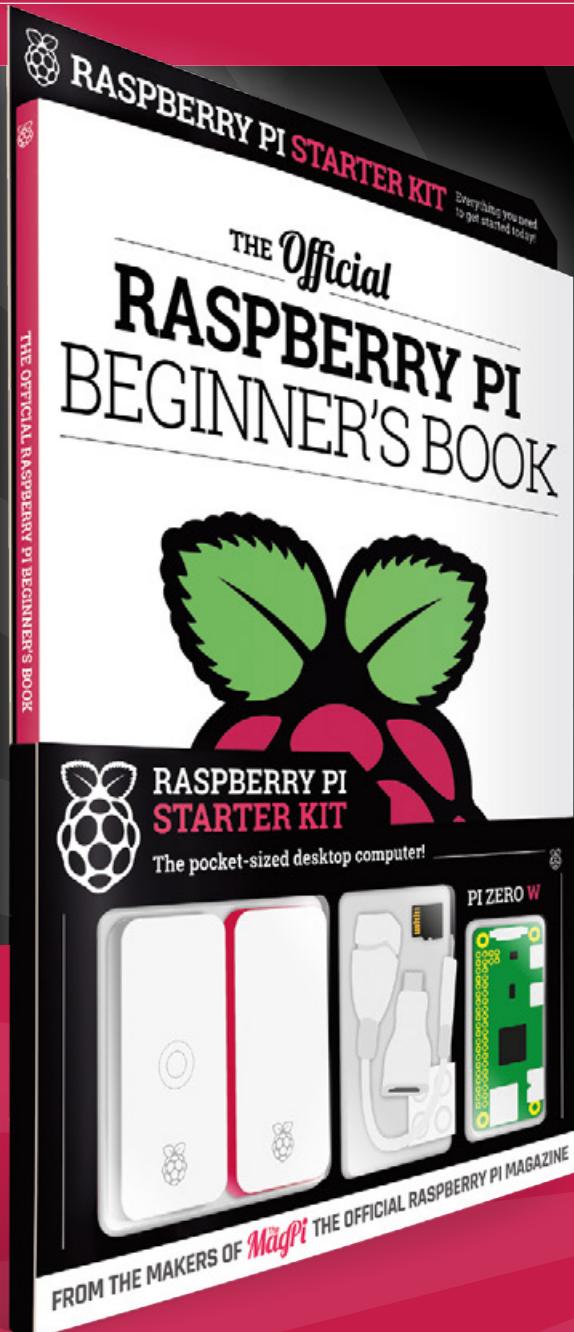
**Figure 5** ◇  
During Alexa skill setup, use the address shown at the location of the arrow when you run ngrok

## I<sup>2</sup>C WIRING

The I<sup>2</sup>C (Integrated Circuit) protocol is designed to attach one or more slave devices to a master device – in our case, the Pi is the master, the BME280 is the slave. I<sup>2</sup>C uses two wires for communication – a clock line and a data line. Each device uses its own channel. SPI (Serial Peripheral Interface) also supports multiple devices but uses four wires to communicate – clock, data in, data out, and slave select. Instead of using different channels, the slave select line lets you select the device.

THE *Official*

# RASPBERRY PI BEGINNER'S BOOK



**LEARN  
COMPUTING  
THE EASY WAY!**

**Includes**

- Pi Zero W computer
- Official case with three covers
- USB and HDMI adapters
- 8GB microSD card
- 116-page beginner's book

*Available  
now*



**Buy online: [hsmag.cc/store](http://hsmag.cc/store)**

# FIELD TEST

## HACK | MAKE | BUILD | CREATE

Hacker gear poked, prodded, taken apart, and investigated

PG  
**116**

DIRECT FROM SHENZHEN:

### 3D PEN

Embrace the future and move your doodles into the third dimension



PG  
**124**

### CAN I HACK IT?

We pull apart some light-up trainers to see what really makes them tick



PG  
**118**

### BEST OF BREED

We pick out the best cameras for hackers and makers



## REVIEWS

**126** Bare Conductive



**128** Arduino Cloud

**129** The Maker Revolution

DIRECT FROM  
SHENZHEN

# 3D pen

Take your drawings to the third dimension

By Ben Everard

 @ben\_everard

3

**D pens are, essentially, a printing head from a 3D printer put inside a convenient package for holding in your hand.**

Press a button and molten filament is squeezed out the end of the pen, which you can then doodle with. It

is, in a way, like a more precise hot-glue gun, but with plastic instead of rubbery glue.

There's a variety of similar pens for sale on most direct-from-China websites. We got one from Hua

Qiang Bei electronic store, on Ali Express for £14.91 including UK delivery, and three colours of PLA filament (red, white and grey). The pen takes 1.75mm PLA or ABS filament, which are the most common types of 3D printer filament available, so you can stock up on more colours from just about any filament shop around. The temperature range is usually a little lower than 3D printers (ours goes up to 210°C).

Anyone who has ever had to configure a 3D printer will understand that they need to be very accurate



## DIRECT FROM SHENZHEN

to get usable prints – this level of accuracy is almost impossible to achieve by hand. Instead of building up 3D objects layer-by-layer as printers do, these pens are usually used for creating 2D designs that can be joined together to make more complex objects. The easiest way of getting started is by tracing out a design from a template, (there are plenty of these available online that you can print out and use, for example: [hsmag.cc/WrgUsk](http://hsmag.cc/WrgUsk)). These templates are typically comprised of several parts that are traced separately and joined together. For example, the flower you see below is made from three parts.

It is possible to do some free-hand drawing with the 3D pen as well. For example, adding textures to a hard surface or even filling in a gap, and this sort of thing could be particularly useful to cosplayers adding little details, or for applying a quick fix.

You can, in some circumstances, draw lines going vertically off the surface, but there are some pretty major caveats. These vertical lines need to be well anchored to the ground and it's only possible to

do straight lines (well, saggy lines are more likely), and generally, these lines have to be between two objects, rather than just terminating in thin air.

Another use for the pen is finishing 3D prints. Since it uses the same filament as printers, you can use the pen to join parts together. If you use the filament that you print with, the colour will match perfectly, and the temperature should be hot enough to weld the parts together creating a strong, seamless joint.

### GETTING STARTED

The control options on our pen allow you to control the speed the filament feeds out and the temperature of the print head. The model we got requires the pen to be plugged in while in use, though there are other models that run on battery power. It might be a little easier to use the pen unhindered by wires if you're drawing large or complex models, but we didn't feel particularly restricted by being tethered to the socket.

After about five hours of use, our pen stopped working. A bit of investigation showed that the panel that covered the buttons and display had slipped out of place and was holding one of the buttons down. This panel had only been glued on using weak glue. We pried it off and the pen continued to work, but it highlights the quality – or lack thereof – of the materials used in the manufacture of such a cheap device. Perhaps, if we had two pens, we could have doodled a new cover for the display, but we left it naked and didn't run into any further issues.

The pen we bought was marketed as 'For Kids Birthday Christmas Gift'. However, while children would no doubt enjoy the process of building up models using such a pen, the high temperature of the plastic extruded from the nozzle makes it impossible to recommend for younger makers. PLA comes out around 170 °C, and we lightly burned ourselves more than once while testing this out, (mainly due to our ability to forget that molten plastic is hot).

We don't consider the 3D pen particularly dangerous (particularly when compared to other maker tools, such as power drills and soldering irons), but it's certainly possible to hurt yourself if you're not careful. Low temperature pens that either use a different plastic or use UV curing are available.

Overall, the 3D pen is fun to use and it can be used to create some interesting looking things. However, it's not particularly suited to making anything practical. They're often sold as 3D printer pens, and this oversells the usefulness of the device. □



**Left** It's easy to change colours with the pen, so multicoloured finishes are far easier than with 3D printers

**ONLY THE  
BEST**

# Hackable cameras

Capturing the world around you

By Marc de Vinck

 @devinck

We have come a long way since the invention of the first cinema camera over a hundred years ago. The technology behind capturing and displaying a moving image dates back to the late 1800s. Flash forward almost a hundred years to the early 1990s and the first webcam was put online, and just about ten years later, movies were being captured in pure digital ones and zeros.

Cameras are everywhere, and now more than ever it's easy to integrate them into your next project. You don't need to have a powerful single-board computer to capture video: even simple microcontrollers can take advantage of a camera's usefulness. We're now going to be looking at several products, ranging from the fairly simple to the latest cutting-edge camera systems, all of which you can easily add to whatever project you are working on that requires some variety of image-capturing technology.

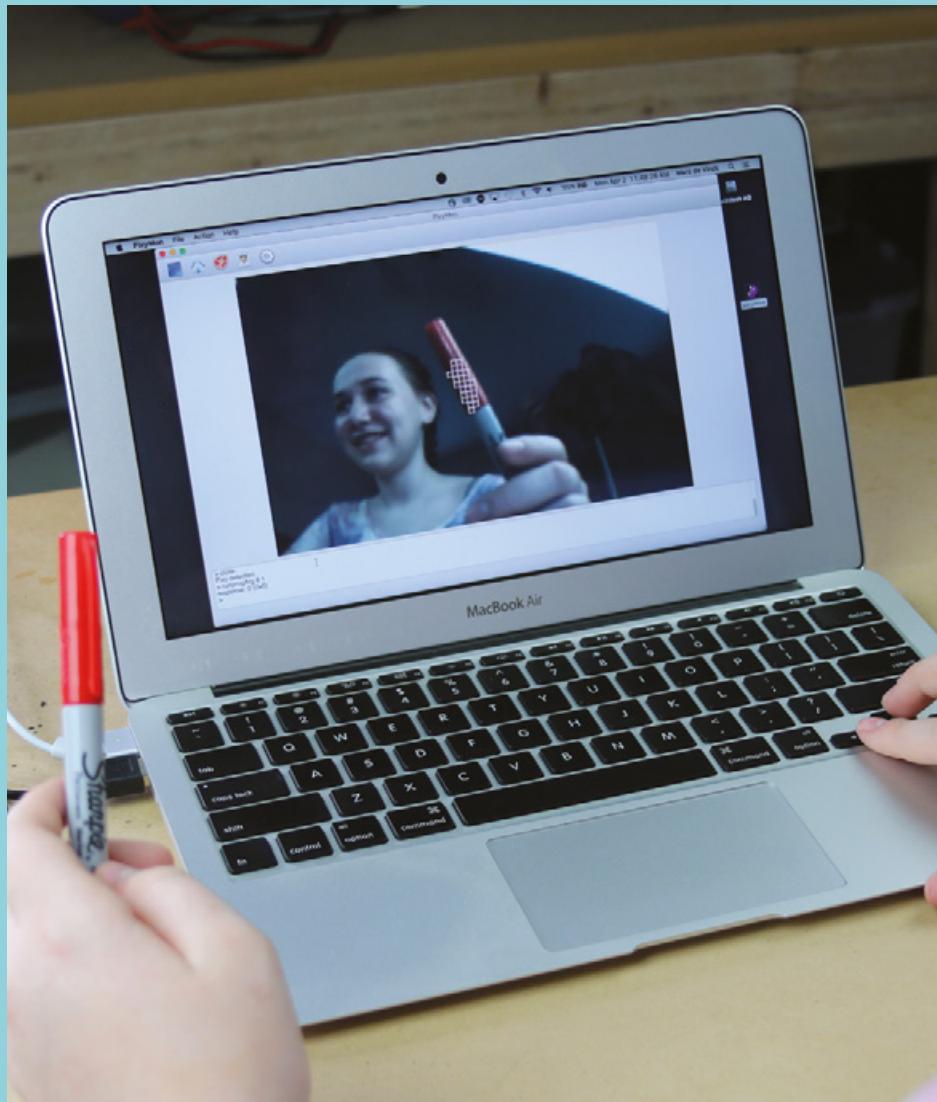
When considering what camera would work best for your project, you need to think about what else is needed. In general, a camera is just a camera, and it needs to be told what to do, and how to do it. You will also want to consider what the camera is going to do with the images it is capturing. Do you need to store video or pictures? If so, on what? Do you need to detect motion? Does it need to be high resolution, or is low resolution just fine? There are so many variables to consider, and so many different cameras available to purchase. We chose just a few of our favourites for these reviews, to cover the best available options for a wide range of projects.

**Right ↗**

The Pixy Camera has on-board object tracking abilities

**Credit**

Marc de Vinck





FIELD TEST

# Adafruit Spy Camera for Raspberry Pi vs Pimoroni Camera Module for Pi Zero

ADAFRUIT \$39.95 | [adafruit.com](http://adafruit.com)PIMORONI £14 | [shop.pimoroni.com](http://shop.pimoroni.com)

**Above** Don't let its small size fool you: it's a capable little camera

**Credit** Adafruit licensed under CC



**Above** A small Pi Zero deserves a small Pi camera

**Credit** Pimoroni LTD

**D**on't spy on my Pi might have been a better title for this article, and these cameras certainly fit into the spy-sized category. First up is the Spy Camera for Raspberry Pi from Adafruit. The actual camera module measures a mere 8.5mm x 11.3mm, allowing it to fit into small spaces. We're not suggesting you spy on anyone unknowingly, but there are times when an innocuous camera can be really helpful. Think about building your own doorbell camera, nanny-cam, or better yet, a camera to keep an eye on your pet when you aren't home. But then again, do you really need to hide a camera from your dog? In any case, this camera is perfect for embedding into your next project, especially when space is limited.

What we like about this particular camera is the extremely long ribbon cable connector which is included. The camera easily attaches to your Raspberry Pi via a 289mm-long cable, and uses the specialised CSI bus, allowing for high-speed data transfer. Just keep in mind, because of the specific cable included, the Spy Camera for Raspberry Pi will not connect to a Raspberry Pi Zero.

**“**The camera features a native resolution of 5 megapixels<sup>■</sup>

The camera features a native resolution of 5 megapixels, with a fixed focus lens on board. It can capture still images at a resolution of 2592 x 1944 pixels, and it can also capture video at 1080p30, 720p60 and 640 x 480p60/90. Those are some pretty good specs for a camera this small, which is simply plug and play. □

**W**hen your project needs a camera, and it needs to be small, it's hard to beat the Camera Module for Raspberry Pi Zero from Pimoroni. Don't let the generic name fool you: this little camera is great when you need to capture still images or video with your Raspberry Pi Zero or Zero W with a CSI camera port. This camera won't work with your standard Raspberry Pi, making it a bit limiting, but it's still a great spy-sized camera module, and easily coupled with the ever-so-small Pi Zero.

One thing to note about this particular camera from Pimoroni is that it's available in three different flavours. There is the standard version; another without an infrared filter, enabling you to explore IR photography and make your own night vision systems; and lastly a 120° wide-angle version, which is capable of capturing a wider field of view. And don't forget about the available accessories, making the camera even more versatile, like a variety of available lenses, including a telephoto lens for getting up and close to your subject.

All the different varieties of the cameras feature a 5MP sensor capable of capturing pictures at a pixel dimension of 2592 x 1944, which is the same as the version 1 official Pi camera. You can also grab video at a resolution of 1080p at 30fps, 720p at 60fps, and even 480p at a high rate of 90fps. Not bad for a camera that only measures about 11mm square, not including the ribbon connector. One last thing to consider prior to purchasing this camera: because of the way the connector is mounted to the Pi Zero, adding a camera may or may not interfere with a few of the ready-made enclosures. But then again, if you're building a custom electronics project, why not build your own case too? □

## VERDICT

Spy Camera for Raspberry Pi

Tiny, really tiny, and great!

**9/10**

Camera Module for Raspberry Pi Zero

A great camera, even if it's only compatible with the Pi Zero

**8/10**

# Pixy CMUCam5

**PIXY** \$69.95 | [charmedlabs.com](http://charmedlabs.com)

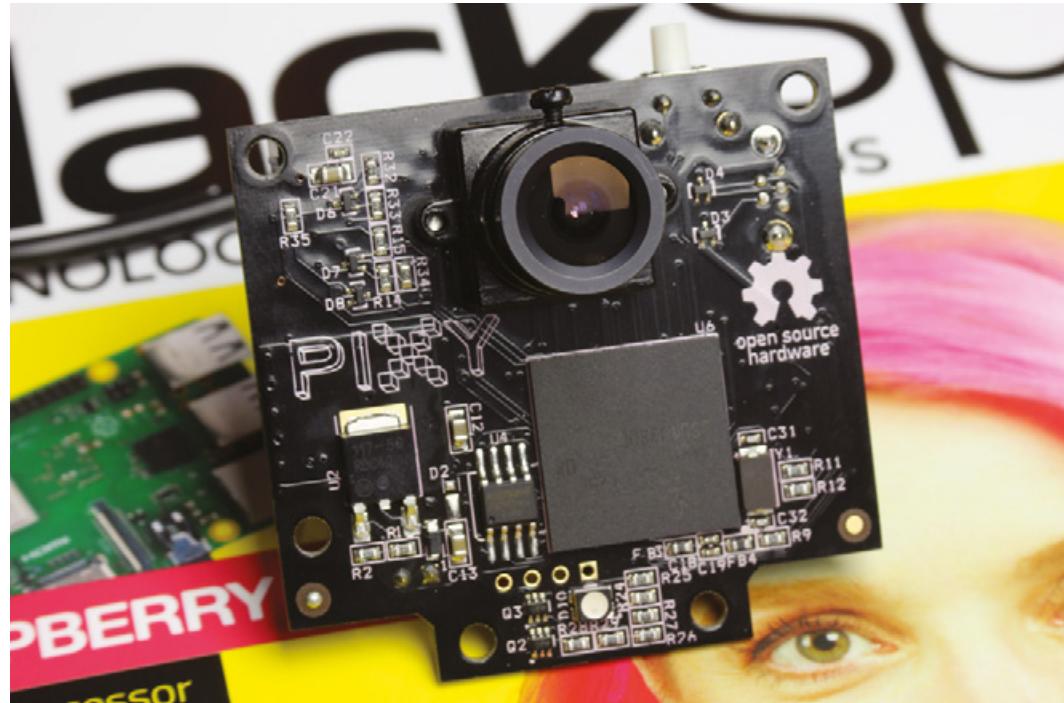
**W**hen your project needs object recognition, you can go in a few different directions. First, you can rely on your computer to do all the high-level computations.

But that means your project needs a big bulky computer, and that's not always convenient. You could also use the power of a Raspberry Pi and get similar results in a smaller, and more affordable, form factor. Lastly, there is the Pixy camera from Charmed Labs. This amazing little camera has on-board object recognition, and it can work independently of a microcontroller or computer.

The Pixy CMUCam5 allows the user to set the object colour to be tracked, simply by pressing a button. It can then analyse the incoming images and identify the object, or hundreds of objects, at a distance of up to ten feet. Yes, hundreds of

objects. And all of this at 50 frames per second. That enables it to update and process an image in just 20 milliseconds – not bad, especially considering it does this all without the need for any additional microcontrollers or computers. Keep in mind it is tracking colours, which can be really useful, but it cannot reliably track faces. At least, not yet.

There is so much information on their included wiki, you are inevitably going to find the answer to whatever question you may have about using the Pixy camera. They include everything from connecting it to your Arduino, Raspberry Pi, and BeagleBone, to using the specific Lego version of the camera and everything in-between. You can even plug two servos directly into the back of the board, allowing for smart pan and tilt functions while tracking objects. It's a great camera to experiment with, even if you don't have a particular project in mind. □



**Left** One of the smartest cameras available

**Credit**  
Marc de Vinck

**VERDICT**  
Incredibly powerful and fun to use

9/10

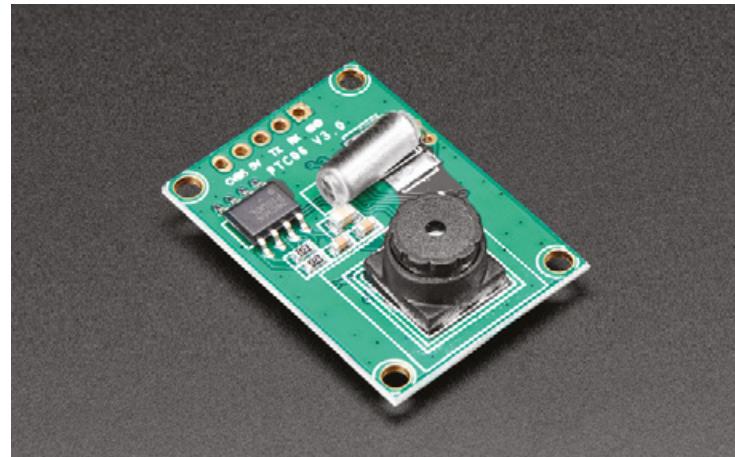
# Adafruit Miniature TTL Serial JPEG Camera with NTSC Video

**ADAFRUIT** ◇ \$35.95 | [adafruit.com](http://adafruit.com)

The Miniature TTL Serial JPEG Camera from Adafruit is a fairly common component used in many different products, like always-on video security cameras. It's a fully vetted technology that allows even a microcontroller to easily control the camera and the images and videos it outputs.

The camera has automatic exposure, gain, and white balance. This can be really handy, but it also limits its use in some situations. Its effective monitoring distance is between 10 and 15 metres, and has a few different image sizes, the highest being VGA at 640 × 480 pixels. It's also powered by 5V DC, which makes it a perfect addition to your Arduino project.

One of the really interesting things about this camera is its ability to be triggered by motion. Once you get the camera set up and running, try uploading the MotionDetect code. Once you power up your camera, it will immediately snap a picture any time motion is detected. Just remember that transferring the images to an SD card breakout board, which you will need to purchase separately, is slow and can take up to 30 seconds, since the data is transferred



over a serial connection from the camera and an SPI connection to an SD card breakout board.

Adafruit has a great tutorial explaining everything you ever wanted to know about TTL cameras using the Arduino IDE or running CircuitPython. It's simple enough to hook it up, install some libraries, and wire up an SD card breakout board for image storage. □

## VERDICT

A great camera  
for your Arduino  
vision needs

**8/10**

## YOU DON'T ALWAYS NEED A CAMERA TO SEE!

### WALABOT STARTER & WALABOT DEVELOPER

◇ \$149.95 - \$599.95 | [sparkfun.com](http://sparkfun.com)

Sometimes you need to see something that's not easily accessible, or only occurs in a low light situation. That's where the Walabot Developer and Walabot Starter come into play. These unique programmable sensor tools are capable of visualising the surroundings through the combination of radio frequency technology, sophisticated antenna arrays, and an A3 system-on-chip integrated circuit.

The Walabot Starter sensor board features three antennas capable of 2D visualisations. Keep in mind that you will have to build your own enclosure to protect the components. The more powerful 18-antenna Walabot Developer includes an enclosure, and is capable of advanced 3D visualisations. You'll also need a host device for the visualisation with either of these systems.



**Left, above** ◇  
A classic TTL camera  
breakout board for  
your Arduino

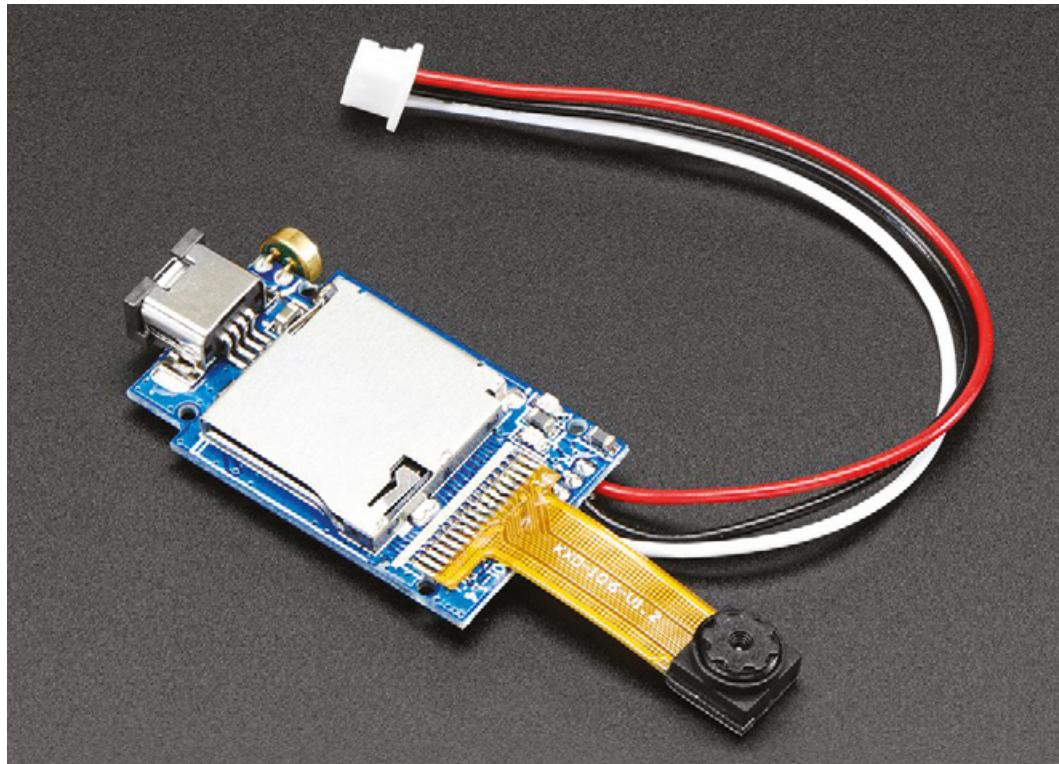
**Credit**  
Adafruit licensed  
under CC

**Left, below** ◇  
Walabot Starter  
uses three antennas  
instead of a camera  
to allow it to 'see'

**Credit**  
Sparkfun Electronics  
CC BY 2.0

# Mini Spy Camera with Trigger for Photo or Video

**ADAFRUIT** ◇ \$12.50 | [adafruit.com](http://adafruit.com)



**T**here are times when you just need a simple camera to take a few pictures, and that's when the **Mini Spy Camera with Trigger from Adafruit** comes in really handy.

You can easily hook this up to your Arduino and trigger it to take a picture every few seconds, minutes, or even hours to create a beautiful time-lapse video. Or how about having it hooked up to your doorbell to snap a quick pic every time it's pressed? And yes, you could use it to spy on something, like taking snapshots of your animals while you are away from home. Just make sure your cat knows to look busy!

The camera, which is really a complete camera system, includes a microSD card slot where it can store JPEG images at a resolution of 1280 × 720, which is pretty good considering its size and cost. You can also shoot videos with audio at a resolution of 640 × 480, which isn't a very high, but it's still useful for a lot of projects. Best of all, when you need to retrieve those images and videos, you can simply plug the camera into an available USB port on your computer with the included USB cable, and it will show up as a drive. Not only is this camera system simple to use and small, but it's extremely affordable, especially when you consider everything that is included. ☐

**Left** ◇  
A tiny camera capable of generating fun time-lapse movies

**Credit**  
Adafruit licensed under CC

**VERDICT**  
Small and very affordable, what's not to love?

**L**10 /10

# The Raspberry Pi Camera Module v2

RASPBERRY PI ◇ £24 | [raspberrypi.org/products](http://raspberrypi.org/products)

**T**he Raspberry Pi Camera Module v2 is a high-resolution 8-megapixel camera that features a Sony IMX219 image sensor that was custom-designed as an add-on board for Raspberry Pi.

It's capable of capturing pictures as high-res as 3280 × 2464 pixels, and it can also capture video in multiple resolutions from 1080p30, 720p60 and 640 × 480p90. That's pretty impressive for an inexpensive camera that simply plugs into your Raspberry Pi board.

What makes this camera module even more versatile is the ability to add different lenses and enclosures. Pimoroni sells a compatible three-lens pack that includes a fish-eye, wide angle, and

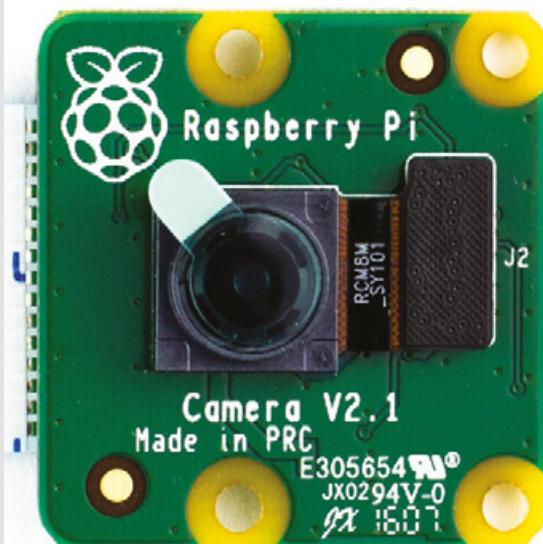
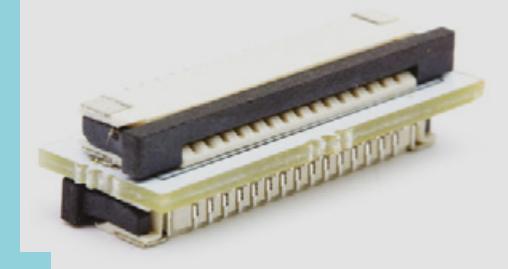
macro lens. That adds a lot more possibilities for not only practical projects, like adding a wide-angle lens for security systems, but also having a little fun with a fish-eye lens in a Raspberry Pi-powered photo booth.

You can pick up the Raspberry Pi Camera Module v2 from Pimoroni in two different flavours: the standard, as pictured, and the NoIR, which does not have an IR filter, making it perfect for taking pictures in low light, or exploring infrared photography. You can also pick up a very affordable camera mount that makes implementing this camera into your next project a snap. □

## A LITTLE EXTRA CABLE IS NEVER A BAD THING

CAMERA CABLE JOINER/EXTENDER FOR RASPBERRY PI ◇ \$2.92 | [shop.pimoroni.com](http://shop.pimoroni.com)

What do you do when the ribbon cable on your Raspberry Pi camera isn't long enough? Wouldn't it be nice if you could easily add another extension cable? Well that's where the Camera Cable Joiner from Pimoroni comes in handy. This little module has a PCB with two cable connectors mounted to it. It allows your Raspberry Pi camera cable to snap into one connector, and an extension ribbon cable can be snapped into the other, allowing you to extend the overall length of your cable. Best of all, it's really inexpensive, and includes a 30cm cable!



What makes this camera module even more versatile is the ability to add different lenses and enclosures. Pimoroni sells a compatible three-lens pack that includes a fish-eye, wide angle, and macro lens

**Left** ◇  
It's hard to beat the latest Pi Camera v2.1

**Credit**  
Pimoroni LTD

**Far Left** ◇  
A nifty little cable extender for your Raspberry Pi camera

**Credit**  
Pimoroni LTD

**VERDICT**  
One of the best cameras for your Raspberry Pi

**10**/10

Some illuminated shoes?

## CAN I HACK IT? ■

# Can I Hack It? Illuminated shoes?

Can glowing footwear be hacked into a project?



**Les Pounder**

@biglesp

Les Pounder is a maker and author who works with the Raspberry Pi Foundation to deliver Picademy. He also helps teachers/learners to become creative technologists. He blogs at [biglesp](http://biglesp.com)

**T**hink wearable technology and we instantly think of jewellery, sewable LEDs, and light up hoodies. But a few years ago we saw illuminated shoes enter the world, most notably from our very own Dr Lucy Rogers, who wore them to an evening with HRH The Duke of York at an event to celebrate the Raspberry Pi community. Lucy certainly made an entrance and lit up the room with style.

So how easy are they to hack, and can we use them in a project? Well there is only one way to find out. Let's take one apart!

### GENERAL CONSTRUCTION

Made of synthetic 'leather', the uppers are pliable and easy to work with using conventional hand tools and knives. The soles of the shoes are a

rubberised plastic which is used to diffuse the LEDs and provide an adequate level of grip. Moving inside the shoe, and the synthetic leather is protected from sweat via a semi-breathable fabric that has an almost 'rubbery plastic' feel. Between the outer and inner layer is a foam layer that offers comfort to the wearer and rigidity for the shape of the shoe. Still inside the shoe, and the sole is a two-part construction, with a replaceable sweat barrier on top of a non-replaceable membrane used to keep the electronics in place. Inside the sole are the electronics and the LEDs, as well as plenty of space! The soles are hollow with a series of inner support structures used to contain the electronics and prevent the weight of the wearer from crushing everything. The internal support structure is easy to work with using hand tools; rotary 'Dremel' tools would also make short work of the plastic used, but remember that you will need to add support to stop the wearer crushing the electronics – oh, and for comfort!

Talking of crushing, the electronics are 'potted' in a block of resin. This means that they are not easily accessible, if at all. The resin is tough and it surrounds the battery and electronics. The best way to get access is to use side cutters, not your best ones, and snip away at the plastic container to reveal the resin. The next bit requires a



#### Right □

Available in black, silver, gold, and white, these shoes were made for hacking!

#### YOU'LL NEED

❖ **PADGENE unisex light-up trainers (Amazon - see link below)**

#### COST

❖ £26.98

#### WHERE

❖ [hsmag.cc/POxSPt](http://hsmag.cc/POxSPt)

little ‘brute force and ignorance’ to get into the resin. A rotary tool will make the task much easier, but be very careful, as LiPo batteries do not like being punctured and can cause fires, so only try this if you know what you are doing!

## POWER

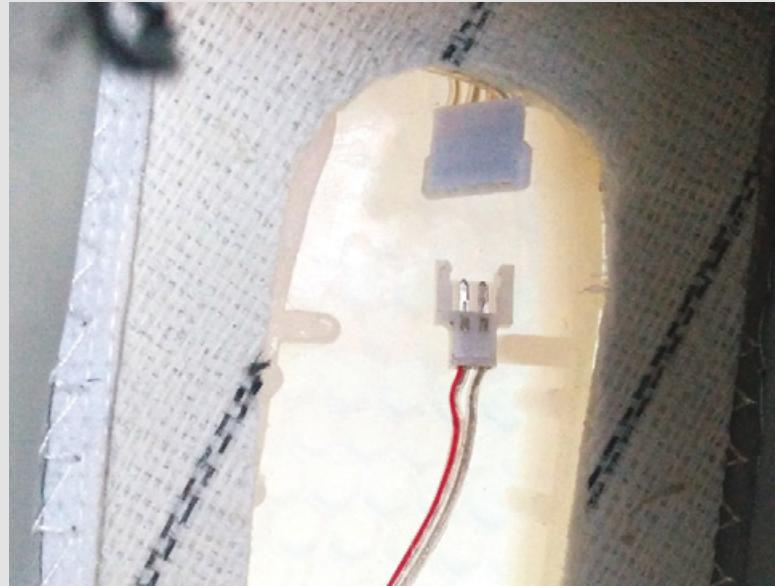
Each shoe has a non-replaceable LiPo battery, rated at 3.7V 450mAh, which is connected directly to the electronics that control the LEDs. Charging the battery is handled by a micro USB interface that is sewn into the layers inside the shoe. Both shoes can be charged at once using the included USB ‘A’ to micro USB connector.

## ELECTRONICS

The circuit board is identical for each shoe, and it features an unknown eight-pin chip that is used to control the LiPo battery charging and boost the 3.7V voltage from the LiPo to the 5V used by the LEDs. The LEDs themselves are not WS2811 / WS2812 ‘NeoPixels’-intelligent LEDs, instead they are 5V RGB which can display red, green, and blue, or a mix of each colour. The colour is controlled from the circuit board by use of a common 5V power, requiring the choice of colour by pulling any of the RGB pins to Ground. So, while this may be ‘primitive’ compared to NeoPixels, it does mean that we can control the LEDs using any single board computer or microcontroller, for example Adafruit’s Trinket M0, or Feather range of boards. But this would also mean sourcing a replacement LiPo battery and charge circuitry, which can easily be found in dollar store / pound shop LiPo batteries. There is plenty of space inside the sole of the shoe in which we can add our electronics; keep in mind that any electronics inserted into the shoes will need to be enclosed in a barrier to prevent moisture, and stop the weight of the wearer from crushing the kit.

## INTERNET OF SHOES

Internet-connected shoes, yes I did write that. Imagine if your shoes could tell you which way to go! Using a Pi Zero W, GPS, and an internet connection, our shoes could tell us which way to go to reach a destination. Green means we are on the right path, red and we have gone astray. Remarkably, these shoes could form the heart of that project. There are Python libraries for working with GPS and the Google Maps API, using these with a Flask web application running on the Pi Zero W, and accessed via our phone we can plot a destination, and have our shoes light the way!



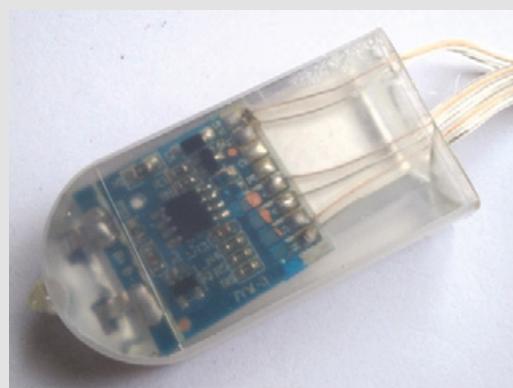
## CONCLUSION

These shoes were made for hacking, and that’s just what we’ll do. These shoes are your route into cheap wearable hacks. They do not need to be torn apart to get access to the electronics, so a sympathetic hack is possible. For those more daring, the liners inside the shoe can be torn away to reveal the internal support structure and with that we can place our own electronics. Two great hacks for these shoes would be to use a controller with Bluetooth support, and then use an application on our phone to control the LEDs in our shoes! Also using a sensor to detect a footstep, and then causing the lights to pulse, would make for an effective and useful hack for joggers.

Replacing the LEDs with NeoPixels is also possible, for those who are daring! But replacing the LEDs with NeoPixels would offer much more flexibility and control. Imagine a scrolling ‘marquee’ or rainbow effect as you walked.

No matter how you hack these shoes, you will make an entrance at any party! Be bold, stay safe, and keep on hacking! ☐

**Above** Inside the sole lies a cavity, just large enough for the electronics, be it the default, or your own. It also has connections to the LEDs and the micro-switch controls



**Left** Potted in resin, the electronics and battery are protected from impact and the crushing weight of a human being, and smelly feet!

# Bare Conductive Touch Board Starter Kit

Throw away your wires – paint your circuits instead

£92 | [bareconductive.com](http://bareconductive.com)

By Ben Everard

 @ben\_everard

**Below**  The smaller pins can be soldered on to open up the GPIO options on the touch board

**B**are Conductive makes conductive paint and the bits and pieces needed to make use of this paint. Its touch board is an Arduino-compatible microcontroller, with an added set of touch-sensitive pins. These pins can work with the conductive paint to turn almost anything into a touch sensor. Just place the touch board flat on a surface and paint directly from the touch pins to add touch sensitivity to almost any build. Of course, you can use the paint to wire up all sorts of circuits.

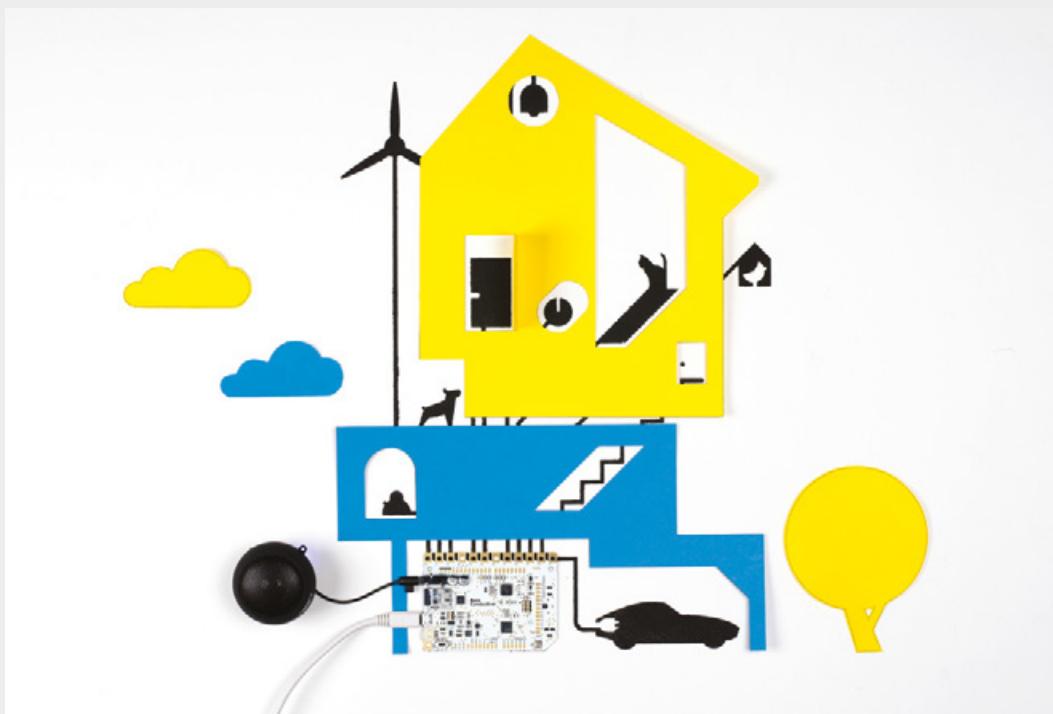
The technical specifications of the paint claim that the resistance of a 50 micron layer of paint is 55 ohms for a square. This might seem slightly odd to people working with things that have a different resistance for different lengths

(such as wires), but a square sheet of material will have the same resistance regardless of how large it is. Basically, this is because as you make something wider, you decrease this resistance, but as you make something longer, you increase it, and in a square, these exactly cancel out.

## DON'T BE SO SQUARE

Of course, you rarely need to work in squares, so we measured the resistance of traces from the paint tube. We found them between about 60 and 150 ohms per centimetre, depending on how thick a line of paint was used (as the paint is quite viscous, it's easy to create a thick bead). This is slightly higher than silver-based conductive paints, but these are generally more expensive. At this resistance, you can create a capacitive touch sensor or light an LED with a moderately long line, but if you're planning on doing something that requires more current (such as running a motor), you may run into problems.





**Left ↗**  
The blue and yellow cut-outs (provided in the kit) give you a backdrop to your touch-sensitive projects

In the Touch Board Starter Kit, you get a touch board, a tube of paint (10ml), a jar of paint (50ml), a speaker, some crocodile clips and some stencils, cut-outs, and other bits and bobs to join it all together.

### TOUCHY FEELY

The touch board has a micro SD card for storing sounds and a 3.5mm audio jack for playing them, (such as through the included speaker). When you first power up the board, it has a brief audio guide loaded that you can page through by touching the input pins in order. It's easy to change the sounds to make different noises when things are touched. The touch inputs are in large pads at the side of the board, so you can either use crocodile clips, or lay the flat-bottomed board on paper and use the conductive paint to create traces directly from the board.

Slightly unusually for an electronics kit, there aren't any components – no LEDs, buttons, motors, or anything else. It's not actually that straightforward to wire bits like this up to the touch board. Although it does have some GPIOs exposed, they're not on the main connectors and you'd have to solder onto them to be able to access them. This wouldn't be a particular problem, except that the board needs to have nothing protruding from the base if you want to be able to connect directly to the paint, (though you can still use crocodile clips if there are headers soldered on). Bare Conductive no doubt has its reasons for this layout, but we can't help but feel that the kit could have a lot more uses if just a few GPIOs were exposed in the same way as the touch pins.

Part of the problem may be that the resistance of the paint traces means that you need to be a little careful with voltages to get things to work properly, because the resistance of the paint traces is so much higher than metal wires. We found that we needed at least five or six volts to make LED circuits reliable. This could lead to confusion among beginners when circuits didn't work.

### GOING IT ALONE

Of course, you don't have to use the touch board; you can just use the conductive paint (this is sold separately at £6 for a 10ml tube, or £18 for a 50ml jar). If you've got a steady hand, we found this worked well with 1206 SMD surface-mount LEDs, as they could be placed (using tweezers) in a gap in a trace, and the paint had enough adhesion to hold them in place when it dried. The paint also works well with through-hole components, and Bare Conductive also sells a Glowing House kit for £22, which includes a fold-out house, and the bits you need to light it up using conductive ink traces.

The Bare Conductive Touch Board Starter Kit does one thing really well – playing sounds when a painted pattern is touched. There is the capability to go beyond this and try more things out, but it's a bit of a step up between the basic usage and the more advanced options.

If you're more interested in playing around with paper-based circuits, then Bare Conductive paint can work well (with a caveat about the resistance), and you can get started with this without the touch board. □

### VERDICT

A fun and easy way of adding interactive elements to builds, but we'd like easier access to the GPIOs

L 8/10

# Arduino Cloud

Networked programming for little boards

free | [cloud.arduino.cc](http://cloud.arduino.cc)

By Ben Everard

 ben\_everard

**P**erhaps the best thing about the Arduino isn't the hardware, it's the ecosystem of software and compatible devices that gives users a familiar interface to different boards, and enables hardware developers to easily develop software for new boards. This is all centred around the Arduino IDE, which is a beginner-friendly development environment which helps users get started with any Arduino-compatible hardware. However, Arduino started before makers were commonly connecting their hardware to the internet, and creating web-enabled devices isn't particularly straightforward on the traditional IDE.

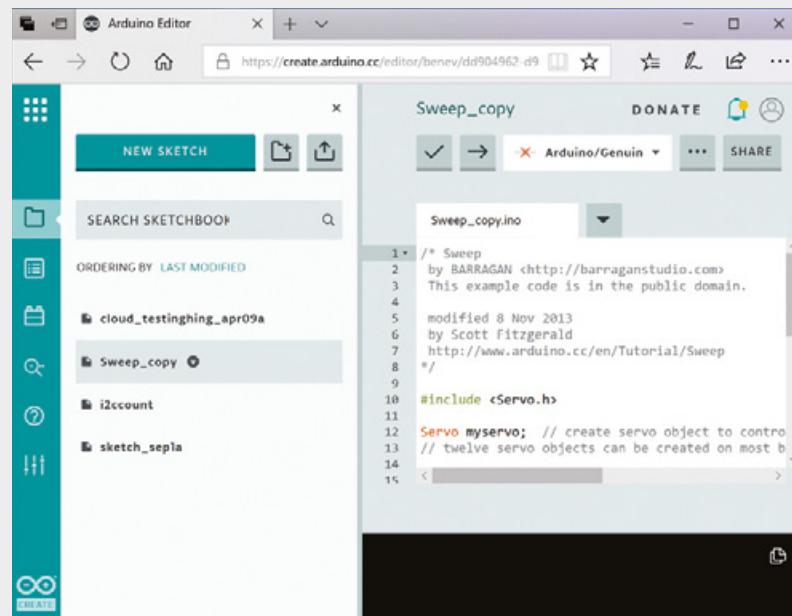
This is where Arduino Cloud comes in – it's part web IDE and part IoT back end. Point your web browser to [cloud.arduino.cc](http://cloud.arduino.cc), install a plug-in, and you can use your web browser as an Arduino IDE. This gives you instant access to a wide variety of boards and libraries (and you can upload more if you need). This is a nice addition to the Arduino suite of tools, but not really revolutionary.

The second thing that Arduino Cloud does is give you a back end to allow you to transfer data between your devices. This is, essentially, an MQTT broker housed on the Arduino servers. Again, this isn't particularly new technology, but it's something that's been a little fiddly for beginners to set up. Using Arduino Cloud, it's just there with no setup needed.

As one final bonus, the new Arduino Cloud also lets you write code for Linux devices such as the Raspberry Pi, as well as microcontrollers. While it's hardly revolutionary to let you program Linux devices, it is a bonus for anyone who programs in Arduino's Processing language as it means they don't need to learn a new language to get started with this class of hardware.

The big thing about Arduino Cloud is that it brings things together. It was perfectly possible to do all of

**Right ♦**  
Arduino Cloud now sits alongside the device manager and hub in an online environment that aims to provide all your Arduino needs



this before – and no doubt many readers were – but putting everything into one place makes it much easier to get started. Using a single language and a single environment, you can create a fully networked system of smart devices. In many ways, this is an expansion on what made the original Arduino IDE so popular. It doesn't overwhelm the user with options, while still providing plenty of power for common uses – of course this means that it's not going to be perfect for advanced users, but that's not the point of this environment.

If you're happy with your coding setup, and networking devices, then there's probably nothing in Arduino Cloud to make you want to switch. However, if you're interested in networking sensors, but have been put off by the back-end setup, then Arduino Cloud may be exactly what you're looking for. □

**Above ♦**  
The web IDE will be familiar to anyone who has used the offline version

**VERDICT**  
Ideal for beginners, it puts everything in one place for ease of use.

**8/10**

# The Maker Revolution

Mark R. Hatch ◇ £24.99 | [wiley.com](http://wiley.com)

By Richard Smedley

 RichardSmedley

**B**OOM!" says Hatch, "the greatest explosion of innovation and creativity in all human history is upon us." And it is, as access to tools to make almost anything that can be imagined spreads at prices unimaginably low a few short years ago.

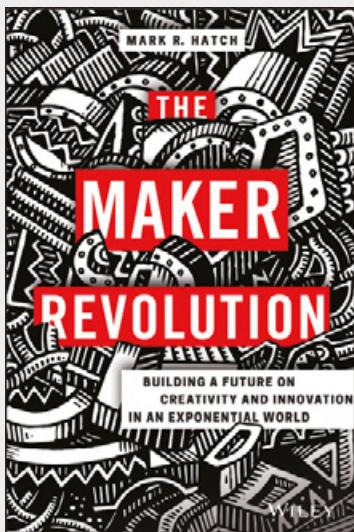
Other writers speak of the coming of AI, and new manufacturing techniques transforming manufacturing in terms of a seismic shift in society,

as labour and employment patterns we have lived with since the early days of the industrial revolution disappear. Hatch – who has been a leading figure in the maker movement since setting up TechShop in California a decade ago – looks beyond, to the creativity unleashed with widespread and marginal-cost access to the tools which can make visions into products, or artworks.

Hatch is a lively, informal writer, his discourse littered with explosions of 'BOOM!' in the same way he peppers his talks. It makes for a passionate, driven polemic in favour of the potential for makerspaces to transform lives. *The Maker Revolution* first outlines the trends that are driving the changes: the exponential technology changes that leave us dizzy as we try to keep up with them; the 90%+ fall in the costs of tools in recent years – in laser cutters, CNC machines, 3D printers; and changes to capital, manufacturing, and marketing. All with examples that really highlight that we're living in a new world, whether we've noticed it or not.

In turn, the growth of makerspaces – particularly in the United States, and Hatch's work is very US-centric – has begun to make major changes in people's lives. From military veterans to grandmothers and tattooed bikers, Hatch recalls working alongside a wide variety of people on their projects. Through education, enabling innovation, and very real – and startling (\$12 billion in estimated shareholder value from TechShop alone) – economic improvements.

The strong and detailed call-to-arms of the last section outlines just what US readers should be doing to spread this revolution. You won't get this far in the book without it kindling an enthusiasm to do something to help the changes along, wherever you live, and there are plenty of ideas you should be able to enact in your own community. BOOM! □



## VERDICT

A breathless tour of the staggering changes happening everywhere as makerspaces bring access to tools and unleash potential.

**L** 8/10

Issue #7

issue  
#7

ON SALE  
24 MAY

# FEATURING INTERNET OF THINGS

---

## ALSO

- THE OPEN SPACE AGENCY
  - WIRING UP THE Z80
  - LASER CUTTING YOUR OWN GEARS
  - FROZEN REVERSE SPHERIFICATION
  - AND MUCH MUCH MORE
- 

DON'T MISS OUT

[hsmag.cc/subscribe](http://hsmag.cc/subscribe)

# HackSpace

TECHNOLOGY IN YOUR HANDS



**“TO INVENT YOU NEED  
A GOOD IMAGINATION  
AND A PILE OF JUNK”**

**Thomas Edison**

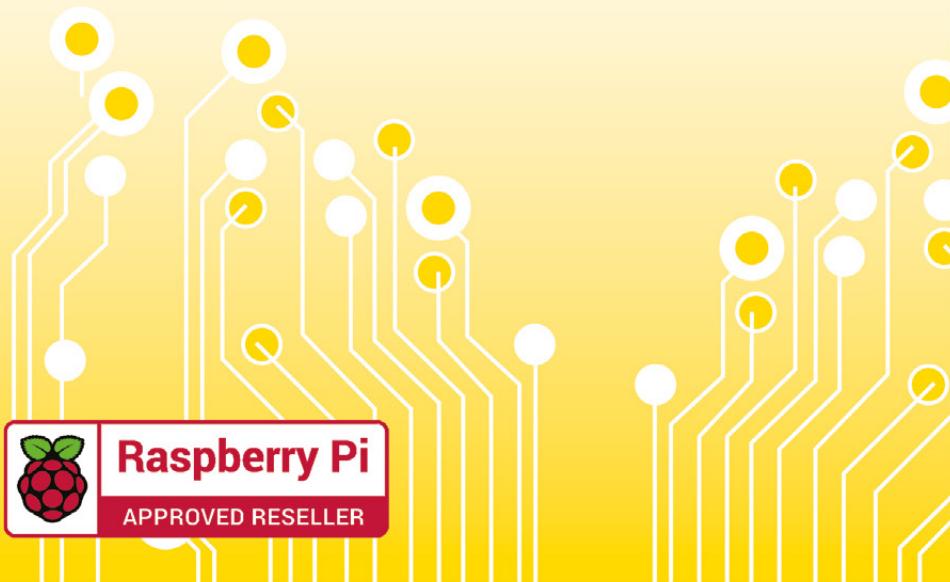
— hsmag.cc —

## CanaKit Raspberry Pi 3 Complete Starter Kit



### Kit Includes:

- ✓ *Raspberry Pi 3 Model B  
Quad-Core 1.2 GHz 1 GB RAM*
- ✓ *On-board WiFi and Bluetooth*
- ✓ *32 GB MicroSD Card (Class 10)*
- ✓ *CanaKit 2.5A Power Supply*
- ✓ *High Quality Case*
- ✓ *HDMI Cable with CEC support*
- ✓ *MicroUSB Reader*
- ✓ *Set of Heat Sinks*
- ✓ *GPIO Quick-Reference Card*
- ✓ *CanaKit Quick-Start Guide*



Available for worldwide shipping at:

**WWW.CANAKIT.COM**

**\$74<sup>.99</sup>** **£59<sup>.99</sup>** **€64<sup>.99</sup>**  
US DOLLARS EXCLUDING VAT EXCLUDING VAT

Raspberry Pi is a registered trademark of the Raspberry Pi Foundation.  
CanaKit is a registered trademark of Cana Kit Corporation.