

MAKE | BUILD | HACK | CREATE

HackSpace

TECHNOLOGY IN YOUR HANDS

hsmag.cc

November 2018

Issue #12

BUILD YOUR FIRST

ROCKET

Reach for the stars



› Open Science Hardware

Making experiments accessible

› Lots of LEDs

Control your blinky masterpiece

› Amie DD

Body hacking and
demonic technology

Nov.2018
Issue #12 £6



9 772515 514006 128



Get launch pad ready

Design your own craft

Thrust vector control

DIY space travel



JOSEF PRUSA

Bringing resin 3D
printers to everybody

DIODES ARDUINO LAMPS MULTIMETERS

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



PRE-LOADED
WITH OPERATING SYSTEM

USB MICROSD
CARD READER



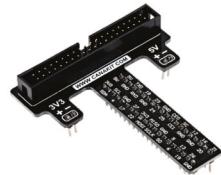
PREMIUM
HDMI CABLE



QUICK-START GUIDE



GPIO TO BREADBOARD
INTERFACE BOARD



RIBBON CABLE



FULL-SIZE
BREADBOARD



JUMPERS



MALE TO MALE &
MALE TO FEMALE

LEDs

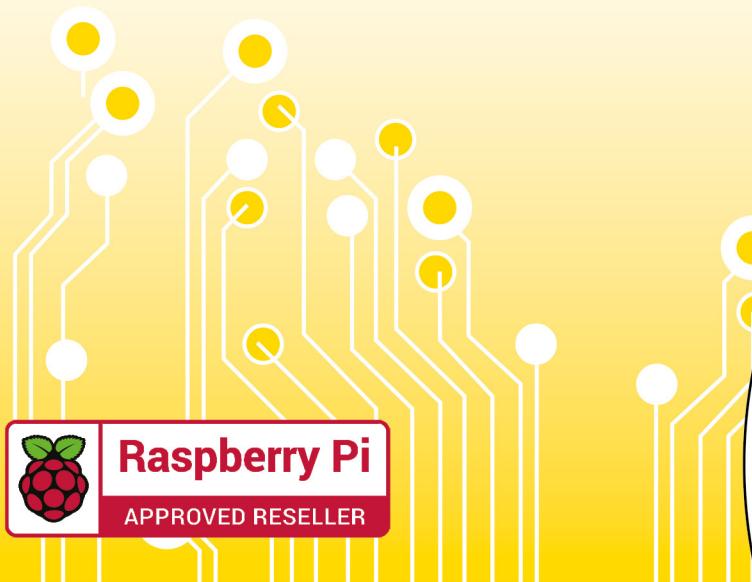


RESISTORS &
PUSH-BUTTONS



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

This month I had the privilege of joining the Midland Rocketry Club for their annual Midland Sky event, as both a spectator and a participant. You can see the full article, with some information on how to get started with model rocketry, on page 32. What really stuck with me though, was how welcoming this group of makers was. They were passionate about rockets and wanted to share this with people,

If you're looking for an outlet for your
3D printing, laser cutting, and electronics skills,
this could be the hobby for you

regardless of skill or experience.
While I've been to plenty of meetups, festivals, and

spaces over the years, and while they've almost all been welcoming, there's few I can remember where people of all skill levels came together in the same way.

If you're looking for an outlet for your 3D printing, laser cutting, and electronics skills, this could be the hobby for you. You get to lovingly craft an object to the best of your abilities, then put an engine in it, and launch it upwards at very high speeds (it's probably the only maker discipline where you can exceed the speed of sound), and hope that it comes down in one piece. It's easy to start, and there's probably a club near you that'd be happy to help you launch. Begin countdown ...

BEN EVERARD

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

Got a comment,
question or thought
about HackSpace
magazine?

get in touch at
hsmag.cc/hello

GET IN TOUCH

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

ONLINE

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

Available on the
App Store

GET ON
Google Play



PAGE 44
SUBSCRIBE TODAY

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

David Higgs, Nicola King

DESIGN

Critical Media

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

Head of Design

Lee Allen

Designers

Mike Kay, Harriet Knight

Photography

Brian O'Halloran,
Fiacre Muller

CONTRIBUTORS

Lucy Rogers, Andrew Huang, Jo Hincliffe, Cameron Norris, Mayank Sharma, Graham Morrison, Dave Astels, Simon Stevenson, Alex Eames, Brian Lough, Andy Clark, The Kuroiologist, Paul Freeman-Powell, Zack Akil, Les Pounder, Marc de Vinck, 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

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

122



06 SPARK

- 06 Top Projects**
You too can build something amazing!
- 16 Objet 3d'art**
3D printing for pros
- 18 Meet the Maker: Josef Průša**
The king of open hardware printing speaks
- 22 Columns**
What qualifications do you need to be a maker?
- 24 Letters**
Beer and Zeppelin, two of the greatest things ever
- 25 Kickstarting**
Resurrecting good old Ethernet
- 26 Cool Build As We Are**
The world's biggest and best selfie machine

31 LENS

- 32 Build your first rocket**
The Space Race continues
- 46 Global Open Science Hardware**
How makers are bringing science out of the classroom
- 52 Interview AmieDD**
We meet a real life cyborg. Yes, really!
- 60 Improviser's Toolbox** Light bulbs
Make lovely things with glass spheres of magic

Cover Feature

Build Your First
ROCKET

Design, build and
fly your first rocket.
Space Force!

32

88



Tutorial

Alexa wheel of fortune



82

Alexa! Spin the polygon
of destiny!



26



46

Interview

AmieDD



52

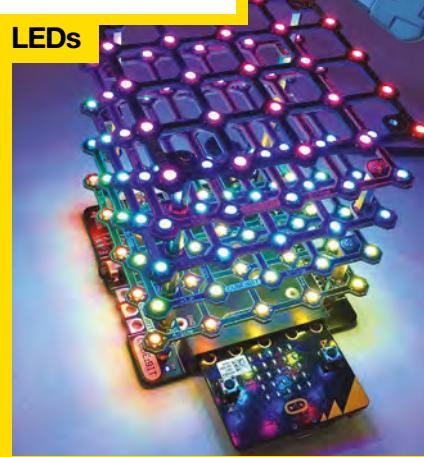
What links this person with the Xbox One? Demons, of course...



78

Best of Breed

LEDs



116

Make Piccadilly Circus/Times Square in your living room

FORGE**66 SoM Arduino**

Make noise accurately with interrupts

72 SoM Electronics 101: Diodes

The gate-keepers of electric current

78 Tutorial Drills

Everything you need to know about making holes

82 Tutorial Alexa wheel of fortune

Use Amazon's voice assistant to tell the future

86 Tutorial Rubber stamps

Create custom-designed stamps

88 Tutorial Way-home meter

Let your loved ones know where you are

94 Tutorial Skull bowl

Is it an ancient curse or capacitive touch?

98 Tutorial Copper piping

Steampunk lighting – the HackSpace way

102 Tutorial Number-plate reader

Teach your garage to recognise your car

108 Tutorial Machine learning

Create an AI botanist out of a Raspberry Pi

FIELD TEST**114 Direct from Shenzhen Multimeters**

Cheap and cheerful electronics essentials

116 Best of Breed

Easy ways to add shiny LEDs. More blinkenlights!

122 Can I Hack It?

What can we make with an IoT light bulb?

124 Review Anet Prusa A8

Want to learn every facet of 3D printing? Get one of these

126 Review SparkFun Deluxe Tool Kit

An electronics workshop in a box

128 Review Simulant Retro WiFi SI

Access the modern internet on ancient hardware

129 Book Review Collins Woodworker's Manual

The bible of making things out of trees

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.

Earth Clock

By Simon Robert

 hsmag.cc/eqtcnn

wanted to build something linking electronics, mechanics, and astronomy. I first thought about making an orrery, but a lot of people had already made a huge amount of them. I wanted to create something that has never been done before, something cool and new.

I was wondering what I could do, searching for ideas on Instructables, when I saw a contest named 'clocks', I started to think about a clock with an Earth globe on top of it, spinning at the same speed as the Earth in order to see the face of our planet exposed to the sun.

The realisation was trickier than I thought because there were a lot of parameters to include, (the inclination of 23° of the Earth's axis, the difference between a stellar day and a solar day, the modelisation of the sun, finding a way to include the rotation of the Earth around the sun) – turns out astrophysics is hard! □

Right ↗

The sun is represented by a ring, rather than a single LED, to better simulate the angle that the sun's rays reach Earth



Floating fibres

By Johana Hartwig

hsmag.cc/DAbqQY

coined the phrase ‘domesticating data’ to describe the process of putting data into more lay, or accessible modes. The ‘Floating Fibres’ project uses environmental data and the water levels of the River Taff, to inform the weave/pattern of textiles.

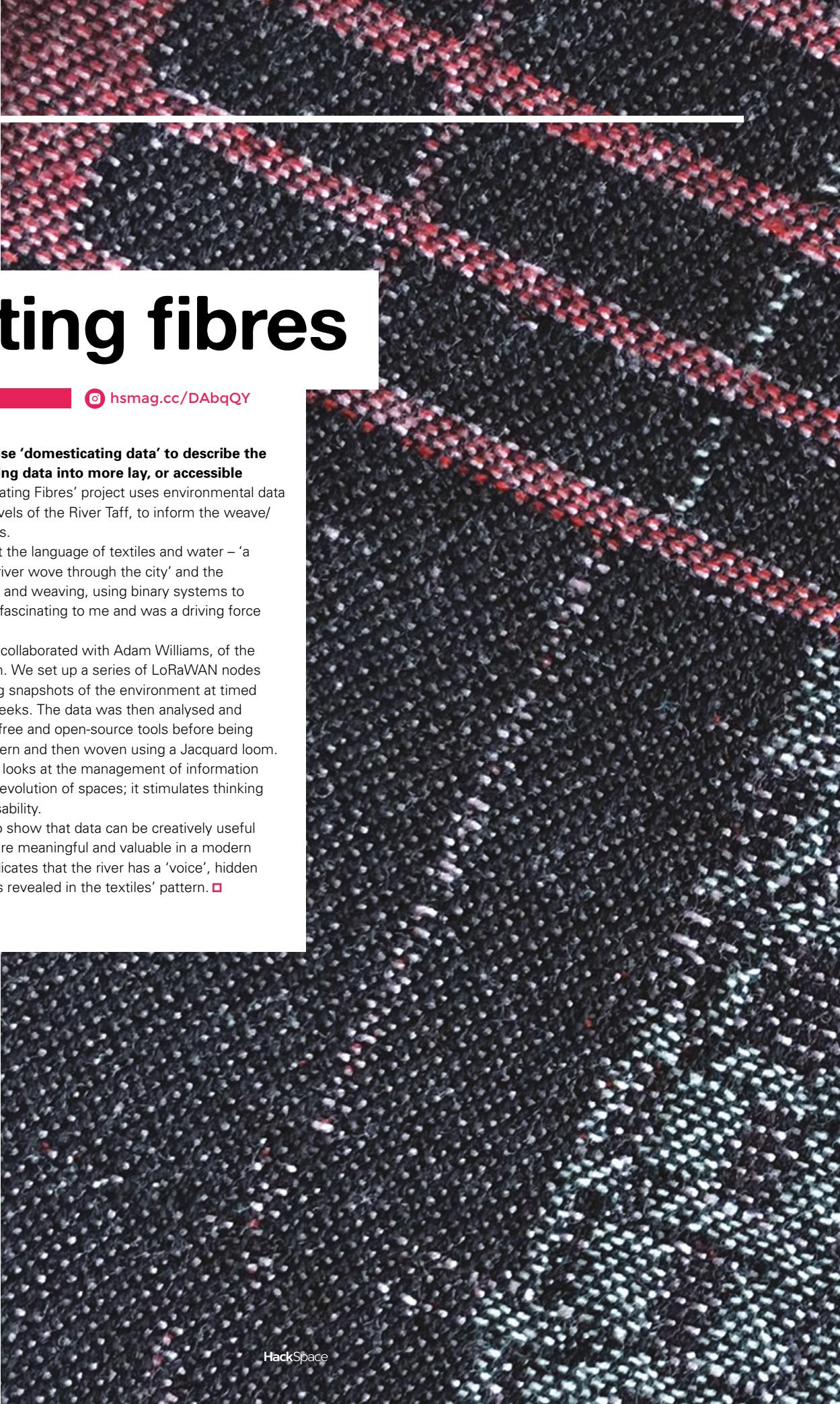
Thinking about the language of textiles and water – ‘a blanket of water’ / ‘the river wove through the city’ and the shared language of data and weaving, using binary systems to communicate, is totally fascinating to me and was a driving force behind the project.

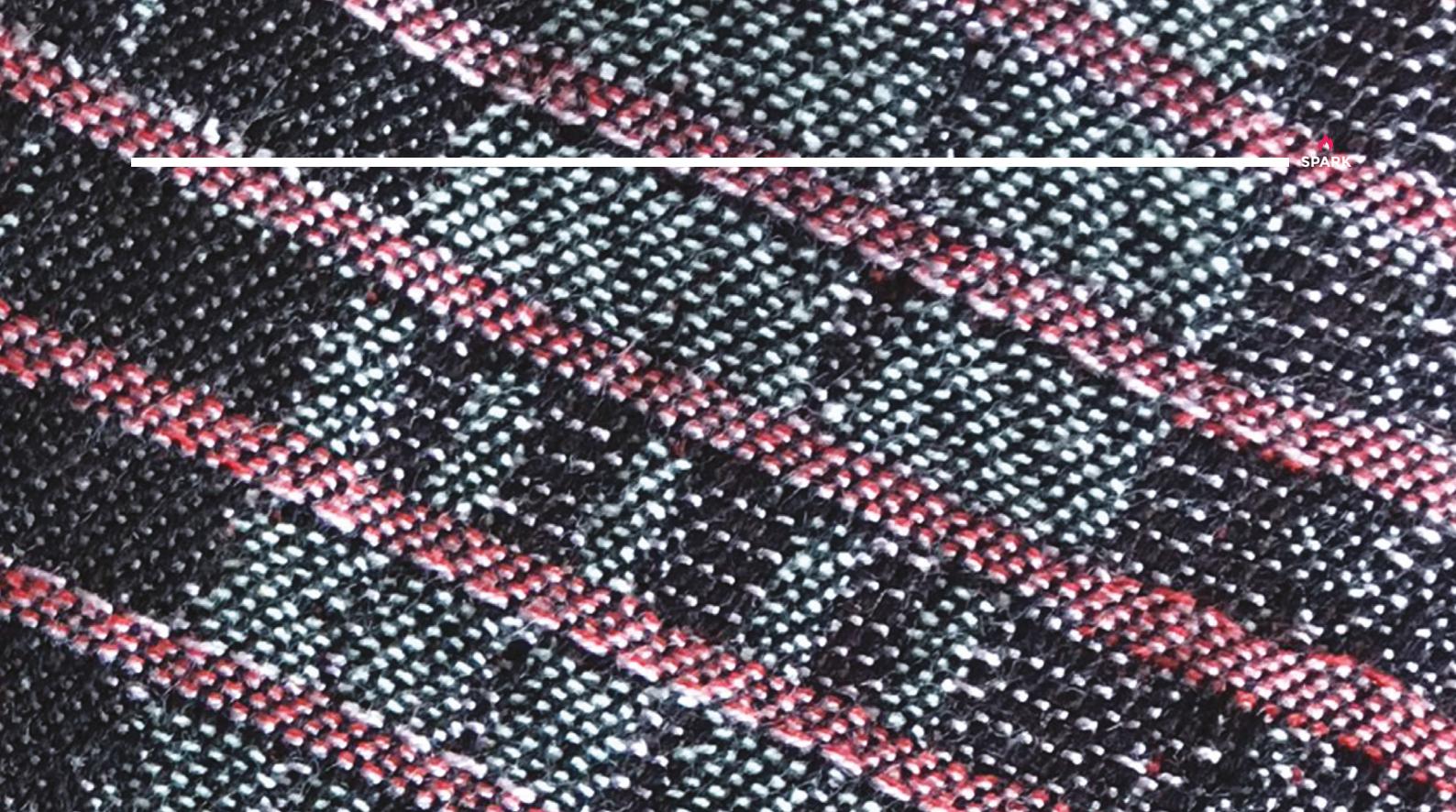
To harvest the data, I collaborated with Adam Williams, of the Raspberry Pi Foundation. We set up a series of LoRaWAN nodes along the river, capturing snapshots of the environment at timed intervals over several weeks. The data was then analysed and cross-referenced using free and open-source tools before being interpreted into the pattern and then woven using a Jacquard loom.

My wider art practice looks at the management of information and the construction or evolution of spaces; it stimulates thinking around value and disposability.

This project set out to show that data can be creatively useful and that our city rivers are meaningful and valuable in a modern context. The blanket indicates that the river has a ‘voice’, hidden communicators/rhythms revealed in the textiles’ pattern. □

Right We appreciate the throwback to early computing represented by the use of a Jacquard loom





Laptop cooler

By Theo Tutuiaru

 hsmag.cc/XihemM

My laptop was constantly overheating after playing video games for hours at a time, no surprise there, but I wanted to fix the problem anyway without having to fix my addiction. I got two computer fans and began testing which direction of airflow was better. My computer sucked air in from the bottom, so I made sure the fans also blew the air into it. I began building the base around the two fans. I cut a base out using a jigsaw and started focusing on the electronics.

Since computer fans use three-point connectors and 15 volt charges, all I had to do was find a computer charger with the same specifications and buy a lamp switch to be able to turn the pad on and off. I glued all the pieces together and stuck the wiring inside. The fans are doing a great job and, after only 20 minutes of gaming, lower my CPU temperature by 60 degrees. □

Right  Theo's first attempt sent airflow the wrong way – we all make mistakes



Mammoth Beat Organ

By Sam Underwood and Graham Dunning

 MammothBeatOrgan.co.uk

The Dunning Underwood Mammoth Beat Organ is a modular, mechanical music contraption created by Sam Underwood and Graham Dunning.

Designed as a two-player, semi-autonomous musical instrument, it plays unusual, sometimes erratic compositions drawing on drone music, minimalist repetition, and fairground organ techniques.

The pair iterated through various power sources, using a roller from the back of a hearse and a motor out of an old treadmill, and settled on a roller from a conveyor belt. This sits at the centre of the device. This turns bike wheels attached to different instruments, and the system is modular, so Sam and Graham can swap components in and out mid-performance to create a unique sound every time it's played.

The valves that distribute air to the various organ pipes are repurposed ballcock valves from toilets.

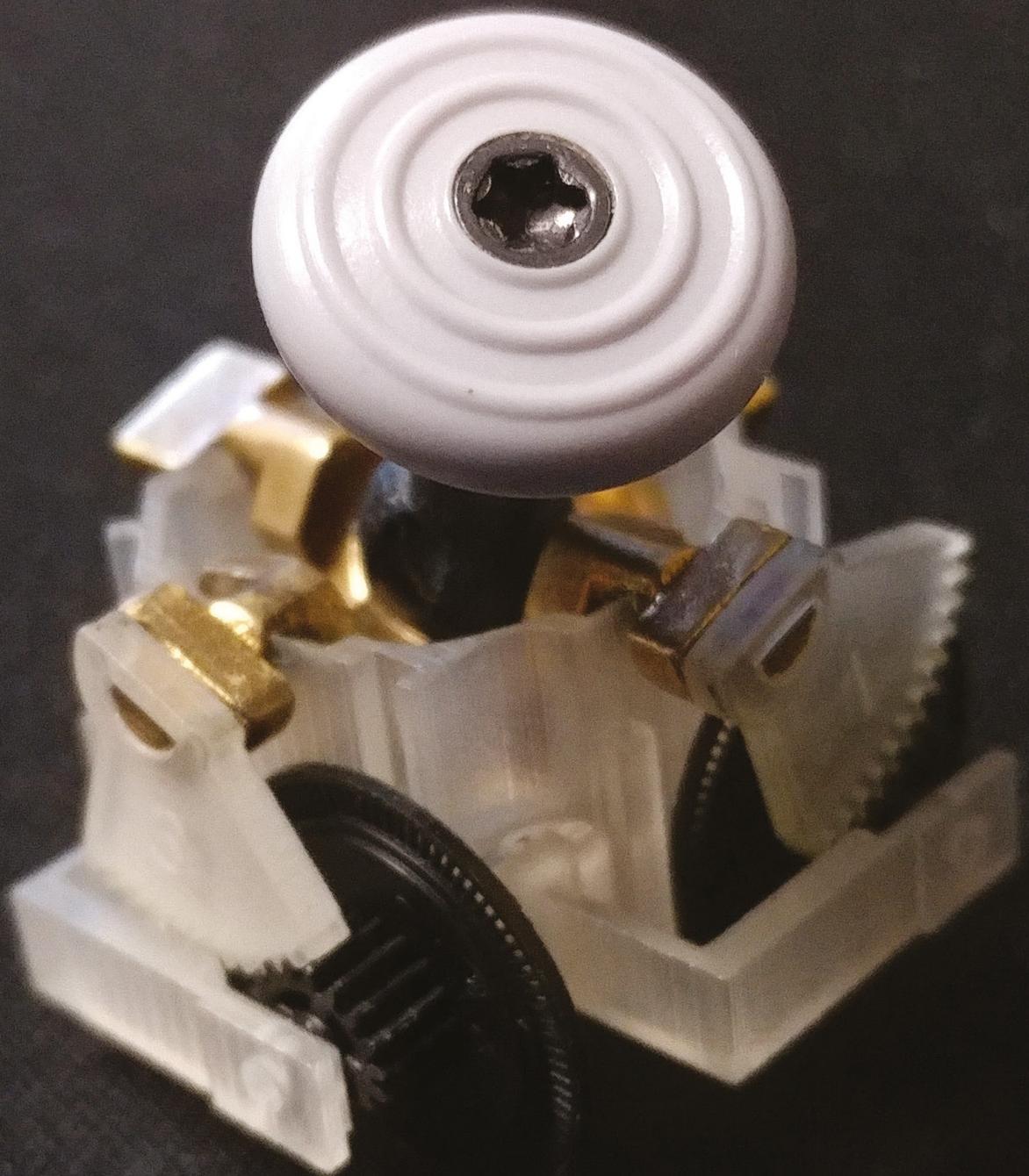
Says Graham: "It's slightly dangerous... there are various parts where you can trap your fingers in the gears. I think you'd have to really know what you're doing to not injure yourself."

"It's heavy, cumbersome, and completely fills my van," says Sam, "but it takes the music in a different direction. We never know where it's going to lead." □



Right  This semi-autonomous modular mechanical music instrument had its debut performance at the Supersonic Festival





Right ↗
The plastic parts
are absolutely
mashed compared
with the new
brass components

N64 Controller

By Nam Le

 imnam.com



am Le is a student at California Polytechnic University, and just like the students of 1998–2002, he's a big fan of the Nintendo 64.

He's such a big fan that he goes through game controllers at the rate of around one every two months – so instead of wasting time on eBay, he's reverse-engineered them.

First, he took an old one to bits, measured every part, then reassembled it in the SolidWorks CAD program. During this step he found that the reason they break down is that the moving parts eventually grind away to dust. The way to fix this was to upgrade the materials – cast brass for the gear housings and socket that the joystick fits into, 3D-printed VisiJet M3 Crystal for the gear teeth and the joystick housing, and 3D-printed nylon for the joystick itself.

"Generally, harder materials won't be worn by materials of a lower hardness," says Nam. "The result is having only one part wear compared to every old component wearing. Every once in a while the stick will have to be replaced, but it takes a longer time to wear, and is a better cheap alternative than buying a new controller." □



Objet 3d'art

3D-printed artwork to bring more beauty into your life



Head to 3dhubs.com/book
to check out the
#1 3D printing book on Amazon

We've seen a couple of 3D-printed clamps so far, but this is the one that immediately leapt out at us as being

potentially useful. First of all, it went together extremely easily, and feels solid. There's one handle to turn, which is geared to two screws pushing the vice

together, so you get even pressure across the whole thing. The nature of 3D-printed plastic means that it's nowhere near strong enough for metal or woodwork, but for holding PCBs securely for surface mount components, it's perfect.

Thingiverse user Christoph Laimer (hsmag.cc/blhjhA) made this – download the files from the link below. □

↗ hsmag.cc/njFzMv





3

D-printed objects are usually made of layers of hot plastic, but this Hulk is an example of something quite different: it's printed out of resin, and set not by a temperature change, but by exposure to UV light. Resin prints are smoother and more detailed than thermosetting plastic (you can just about see the veins on Hulk's arm), but our favourite difference is watching the process: the object emerges out of a pool of resin like the Terminator at the end of *T2 Judgment Day*, but in reverse.

For more on resin printing, turn to page 18 and our chat with Josef Prusa. □

Meet The Maker: Josef Průša

Taking home 3D printing to the next level



3

Dprinting has come a long way in its short history. We've seen various printers at all price points, but the one thing they have in common is that they're all better than their equivalent models of just a couple of years ago.

The factors that have driven this are the collaboration inherent in open-source hardware and the constant drive to provide something better. Both of these are found in the work of Josef Průša, creator of (among other things) the most widely hired home 3D printer on the market, the Prusa i3.

We met up with Josef to find out how the magic happens, what's next in the pipeline, and how he got started.



Right ♦

This level of detail was unimaginable on hobbyist printers only a few years ago

"I was playing with music and I started to build my own controllers," says Josef. "So I needed to make knobs and faders, and that's when I found 3D printing. So I went ahead and started to build a RepRap. And because I'm a lazy person, I found it really complex. I wanted to make it easier and simpler during the process."

"When I shared it back, people started to use it instead of the original design. It became my hobby, and after two more years, when people were constantly asking me to build a printer for them, I turned it into a business. And from there it's grown."

"I went full time in, I think, 2012, so six years ago. I dropped out of college and started a company."

"A lot of people in Czech use me as an excuse, 'Hey, he also dropped out, and look at them.' But you have to have something before you drop out. If you just don't like studying it's a bit tougher."

PRINTING WITH ULTRAVIOLET

You could say that things are working out OK for Josef; his company has sold around 100 000 printers so far, with most of that coming in the last three years. So what's next on the horizon?

"We acquired an SLA resin printer company and, for last year, we were working on making a new one, the Original Prusa SL1. That's a resin-based 3D printer, which is a completely different technology than we were doing before. It's pretty exciting."

Resin printing sounds a bit sticky, but according to Josef, the process is similar to using a traditional material such as PLA. The only difference is that you use light rather than heat:

"PLA is a thermoplastic, so you heat it up and then you lay down the layers. But this uses liquid resin: when you shine a specific UV light on it, it cures ➤



Above □
Josef's Original Prusa SL1
makes great Hulk prints

into hard plastic. So the layers are not done by melting the plastic, but by putting light in specific places, and it cures the resin.

"It's pretty similar to a lot of UV coatings. You put on a coat of varnish and then you harden it with the UV light. If you could put the varnish in a specific space, cure it, put down another layer of varnish, cure it, put another layer of varnish, cure it, you would have 3D printing.

"It has a different use [to PLA], it's high resolution, and in some instances faster, and it's usually used by jewellers.

"Basically if you want to do something that's very small and high-detailed, this technology is for you.

“If you want small detail, like you have on your glasses, these things, they would be very hard to print on FDM; they would not be as detailed”

"If you want small detail, like you have on your glasses, these things, they would be very hard to print on FDM; they would not be as detailed. But on SLA you can print them with no problems."

THE POSTER CHILD OF 3D PRINTING

3D printing is growing up: "It's interesting to see the change in the last year, year and a half. There were a lot of companies who were small, but at some point the market started consolidating, and I'm very happy that I started to invest in development a lot a couple of years ago, so we do our own slicer, our own firmware, we can develop new stuff. If you are a company just building Marlin [firmware] and Cura [slicing] compatible machines, and if you are just building printers to be compatible with these, you are never going to make something new. So you know, it's market disadvantage. At the end of the day, we are open-source and everything, but if I need to push the company forward, you need money, you need the sales." □



**Left ♦**

Josef's been busy: "Last month, I went to Detroit for one week, was home for two days, and then went to Japan, which is around 20 hours of time difference."

Learning the ropes

Do you need qualifications to be a maker?



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

Leonardo da Vinci was apprenticed aged 14 as a 'studio boy' to the sculptor and painter Andrea del Verrocchio in Florence. In the workshop, he would have had practical experience of drafting, chemistry, metallurgy, metal work, plaster-casting, leather work, mechanics, and carpentry, as well as the chance to improve his creative skills in drawing, painting, sculpting, and modelling. At the age of 26 he became an independent master – he had all the skills he needed to set up on his own.

Today, this type of 'education' is unavailable – and probably, at the time, it was only available to a very few. So what should you do if you want to be a maker today? Do you need qualifications?

When I started thinking about this, I had clumped 'qualification' and 'licence' together in my head. However, a qualification shows that something has been studied to a certain level, and is awarded forever. For example, 'A' Levels, a Higher National Certificate (HNC), or a vocational City and Guilds (cityandguilds.com) are all qualifications. A licence is usually a permit from a relevant body that allows you to do something, is often temporary and can be revoked. For example, in the UK, the Health and Safety Executive (HSE) can issue a licence for the manufacture of explosives, and Ofcom can issue a 'spectrum innovation licence' in order to use cellular bands in research and development. Surprisingly, you don't need a licence to operate a high-powered

laser in the UK. Some licences require you to pass a test first. Probably the most common licence is a driving licence.

For some professions, such as in medicine, aviation, and civil engineering, qualifications are very important – and you can't get a licence to perform those jobs without them. However, for the majority of making, the piece of paper is not as important as the skill.

I have some vocational qualifications, such as the City and Guilds in Wood Turning. So why did I do the course? At the time, it was the cheapest and easiest way for me to become proficient at that skill. I much prefer to be taught than to try to teach myself – and YouTube wasn't invented back then!

There are many opportunities to learn in unofficial environments such as meetups, workshops, and clubs, as well as the many things you can find on the internet – from courses and videos, to forums, and even Twitter.

Some skills require specialist equipment though – and formal training establishments are great for this. I am envious of the students on the BA (Hons) in Model Making (hsmag.cc/TiPICv) course at the Arts University Bournemouth, and those studying at the School of Jewellery at Birmingham City University (hsmag.cc/ysZULC).

Qualifications are often a stepping-stone that give you opportunities, such as working for others. But in many cases for many makers, formal qualifications are not actually required. There's only so much you can learn from studying. However, practical experience and regular practice are essential. □

So what should you do if you want to be a maker today? Do you need qualifications?

Trade war stories

Picking your way through red tape



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.

M

any readers of this journal may have already felt a direct impact from the rapidly escalating trade war, as distributors like Digi-Key have started passing the cost of tariffs directly to customers located in the US. Being on the front lines myself, I already have a couple of war stories to share in case they might be helpful.

Tariff codes (also known as HS codes – essentially a numbering system to describe every imaginable product) are often ambiguous. While one should always endeavour to pick the most appropriate code, multi-featured devices often

fit several tariff code descriptions. For example, I’ve seen a Raspberry Pi classified as everything from a computer to a circuit board, which can have significantly different tariff outcomes for certain jurisdictions. Thus, it pays to study the tariff lists to try and find a tariff code that accurately represents your product, that is also not taxed. My observation is that the more ‘finished’ a good is, the less likely it is to be taxed by the US. So, circuit boards and components are taxed, but as of writing, certain toys, A/V appliances, video games, and mobile phones are not.

You may have also heard news commentators quip that it’s futile for

a single country to engage in a trade war. This is because the 1974 Kyoto Convention specifies that the country of origin can be set by the last country where a product’s HS code has changed – known as the ‘substantial transformation criterion’. So, even if 99% of the value-add could occur in country A, as long as the last 1% value-add in country B legitimately changes the HS code of

the product, the country of origin is B.

This is why ‘last screw’ operations are getting so much attention now. China can still produce the circuit boards and plastic case for a typical consumer electronic product, but the country where these get screwed together

– a simple, low-value operation – can be legitimately claimed as the country of origin under these rules. Ironically, the tariff policy, as implemented, makes it extremely unattractive to move ‘last screw’ operations into the US, because taxes are highest on the parts, components, and tools necessary to implement a last-screw operation.

If you’re planning a new product, I’d encourage taking a deeper look at these trade rules, as tweaking your product’s function or packaging, or perhaps a small adjustment to your supply chain, could significantly alter the cost structure for your product. □

I've seen a Raspberry Pi classified as everything from a computer to a circuit board, which can have significantly different tariff outcomes for certain jurisdictions

Letters

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



FLY ME TO THE MOON

The article from Jenny List about quadcopters reminded me of my childhood summers making rubber band-powered planes from balsa wood. I wondered if these could now be made using a laser cutter and controlled via a tiny microcontroller?

Andy Clark

London

Ben says: I bet they could you know. And now that we've done quadcopters and rockets (turn to page 32 if you haven't already seen this issue's cover feature), we're thinking about putting a microcontroller into a Zeppelin. Gravity will not defeat us!

TIME AT THE BAR

I take my hat off to your interview subject (Al of Al's Hack Shack YouTube fame) in issue 11. A workbench with a built-in fridge, with a built-in beer ejector... it's genius. As it only holds one, I'm not sure how much use it would be in my house, but still – there's your next iteration!

James Atherton

Oldham

Ben says: The only problem with voice activation, that I can see, is that it's so damn creepy. I really don't want Google, Amazon, or whoever, listening in all the time (even if it is only in the shed). That's why the Mycroft server project is so appealing (hsmag.cc/fRMAKW). Rather than send your queries to a company whose raison d'être is spying on you, the Mycroft software can run on a server in the home, so it's completely private. And it's open-source – double win.



INTERNET OF BUBBLES

For your next IoT project, I'd like to see a device that lights up when I've got a message. I find beeps and notification noises extremely distracting, so I usually have the volume turned down on all my devices. It would be great if my devices could let me know when I've got a message using some less intrusive method



– a diffused light maybe, that could fade in without breaking my concentration?

George McKenzie

Oakland, California

Ben says: I might be wrong, but DoES Liverpool makerspace showed off a device at Maker Faire UK that did something similar – when a particular account was tweeting, it would blow bubbles. That's what the Internet of Things was made for.



When backing a crowdfunding campaign, you are not purchasing a finished product, but supporting a project working on something new. There is a very real chance that the product will never ship and you'll lose your money. It's a great way to support projects you like and get some cheap hardware in the process, but if you use it purely as a chance to snag cheap stuff, you may find that you get burned.

CROWDFUNDING NOW

wESP32

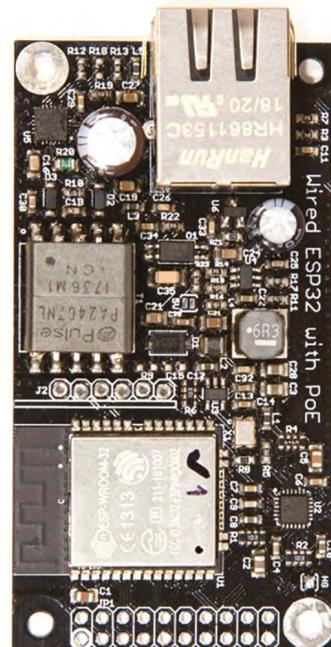
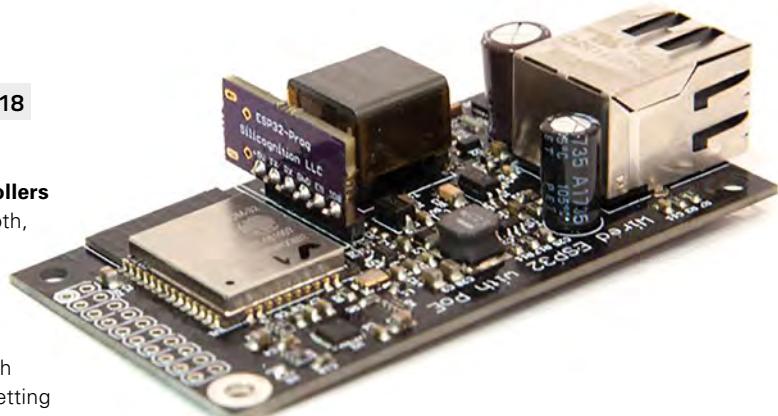
The internet of wired things

From \$55 | hsmag.cc/xAEwDP | Delivery: December 2018

There's an abundance of connected microcontrollers available these days. Whether you need Bluetooth, WiFi, LoRaWAN, or almost any other way of sending data across the airwaves, you can find a board with access built-in. However, what about the humble wired Ethernet? Many buildings have this already wired-in, so attaching more devices is trivial, and with Power over Ethernet, you don't even need to worry about getting electrons to your device.

There is, of course, the Raspberry Pi, but if you want a simpler controller, you'll have been limited to older boards, until now. The wESP32 is, as you might have guessed, an ESP32 dual core microcontroller with a wired Ethernet connection. WiFi and Bluetooth are also available (as on all ESP32 modules), and 15 GPIOs are exposed for you to connect your own hardware, sensors, or anything else you need.

“The wESP32 is, as you might have guessed, an ESP32 dual core microcontroller with a wired Ethernet connection”



Above There's also a programmer board that can help upload code and work as a serial connection

Left Ethernet is inspired by ALOHAnet, which was a wireless network at the University of Hawaii. It has its roots in wireless, even though it goes down wires

Ethernet is – perhaps surprisingly – a good choice for IoT as it's much easier for devices to 'just work'. There's no pairing or setting up of SSIDs – just pop in the cable and you should have network access. Admittedly, this probably isn't a convincing enough case for anyone to set up a physical network if it doesn't already exist, but if you've got a network in place, why not use it for your IoT? □

As We Are

Ever wanted to see your face up in lights?

By Andrew Gregory

 @AndrewGregory83

This is **As We Are**, a 14-foot tall interactive sculpture imagined by Matthew Mohr and built by DCL, a Boston-based company, specialising in fabricating really cool things.

It's built out of layers of contoured aluminium, covered in custom SANSI LED modules, comprising 850 000 individual LEDs. In the back of the head there's a photo booth, with 32 Raspberry Pis and Camera Modules, controlled by custom facial recognition software that finds your face, flattens it, and maps it onto the LEDs on the exterior of the sculpture, making your face twice as high as André the Giant.

As We Are is located in the Greater Columbus Convention Center in Columbus, Ohio; it was inspired when Matthew wanted a project to show off the diversity of the city.

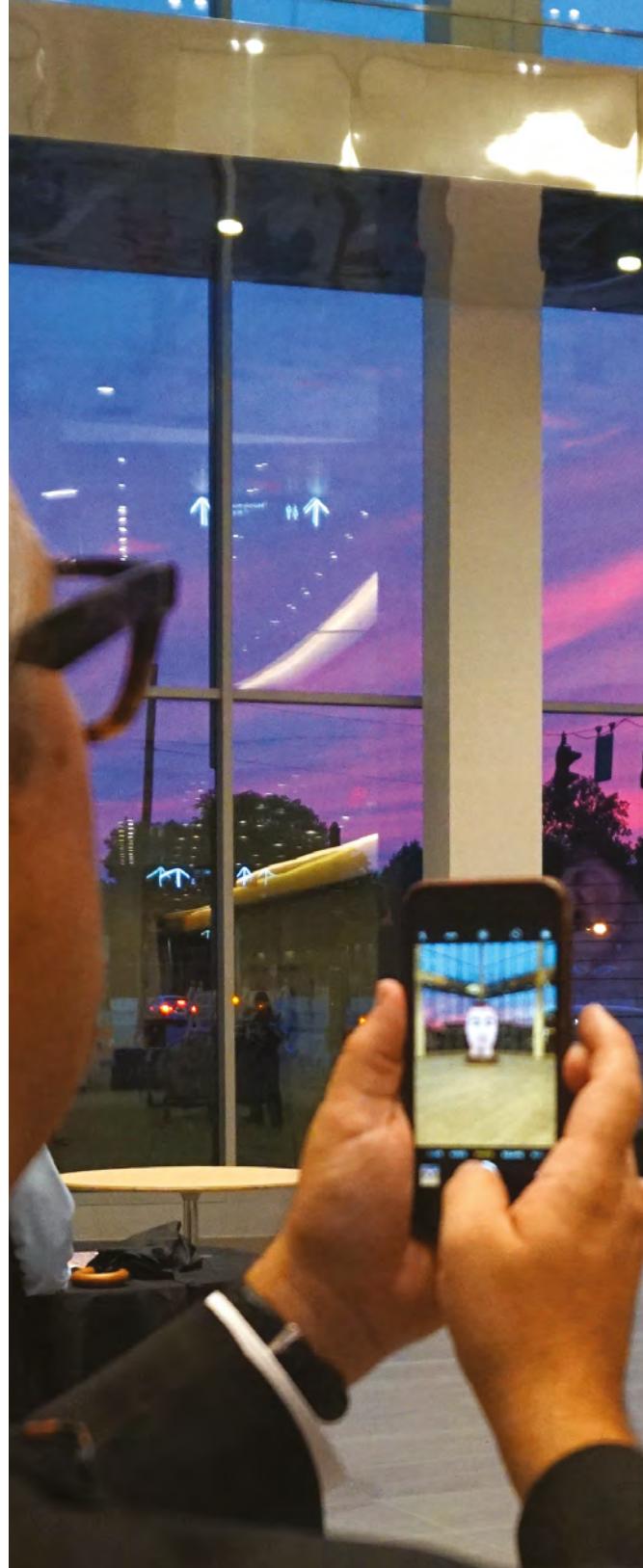
This struck us as a brilliant project for a couple of reasons. The first is that it's a giant version of the Frank Sidebottom head. The second is that, at its heart, it's just some Raspberry Pis, Camera Modules and LED screens, meaning that it should be possible to make one of these yourself (OK, a much smaller, lower resolution one, but technically similar). Go forth and give it a go. □

 hsmag.cc/eYjPdW

Right ♦
Every subject in As We Are is also a participant – this is really public art



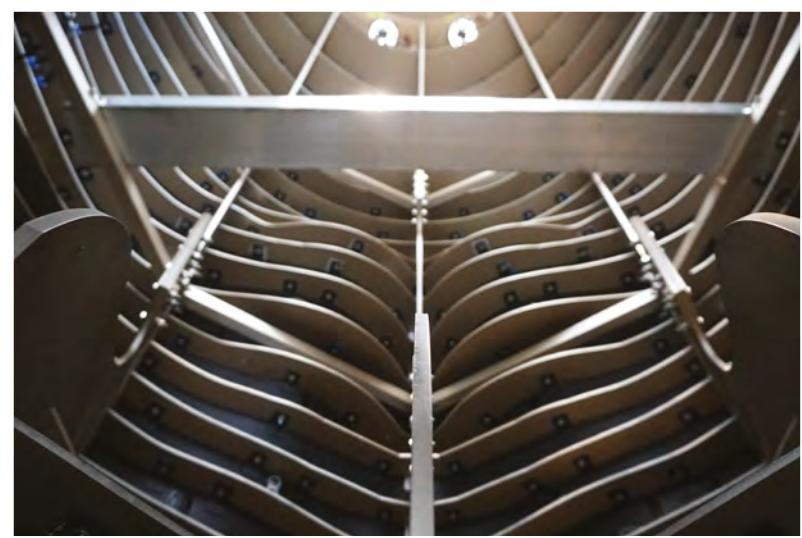
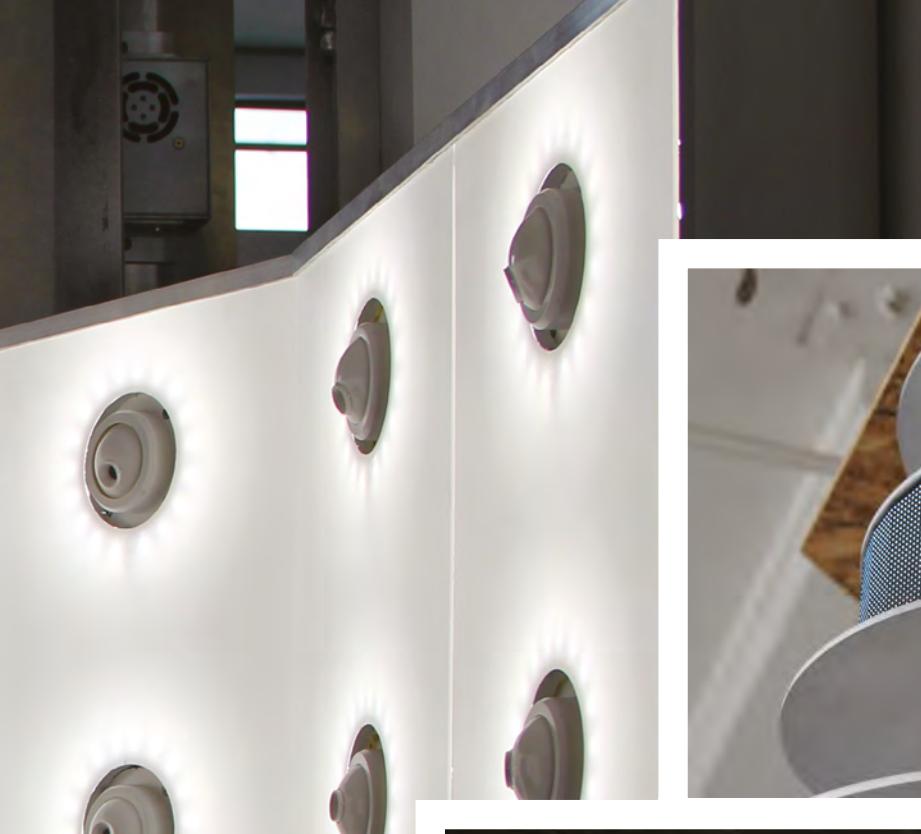
Right □
During the day, the sculpture faces into the atrium. At night, it rotates to face outward to the street







Right ↗
Each of the circular
mounts holds a
Raspberry Pi and a Pi
Camera Module



Above ♦

The frame is made from 24 layers of curved aluminium, stacked to form a human face

Left ♦

The eyes and nose of each subject are altered slightly in software to better conform to the physical sculpture

Below ♦

Inside the head

Over 10,000 Different Boards and Modules In Stock

NO ONE KNOWS BOARDS LIKE DIGI-KEY!

- Open Source Community Boards
- Evaluation and Development Boards
- Wireless Modules
- Single Board Computers
- Reference Designs
- Programmers, Emulators and Debuggers

**DESIGNS START HERE!
DIGIKEY.COM/BOARDS**

Digi-Key
ELECTRONICS



#MAKEWITHDIGIKEY

Digi-Key is a franchised distributor for all supplier partners. New products added daily. Digi-Key and Digi-Key Electronics are registered trademarks of Digi-Key Electronics in the U.S. and other countries. © 2018 Digi-Key Electronics, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA



LENS

HACK | MAKE | BUILD | CREATE

Uncover the technology that's powering the future

PG
46

GLOBAL OPEN SCIENCE HARDWARE

Science is about openness,
so the tools should be too



PG
52

AMIEDD

We chat about body hacking
and what it means to be a cyborg

PG
60

LIGHT BULBS

A great source of light, heat,
and glass compartments for
your next build

PG
32

BUILD YOUR FIRST ROCKET

Shoot for the moon in
your homemade
flying machine

HackSpace
TECHNOLOGY IN YOUR HANDS



Build your first rocket

FEATURE

Build Your FIRST ROCKET

Reach for the stars



Ben loves cutting stuff,
any stuff. There's no
longer a shelf to store
these tools on (it's now
two shelves), and the
door's in danger.

I'm heading to Midland Sky for my first taste of model rocketry. Just off the M42 in the West Midlands, we follow a farm track to a strip of grass alongside the field that serves as car park, camp-ground, and launch site.

Occasional clouds push across the post-harvest fields and a group of rocketeers gather around, readying their crafts.

There are two groups of launch pads – one for High Power Rocketry (HPR) and a smaller one for smaller rockets, closer to the group of spectators. In both cases, the launch pads look like poles, supported at the bottom and poking at the sky – although later, when I see them up close, I find that the model pad is thin metal poles, while the larger ones are slotted shafts (the rockets have pins that slot into these).

A group of people approach and fill the pads with rockets, then retreat to a safe distance. Paul Carter, the Range Safety Officer (RSO), summons each rocketeer in turn to launch. Checking that there's nothing overhead (we're between two airports, and adjacent to a microlight airstrip), he issues a countdown.

**Five
Four
Three
Two
One**

VVVVWOOSH!

The engine lights and, leaving a trail of smoke, the craft flies skywards. Even though I'm not involved in the rocket, the roar, the smoke, and the flash of flames induce a tingle of exhilaration in my veins. The closest thing that I can describe it to is a penalty shoot-out. Although it's non-competitive, there's the same build-up, the same explosive release, and the same sudden success-or-failure.

Most engines burn for only a few seconds, but that's enough to send them hundreds, or thousands, of feet into the air (the club uses feet rather than metres).

The crowd of spectators (mostly, but not entirely, fellow rocketeers) crane their necks upwards. Even the larger rockets (over two metres in length), can be hard to spot at high altitude,

and having more eyes means it's more likely to be tracked.

Once it's reached its high point (the 'apogee' in rocketing terminology), it starts to fall. The simple, smaller rockets have a small charge that detonates a few seconds after the motor; this pops out a parachute and it glides down safely. However, the larger and higher-flying rockets don't use this automatic method, as it would deploy the parachute high up – this means that there'd be a long drift as the wind carried it across the field, and possibly into a neighbouring farmhouse. These use an altimeter to automatically release the parachute at a preset height.

As the rockets glide safely to earth, the crowd applauds – both in congratulations of a successful launch, and in appreciation for the show that's been put on for them.

Once all the rockets have launched, the rocketeers set off across the field to retrieve their craft, and a new group of people begin setting up their rockets.

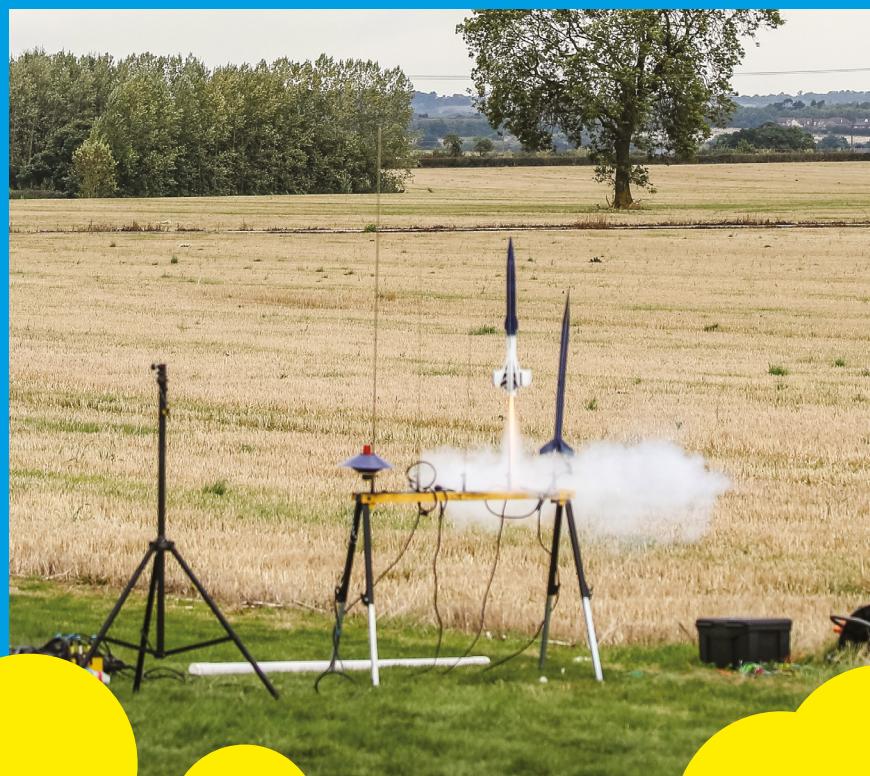
At least, this is how the rockets should launch. A few go wrong. Mostly, they fail to ignite the motor – usually due to a duff igniter. A few spin spectacularly, or fail to deploy their parachutes, and end up coming back in more pieces than intended. →



Above Once you've learned the basics, you can improvise with different materials



Below A line-up of model rockets ready for launch



Our First ROCKET

Start simple and then develop your skills

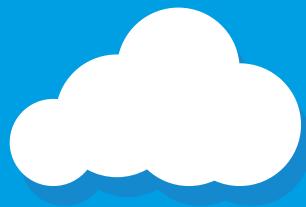


W

e couldn't turn up to the event empty-handed, so in the week before, I set about building my own rocket to take to the event.

There are lots of ways to build a rocket, but by far the best option for beginners is to use a kit. This way, you don't have to worry about things like the centre of pressure or the structural integrity. You just have to put the bits together and, once you've gained the experience of getting this working properly, you can progress to custom designs, should you wish.

Below ▶
Setting Starship
HackSpace mag on
the launch platform



There's a wide variety of model rocket kits available, but almost all of them follow the same basic design – cardboard tube for the body, balsa wood fins at the bottom, and a plastic nose cone on the top.

We built and flew two kits: the Wizard and the Baby Bertha, both from Estes (estesrockets.com). While they aren't hard to put together, they're not completely trivial. The most complex part is attaching the fins to the body. Use the paper guide to mark the appropriate place on the tube, and then hold it in place while you glue the fins on. Once this is done, you attach a piece of elastic to the nose cone and the main rocket body.

A parachute or streamer attaches to the nose cone and, once the main engine has finished burning, it ignites a second charge that pops the nose cone out, and this deploys the parachute. As the Wizard is a narrower rocket, there's no space for a parachute – instead, it uses a streamer, which is enough to slow it down to a safe speed.

We had both kits built, painted, and adorned with our logo in three hours, and most of that was waiting for the glue to dry so we could paint it.

READY FOR LAUNCH

Paul Carter, the Chairman of Midland Rocketry and RSO of Midland Sky, gives my rockets a look-over. The first test is to make sure the nose cones are held in place. They friction-fit into the body tube – too tight and they don't pop out, too loose and they can slip during flight. He determines that they're too loose. A couple of strips of electrical tape make the nose cone a little wider and it then fits securely in the rocket.

The Wizard kit also doesn't secure the motor, so Paul suggests another strip of electrical tape to ensure it doesn't fall out once it's finished burning.

CLUBS

Launching your rocket as part of a club is a great way to get advice (Paul's help tweaking our kits undoubtedly helped them fly better), and also to see what other people are doing, and it gives you a chance to see more powerful rockets fly.

However, it's not essential, and you can launch rockets by yourself. The UK Rocketry Association (UKRA) maintains a safety code that you should follow if launching in the UK (hsmag.cc/tHLWqR). In the US, the National Association of Rocketry maintains a similar code (hsmag.cc/bMzxLE). Many other countries have similar organisations, so make sure you're following your national body's advice.

The final bit of prep involves adding an igniter. There's a hole in the bottom of the motor, into which I poke the electrical igniter. It's a small blob of translucent material with two wires coming out. A plastic plug holds this in place, and the ends are bent over to make it easier for the crocodile clips to connect.

ALL SYSTEMS GO

The rocket then slips down the guide-rod on the launch pad (there's a thin plastic tube glued to the side of the rocket), the launch controller is clipped onto the igniter (making sure there's no short circuit), and we're ready for launch.

Below ♦

The first of our rockets heading (almost) straight up to the clouds



We retreat to a safe distance, and get ready. Paul turns a key on the launch controller. A faint whine comes out, indicating that the Wizard is ready to go.

I become aware that a small crowd has gathered around; all experienced rocketeers, but they still turn up to support me and watch my simple rockets.

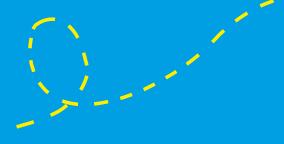


Above ♦
A launch controller lets you deliver power to ignite your motors safely

Five
Four
Three
Two
One

Nothing. I hammer the launch button a few times, but it's futile. We switch to Baby Bertha and get ready again.

Five
Four
Three
Two
One
FFFFWWOOOSH!



Success! It shoots off, leaving a trail of smoke. It becomes a faint dot in the sky, then the parachute deploys. The crowd still applauds the successful launch; the first time anyone's applauded anything I've made.

We go and take a look at the Wizard. Paul gives the crocodile clips a sanding to make sure the accumulation of soot hasn't stopped the circuit, and we try again.

This time it works. The two rockets have the same engine, but the Wizard flies far higher than the Baby Bertha. Again, the nose pops off, and the streamer comes out, helping it sail safely back to earth.

The noise, smoke, the rockets' raw power, coupled with the nerves of it not firing, combine to make it an exhilarating experience. I can't claim to be a rocketeer, but I have built and launched a couple of rockets! ➤

Below ♦
The final preparation for launch is to connect the igniter to the launch controller



Design

YOUR OWN

Personalise your rocket using specialist software



Jo Hinchliffe

[@concreted0g](#)

Jo Hinchliffe is a contributor to the Libre Space Foundation and sits on the UK Rocketry Association council. He loves designing and scratch-building both model and high-power rockets, and releases the designs and components as open-source.



nce you've tried a rocket kit, you might want to move up to something a little more personalised. Designing your own rocket can be as simple or complex as you like. There are a few bits

you need to take into consideration, such as the estimated height and the stability. Beyond this, you can complicate things with multiple motors, or by making scale models of other craft, or you can keep things simple. You can go for ultimate power, or you can keep things at a smaller scale. However you do it though, it's easier with some design software to help you. Let's take a look at how to design a rocket using OpenRocket, one of the most popular bits of design software for hobbyists.

OpenRocket helps you understand how different parts will fit together and how these will affect the flight characteristics. You can even put the rocket through a simulation that estimates how high the rocket will go. Let's get started on our rocket.

First, we'll need to install the relevant software. OpenRocket runs on Java, so you should be able to run it on Windows, Mac, or Linux, provided you've got the runtime environment installed (you can get this from java.com/en/download if you don't have it already). With this, you can download OpenRocket from openrocket.info. It's a JAR file, so there's no need to install it. Just click on the downloaded file to start.

When you first start OpenRocket, it'll start a new rocket project for you and you'll be greeted with a screen similar to **Figure 1**.

Traditionally, rockets have a pointy bit at the front, a long straight bit in the middle, and some fins at the

Right The Open Development rocket on the launch pad at Midland Sky



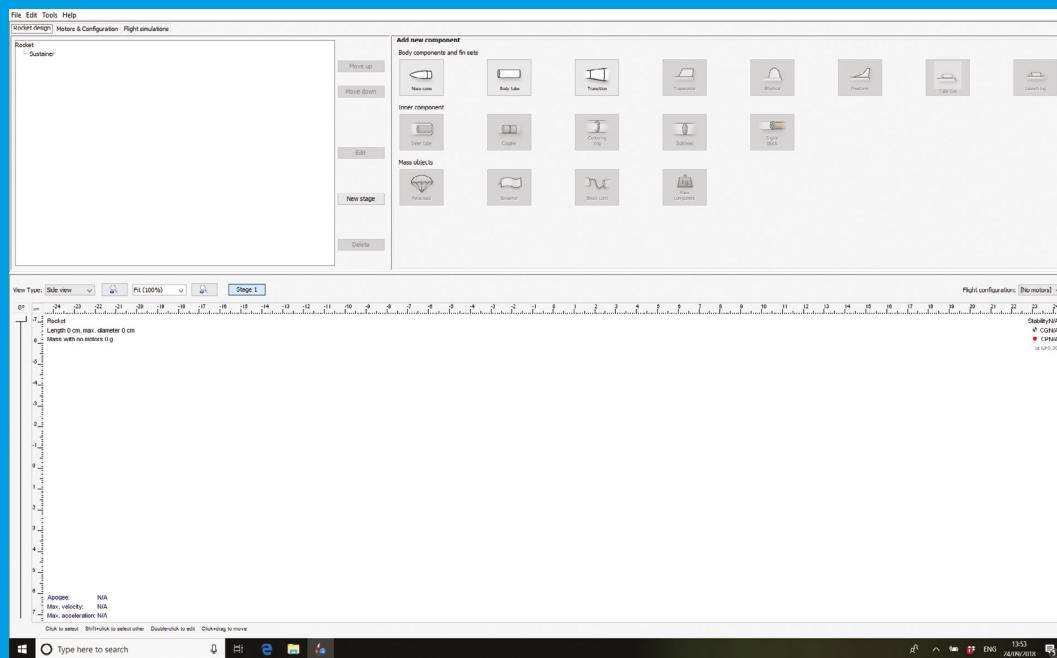


Figure 1 The layout of OpenRocket has all the key information split up into three panes

back, and ours will be no different. Let's start with the pointy bit at the front (technically, this is known as a nose cone).

Click on Nose Cone in Add New Components, and you'll see a roughly triangular shape appear in the side-view panel. You'll also get a dialog box that lets you customise this. Enter the following options:

- **Nose cone shape:** Ellipsoid
- **Nose cone length:** 8cm
- **Base diameter:** 3cm
- **Wall thickness:** 0.2cm
- **Material:** PVC

We'll also need to add a shoulder that lets it fit into the main body tube. Click the Shoulder tab and enter the following:

- **Diameter:** 2.6cm
- **Length:** 2cm

Click Close to get back to our design.

We'll now create the main part of our rocket. Click on Body Tube, and enter the following in the dialog:

- **Length:** 45cm
- **Outer diameter:** 3cm
- **Set material:** Cardboard

The inner diameter should automatically set to 2.6cm, giving a wall thickness of 0.2cm. Double-check that this is correct, then click Close to return to our rocket.

The final part of our rocket outline is the fins at the back. To add this, click the Trapezoidal icon. You will need to have the body tube highlighted in the plan view for the fin set icons to be available to select. Either click on the body tube itself in the plan view area, or click its name in the component tree area.

In the dialog box, set the following:

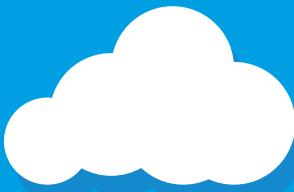
- **Number of fins:** 3
- **Root chord:** 4cm
- **Tip chord:** 4cm
- **Height:** 2cm
- **Sweep:** 1.5cm
- **Sweep angle:** 36.9
- **Fin cross section:** Rounded
- **Thickness:** 0.3cm
- **Material:** Birch

Then click Close.

You should see now that the centre of pressure (CP) symbol (red dot inside red circle) is behind (toward the fin end) the centre of gravity (CG) symbol (blue quartered circle). The top right-hand corner of the plan box contains some text about their positions. The 'Cal' number is to do with stability. It is the number of calibres (diameter of the body tube) behind where the CG the CP is. We ideally aim for the CP to be between 1 and 2 calibres behind the CG. Over 2 calibres is OK, but is classed as 'overstable' and may turn more into any prevailing wind, resulting in a ballistic flight. At the moment, the stability is 2.26 Cal, but this will change as we continue to alter the design. →

EXAMPLE ROCKETS

We've built a simple rocket, but you can browse the other example rockets for some other designs. Click on File, then Examples, to see the list of available designs. There are some showing more advanced capabilities such as clustered motors and multiple stages. You can tweak these, or build new rockets based on them, but make sure that you match your designs with your skill as a rocket builder – don't jump straight to advanced builds without gaining the necessary skills first. Rocketry clubs are a great place to get guidance on how to proceed.



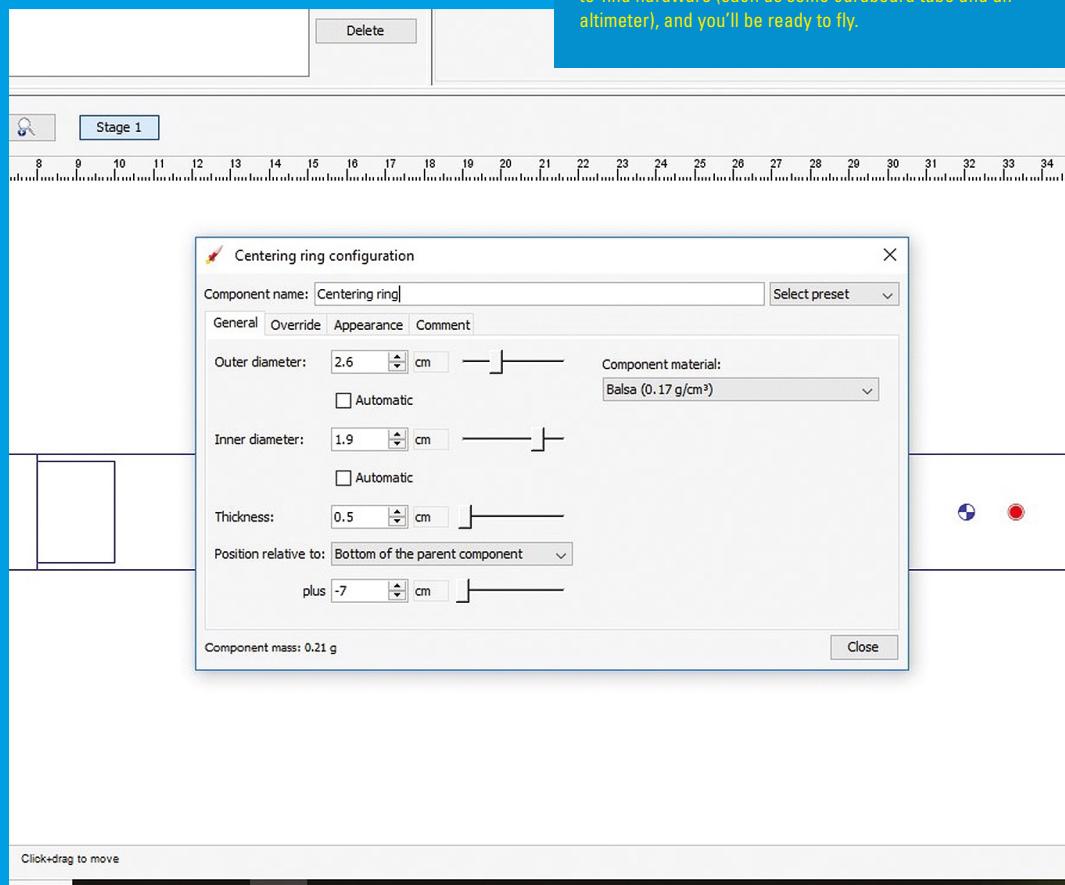
Right □
Selecting the correct material helps OpenRocket understand how the rocket will behave in flight

We've got the outside of the rocket designed. It's time to move inside and look at how we'll secure the motor. With the Body Tube highlighted, click the Inner Tube icon and, in the dialog box, set the following parameters:

- **Outer diameter:** 1.9 cm
- **Inner diameter:** 1.8 cm
- **Wall thickness:** 0.05 cm
- **Length:** 7 cm
- **Material:** Cardboard

In order to make sure that OpenRocket knows where to put our motors, we need to let it know that this is where we want to put one. In the Motor tab, check 'this is a motor mount', then click Close.

The inner tube is just floating around inside the rocket at the moment; we need to add some bits to secure it to the main tube. With the Inner Tube (motor mount tube) highlighted (you'll need to select it in the component list), click the Centring ring icon, and enter the following in the dialog:



OPEN-SOURCE ROCKETRY

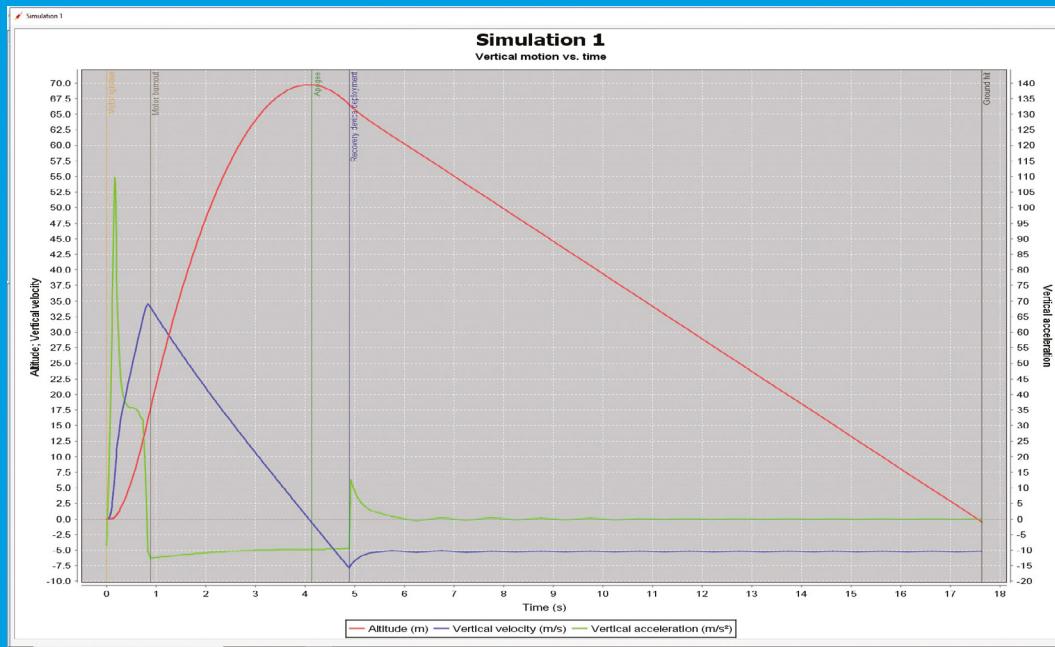
If you're looking to get started at designing your own rocket, an open design is a great place to begin.

The Open Boat Tail rocket takes low-power 18mm engines, so is great for people getting started in rocketry. You can find the designs at hsmag.cc/AOudlm.

If you're looking for something with a little more 'oomph' (mid- to high-powered rocketry), then the Open Development Rocket is a great place to start. It's a rocket designed to be made using a 3D printer and laser cutter (as well as a few commonly available parts, such as cardboard tube). It takes a 38 mm motor in the mid-to-high power range (G to I).

On its maiden flight, the ODR reached 3255 feet, but deployed the parachute a little bit too early (possibly due to a loose nose cone shaking as it fell, rather than waiting until it had reached the preprogrammed deployment altitude), so if you base your design on this rocket, you may need to make adjustments to take this into account. You can find the designs at hsmag.cc/TFEFJa.

In both cases, you'll find everything you need to 3D-print and laser-cut your rocket. Couple this with some easy-to-find hardware (such as some cardboard tube and an altimeter), and you'll be ready to fly.



Left □
The simulation shows you the speed, height, and vertical acceleration of your rocket

- **Outer diameter:** 2.6 cm
- **Inner diameter:** 1.9 cm
- **Thickness:** 0.5 mm
- Set 'position relative to Bottom of the parent component' to -7 cm
- **Material:** Balsa

Repeat this to add a second ring with the same values, except the 'position relative to Bottom of the parent component' as 0. This will put a mount at either end of the motor tube to fully secure it to the main body.

We're almost there now. The only thing left is to ensure that our rocket can come safely down to earth. Highlight the Body Tube and click the Parachute icon. Then enter the following in the dialog:

- **Diameter:** 30 cm
- **Position 'top of the parent component':** 3cm

Now is the time to save your work. Click File, then Save As, and give your work a unique file name if you haven't done it already.

READY FOR LAUNCH

We've got our rocket designed, so now it's time to see what it'll do in flight. Click to highlight Inner tube/motor tube, and then click the 'Motors and configuration' tab in the tabs area. Click New

Configuration, then double-click on None in the Inner tube column.

There are all kinds of ways you can constrain the lists of commercially available motors. For example, you could set the software to search for an individual motor manufacturer or for the motor diameter. For our run-through we need to find an 'Estes B6-4' motor and select it, making sure to add a four-second delay in the motor selection dialog box, then click Close.

Now everything's set up, let's see how this flies! Click the Flight simulations tab from the main three tabs.

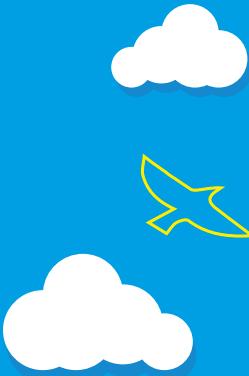
You should see that there is a single simulation listed that hasn't been run with your motor details in it. Highlight it, and then click the 'run simulation' button. The slots will fill with data. Then, click the Plot button and OK the next dialog, and you should get a flight simulation graph.

Congratulations! If you made it this far, you have designed and simulated a small, and not terribly efficient, rocket. Time for you to experiment and play and design something amazing. →



JOE BARNARD

Meet the maker attempting to land model rockets



Below
The electron rocket that Joe's been attempting to land vertically. Note that the fins are on the top to help stability when coming down

M

ost model rockets follow the basic flight process that's been unchanged since the Space Race.

A motor accelerates the rocket upwards. At some point, the engine stops burning – the rocket then slows, and falls to earth. A parachute is deployed and glides the craft down as safely as possible. However, since SpaceX pioneered vertical landing (where a second burn of the engines slows the rocket down, allowing the rocket to land vertically on the pad), hobbyist rocketeers have been keen to emulate this feat.

Joe Barnard is attempting to do this using solid-fuel rockets, and regularly uploading videos on his progress to YouTube at: hsmag.cc/pZALEA. These engines are cheap and simple to use, but don't have any control over the thrust. Once they're lit, they



keep burning until they're spent, unlike hybrid or liquid engines that can vary their thrust. This means that it has to be lit at exactly the right moment (or at least within about 20ms of the right moment) in order to land smoothly.

PRACTICE MAKES PERFECT

"It's not perfect, but as soon as you start adding actual throttle control, everything gets about 1000 times harder because then you need plumbing, you need much stricter safety procedures. If you're willing to accept that – just by luck – some of your flights aren't going to work, then you can get really close with solids."

While the amount of thrust can't change, the direction can. Joe developed a thrust vector control system that can angle the motor in order to keep the rocket upright as it takes off and returns to earth.

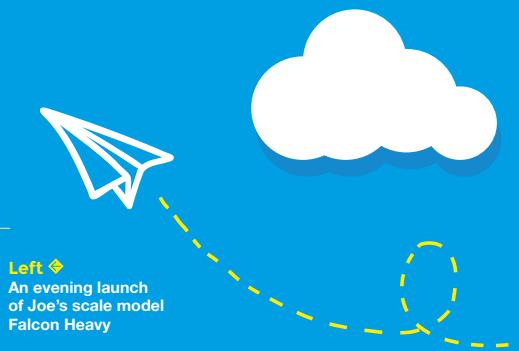
In order to simplify the problem, Joe's currently working on just the landing, by dropping rockets from drones.

"I think I'm getting pretty close to getting it. I don't know if it's going to be a couple more tests, or if it's going to be another three years. We'll see."

Joe's work is unashamedly inspired by SpaceX. In fact, Joe first started working on thrust vector in order to hopefully impress SpaceX enough to get a job there. He's no longer working towards this aim, but you can still see this inspiration in some of his rockets, including the 1/48th scale model of a Falcon Heavy that he's working on, complete with thrust vectoring in the boosters and second stage.

Looking to recreate Joe's style of rocket? He sells the Signal thrust vectoring system, which comprises a flight computer and 3D-printed mount that uses two 9g servos to control the motor. He's also publishing a series of videos, entitled 'Landing Model Rockets', that go through the technicalities of doing exactly this. →

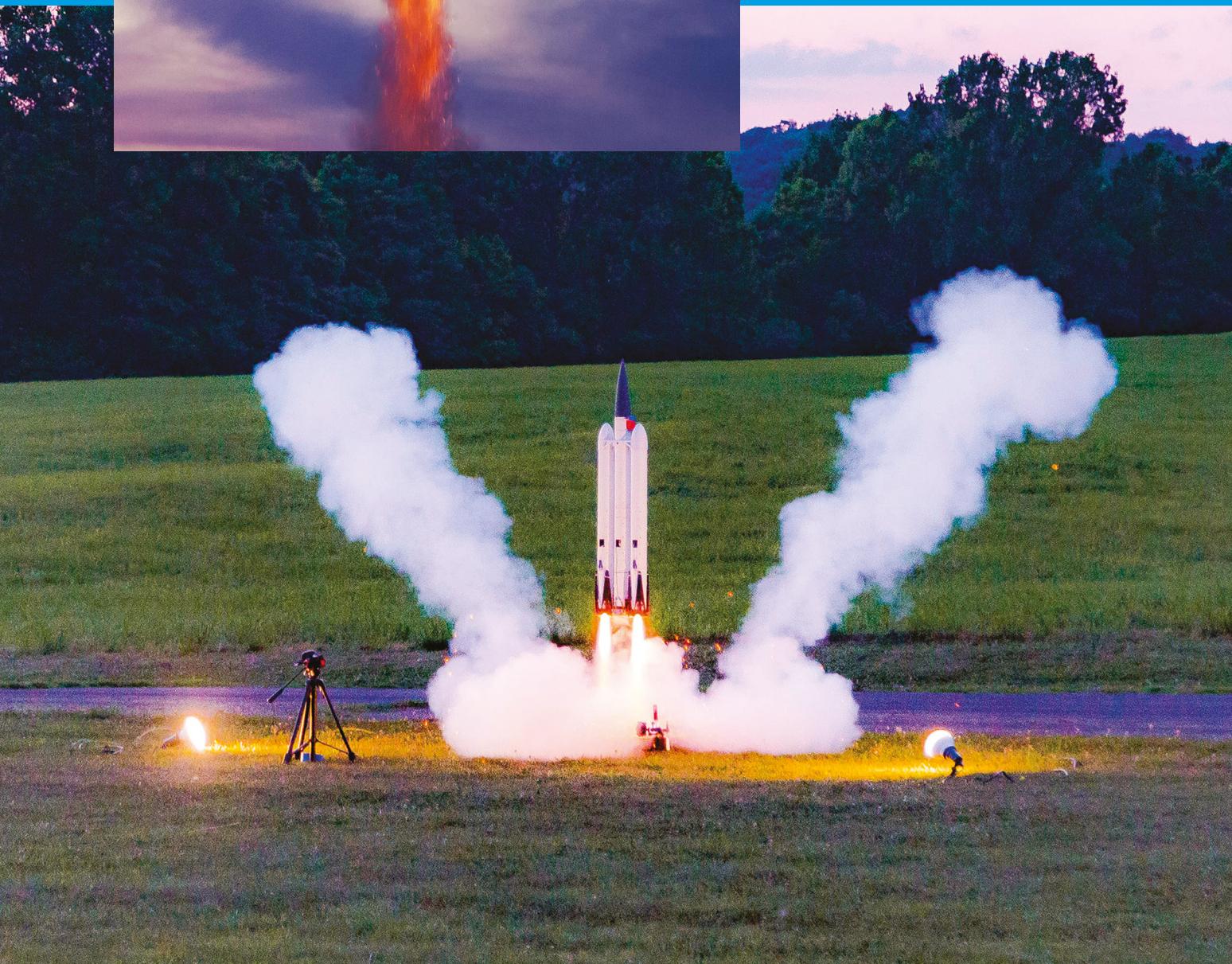




Left ↗
An evening launch
of Joe's scale model
Falcon Heavy

Below ↘
Using thrust
vectoring allows
slower launches,
which are more
realistic compared
to full-sized rockets

Credits
Joe Barnard,
BPS.Space



Build your first rocket

FEATURE

COPENHAGEN SUBORBITALS

Inside the amateur space programme run by hobbyists

Right ↗

The Heat 1X leaving the launch pad. At the top you can see the capsule for an astronaut

Credits

Copenhagen Suborbitals – Thomas Pedersen





Left ♦
The Nexo II rocket being safely recovered from the Baltic Sea

Credits
Copenhagen Suborbitals – Mads Stenfatt

Copenhagen Suborbitals (CS) is an amateur space programme aiming to put a person in space. That's an ambitious target – you need a powerful rocket, a reliable launching system, and a way of getting an astronaut safely back to Earth. This has obviously all been done before, but only by huge space programmes run by the world's largest nations. Replicating this achievement with a completely volunteer staff of hobbyists is a massive challenge, but it's one that CS seems determined to meet. The CS team have already been working on this for seven years, and have launched seven rockets, each one getting them a little closer to their goal.

CS's first rocket, HEAT 1X stood 9.38 metres tall and weighed 1680kg on the launch pad – an enormous rocket by amateur standards. As a testament to the team's ambition to launch a person into space, the HEAT 1X included a capsule with a life-size crash test dummy. It was powered by a hybrid engine which used a liquid oxidiser (nitrous oxide) and a solid fuel (polyurethane).

Unfortunately, the rocket started to pitch to one side immediately after launch, and ended up at an angle 30 degrees off horizontal, and had to be manually shut down (via the radio link). A post-mission analysis concluded that a misaligned engine nozzle caused the problem. As HEAT 1X was passively stabilised, there was no process to correct this problem in flight.

Since then, CS has been launching smaller rockets and building up the technology bit by bit. The most recent rocket, Nexo II, took off from CS's floating launch platform in the Baltic Sea on 4 August 2018.

Powered by liquid ethanol and oxygen, with thrust vectoring to able it to adjust its flight, this is a far more advanced rocket than HEAT 1X. It reached an altitude of 6500 metres, before returning to Earth successfully via a parachute and a splashdown. Although this was far too small to carry a person (6.7 metres long, 30cm wide, and weighing 292 kg at take-off), this is the first CS rocket to achieve a landing soft enough that a human could have survived. This launch wasn't without problems, though, failing to reach its intended altitude of 12 000 metres. The team are still investigating why this happened.

BIGGER AND BETTER

The final goal of CS lies in the Spica rocket, which will stand 13 metres tall, 95.5cm wide, and weigh 4000kg. This will be powered by a liquid-fuel engine similar to those in the Nexo II, but scaled up to provide 100kN of thrust (compared to 5kN on the most recent launch). Hopefully, the Spica rocket will be able to reach an altitude of over 100 km above Earth (known as the Kármán line, and generally accepted as the limit of space) and deploy a capsule which will contain an astronaut who will then come safely down to sea.

If you're interested in what it takes to build and launch rockets of this scale, CS has an informative set of videos that take you through the things the team are building and the problems they encounter. You can take a look at hsmag.cc/DJOviq.

CS is financed by supporters who either sign up to contribute monthly (in a Patreon-like fashion), or buy merchandise. If you're keen to see them reach their goal, you can help sponsor the mission on their website: hsmag.cc/XSGpAt. □



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

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

Global Open Science Hardware

Inside the fight to provide global access to scientific hardware by 2025



Cameron Norris

✉ @cameronsnorris

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

M

embers of the Global Open Science Hardware (GOSH) community believe that scientific progress is being held back by high equipment costs, proprietary designs, and suffocating intellectual property

restrictions. However, a new report, crowdsourced from over 100 contributors in 30 countries, suggests that open-source hardware, digital fabrication, and collaborative research tools can provide academics and citizen scientists alike with a promising new solution.

The community gathered in Santiago for GOSH 2017, to discuss strategies on how open scientific technologies can be made accessible to all levels of society, particularly researchers in developing countries and communities wanting to gather and analyse data about their environment. The event, organised by Dr Max Liboiron, an environmental scientist from the Memorial University of Newfoundland, resulted in the completion of a "Roadmap for making Open Science Hardware ubiquitous by 2025."

The GOSH Roadmap includes detailed guidelines for providing global access to scientific hardware that can be fabricated using open-source designs, rather

than ordered from a catalogue. "Reproducibility is a hallmark of good science, and Open Science Hardware allows for greater reproducibility," explains GOSH's community manifesto, which was co-authored by Dr Liboiron and Greg Austin, in consensus with attendees from the first GOSH event in 2016 at CERN in Geneva.

GOSH defines Open Science Hardware as any piece of hardware used for scientific investigation that can be obtained, assembled, used, studied, modified, shared, and sold by anyone. This includes standard lab equipment, as well as auxiliary items such as sensors, biological reagents, and electronic components.

Dr Liboiron believes that many scientific endeavours are being held back by lack of access to the tools/materials required for even routine experimental techniques, which severely limits the ability of groups to engage in the scientific process.

Even with access to existing proprietary solutions, customising hardware to meet the needs of individual experimental setups is challenging, which can restrict the implementation of experimental designs and push back scientific progress. "The fact that a lot of modern scientific equipment is a consumer product that is patented, not supplied with full



ULTIMATE IMAGING

Hyperspectral imaging systems are increasingly being used in precision agriculture to monitor the health of crops. These devices cost from £15 000 to £80 000, or more when purchased commercially, but Prof. Tom Baden and PhD student Nora Nevala have developed an open-source version that costs just £1500 and even works underwater!

design information, and difficult to repair also blocks creativity and customisation," Dr Liboiron explains.

Open Science Hardware can increase access to tools by lowering financial barriers to participation in research, and by providing flexibility for hardware customisation. The GOSH community hopes these efforts will help democratise scientific practice by increasing the diversity of people with the tools required to perform research for knowledge discovery, education, innovation, and civic action. Also, decentralised production chains, enabled by digital fabrication methods, could create markets where it is difficult to import scientific instruments.

FRUGAL SCIENCE

For example, Prof. Tom Baden (University of Sussex) and colleagues published plans for FlyPi, a 3D-printable fluorescence microscope. He estimates that FlyPi costs around five times less than comparable models. "All of our designs are available on GitHub, and we also put the best ones on Thingiverse and Open-Labware.net so they are freely accessible to anyone who wishes to use them."

The GOSH Roadmap's purpose is to create a focused plan for addressing the primary barrier to Open Science Hardware achieving its full potential: communities that use and develop open hardware are diverse groups, often separated by geographical and

disciplinary borders. This separation limits their ability to collaborate and collectively support Open Science Hardware as a single cohesive community.

THE ROAD TO 2025

Despite this, many developers of open hardware for science are highly active when it comes to sharing designs and information online, but poor documentation means that many Open Science Hardware projects never go far beyond the research lab or small community that initiated them, which significantly limits their potential use and impact.

GOSH has identified three critical activities that will enable achievement of their ambitious goal by 2025. These include ensuring that information about Open Science Hardware is accessible to a broad audience, securing support from existing institutions, such as colleges and universities, and expanding the existing community. Also, the introduction of documentation standards and quality control guidelines will help ensure that projects are shared appropriately.

GOSH members will move forward with plans for scaling both the community and the reach of open hardware distribution, at GOSH 2018 in Shenzhen, China. Find out more at: hsmag.cc/IuSFbQ.

OPPORTUNITIES FOR ALL

"Scientists in developing countries, grassroots community organisations, and citizen scientists can struggle to obtain and maintain the equipment they require to answer their own research questions. The result of this exclusion from participation is that scientific research becomes ever more elitist, as a small number of people decide what the worthwhile and valid projects are."

Dr Max Liboiron and Dr Jenny Molloy



Above ♦
Open Science
Hardware



Left ♦
GOSH 2017
electronics workshop

Trawling the Depths

DIY citizen science projects
for monitoring marine plastics

Researchers believe that of the 5.25 trillion pieces of marine plastic in the world's oceans today, 92% are microplastics less than 5 mm in size. These tiny plastic particles can easily be ingested by marine life as small as plankton, and studies have shown that they are capable of absorbing up to a million times more chemicals than their surrounding waters. Scientists are becoming increasingly concerned that widespread ingestion of these microplastics is causing toxicants that accumulate in animals to magnify up the food chain, causing a potential threat to human health.

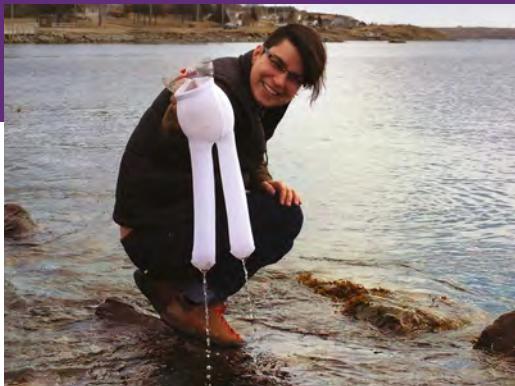
Below ◊
The BabyLegs plastic pollution monitoring device in action

Images
Clear

In response, Dr Max Liboiron created Open Science Hardware project 'BabyLegs' as an inexpensive means to trawl for floating marine microplastics using nylon tights, soda pop bottles, and other readily accessible materials. Dr Liboiron designed the simple sampling system to mimic the \$3500 Manta Trawl, a commercial net used by marine researchers to sample the surface of the ocean. "In essence, BabyLegs looks like a little person or jellyfish, and during presentations and field research, we treat the technology like a doll because it encourages people to approach us, talk to us, and share images of the technology," she explains. "The goal of the project is to be as accessible as possible so that people in Canada's remote northern communities can monitor plastics in their fishing and hunting areas."

The province of Newfoundland and Labrador has over 18000 miles of rugged coastline, which is about twice as much as the entire United Kingdom. Many of those living outside of St. John's, the province's major city, do not have a regular income and lack year-round access to external supplies. This causes them to depend heavily, and even at times exclusively, on the ocean for survival.

"Rural, low-income, and Aboriginal communities rarely have control over the type of scientific questions that are asked about their areas," says Dr Liboiron. "Our goal is to increase the ability of underserved communities to identify, redress, remediate, and create awareness and accountability around environmental concerns."

**Above**

Dr Max Liboiron collecting microplastics from a raw sewage outflow from a rural community in Newfoundland, Canada

#BABYLEGS AT SEA

In March 2015, BabyLegs was first used in a raw sewage outflow from a rural community in Newfoundland, Canada. "We caught microplastic fibres from kitchen scrubbers and fibreglass strands from cigarette filters," explains Dr Liboiron. "In Newfoundland, where I live and work, there is almost no information on the state of marine plastics, even though the province's cultures and livelihoods are focused on the ocean."

Dr Liboiron stresses that the data gathered by BabyLegs is qualitative rather than quantitative, meaning that you can identify types of plastics, as well as a general ratio of plastic types in your sample, but you cannot estimate the total amount of plastics in any given area. This is because as the flow of water picks up speed, the holes in the tights expand, enabling some of the smallest microplastics already collected to escape.

This type of research in Northern Canada is very tough. Most beach survey protocols – where volunteers and scientists count plastics that wash up on beaches – assume warm, sandy beaches. Currently, there are no established protocols on how to gather and study ocean plastics in an often frozen and rocky setting, like Newfoundland. "Plastics less than 5mm in size, the most plentiful of marine plastics, disappear between rocks, making it seem as though our main type of plastic pollution is large fishing gear," says Dr Liboiron.

ATLANTIC CURRENTS

Beachcomber Martin Gray discovered fish tags originating from Newfoundland near his home in Scotland. Based on his findings, scientists believe that currents are carrying plastic waste from Canada across the Atlantic. CLEAR's Fish Tag Citizen Science project aims to create a database of where these tags have been found, and where they came from.

LITTER BUGS

According to the U.S. National Oceanic and Atmospheric Association, nearly 100000 marine mammals have litter-related deaths each year, and an estimated 80 per cent of marine litter gradually makes its way to the ocean from land-based sources via storm drains, sewers, and other routes.

To overcome this problem, a group of Dr Liboiron's students, including Cian Kavanagh, Colin Grenning, and Nicolas Brouard-Ayres, began designing a "plastic eating device for rocky ocean coasts", to catch microplastics that would otherwise disappear between rocks before a sample could be taken. Should an unsuspecting beachcomber stumble across the device, P.E.D.R.O.C. (as it's affectionately known) includes a flag that describes what the technology is being used for and how passers-by can make their own. Although still in development, the P.E.D.R.O.C. 2.0 prototype was successful in detecting marine microplastics and surviving a short period of deployment on Topsail Beach, Newfoundland, an area of extreme kinetic wave energy.

ICE CREAM

Another notable project from CLEAR is the 'Ice Cream Scoop', an educational tool to help children learn about the marine environment by allowing them to see the presence of plastics in the ocean first hand. Additionally, there is the 'Plastic Entanglement Trap', a static ocean plastic monitoring device, which can be anchored in the water near a shoreline, where plastics are commonly found.

If you want to start gathering your data or contribute to one of these projects, there are many ways you can get started. A list of open-source tools and protocols to guide you through creating your studies on marine plastic pollution is available from the Civic Laboratory for Environmental Action Research website at: hsmag.cc/eqUPe.

**Above**

A fully assembled BabyLegs plastic pollution monitoring device

**Below**

Plastic fishing waste such as this is a common cause of ocean pollution

FlyPi Modular Microscopy System

Capturing tiny details

Prof. Tom Baden, a neuroscientist at the University of Sussex, worked with André Maia Chagas of the University of Tübingen to develop the FlyPi, a 3D-printable open-source microscopy system. The FlyPi is based on a Raspberry Pi 3, Arduino Nano microcontroller, high-definition camera, and a range of off-the-shelf electronic components. All of the mechanical parts of their device are 3D-printed and allow for modular placement of additional components, including holders for Petri dishes and microscope slides. In its simplest form, the FlyPi can be assembled for well under £100. This remarkably low price means that the FlyPi can be used by under-funded labs across the world, for classroom teaching in schools, and by enthusiasts interested in participating in citizen science.

ALL-IN-ONE BIOLOGY LAB

Tom explains that the average assembly time for the FlyPi, including the complete software setup, usually takes around two to three hours. However, those without previous soldering experience should expect it to take up to five hours. "FlyPi is not our only creation; we have actually built several pieces of equipment – pipettes, micromanipulators, and "even a pico-injector," he says. "But some form of microscope is at the core of most biomedical labs and, as such, it's an obvious starting point."

According to Tom, the ever falling price of high-performance charge-coupled device (CCD) chips and optical components means that a functional neuroscience laboratory, capable of delivering high-

Below A 3D-printed FlyPi, with the Raspberry Pi 3 clearly visible

Images
Open-Labware.net



Above Prof. Tom Baden guides a FlyPi fabrication class

quality research data can be built from scratch for magnitudes less than the cost required to purchase a single commercial scientific instrument. "If you break the lens out of a cheap laser pointer and tape it over a mobile phone camera, you already have a pretty powerful microscope in your pocket," Tom explains.

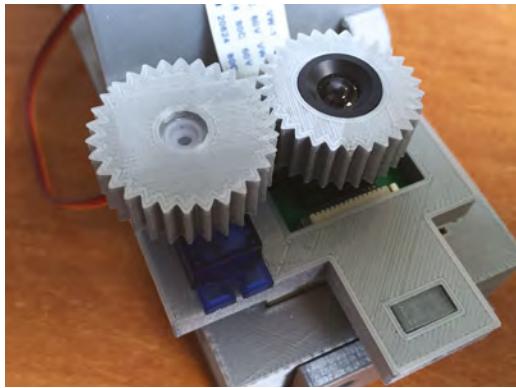
While developing the FlyPi, Tom and André discovered that a popular 12 mm adjustable-focus camera module for the Raspberry Pi could already

DAYGLO

Fluorescence microscopy has been used in the study of cells and living tissues since it was first developed by Otto Heimstaedt and Heinrich Lehmann in the early 1900s. By observing tissue samples prepared with a fluorescent DNA stain, the organisation of the DNA within the cells can be revealed.

PEERING BACK IN TIME

Historians have long disputed the inventor of the microscope. In the late 16th century, several Dutch spectacle makers designed devices for magnifying objects, with Hans Lippershey most famous for filing the first patent for a telescope. However, it wasn't until 1609 that Italian polymath Galileo Galilei perfected the first device we recognise today as the microscope.



Above

A 3D-printed FlyPi with motorised focus

be used as an effective microscope by screwing the objective lens all the way out and aligning the optics to bring nearby objects into focus on the CCD chip. "We stumbled across this when using the camera system in an attempt to film some fruit flies in the lab, so we just integrated the feature into our microscope design," says Tom.

And although the FlyPi doesn't match the image clarity of most commercial microscopes, it definitely outperforms its rivals in regards to flexibility. "If you want to, you can point the camera at a tree, zoom out, and do a time-lapse over several days. Or, you can zoom all the way in, point it at a blood smear, and count macrophages," Tom explains. When coupled with coloured sheets of plastic commonly used for theatre lighting, the system also allows for some forms of fluorescence microscopy, which can be used for tasks such as spotting parasites in tissue samples, or identifying different types of white blood cells.

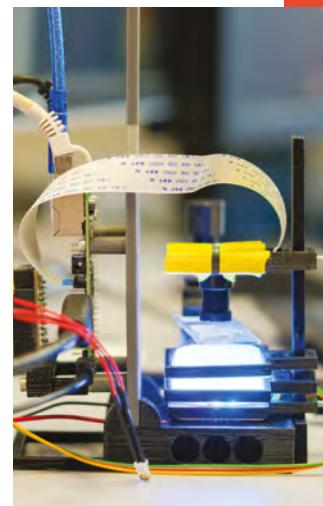
To survey to what extent the FlyPi may be beneficial in a classroom scenario, Tom and André ran a series of multi-day workshops at universities in sub-Saharan Africa. In one workshop, 3D-printed parts, custom PCBs, and off-the-shelf electronics were provided to students that were then guided through the entire process of assembly and installation. Although these

students had no previous experience with electronics or soldering, all of them were able to successfully assemble a working FlyPi. Assembling, using, and maintaining Open Science Hardware like the FlyPi can help students and citizen scientists expand their confidence, ideally inspiring them to build and modify other pieces of equipment themselves, which in a teaching scenario, as Tom quite rightly points out, is perhaps the most significant benefit of all.

CITIZEN SCIENCE!

The current version of the FlyPi only scratches the surface of possible applications, and a recent community-driven modification to the 3D-printed frame repositioned the camera and focus motor below a closed stage, resulting in a substantially more robust design, which is far better suited to classroom teaching. Other community-driven modifications include a version where all the 3D-printed parts have been replaced by LEGO bricks, as well as several forks geared towards optimising the code, 3D models, and expanding the electronic control circuits to include additional modules.

Tom and André have also set up a centralised public repository for anyone to access their Open Science Hardware projects: **Open-Labware.net**. This repository includes the Openspritzer, an open-source 'Picospritzer' that reportedly performs just as well as four-figure commercial models, and the Spikeling, a low-cost hardware implementation of a spiking neuron for neuroscience teaching and outreach. "We think it is very important that neuroscientific training and research is opened up to larger numbers of students and junior scientists around the world," says Tom. □



Above

The 3D-printed FlyPi designed by Prof. Tom Baden and André Maia Chagas

Below

Photograph of a bipolar cell taken using the FlyPi



HackSpace magazine meets...

AmieDD

Cosplayer, developer, open electronics enthusiast... and unicorn

Amie Danielle Dansby has one of the best job titles we've ever seen: **Full Stack Unicorn**. She's a game developer, software engineer, cosplayer, and 3D printer-er. While her CV is impressive (3D-printed electric swords are pretty damn cool), we wanted to ask her about body hacking. This is a growing scene, confined pretty much to early adopters, but it raises questions about hacking, ownership of data, and sharing, that affect anyone who's ever wanted to make their life better and share the results.

We spoke to Amie at World Maker Faire in New York, about *Black Mirror*, healthcare data, and what it's like to have 880 bytes of data implanted in your hand. ↗





Above □

What's the difference
between a chip implant
and a knee replacement
or a pair of glasses?

HackSpace Let's start off with the basics; you've got a chip in your hand. As I understand it, there are two sorts of tech that go in these implants: NFC and RFID.

ADD Correct. The one I have does both. They're both mainly like radio devices. That stuff has been around forever, right? It's a passive read device. If you have a badge to get into your building, that's RFID right there. It's not a long-range thing, it doesn't access the internet, it does one thing: access control. That's the simplest way to put it.

NFC is a more modern version. It's still the same kind of thing, it's reading information over short distances.

If you have Apple Pay, Google Pay, all of that type of information on your phone, you already have NFC on your phone. If you have a clipper to get into your building, that is RFID. It stands for Radio Frequency Identification. It's unpowered – you do not need a charge in your hand to pass any information. It's primarily used through magnets.

HS How big is it?

ADD It's about the size of a grain of rice. If you have pets, the chip in the pet is pretty much the same technology. It gets injected with a hollow needle. In terms of storage, it's 880 bytes.

HS That's far more storage than I thought it would have. Is this consumer-level stuff? Are companies adding compatibility with their services?

ADD That chip is from Dangerous Things. I guess you could go to your vet and ask for one of those chips, I don't know whether they should give it to you. [As for] readers, NFC is already built into your phone. I don't think it's going to go away or anything – the security may change on it a little, but it's here to stay.

I just tried to get your phone to read my hand, and it couldn't because it was

locked. The bigger security questions I get asked are more to do with tracking the individual with a chip in them, and no, you can't track me on the internet. I've also had arguments of, "Well, if I could track my kids, should I?" There are different opinions on that. There was a big issue in the US where these three girls went missing. They were missing for a number of years and they weren't too far away. Would you put one in your kid on the proviso that you aren't going to be watching all the time, but just in case something happens?

HS There's a definite air of *Black Mirror* about it.

ADD Right, and there's a whole bunch of stuff that's *Black Mirror* about body hacking. I think everyone has a different

Would you put one in your kid on the proviso that you aren't going to be watching all the time, but just in case something happens?

life experience that turns into ethics at some point. Would you do that to a child, knowing that they didn't have a choice? Even though they know that they can track your mobile device?

HS Well, you can turn your phone off, or leave it at home when you go out. You're not compelled to carry it around with you all the time. My dog has a chip in him, but he has no rights.

ADD My dog has one too, but it's more because the government requires him to have a rabies tag and identification.

If you got one, what would you use it for? That's the question I always ask people. Me saying what I use it for is totally fine, but asking what somebody else is going to use it for, that's what's

interesting. A lot of people wouldn't get one, but I find that I use mine for more than I think now I've seen what I can do with it, and that's part of contributing back to what you can do or what you can create with it.

HS I guess if there are devices that are used by more than one person in a household, that device could instantly know who's holding it and adjust its settings accordingly?

ADD Yes. For example, at Dallas Makerspace, the largest makerspace in the United States, we have so many members, but some of the tools need an RFID identification to start the machine, because you need to have taken a safety class for that. So, instead of carrying around your little key fob that's RFID, it's written to my hand. If I want to use the laser cutter, I scan my hand, it knows that it's me, because it recognises my tag, verifies that I am allowed to use it, then lets me use it. As far as turning on the machine, that's not the chip; it's the access control that it's giving me. We use them for that.

Myself and a couple of other people are working on transferring the data from a Tesla valet key into an implant. As far as the information you can read from them, we've scanned everything that's in there. Tesla does have a bug bounty programme, so they do give you access to some of that stuff, because if you can expose a vulnerability and report it to them, they can then patch it. But I don't really see it as an error; it's more like an alternative way of carrying a key.

So, we took apart a few of the valet keys, and you can take out the chip which pretty much is just a flat NFC. Have you ever seen an NFC chip taken out and it looks like a flat little sticker? It's just like that, so it's very tiny, it's about 2 mm by 1 mm. Put it in a glass-encapsulated chip, just like I have, and you could just insert it and it could work. ↗

Below ■

You can just about
see the scar from
where Amie's chip
went in

|





Above ■
Open data, open
source and sharing
are essential for body
hackers – we'll all
benefit from their
work sooner than
we realise

We're trying to test the range on it though, because what happens when you put a chip implant in here? Say this read range is 5 mm; well now it's in my skin, so read range may only be 2 mm. You want to test it really well before you put it in your body.

HS What about props – you're a cosplayer, you must have used it for something fun like that?

ADD We were talking with my friends and they were just like, why don't you do something with your sword? So, I have a cosplay sword and I want it to light up when I hold it. Or, you could do something based on an accelerometer with all the fancy microcontrollers. What if I just have it that my hand is the key, and it only lights up when it recognises the key?

And I'm like, I should do that, it would be fun. It's a very simple idea. It's just like taking copper tape, an LED, and battery, and you can make something light up. It's cheap and simple, and this is kind of the same thing.

HS I thought Thor's hammer would be a good idea: get a strong magnet to hold it down, which switches off when the right person takes hold of the handle.

ADD That would be a fun thing to do. And I guess not everyone's going to go and get a chip in their hand because of cosplay, I'm just trying to show that they're not necessarily a malicious thing.

I want to educate people on what it is and what it isn't. So, my dad's wife is very religious, and when I told her about it, the first thing she said was like, "666 mark of the beast!"

I have had some hate mail from a religious group in Greece sent to my house. Amy Danielle Dansby is my real name; I own a house in Texas, where all property ownership is public

record, so all you have to do is look up public records of your house, and they found me, and they sent me something. They compared me to "Antichrist technology like the Xbox One". It's kind of a compliment. I laugh about it now, but it's different when things come to your own home, to your family.

HS How affordable is it to get an implant like this? Apart from yourself, the only person I've met with one of these is a Bitcoin millionaire.

ADD It was \$99, US. And then the guy who did it – he's a body modification person, so the sort of person you'd get a piercing from, and he charged



I got some hate
mail from a religious
group in Greece...
They compared me to
"Antichrist technology like
the Xbox One"

it out or get a new one if I want to, when a new one comes out. So, that was my question: can I get a new one? I want to get an upgrade if a new one comes out, to upload more information.

iPhone is a little more closed off than Android, so I found, in that community, people were creating an app to be able to read the chip data, stuff like that. I mean they're helpful too, but I think the Dangerous Things Forum has been a more positive experience.

There are a few body hacking [conventions] too. They'll have medical devices, there's DARPA (Defense Advanced Research Projects Agency), which is for the US military...

HS Replaceability is a big thing for any tech. You said earlier that spectacles are a body hack, but I get these checked every two years, and I can put them on and off. But that's also a pain, because they fall off, and get lost, and get sat on sometimes.



ADD It's still something to improve your life. It's like when someone needs a heart valve replacement. Can they live without that? Do they have to get that valve replaced every however many years?

I see body hacking as anything that you're using to modify or enhance your way of life. So, say you have a toothache and you go to the dentist and get a filling – it reminds me of *Castaway* with Tom Hanks, and he's on the island and he has to break his tooth with the ice skate. That was such an awful thing for him, solely because he had a tooth infection. To have a dentist able to do that for you, and get it done without pain, is an enhancement to our way of living.

And because everyone does it, like everyone has glasses, you don't think of it as being bizarre or odd, right? You're like, "Oh, I'm getting old, I need glasses". What about getting laser surgery? That's permanent, you weren't born with it, and it's an enhancement. ↴

\$50. I think probably the prep time of him sanitising everything, cleaning everything by hand, probably took longer than the actual procedure. He came up to our makerspace and did it there.

HS Is there an online community where people can share information about their experiences of body hacking?

ADD Yes, the Dangerous Things Forum has really good things. People share information like, "Hey, I went for an MRI with metal in my body. This is what happened, this is what didn't happen".

I feel like everyone in the body hacking community who ends up getting an implant has different questions.

Some people will get one because they're intrigued, they're an early adopter. I like technology and new things. It's under the skin, but I can take



Above ↑
NFC is commonplace
tech: any Android
phone already has it
built-in

Left ←
Why shouldn't you be
able to unlock your
car with your hand?

Right →
One malicious
Android Package
kit successfully
delivered to an
unsuspecting victim





HS It's also just a bit too scary for me. What's the future for these? At the moment, it's just early adopters, like you say.

ADD I think the biggest thing is the medical industry. I have this friend who is diabetic, and he has it to monitor his insulin pump. He works for Microsoft. His insulin pump is actually hooked up to his Azure cloud, and the Azure cloud monitors his blood sugar levels. So, for people who need [it], that is something that's very interesting. That, if anything, is the definition of some kind of bio enhancement.



Above ▶ Other electric car makers are available

I got it because I like technology; he got it because it improves his life. It's where they come together that's going to help. Whatever's going to come to be, whether it's for diabetic people, or something that can monitor your vital statistics to send back to your doctor, your heart rate, and things like that.

But I think there needs to be a push to educate people on what it is, because fear is a big part of anyone's decision-making process.

If I said, "Right, now we're going to go get you laser surgery", you'd be like, "no, no why?" If you don't know the technology behind it, you're going to have loads of questions. It's a slow process to answer those questions.

My friend's part of Open Bionics, they have open-source prosthetics. They're trying to provide not just the 3D print, but also the electronics and the schematics, so that it's affordable for people who need it.

It's not easy, especially when you're trying to do stuff that's open-source – people need to eat and live still – but

if you can get the community behind it, open is a huge win. Most people's response to Open Bionics is, "Oh, I have both my arms – but that person needs an arm". The advantage is clearly visible. But, if you don't have a chip, the question is, "What am I going to use it for?"

It's not easy, especially when you're trying to do stuff that's open-source – people need to eat and live still – **but if you can get the community behind it, open is a huge win**

I think if anything were to push body hacking forward, it would be payments working with this. I love it when I go to the UK, because everyone uses Apple Pay, everyone, including the

Tube. Over here, not everyone takes it, and yeah, I think if the payment systems and the banks got on board, it might be a different story.

Banks and the medical industry are what will change it. We have the capacity to build a tiny device that will collect your vitals, send it to your doctor, warn you if you're about to have a heart attack, or monitor your loved ones if you're not around them – parents or grandparents, to notify you if someone's had a fall. That's where we're going. □

LIGHT BULBS



Light up your projects with this illuminating tool



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.

H

How many inventors does it take to design the light bulb?

The incandescent light bulb is energy-inefficient by design. It converts less than 10% of the supplied electricity into visible light and loses the rest as heat.

Despite this major shortcoming, light bulbs are still one of the most popular means of illumination all over the world, primarily because of their ability to easily adapt to electrical systems and operate even in low voltages.

At the core of the bulb is a wire filament that's heated to such a high temperature that it glows with visible light. The emission of light from a hot object is known as incandescence. The glass bulb that helps protect the wire filament from oxidation is filled with an inert gas. The concept of heating a wire to produce light was first demonstrated in 1761. Over the next several decades, many inventors produced their variation of the light bulb, but none of the designs were practical enough for commercial application.

One of the major milestones in the evolution of the light bulb came in 1850 with English physicist Joseph Swan, whose design enclosed carbonised paper filaments in a vacuum glass bulb. It lasted longer than previous iterations, but it wasn't until 1878 that he had

something viable, thanks to better vacuum pumps and the use of a treated cotton thread that helped prevent the bulb from blackening.

Thomas Edison was one of the two dozen inventors working on perfecting the light bulb. He experimented with different types of elements as filaments, including platinum before settling on carbon. He patented his improvement in the late 1870s and began marketing it soon after. The cargo and passenger steamship SS Columbia became the first ship to use electric lights instead of oil lamps.

The next major evolution in the bulb's design came from the team of Hungarian and Croatian chemists who were the first to patent the use of a tungsten filament, instead of the carbon one, in 1904. Then, in 1913, Irving Langmuir doubled the bulb's luminous efficacy by filling it with an inert gas.

But the days of the traditional light bulb are numbered, mostly because of the falling prices of energy-efficient alternatives. While it may have taken over two dozen inventors to design the light bulb, there's still enough room for makers to use them creatively. So, the next time one blows out on you, instead of chucking it away, use it to brighten your builds.

WINDOWSILL PLANTER

A

student of industrial design, Yasmine Riachi came up with this project during the Christmas break when she “was going through a bit of a light bulb phase, particularly

vintage light bulbs.” She applied her skills to turn used light bulbs and some wood into a planter that can be used to grow herbs. The project was a runner-up in the indoor gardening contest on Instructables.com but, more importantly, got her creative juices flowing: “Being in that creative headspace is a unique state of mind and I find little projects are very good at helping myself clear my mind from stress.”

“I FIND LITTLE PROJECTS ARE VERY GOOD AT HELPING MYSELF CLEAR MY MIND FROM STRESS”

For the project, she first built a stand in two parts with a thick piece of wood for the base, since it needs to support the weight of the bulbs. To hang the bulbs, she drilled holes in another piece of wood after measuring the neck size of the bulb. The piece was then cut using a jigsaw before it was attached to the base. Finally, she used a drill press to turn wood scraps into ring braces to fit the light bulbs and hold them in place. The entire process has been explained and illustrated in her Instructable. □

Project Maker
YASMINE RIACHI

Project Link
hsmag.cc/xNrrp

Below ♦
Yasmine, who is looking for industrial design internships, hosts an impressive design portfolio at yasmiriaci.com



SNOW GLOBE

Christine had a lot of bulbs to spare when she swapped them with new compact fluorescent lamps (CFLs) soon after moving houses. She was saving them for a project, and since Christmas was approaching, she decided to repurpose the bulbs as snow globes. She suggests using a kitchen scrubber or steel wool to remove the label on top of the bulb. The toughest (and most dangerous) part of the build is to disassemble the bulb, so make sure you use safety glasses and gloves. Christine has detailed the entire process, along with helpful images, to guide you through the process of using pliers and a knife to hollow out the bulb without breaking it. To fill the bulb with snow, add about one teaspoon of some crushed shell of a hard-boiled egg, along

with an equal amount of glitter. Then fill 3/4 of the bulb with distilled water and add a couple of drops of glycerine, before topping it up with more water. In another version, she stuck some mini figurines inside the hollowed glass with some glue before filling it with the egg-shell snow and water. The bulb globes were then sealed with bottle caps and a dollop of silicone sealant. □

Project Maker

CHRISTINE NAST

Project Link
hsmag.cc/tGGWKu

Below ◆
 Christine says
 an egg carton
 makes a great
 support stand for
 working on a light
 bulb, thanks to
 its ridges



OIL LAMP

Want to use a blown-out light bulb as a source of light? You can apply the skill acquired from the previous snow globe hack to transform a light bulb into an oil lamp. Inspired by industrial designer Sergio Silva's design, Instructables.com user Magnelectrostatic has detailed the process to create one with easily accessible items. The most cumbersome part of the build is the removal of the base of the lamp, which must be kept intact to augment the final look of the lamp. The Instructable offers suggestions and tips, along with illustrated images to help you with the process. Use a cylindrical file to make a hole in the base that's big enough to pass the wick. Once the bulb is hollowed, glue a couple of small magnets inside the bulb. When it's dry, fill it with paraffin oil. Screw the base back on and place it on the wooden slab. □



Project Maker
DANIEL WINKER

Project Link
hsmag.cc/rOKfTV

Above The bulbs rest on a wooden base in which you've drilled holes and placed magnets

BULB BUGS

Ceshire-based jewellery designer Fern uses broken light bulbs in a very creative fashion. With a simple aluminium wire she transforms the bulbs into a range of bugs and insects that can style all kinds of tables. Begin by first straightening a 1.5 mm-thick length of aluminium craft wire, with your fingers or using pliers. To create the hind legs, wrap some length around the base of the bulb in such a fashion that you end up with two legs sticking out at the ends. Bend these downwards at an angle part-way, and use pliers to spiral the ends for the feet. Similarly, create the forelegs by wrapping another

length across the base, this time bent forwards before being bent downwards. Finally, use the pliers to create the loops for the feet. Add a pair of feelers in the same fashion to complete the look. □

Right Fern has shared a couple of designs for other insects on her Instructable, along with a video



Project Maker
FERN

Project Link
hsmag.cc/sGkzFr

HackSpace

TECHNOLOGY IN YOUR HANDS

Download the app

Out now for smartphones & tablets



£2.29

rolling subscription

£26.99

subscribe for a year



FORCE

HACK | MAKE | BUILD | CREATE

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

PG
78



DRILLS

How to make holes
in (almost) anything

PG
82



WHEEL OF FORTUNE

Alexa! Spin the wheel
and reveal my fate

PG
86

RUBBER STAMP

Laser-cut your own
personalised ink-stains

PG
88



PERSONAL TRACKER

Let loved ones know when
you'll return from work or Mordor

PG
66

SCHOOL OF MAKING

Start your journey to craftsmanship
with these essential skills

66 Arduino: Interrupts

72 Electronics 101: Diodes

PG
94

SKULL BOWL

Protect your sweeties with
the curse of the monkey skull

PG
98

LAMP

Illuminate your home –
steampunk style

PG
102

NUMBER-PLATE READER

Let your smarthome
prepare for your arrival

PG
108

LEARN AI

Build a robotic botanist
with a Raspberry Pi

Arduino programming: Sound, envelopes, and interrupts

Build a simple sound generator that allows you to control the pitch or timbre of the sound over time



Graham Morrison

@degville

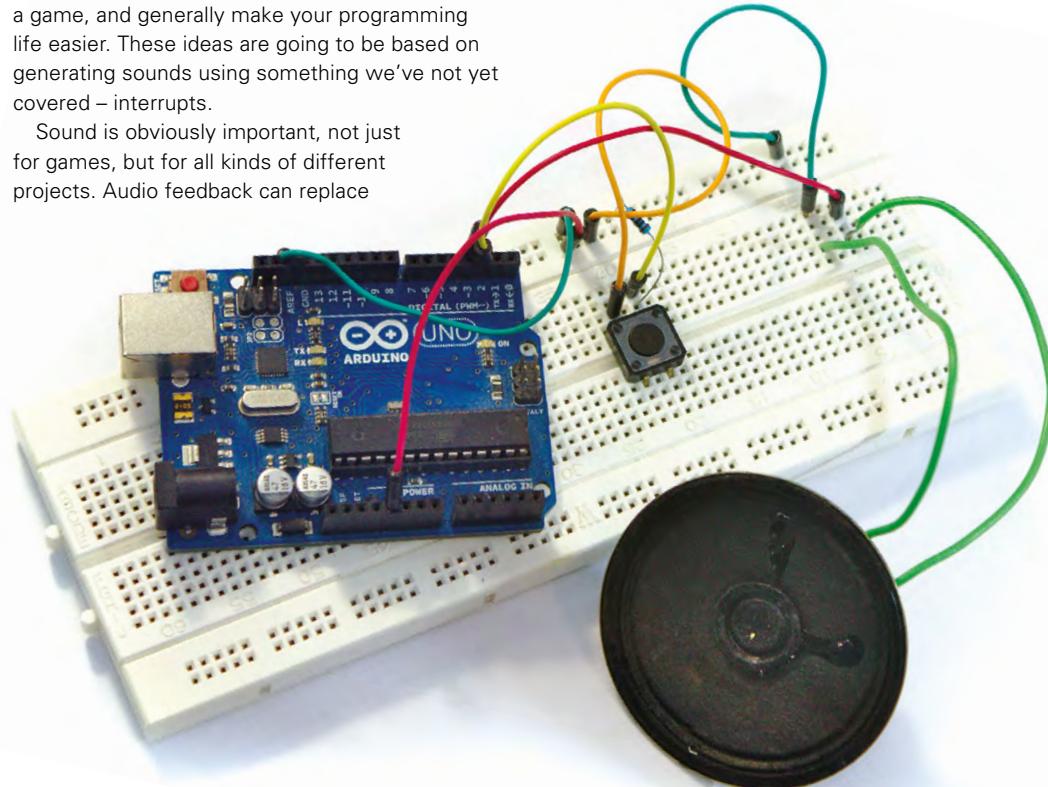
Graham is a veteran Linux journalist who is on a life-long quest to find music in the perfect arrangement of silicon.

We spent the previous couple of tutorials putting some of our programming theory into action, creating a simple game where the player flew a craft through

an ever-decreasing tunnel. In this tutorial, we're returning to practical theory, but we're going to explore some ideas that could be used to expand a game, and generally make your programming life easier. These ideas are going to be based on generating sounds using something we've not yet covered – interrupts.

Sound is obviously important, not just for games, but for all kinds of different projects. Audio feedback can replace

the need for a visual element, such as a screen, and sound can be more intuitive and accessible. You don't need to explain the UI of an audible alert or alarm, for example, and if the sound is annoying enough, it can demand your attention in ways an on-screen notification can't. But the best thing about sound is that it's incredibly cheap and easy to implement. Even a basic Arduino with no specific



Right ♦

You can use almost anything you have around to generate sound from some input trigger with an Arduino



Left A speaker is remarkably resilient, and can sound good even when in a poor state of repair

audio hardware, like the Uno we're using for our projects, can generate sound, because sound is generated by moving a speaker coil using nothing more than fluctuations in current.

The trigger to start the sound could be almost anything. An in-game event, for example. But for our purposes, and to make this project standalone, we're going to use an equally simple momentary switch or button. When the button is depressed, we'll generate the sound. When it's released, we'll stop the sound. Two things are going to make this different to how you might expect. The first is that we're going to use an interrupt to automatically wait for the button state to change, and the second is that we're going to modify the sound as it's being played. This is called 'modulation', and it's essential if you want your sound to be more interesting than a simple beep.

INTERRUPTS

Up until now, we've used the ever-running `loop()` function to look for changes in the state of things we wanted to monitor. If a button is pressed, or a joystick pushed, a variable would change and we could safely assume an event had taken place. This approach is typically called 'polling', because we're constantly waiting and watching, looking for

“ An interrupt allows the programmer to define a function to run when there's a change in state without manually waiting for it ”

a value to change. Polling is a great solution on an Arduino because the device is always on, always running at full speed, and always iterating through `loop()`. Adding extra checks, or polls, shouldn't add to the overall processing burden. And if it does, it's something the programmer can manage by carefully prioritising those checks, or reducing the frequency of less important checks.

But there are strong use cases for not continually checking for changes in state, and instead waiting to be informed that something has changed. This is what an interrupt does. An interrupt allows the programmer to define a function to run when there's a change in state without manually waiting for it. Just like tapping someone on the shoulder, an interrupt is often triggered faster than the equivalent polling code, and the amount of time it takes to respond to an interrupt is more predictable. Polling response times can be unpredictable. It could be that you check for changes in state just as →

QUICK TIP

As you might imagine, one thing you can't do within the function triggered by the interrupt is wait. `delay()` won't work because the function is being executed outside of the main loop, and `millis()` won't be incremented either.

SCHOOL OF MAKING

Below  Momentary switches only stay connected as long as the user presses the button



Below  We salvaged our speaker from an old PC, but they're easy to find by taking apart almost anything that used to make a sound



something has changed, and the response will be fast. Or something changed just after the previous check and won't now be serviced for a longer duration. This induces jitter, which is variations in the delay between when something happens and when your code can respond to it. Of course, we're talking about differences in milliseconds, but it can make a difference in time-critical situations, or when jitter can be easily detected such as with strobing lights or audio playback.

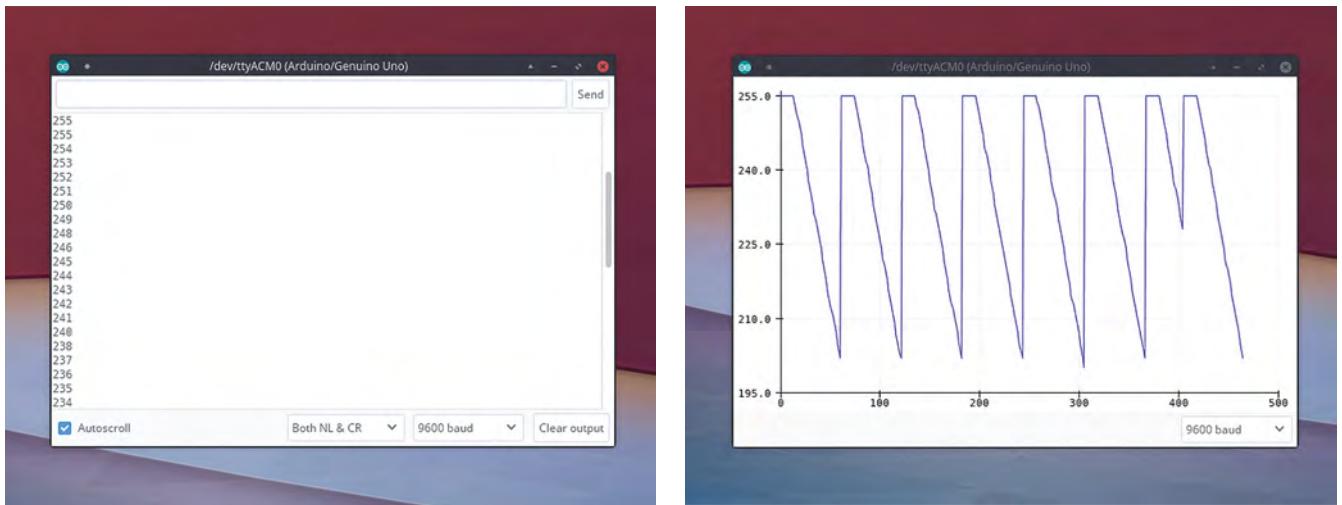
Let's get started by writing the code for the interrupt:

```
const int interruptPin = 2;
const int piezoPin = 3;
unsigned long note_time;
bool trigger = false;
void setup() {
    attachInterrupt
    (digitalPinToInterrupt(interruptPin),
    triggerSound,
    CHANGE);
}
```

All we're doing in the above chunk is first declaring a global constant variable to hold the value of the pin connected to our button, and then using this value within `setup()`. We also create an 'unsigned long' variable to hold up to 4 bytes of data with

“attachInterrupt” is the important part, as this is the Arduino magic that tells your hardware to automatically launch a function

no negative numbers, which we'll use to hold a timestamp, and a `bool` to hold the press state of the button. `attachInterrupt` is the important part, as this is the Arduino magic that tells your hardware to automatically launch a function, `triggerSound`, when it receives a signal corresponding to the final argument in the `attachInterrupt` function call. We've gone for `CHANGE`, as this triggers the interrupt when the button is being pressed and released. We could also have used `RISING` to trigger the interrupt when the button is pressed and `FALLING` when the button is released, but we can handle both of those states with `CHANGE` without using our one remaining interrupt, as we'll show. There's also `LOW` (and `HIGH` on selected



boards) to trigger the interrupt when the input changes to that particular state.

The `triggerSound` function that's called by the interrupt is actually very simple:

```
void triggerSound() {
    if (trigger = !trigger) {
        note_time = millis();
    }
}
```

As we're detecting a change in the button state, and not whether it's being turned on or off, we use a Boolean called `trigger` to flip between `true` when the button is pressed and `false` when the button is released. This isn't obvious in the above code, and we're perhaps guilty of needless obfuscation here, but the `if (trigger = !trigger)` line is both the assignment and the comparison. This isn't a comparison, using `==` or `!=`, as you'd usually expect to see with an `if` statement, it's actually an assignment. We're assigning the `not` value of `trigger` to `trigger` because the exclamation is the `not` operator. This makes `not true = false` and `not false = true`. If `trigger` is true after the assignment, the `if` statement will see the expression resolve as true and `note_time = millis();` will be executed. This line adds another new command, `millis()`, which assigns the number of milliseconds the Arduino has been powered on to `note_time`, the unsigned long variable we created earlier.

PLAYING A SOUND

Playing a sound on an Arduino is remarkably easy, partly because there's a built-in function, `tone()`, so you don't need to worry about pitch and partly because all the Arduino has to do is send pulses of →

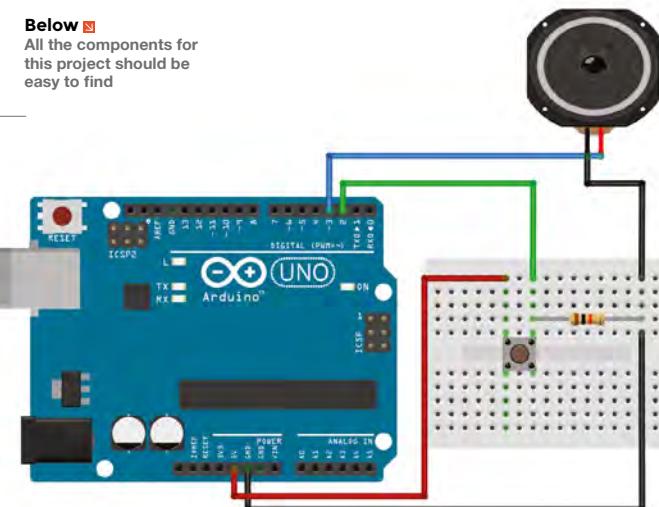
HARDWARE

The great thing about this project is that you likely already have everything you need. You can use almost any old speaker, for example, although the better the speaker, the better the quality of sound – we took one from an old PC. You could also use a small piezo buzzer, often found in component kits. The sound output isn't so good, but the Arduino isn't exactly capable of high quality anyway. It's connected to pin 3 of the Arduino and ground, but if you find the output is too

loud, place a resistor between the positive connection and the Arduino. The higher the resistance, the lower the volume.

Similarly, we plundered an old component box to find a momentary switch to use. One side of this switch is connected to both digital pin 2 on the Arduino and a 10 kΩ resistor, which is itself connected to ground. The other side of the switch is connected to the 5 V pin or rail from the Arduino. And that's all there is to this circuit.

Below █
All the components for this project should be easy to find



SCHOOL OF MAKING

current to the pin connected to the speaker. It can all be done with a single line, which we'll place within its own function:

```
void playSound(int pitch) {  
    tone (piezoPin, pitch);  
}
```

We'll pair the above function with another to turn the sound off:

```
void stopSound(){  
    noTone(piezoPin);  
}
```

All we now need to do is write the simple `loop()` function to trigger either the `playSound` function or the `stopSound` function depending on the state of the `trigger` Boolean:

```
void loop()  
{
```

Below ♦
A piezo buzzer is a handy component. It works reasonably well as a speaker and can also be used as a crude microphone

```
if (trigger) {  
    playSound(261);  
} else {  
    stopSound();  
}
```

If you now run all the code we've just written, you should find that your Arduino generates a tone at pitch equivalent to a middle 'C' on a piano keyboard. But this is only part of the project, because a simple tone isn't all that exciting. To solve this, we're going to change the sound during playback using something called an 'envelope' to modulate the playback pitch.

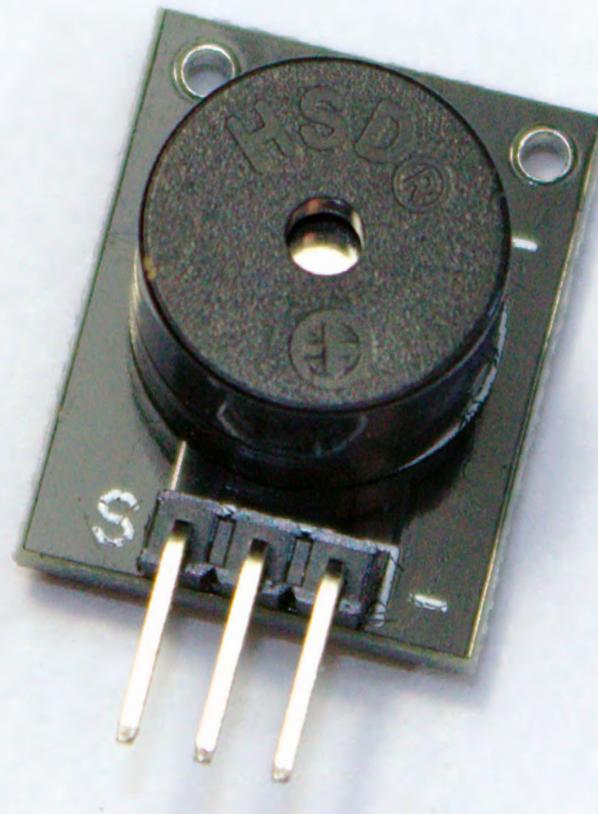
An audio envelope describes how much a sound changes over time, from the moment it's triggered to when it's released. Envelopes are typically used to change the amplitude and pitch of a sound over the duration of a note, and the most common envelope type consists of four stages: attack, decay, sustain, and release, also written as ADSR. Attack, decay, and release are time durations that indicate how fast or slowly the audio changes, whereas sustain is a level that is held while the note is being triggered.

ENVELOPE GENERATOR

Before we start creating our own envelope, we need to add a few global variables:

```
const int pitchEnv[] = {500, 250, 200};  
const int pitchMax = 255;
```

The array is going to hold the attack, decay, and sustain values, with the first two being durations and the final element being level value. As we're going to use this



“
Envelopes are typically used to change the amplitude and pitch of a sound over the duration of a note
”

envelope to vary the pitch of our sound, we've called it `pitchEnv`, along with `pitchMax` to hold the maximum value (amplitude) we want the envelope to reach on the initial attack. Apart from its name, though, there's no reason why the envelope can't be used to control any other audio-related value to modulate the sound. Before we write the envelope generator code itself, we need to patch the envelope effect into our current code. This is as simple as adding the following to the beginning of the `playSound` function:

HARDWARE INTERRUPTS

ARDUINO	INTERRUPT PINS
328-based, Uno, Nano, Mini	2, 3
32u4-based, Micro, Leonardo	0, 1, 2, 3, 7
Due	all digital inputs
Uno WiFi v2	all digital inputs
Zero	all digital inputs except 4
Mega, Mega2560, MegaADK	2, 3, 18, 19, 20, 21
MKR boards	0, 1, 4, 5, 6, 7, 8, 9, A1, A2

Arduino interrupts work at the hardware level and can respond to detected changes on specific pins, such as the rising or falling signal you get by pressing a button, but which pins you can use is restricted, and different Arduinos support different numbers of pins. On our Uno (and other 328-based Arduinos), pins 2 and 3 are the only two capable of generating interrupts, and we've settled on pin 3 for the button connection. But to get pin 3 to generate interrupts requires an extra step we wouldn't ordinarily take, and that's to convert this pin number into an 'interrupt number'. This is because most Arduinos support a restricted number of interrupts, just two on the Uno, and the number for each interrupt won't necessarily align with the pin being used to generate the input. If your project needs more interrupts, the best thing to do is upgrade your Arduino.

QUICK TIP

Just like writing English, the simple approach should always be taken when writing code, even if you know it can be compacted. This makes it much easier for other programmers, and your future self, to understand.

```
pitch += envMod();
```

The above operator is adding the value returned from the `envMod()` function we're about to write to the current value of `pitch`.

```
int envMod() {
    unsigned long current_dur = millis() - note_time;
    if (current_dur <= pitchEnv[0]) { // Attack
        return ((pitchMax * (100 * current_dur) / pitchEnv[0]) / 100);
    } else if (current_dur <= (pitchEnv[0] + pitchEnv[1])) { // Decay
        pitchEnv[1] / 100;
        return (pitchMax - (pitchMax - pitchEnv[2]) * (100 * (current_dur - pitchEnv[0]) / pitchEnv[1]) / 100);
    } else { // Sustain
        return (pitchEnv[2]);
    }
}
```

The above code is complicated, so we'll break it down into parts. It starts off by taking another timestamp for when the function is being run. By subtracting the note's start time, which we saved earlier, and by using the time values in the envelope array, we can calculate which stage of the envelope we should be in. This is what the `if` and `else` statements are doing, with the first simply checking to see whether the time is less than the time of

the attack stage, and the second whether the time frame is between the attack and the end of the decay. If it is, we have a long calculation that does the following:

1. Calculates current time frame as a percentage of the whole stage
2. Returns a percentage of changing value

We both multiply by 100 and divide by 100 in the expressions to keep the end values as integers and avoid floating point mathematics, which is a lot slower and resource hungry on an Arduino. With the attack stage finished, the next `if` deals with the release stage. Finally, if we're in the sustain stage, we simply return the sustain value from the array.

With that function written, you can now re-upload the project to your Arduino. When you press the button, the pitch of the sound will now change according to the durations and sustain level of the envelope, making the sound much more dynamic and interesting. You could even build this into a synthesizer, adding potentiometers to control the values for each stage of the envelope, or adding more modulation envelopes to control amplitude, or even pulse-width modulation. But that's another story.

The code for this project can be downloaded from hsmag.cc/sEgZSN. □

Electronics 101.4: Diodes

Electricity's one-way street



Dave Astels

daveastels.com

Dave's career started in the 8-bit days, with the Z80 and 6502, and he's been working with computers ever since. Check him out at daveastels.com and learn.adafruit.com

S

o far we've looked at resistors, capacitors, and inductors. These all work in a pretty straightforward way. With this part of the series, we're going to venture into something quite different: semiconductors. We'll start with the simplest: the diode.

Diodes are physically simple, yet electrically very interesting. Resistors, capacitors, and inductors are what we call linear components. A resistor is the classic example: double the voltage across it and the current through it doubles.

A diode is a non-linear component; the first we've had a close look at (the LED is, in fact, a diode, and we used one in issue 9). It's also the first semiconductor we've had a close look at.

A diode is polarised. If you recall, resistors are not polarised; it doesn't matter which way you connect them. We found that some capacitors are polarised, and some aren't; it depends on how they are constructed. Diodes are always polarised. You may have noticed that LEDs only work one way around. That's because they're diodes.

First, how is it non-linear? Have a look at **Figure 1**. It shows the current vs. voltage for a resistor. You can see that when the voltage is doubled in either direction, the corresponding current doubles. Now look at **Figure 2**. It's the same current vs. voltage curve for a diode. It should be clear that it's not linear. In fact it's a very different behaviour.

Between the two knees of the curve (at V_{bd} and V_d) practically no current flows through the diode.

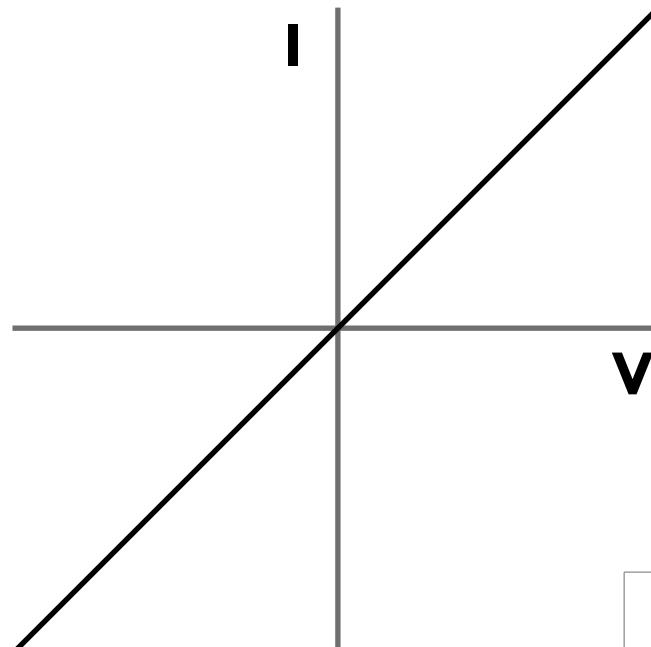


Figure 1 Resistor's linear current vs. voltage curve

YOU'LL NEED

- ❖ Solderless breadboard
- ❖ Two different voltage power supplies (e.g. 5 V and 3.3 V or battery holders with a different number of AAs)
- ❖ A few 1N4001 and 1N4148 diodes
- ❖ Some LEDs
- ❖ Various resistors, from $100\ \Omega$ to $1\ k\Omega$
- ❖ Jumper wires, etc.

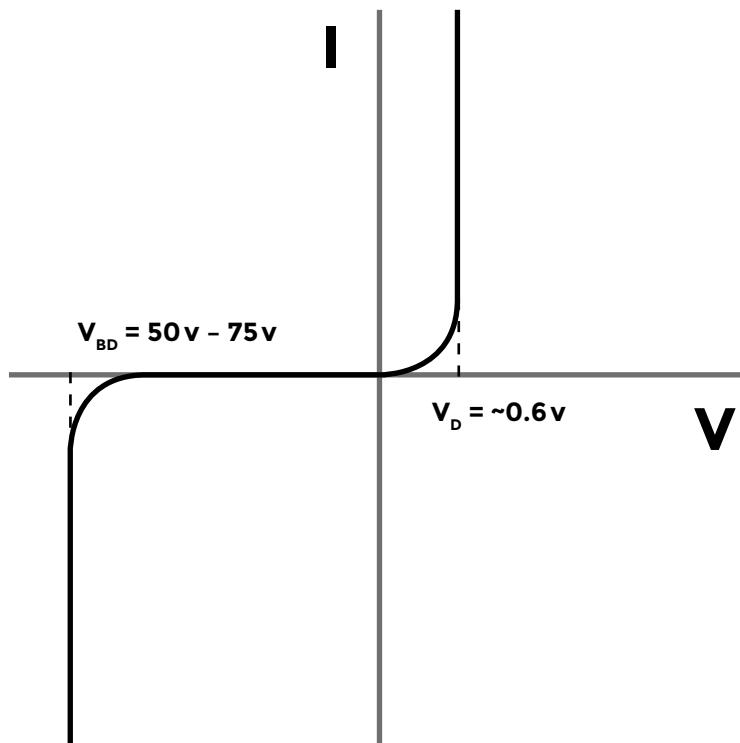
Once the anode is at least V_d more than the cathode (typically around 0.6V), current starts flowing. This is referred to as forward-biasing the diode. Notice on the graph that the current isn't really bounded: as much can flow as needed. The only limit is that the diode will eventually melt from the heat generated. In practical applications, you'll need to include some other components (such as a resistor) to limit the current and prevent the diode from melting. The other knee of the curve is when the anode becomes overwhelmingly more negative than the cathode. When the anode is more negative than the cathode, it's called reverse biasing, and when it reaches that knee (generally -50 to -75V), the diode breaks down (hence the name of the voltage at that point: V_{bd} is the breakdown voltage) and current can flow in the opposite direction. This is generally considered to be not a good thing (except for a Zener diode, as we'll discuss shortly).

The result of all this is that diodes have the incredibly handy property of letting current flow in only one direction when they are forward biased.

ONE-WAY STREET

Technically, what we are discussing is silicon P-N junction diodes. **Figure 3** shows a representation of what a diode is structurally. It's simply two pieces of silicon jammed up against each other, with an external connection on opposite ends. Those connections are the anode (on the P piece) and the cathode (on the N piece). What's that P and N business?

We discussed conductors and insulators in issue 9. With diodes (and other components we'll discuss in later parts) we have something different: semiconductors. As you might expect from the name, they have a conductivity in between conductors (like metals) and insulators (like glass). Silicon is probably the best known semiconductor material, but there are others. By adding impurities to the silicon, different properties can be realised.



Ignoring the chemical details, we generally have two types: N and P. The N-type has extra electrons that can wander around the crystalline structure of the silicon. These are called free electrons because they aren't locked into silicon atoms. The P-type has free holes. You can think of a hole as the absence of an electron.

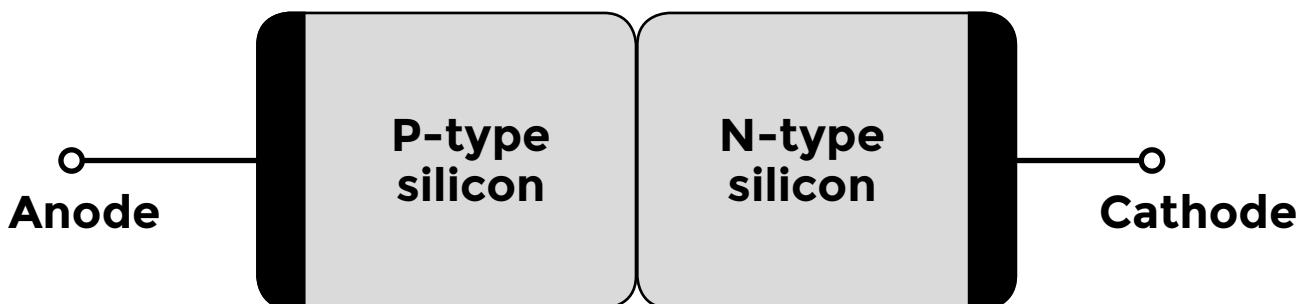
The junction between the types (see **Figure 3**) creates a barrier to the electrons and holes: very few have enough energy to make the jump across. This creates what's called a depletion zone. When we forward-bias the junction, putting a positive voltage on the anode (the P-type), it causes electrons to be pulled toward the junction, and (due to the relatively negative voltage on the N-type) holes to be pulled towards the junction from the other side. The result is that the depletion zone gets narrower. When the voltage difference is big enough (that 0.6V value), the depletion zone disappears, and electrons →

Figure 2 ◇
Diode's nonlinear current vs. voltage curve

Credit
From Wikipedia by Hldsc CC BY-SA 4.0.v

Figure 3 ◇
The structure of a PN-junction diode

Credit
Licensed under GFDL 1.2.



SCHOOL OF MAKING

and holes can flow freely across the junction. Reverse-biasing the junction causes the depletion zone to become wider, stopping any current flow. However, if the reverse-bias voltage is too high, the junction will heat up enough to cause what's called an avalanche effect, which causes the diode to essentially become a short circuit, allowing current to flow in the opposite direction.

“**Chances are, even if you think you've never used a diode before, you've most likely used a light-emitting diode, aka LED**”

TYPE CAST

There are a few different types of diode. Here are some of the most popular:

Signal diodes

This is the basic diode that we've been discussing. It is meant for low voltage, low current use. These are typically small glass components, as shown in **Figure 4**.

Power diodes

Power diodes work identically, but are designed to handle higher currents. You'll find these used for back EMF protection (see below) and rectification in power supplies (also see below). These are larger and typically in heavy plastic or metal bodies, as shown in **Figure 5**.

Zener diodes

Zener diodes have more precise breakdown voltage. In fact, there are Zeners available with one of a large number of breakdown voltages. All diodes have

approximately the same forward-biased voltage drop (~0.6 V), whereas you can get a Zener that has pretty much whatever breakdown voltage you want. You can use a Zener along with a current-limiting resistor (that acts like a constant current source) to roughly regulate to its breakdown voltage. See **Figure 6**. The problem is that the precise breakdown voltage is sensitive to the current and temperature. And they tend to be noisy (i.e. the regulated voltage isn't very smooth). This noise is very random and can be used (along with an analogue to digital converter) to make a true random number generator. The exception to these downsides are Zeners with a breakdown voltage of around 6 V, which are much more stable than other voltages.

Light-emitting diodes

Chances are, even if you think you've never used a diode before, you've most likely used a light-emitting diode, aka LED. This is simply a diode (hence the voltage drop across it that figures into the choice of current-limiting resistor) that emits light (visible or otherwise) when current is flowing through it. The exact impurities used to create the N and P parts of the diode determine the colour (i.e. wavelength) of the emitted light (as well as how much voltage it requires). The brightness of the LED varies based on how much current (on average) is flowing through it. I say 'on average' because using a PWM (pulse-width modulation – a way of rapidly switching a digital pin on and off) signal is the usual way to digitally control the brightness of an LED.

Photodiodes

These can be thought of as the opposite of LEDs. They are diodes that will turn on (i.e. allow current to flow through them) when bombarded with enough



Figure 4 ◇
A 1N4148
signal diode



Figure 5 ✎
A 1N4002
power diode

LED EXPERIMENTS

Hook up an LED in series with a resistor of about 300 ohms, as shown in **Figure 7**. The LED should light. Now change the value of the resistor (either by using a different value resistor or adding other resistors in series or parallel). Be careful when making the value lower. If you go too low, the current may be too much for the LED to handle: it might flash brightly and be destroyed, possibly with a SNAP! Compare the brightness of different coloured LEDs when using the same resistor values.

photons. That means if they have light shining on them, they conduct. In the dark, they don't. The materials used to create the N and P type sections will determine what range of the spectrum the diode is sensitive to.

DIODES LOOSE IN THE WORLD

That's what they are, but what do these electrical one-way streets allow us to do? Here are a few of the most popular uses:

Power rectification

The electricity available at your mains outlet is AC. The details vary depending on what country you're in, but it will be AC, and it will be 120, 220, or the like, at a frequency of 50 or 60 Hz. Far from the 5V or 3.3V or even 12V DC we need to power our circuits. With an AC signal, the voltage alternates between positive and negative. In contrast, with a DC signal it's all one (usually positive). Furthermore, a DC power supply has to be stable (always the same voltage) and smooth (having minimal noise, or jitter).

There are three things that need to happen between the outlet and your circuit:

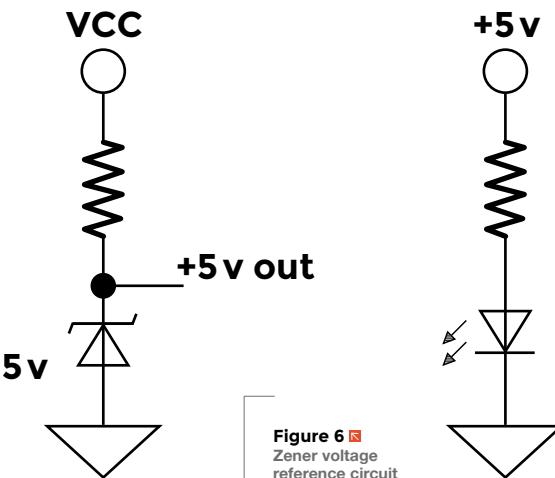


Figure 6 Zener voltage reference circuit

Figure 7 Hooking up an LED

1. The voltage needs to be lowered. A power transformer (see issue 11) can be used for that.
2. AC has to be converted to DC. That can be done with diodes, as we'll see momentarily.
3. The DC voltage needs to be regulated to be a smooth, consistent value. We'll look at that in a future issue.

Power diodes can be used to convert AC to DC. There are a couple of ways to do this. **Figure 8** shows the circuit, input, and output of a half-wave rectifier – so called because only half of the AC waveform is used. The negative part is discarded (since it's blocked by the diode). This is wasteful, but more importantly, the resulting voltage is very variable. By adding more diodes (they're cheap, so why not?), we can capture that negative half of the input. **Figure 9** shows this. It's called a full-wave rectifier, also referred to as a bridge rectifier. This is standard in simple linear power supplies. While most power supplies these days are of the switching variety, if you enjoy working with retro hardware, or salvaged circuitry, you will no doubt run into linear power supplies.

Remember when we talked about capacitors in issue 10? We mentioned that they can be →

QUICK TIP

If you have an oscilloscope, you can try wiring up the circuit in **Figure 11** with different value capacitors and resistors, connecting the input to a square wave (e.g. a PWM output of a microcontroller) and observing the waveforms.

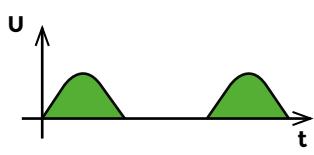
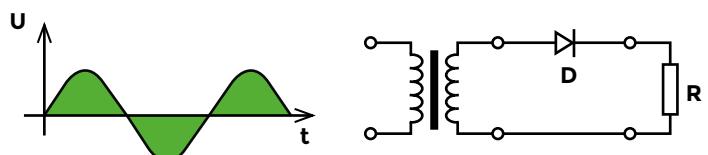


Figure 8 Half-wave rectifier

Credit
By Wikipedia user Wdwd, licensed under CC BY 3.0

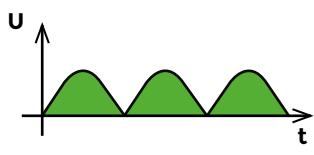
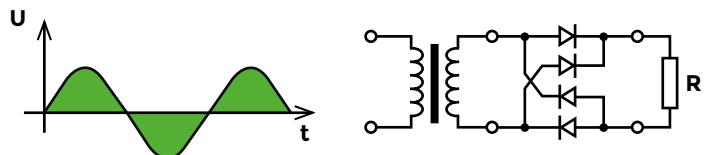


Figure 9 Full-wave rectifier

Credit
By Wikipedia user Wdwd, licensed under CC BY 3.0

SCHOOL OF MAKING

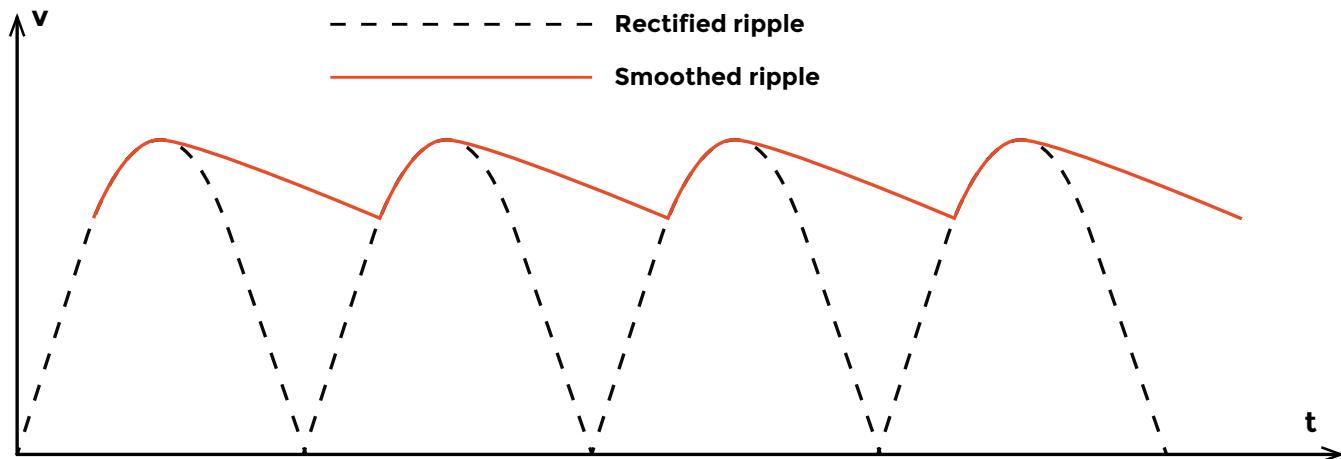


Figure 10 ♦
The effects of power-smoothing capacitors

used to smooth signals: i.e. reduce fluctuations. The fluctuations coming out of a rectifier (even a full-wave) are pretty significant. That's why big capacitors are needed. They store a lot of charge, and discharge slowly. **Figure 10** illustrates the effect of adding a large capacitor to the output of a full-wave rectifier.

Rectification is handy in lower voltage signal processing as well. Let's say you have a square-wave signal and want to pulse every time the input goes from low to high (i.e. a pulse on each rising

edge). If you feed the signal into a differentiator circuit (see **Figure 11**), you will have a positive pulse on each rising edge, as well as a negative pulse on each falling edge. Remember that we said that capacitors basically block steady voltage and let changes through. This circuit takes full advantage of that property to pick out the edges of the input signal. However, those negative pulses could cause problems, so we put that through a diode, which will let just the positive pulses through.

Back EMF protection

In issue 11, we discussed inductors, and in particular electromagnets, solenoids, and motors. Recall that we used a diode in parallel with the coil such that it is usually reverse biased. When the power to the coil is turned off, a reverse current is induced that could result in a very high voltage across it. A diode is used to drain that current off and reduce that voltage harmlessly. Otherwise the controlling circuit (typically a transistor) will quite likely be destroyed. **Figure 12** shows the typical use.

Figure 11 ♦
Rectifying a differentiated square wave to get rising edge pulses

POWER GATE EXPERIMENTS

To show this in operation, you'll need two power sources of different voltages. You can use 5V and 3.3V breadboard power supplies or two different batteries (with different voltages... say, a 9V and 2-3 AAAs). On your breadboard, put together a circuit similar to the one shown in **Figure 13**: you just need the two diodes with a different power source on each anode. The cathodes connect together. You can add a load resistor of $10\text{ k}\Omega$ or so to draw some current. Connect a voltmeter to the cathodes and watch the reading as you connect and disconnect the power sources.

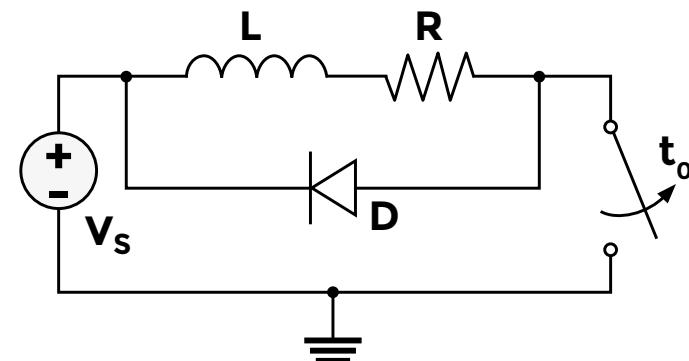
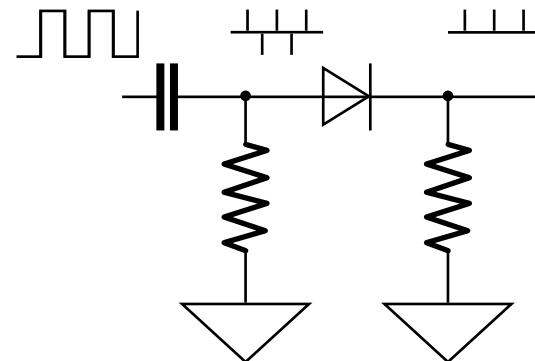


Figure 12 ♦
A diode used to protect against the current/voltage generated when the coil is switched off

When you are working with inductors, remember to either use a driver chip that includes the protective diodes (like the ULN2803), or add them yourself.

Power gates

A good example of this use is a real-time clock (RTC) circuit, like the one in **Figure 13**. When the power is on (i.e. Vin is, say, +5V), the RTC gets power from it. When the power is off and Vin is 0V, the RTC needs to be powered in order to maintain its time-keeping. This is generally done by using a small battery. Diodes can be used to automatically switch

of what IN attempts to do. The resistor serves to limit the current through the diode when it's conducting.

A practical use of diode clamps is on the inputs of static-sensitive integrated circuits (as part of the IC's circuit).

Diodes are the simplest semiconductor device. Even so, they are incredibly useful. The concepts behind their operation are at the foundation of modern digital electronics. In issue 13, we'll look at a slightly more complex, yet vastly more useful device: the transistor. □

Figure 13 ◊ Simple battery backup power switching using diodes

Credit
From StackExchange, licensed by CC BY SA

“ **Diodes are the simplest semiconductor device.**

Even so, they are incredibly useful **”**

between the two power sources. The backup battery is a lower voltage than the main supply, so when the power is on, the cathode of its diode will be at a higher voltage than the anode, reverse-biasing the diode, and keeping it 'off'. When the main power is disconnected, the battery diode's cathode is no longer at that higher voltage and it becomes forward biased, allowing current to flow and power to be supplied to the RTC from the battery.

Clamps

The final use of diodes we'll look at is the voltage clamp. This is handy when you want to make sure that a signal never exceeds a certain voltage.

Figure 14 shows the most basic form of this.

The diode is reverse biased and inactive as long as IN is below 5.6V (5.0V that the diode's cathode is connected to, plus the 0.6V drop across the diode, itself). As soon as IN goes above 5.6V, the diode becomes forward biased and conducts. As long as this is the case, its anode stays at 5.6V regardless

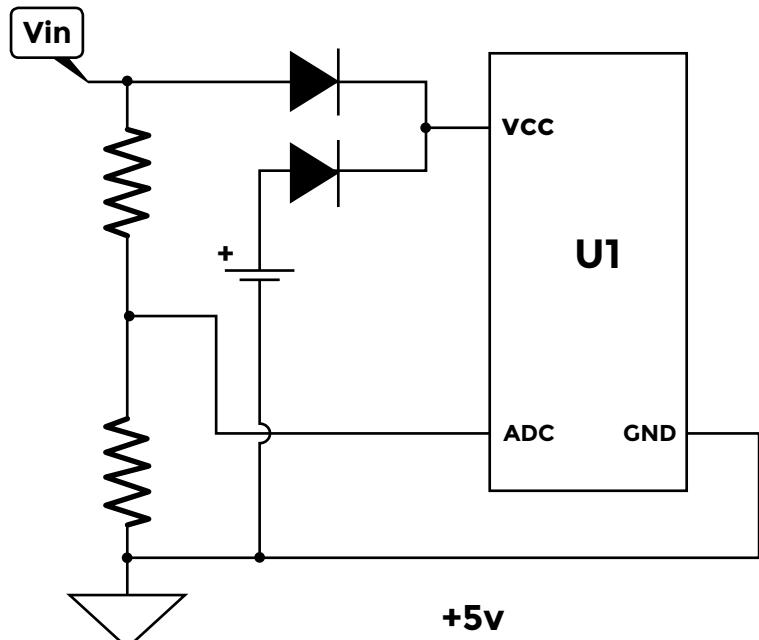
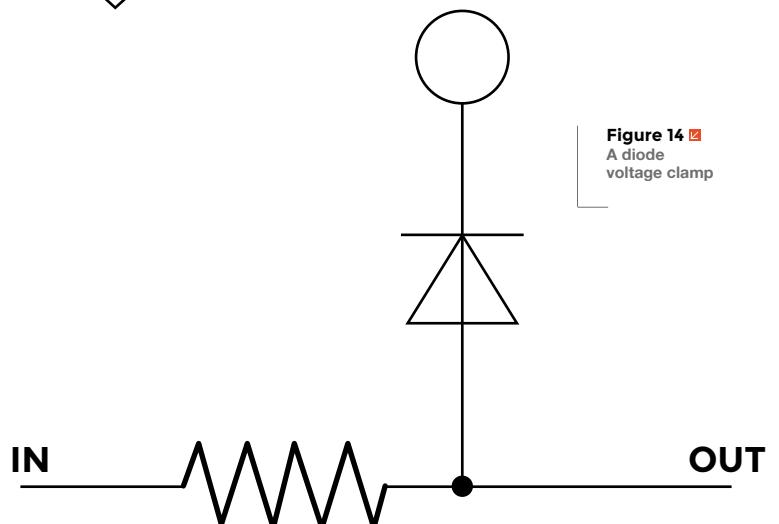


Figure 14 ◊ A diode voltage clamp



CLAMPS IN PRACTICE

Wire up the circuit in **Figure 14** using a 1kΩ resistor. With a voltmeter connected to OUT, vary the input voltage. Using 3V, 5V, or 9V will be adequate to see the operation. If you happen to have a variable supply, you should be able to see OUT stop increasing when IN reaches 5.6V.

Maker's Toolbox: Hand drills

A basic guide to the DIYer's hole-maker



Ben Everard

@ben_everard

Ben loves cutting stuff, any stuff. There's no longer a shelf to store these tools on (it's now two shelves), and the door's in danger.

If you've only used one power tool, the chances are that it's a power drill. This staple of DIYers and makers is now so commonplace, that it's hard to imagine life without it.

At the very basic level, a power drill has a motor that spins a chuck. In this chuck you can put almost anything but, most commonly, drill bits are used to burrow into whatever material you put in front of them. Despite being fundamentally quite simple, there's quite a lot of variation between different power drills on the market. The basic features you'll come across in hobbyist-level drills are:

Speed

Faster isn't always better – you need to match the speed to the bit and the material being used, but sometimes more speed means getting a job done quicker. On modern drills, the speed is usually variable, but some drills are easier to control than others. However, depending on the power (see below), the drill will slow down as you put force on it, particularly with larger drill bits.

Hammer

This action makes the bit go in and out, as well as round. It's mainly used for drilling into masonry



Right ♦

A selection of drill bits. From left-to-right: twist, brad point, masonry, diamond, Forstner

“ Don’t forget that Newton’s laws of motion mean that you have to be at least as strong as your drill ”

or concrete. Drills with selectable hammers are sometimes known as combi drills.

Electric brake

When you pull the trigger, the chuck starts spinning, but what happens when you release this? Some drills continue to spin with the momentum of the chuck, while others will apply a brake to stop the bit almost instantly.

Handles

Just about every hand drill has a pistol-style handle, but some also have a side handle that can be useful when drilling into stubborn material, particularly when using the hammer function.

Clutch

This limits the drill power. It’s mostly used when using the drill as a screwdriver, to avoid stripping screw heads.

Weight

You may find yourself drilling with the drill held in awkward positions, and heavy drills can take their toll on your arms.

Electrical supply

Drills can be corded or cordless. Corded drills are generally more powerful (particularly at the cheaper end of the spectrum), but obviously limit the places you can drill to those within reach of a power socket. Although cordless drills allow you to go anywhere, they do limit the time you can drill to the capacity of the battery. Having more than one battery means you can drill with one while the other charges. Batteries are measured in Amp hours (Ah).

The most confusing aspect of a drill can be its power. The power of a drill is important, but expressed in different (and sometimes confusing) ways. It’s sometimes given in newton meters, and sometimes in the maximum diameter hole that can be drilled. For cordless drills, the voltage of the battery is sometimes used to imply a particular power and, for corded drills, the wattage of the motor is often given.



The power you need depends entirely on what you want to drill. Even the weakest drill should manage light woodwork and being used as a screwdriver, but if you need to get through tougher materials, and particularly if you want to make large holes, you’ll need more power.

Don’t forget that Newton’s laws of motion mean that you have to be at least as strong as your drill. You can’t keep piling on the torque and expect to be able to hold the drill or you’ll end up like a spinning top, so there’s a natural limit for the amount of power that’s useful in a drill. If you need to drill with more power than you can hold, you’ll need a drill-press or some other way of holding everything in place. Having good quality, sharp drill bits can be as important as having more power on your drill.

As a general rule of thumb, going less than 18V on a cordless drill will limit what you can drill (corded drills are almost always more powerful). →

Above The Milwaukee M18CBLPD-402C has got power and speed in a small package

DRILL BITS

Not all bits are created equal, and getting a drill to do what you want often comes down to putting the right bit in it. There's a mind-bending array of different bits on the market, including the simple and general-purpose twist bit and specialist bits for ceramic and glass. A few of the most popular are:

Twist

This is the most basic drill bit. It's simply a rod with a twisted groove along its length and a sharpened point. It's the original drill bit, but still versatile and can be used on a variety of materials, including wood and metal.

Brad Point

The classic wood drill bit has a spur on the end that ensures that the drill bit goes in exactly where you want it to.

Spade

A wide, flat drill bit for making large holes in wood. This drill bit can be a bit rough, so you can get a cleaner finish by drilling from both sides and meeting in the middle.

Forstner

These bits drill wide, flat-bottomed holes in wood that are neater than those created by spade drill bits. However, these can require a bit more force to create the hole.

Step

A single drill bit to create a wide range of hole sizes. Conical step drill bits can be pushed into sheet metal as far as necessary to create exactly the size hole you need.

Hole saw

A central drill bit guides a circular saw blade into the wood. This bit is capable of creating large holes in wood, but has to drill all the way through.

Diamond

Coated in diamond dust, these bits can get through glass or ceramic, but they need water to keep them cool.

Masonry

Designed for use with hammer drills, masonry bits are designed to chisel out the material.

Screwdrivers

These bits turn the drill into a power screwdriver. The potential problem here is that drills are far more powerful and faster than screwdrivers, so you need to be able to control both of these to make a drill work as a screwdriver. Usually, the clutch can limit the power of the drill, and you should be able to limit the speed with the trigger. It's best to start with a much lower power than you think you need, as it's easy to strip screws.

As well as the general shape of the bit, the material it's made from can have a big impact in how it works, and this generally comes down to the toughness of the metal. High-speed steel (HSS), cobalt steel, tungsten carbide, and titanium nitride can all be used to make bits. You often find that you get what you pay for. Cheap bits will quickly blunt, while bits from a reputable manufacturer will last years, if treated properly.

However, if you know you're only going to need to drill softer materials, the lower voltage drills can be significantly lighter and smaller.

WHAT'S UP CHUCK

Drills hold bits in chucks, which are essentially circular vices that fasten three points together to hold the bit firmly and centrally. Almost all chucks follow this basic design, but some require a key to operate, while some can be operated by hand. It's a little easier to use key-less chucks as there's no risk of losing the key, but don't always grip as firmly on the bit.

Not all chucks are the same size. Generally, in drills for hobbyists, you'll get a chuck either 10mm or 13mm, and this is the maximum diameter shaft that you can fit in. In general, you'll find 10mm chucks in less powerful drills that simply don't have the torque to spin the large bits. This limit is only on the shaft size, and doesn't necessarily relate to the size of the hole created.

Although drills are almost always used for the same function (to make a hole in something), their

ability to make a wide range of holes in a wide range of materials is due to the wide range of bits available (see 'Drill Bits' above).

Finding the right drill
is really about
understanding what
you need it for

UNDER TEST

Finding the right drill is really about understanding what you need it for. A drill for woodwork may need to be small and light, while for more general DIY usage, the hammer function might be important. You might need to be able to use it all day (and so have a cord or a large battery), or just for a few holes at a time. In general, you can get any one feature (size, weight, power, etc.) on a fairly cheap drill, but the more general-purpose you need your drill to be, the more it will cost.

**Above ↗**

The risk of running out of power is the price you pay for portability. Make sure you've got enough batteries to complete the jobs you need to do

We put the Milwaukee M18CBLPD-402C power drill through its paces to see what you're likely to get at the higher end of the market. This is a cordless combi drill that can exert an arm-bending 63Nm of torque. It's got two speed selectors, a low range that adjusts between 0 and 500 rpm, and a high speed option that goes up to 1800. These are selected by a toggle on the top of the tool.

What particularly impresses us with this drill is how easy it is to control. Many drills have adjustable speeds, but trigger-style speed controls can make it hard to accurately get the speed you want – the difference between painfully slow and out-of-control fast is often difficult to hit. The trigger on the Milwaukee drill moves smoothly enough to make it easy to find the speed you want. The drill is also well balanced and easy to hold in a variety of orientations, and it's a short drill, which makes it easy to fit inside awkward spaces.

For a drill this powerful, the Milwaukee is quite light, but at 1.7kg, there's still quite a bit of weight if using it for a lot of holes, particularly if you need to drill above head height. Quite a lot of this weight comes from the whopping 4Ah batteries, which can keep the drill going for quite a long time, (unless you're constantly drilling, you should get more than an 80-minute charge time out of each battery, so two should be enough to drill constantly).

At £199.99, it's not a cheap tool, but for that price, you get a quality drill that can take on most jobs you're likely to throw at it. □

**Above ↗**

A Forstner bit, making a 30mm hole. These bits need quite a bit of power to cut cleanly

Alexa-controlled wheel of fortune

Build your own wheel of fortune that can be controlled via Alexa



Simon Stevenson

[@peusste](#)

Simon Stevenson has a shed and isn't afraid to use it. He loves projects that include making, electronics, and coding.

Figure 2 3D parts to download from hsmag.cc/issue12

1. Back plate
2. Stand
3. Pointer
4. Wheel
5. Frame

U

se an Alexa device to spin a wheel of fortune! It not only tells you what your future holds, but also gives you a random number or answers a yes/no question. All you have to do is say, "Alexa, spin the wheel" (or any voice prompt you want). See the complete wheel in **Figure 1**.

Let's get started. The first thing we'll need is the 3D-printed parts. You can download the printable files from hsmag.cc/issue12 (see **Figure 2**).

The circular frame glues onto the back plate; try to make sure it's central and evenly positioned. Then glue the pointer to the top centre of the frame. Finally, glue the frame to stand (see **Figure 3**).

Insert and secure the servo so the shaft is in the centre of the back plate. Secure the shaft to the back plate using either glue or a couple of small screws. My screws stuck out through the back plate, so I filed them down.

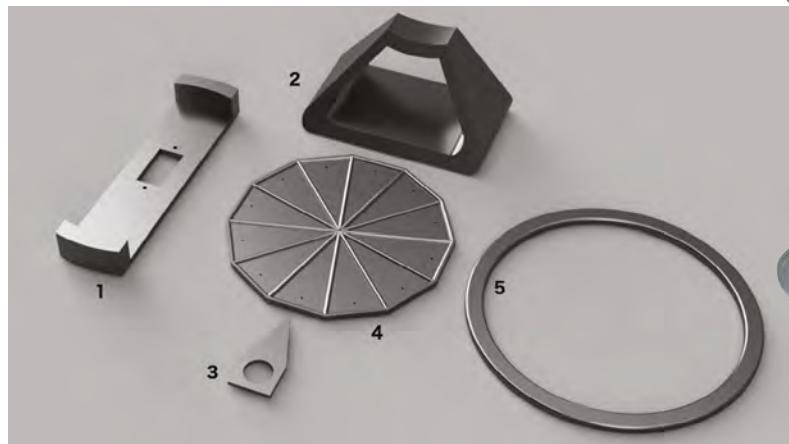


Figure 1 The completed Wheel of Fortune. Dare you ask Alexa to spin it?

THE WHEEL OF FORTUNE

When you say the phrase "Alexa, spin the wheel," the wheel spins for a random amount of time, and when it stops, the pointer will give you your fortune. We can also use the wheel to give us a random number between 1 and 12 (so it can be used as dice for games), or letter A–D (there are four in the author's family, so this is a good way to resolve disputes or to see who goes first in a game). It also has a series of Y/N? symbols, so you can ask it questions, e.g. "should I go to the shops now?"

WHEELY GOOD

Print off a wheel design. You can either use the one we've provided, or design your own.

Ours has twelve segments, but you can really have any number you like. We picked twelve as it meant it was divisible by three and four, so we could get extra features on it like the 'A–D' and 'Y/N?' in even measure. Stick the segments to the 3D-printed wheel or onto a piece of card, 10cm in diameter. Our segments were printed onto adhesive paper and cut out individually, but you could just as easily use a single design if using card.

Stick a servo horn (our servo came with a large round one) onto the back, making sure it is central. There's a small hole in the 3D print to help align this.

SPIN ME AROUND

Push the wheel onto the servo shaft and check it spins without interfering with the frame. You can glue it in place for a more permanent finish.

Connect the servo to the ESP8266 so that the signal wire (orange) of the servo goes to digital pin four, and the positive (red) and ground (brown) go to the corresponding 3.3V and GRN pins next to D4.

We used Velcro to fix the board to the servo, making it easy to remove for programming. Wind the servo cable out of the way, round the servo itself, and you are ready to program your wheel.

IN DISGUISE

We can configure the ESP8266 to pretend to be a WeMo device. WeMo are a series of commercially available lights and switches etc. that can be used for home automation and can be controlled by Alexa. The real magic in this project is the FauxmoESP library. The library, written by Xose Pérez (hsmag.cc/wlgBEG) – based in turn on a Python library written by MakerMusings: hsmag.cc/lzVBWu – allows us to fool Alexa into seeing the ESP8266 as a WeMo device that we can control using voice. The code on the following pages is based on Xose's example code.



Firstly, we have to configure our Arduino IDE to recognise and work with the ESP board. If you don't have the Arduino IDE you can find it here: hsmag.cc/cbJELY.

You'll need to access the board manager from Tools/Board options in the IDE (the board manager appears at the top of the list of boards) and add the library. Full instructions can be found here: hsmag.cc/lgTabu.

You can select the version you install in the board manager. At the time of writing, the current version is 2.4.2; however, we had some issues with Alexa finding the device with the current version, and after some hunting around we found that if you install version 2.3.0 it seems to work fine. At this point, you can upload the blink example code (examples are under the File menu in the IDE) to the ESP8266 to test if everything is correct. If the code compiles and uploads, then all should be fine. →

Figure 3
Fully assembled 3D-printed frame and all the other parts required

YOU'LL NEED

- ◆ **ESP8266 microcontroller**
- ◆ **Continuous rotation 360 micro servo** (I used the SC90 from Olimex)
- ◆ **Wheel frame, pointer, and stand** You could make these out of wood or cardboard if you don't have access to a 3D printer
- ◆ **Micro USB power supply** Like the one you get with a Raspberry Pi, although most phone chargers will probably work
- ◆ **Glue, double-sided tape, or Velcro, and some small screws**

NO PRINTER, NO PROBLEM

If you don't have a 3D printer, this could be made using thin plywood or cardboard. Cut a piece of thick card about 3cm wide and 14cm long. Fold over the card 90 degrees, 1cm from each end, to make the back plate, and cut holes in the back to take the micro servo. Care will need to be given to make sure the servo shaft lines up in the middle. Cut a circle about 12cm in diameter, and cut out the middle, leaving an empty circle about 10.5cm. Stick this to the back plate.

TUTORIAL

QUICK TIP

We used PLA, but any plastic will do, with an infill of about 15%. All pieces are orientated so no support material is required.

FAUX PAS

Instructions for installing the FauxmoESP library via the Arduino IDE can be found here: hsmag.cc/egSrrp. You need to install the two libraries again, and if all is well, there is a FauxmoESP example script that you can test.

Upload the fortune code via the IDE. You can find the code here: hsmag.cc/issue12.

The code works as follows:

Firstly, you need to include the various libraries for WiFi, the FauxmoESP library, and the servo library.

```
#include <ESP8266WiFi.h>
#include <fauxmoESP.h>
#include <Servo.h>
```

You'll need to add your own WiFi credentials for your home router:

Below Servo fitted and secured (making sure the shaft is in the centre)



```
#define SSID "<YOUR SSID HERE>"
#define PASSWORD "<YOUR PASSWORD HERE>"
```

Next, create objects for the Fauxmo device and the servo:

```
fauxmoESP fauxmo;
Servo servo;
```

You only need one variable for the code – **spinning**, which is a Boolean (true or false) indicator of whether or not we have started the wheel spinning.

```
bool spinning=false;
```

In setup, first of all, the code connects to your home WiFi. Then, set the pin for the servo (we used pin 2, which equates to the physical pin 4 that we plugged the servo's signal wire into).

You also need to enable the device and give it a name to identify it to Alexa (we used 'fortune' but you can call it anything that isn't already a device on your network). Continuous rotation servos work differently from ordinary servos, which are usually limited to 180 degrees. As the name implies, a continuous rotation servo can just keep on spinning. To make it turn, you write a value of 0 to the **servo** object and 90 to stop it:

```
void setup()
{
    WiFi.mode(WIFI_STA);
    WiFi.begin(SSID, PASSWORD);
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(100);
    }

    servo.attach(2);
    servo.write(90);

    fauxmo.enable(true);
    fauxmo.addDevice("fortune");
    fauxmo.onMessage(callback);
}
```

The main program loop is quite simple. You have to call **fauxmo.handle()** to keep an eye on the device. If the code sees **spinning** has been set to true, it pauses a random amount of time to let the wheel spin, stop the servo, and set the value of spinning back to false.



```
void loop()
{
fauxmo.handle();
if (spinning){
  delay(500+random(2500));
  servo.write(90);
  spinning=false;
}
}
```

The actual detection of the command to spin is handled by a callback in the setup block. The code to spin the wheel is also fairly straightforward. We just start the servo spinning and set the **spinning** variable to true. This way the servo will keep turning until it is switched off.

```
void callback(uint8_t device_id, const char *
device_name, bool state)
{
if (state)
{
  servo.write(0);
  spinning = true;
}
}
```

ALEXA! SET UP MY CODE

We need to configure Alexa to work with our wheel of fortune.

Make sure you have completed all the above steps, uploaded the fortune wheel code to your ESP8266, and that it is powered on. Using your Alexa app, go to Smart Home in the menu, and click Add Device. The device name you provided in your code should appear. You should now be able to manually switch the device on and off by clicking on it. The WeMo devices we

are emulating are typically appliance sockets or lamps, so can turn both on and off. Our code only needs to detect it turning on, but that does mean you have to turn the device off first before turning it on again. Don't worry, we'll hide all this away in a routine in the next step.

Left Reverse of wheel, showing servo horn glued in place

CREATING A ROUTINE

The final stage is to automate the spinning in Alexa. Go to Routines in the menu, and add a new routine. You can get creative here. I used 'spin the wheel' for voice prompt in the WHEN section. You then have to turn on and turn off the device (fortune in my case), and lastly you can get Alexa to say something whilst spinning! I went with, "Where it stops, nobody knows".

All that is left is to test and spin. I came up with a game with my daughter where we both spun it three times and then used our imagination to come up with a fortune from the three words chosen. Just hope you don't get 'DOOM' three times... ☺

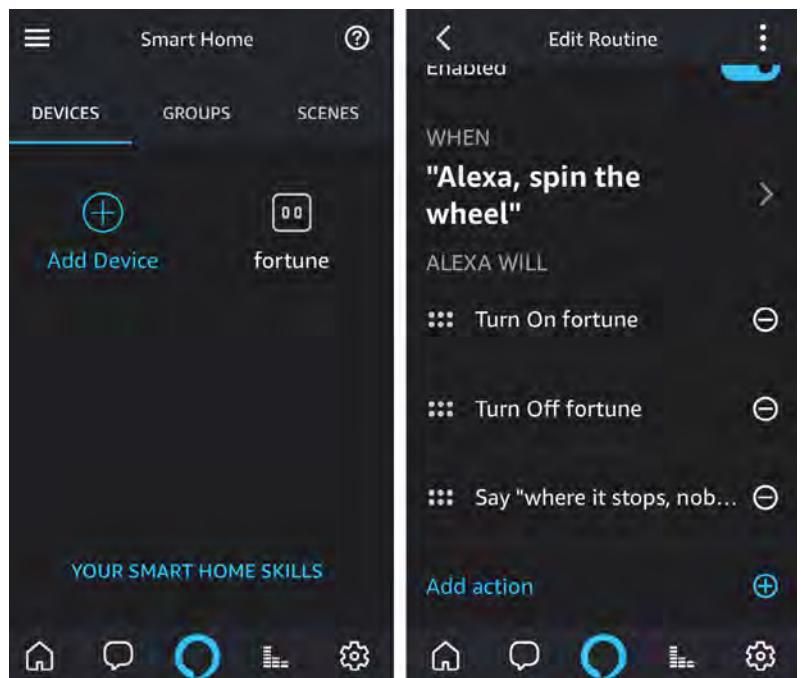
QUICK TIP

There is a handy guide to the ESP8266 pins at: hsmag.cc/YAOAW

TAKING IT FURTHER

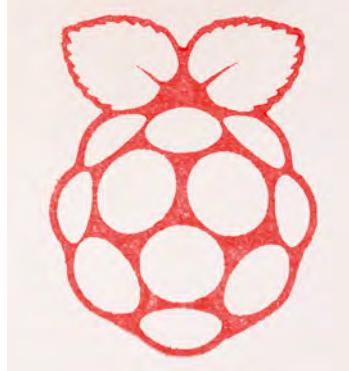
One thing missing from our design is the classic clicker sound of wheels of fortune. You could put a pin in each section and have a clacker mounted on the edge that hits each of these in turn.

Below Adding the device in Alexa (left) and creating the routine (right)



Make a custom rubber stamp

Beautiful, unique, personalised rubber stamps in a couple of hours



Alex Eames

@raspity

Alex Eames loves making things and regularly blogs/vlogs about this at RasPi.TV. He makes a living designing and selling RasPi.IO products. raspi.tv

Rubber stamps are great fun and really useful for great for crafting, decorating, authorisation, event ‘pass out’ verification, and all kinds of short-run print jobs. With widely available access to laser cutters and free design software (Inkscape), you can go from concept to stamping in just a couple of hours.

NEGATIVE THINKING

You’re starting with a flat rubber surface and burning away everything except the ‘printing area’. So, your image needs to be created in negative form. Everything in black will be ‘engraved’ away by the laser. Everything in red will be a cut line (used for the stamp outline).

Everything in white will be the rubber ‘left behind’ that will transfer ink to the paper.

If it’s not horizontally symmetrical, your stamp image needs to be flipped horizontally so it will print the right way round. Lettering should be at least 2–3 mm high or it won’t be resolved.

Create your image, bearing in mind the above design parameters. Save it as a PNG or JPG file.

IMAGE IS EVERYTHING

Open Inkscape (inkscape.org). Select File > Import and choose your image. Leave the import settings at default.

Now define a perimeter cut line for the laser cutter. Click on the rectangle tool (or press **F4**), and drag a rectangle around your design to define the total area of your stamp.

Open the ‘fill and stroke’ pane (**SHIFT+CTRL+F**). Change the stroke paint colour to red (255, 0, 0) and the stroke style to ‘solid 0.2 mm line’. If you want rounded corners on your outline, drag the top-right (circular) drag-handle of the rectangle downwards until satisfied. To fill with black, set the fill colour to black (0, 0, 0). If that makes your graphic disappear, choose Object > ‘Lower to bottom’ to push this black

rectangle to the background. Then Edit > ‘Resize page to selection’ (**SHIFT+CTRL+R**). If you haven’t already done it, now horizontally flip your image, you can do this by pressing **H**.

You should now have your chosen graphic in white within a black (rounded) rectangle surrounded by a thin red line.

Now save your file at full size and then you’ll scale it. To scale your stamp to the exact size you want, Edit > Select All (**CTRL+A**), then lock the aspect ratio of your stamp by ensuring that the little padlock between

“ You’re starting with a flat rubber surface and burning away everything except the ‘printing area’ ”

the width and height adjustment boxes is in the ‘locked’ position (click to toggle it). Change the scale of your stamp to suit your application; e.g. set width to 40mm. The height will scale automatically. Then, Edit > ‘Resize page to selection’ (**SHIFT+CTRL+R**) and save your scaled SVG file with a new name (e.g. MyStamp_30x30.SVG) without overwriting the original.

Below

Lettering should be at least 2–3 mm high or it will just be a blurry mess. The largest here is just about acceptable



QUICK TIP

Some laser cutter software uses different colour schemes for engrave and cut lines, so check which you need before you design your artwork.



LET'S BURN RUBBER

Test your laser cutter settings on a small image or a scaled-down version of your stamp to save wasting lots of material. The settings on our K40 are Engrave: 25% power and 200mm/s, Cut: 25% power at 8mm/s. Yours will be different, but this gives a starting point for experimentation. Air Assist will help here as the rubber will likely want to flame.

Once engraved and cut out, a blast of air or dabbing with a paintbrush will remove any dust or debris.

Then, using the same cut file, cut a piece of 3mm acrylic (17% power, 8mm/s) with the identical

outline to your stamp. This will act as a firm backing, providing even pressure across the stamp. Stick the rubber stamp to the acrylic using either a thin layer of glue or double-sided sticky tape.

Then cut a suitably sized wooden handle 2–4 cm long, with cross-section smaller than the acrylic surface area. Sand the handle edges so they are not sharp or rough. Attach one end to the top-side of the acrylic using glue or tape (a hot glue gun is very quick).

Now your stamp is finished and ready to try out. Enjoy! □

Above

Once completed, you can start decorating and customising everything. Making multiple copies of a stamp for different colours avoids messy cleaning

Below

The acrylic is stuck to the rubber stamp with double-sided tape. Wooden handle to acrylic is fixed with a hot glue gun



STAMPING TECHNIQUE

Press the stamp firmly (but not too hard) onto the ink pad. Remove and press the stamp onto the paper. On a brand new stamp, it may take a couple of iterations until the ink properly wets the rubber. You'll get better results if you don't press too hard, but pressing firmly with a slight rolling action seems to work well. Practice is good. Be prepared to waste some paper and ink learning the best technique.



YOU'LL NEED

- ◆ Computer with Inkscape or Adobe Illustrator
- ◆ Sheet of odour-free laser-cutting rubber
- ◆ Small piece of 3mm acrylic sheet
- ◆ Small scrap of wood
- ◆ Access to laser cutter
- ◆ Ink pad

Way Home Meter

Use an ESP8266 and some NeoPixels to let loved ones know when you'll arrive back home



Brian Lough

@witnessmenow

Brian is a Maker from Ireland who primarily creates projects and libraries for ESP microcontrollers. Check out his stuff on his YouTube channel and blough.ie

“**L**

et me know what time you'll be home – it's a common refrain in homes across the country. We try to give good answers, but it's hard to know how traffic will affect us on the way. Rather than rely on guess-work, let's try to build something to let our families know when we'll make it back.

In this project we'll build a device that will give up-to-date home arrival times, based on the live traffic conditions. To make the device more useful for when it's not being used for that purpose, it works as a clock that automatically fetches its time from the internet and also automatically adjusts for daylight savings.

We've built this using an ESP8266, which is a surprisingly powerful microcontroller with built-in WiFi and can be programmed using the Arduino IDE.

The device makes use of a few different, free internet services:

- **Telegram:** an instant messaging service that allows for the creation of bots that users can interact with. It is a really good way of communicating with your ESP8266 or ESP32 projects from anywhere in the world, for free.
- **Google Maps API:** can be used to get travel time and traffic information between two places.
- **NTP servers:** Network Time Protocol, a way for network-connected devices to get the time. This saves the needs for a real-time clock, and also doesn't require the time to be set.

To use it, the person who is coming home uses Telegram on their phone to share their live location to a Telegram Bot that is running on the Way Home Meter. This will update the Way Home Meter with the person's GPS coordinates every 20 or 30 seconds.

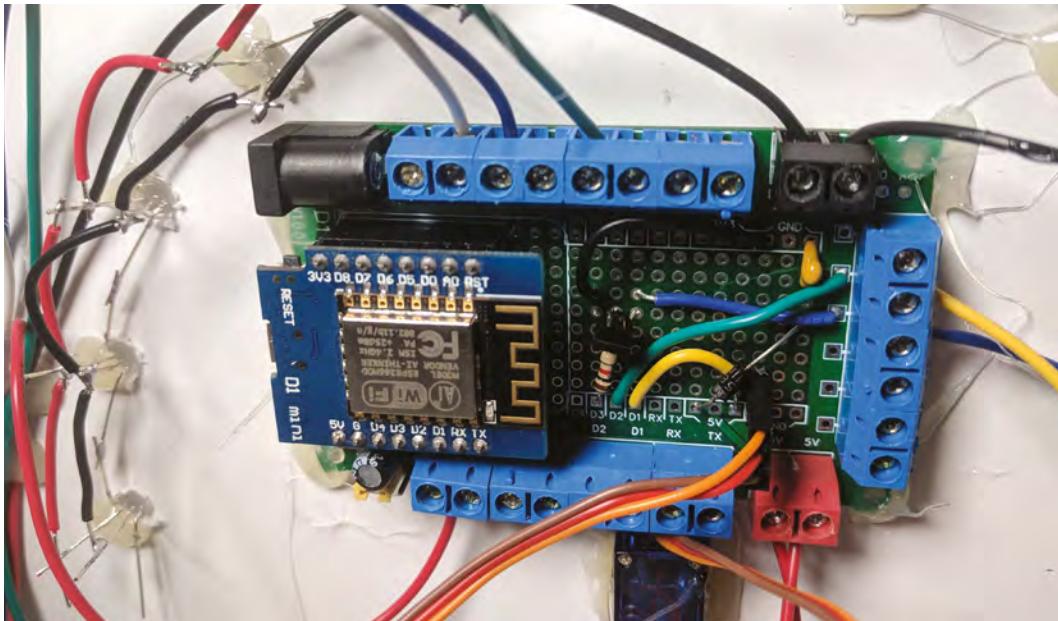
The Way Home Meter takes these coordinates and sends a request to the Google Maps API to get the live travel time and distance between the person's location and home.

The Way Home Meter will then add the travel time onto the current time and display the estimated arrival time of the person and updates the dial and NeoPixels to represent what percentage of the journey (distance wise) has been completed.

CODE IT UP

The code for this project is available on GitHub. Go to the following URL, hsmag.cc/ybAchB, and click the Clone or Download button on the right side of the page, and then Download Zip.





Extract the zip file. Inside the extracted folder, open up the **WayHomeMeter** folder and open the **WayHomeMeter.ino** file.

This sketch requires some additional Arduino libraries to be installed; start by opening the Arduino Library Manager by going to Sketch > Include Library > Manage Libraries.

You will need to add the following libraries:

- **Universal Arduino Telegram Bot** by Brian Lough – for creating a Telegram bot on the ESP8266.
- **Google Maps API** by Brian Lough – for getting the live traffic data.
- **Arduino JSON** by Benoît Blanchon – used by the libraries to parse the responses. Note: There is a breaking change in V6 of this library that will cause it not to work with the Telegram and Google Maps library, so use the drop-down on the left of the window to change the version to V5.13.2.
- **MD_MAX72XX** by majicDesigns – for communicating with the dot matrix display.
- **MD_Parola** by majicDesigns – handles animations on the dot matrix display.
- **Adafruit NeoPixel** by Adafruit – for controlling the NeoPixels.
- **NTPClient** by Fabrice Weinberg – for getting the time from the internet.
- **Timezone** by Jack Christensen – for automatically switching the time for daylight savings. →

PROGRAMMING THE ESP8266

The standard Arduino IDE isn't set up to program the ESP8266, so before we can program the board, we need to set this up (you can skip this bit if you've already used the IDE with an ESP8266).

First, let's get the raw IDE. You can download this from the Arduino website and install it as you would any other software: hsmag.cc/TAfEJp.

Next, you will need to set up the IDE so it knows how to communicate with an ESP8266. Open the Arduino IDE, go to File > Preferences, and paste the following URL into the Additional Boards Manager URLs, then click OK:
http://arduino.esp8266.com/versions/2.4.2/package_esp8266com_index.json

Back on the main screen of the Arduino IDE, go to Tools > Board > Boards Manager. When this screen opens, search for 'ESP8266' and install it; this may take a few minutes depending on your internet connection.

After setting up a new board it is recommended to get the simple example blink sketch before trying anything more complicated; this can save a huge amount of headache down the line! You can find this in File > Examples > 01. Basics > Blink.

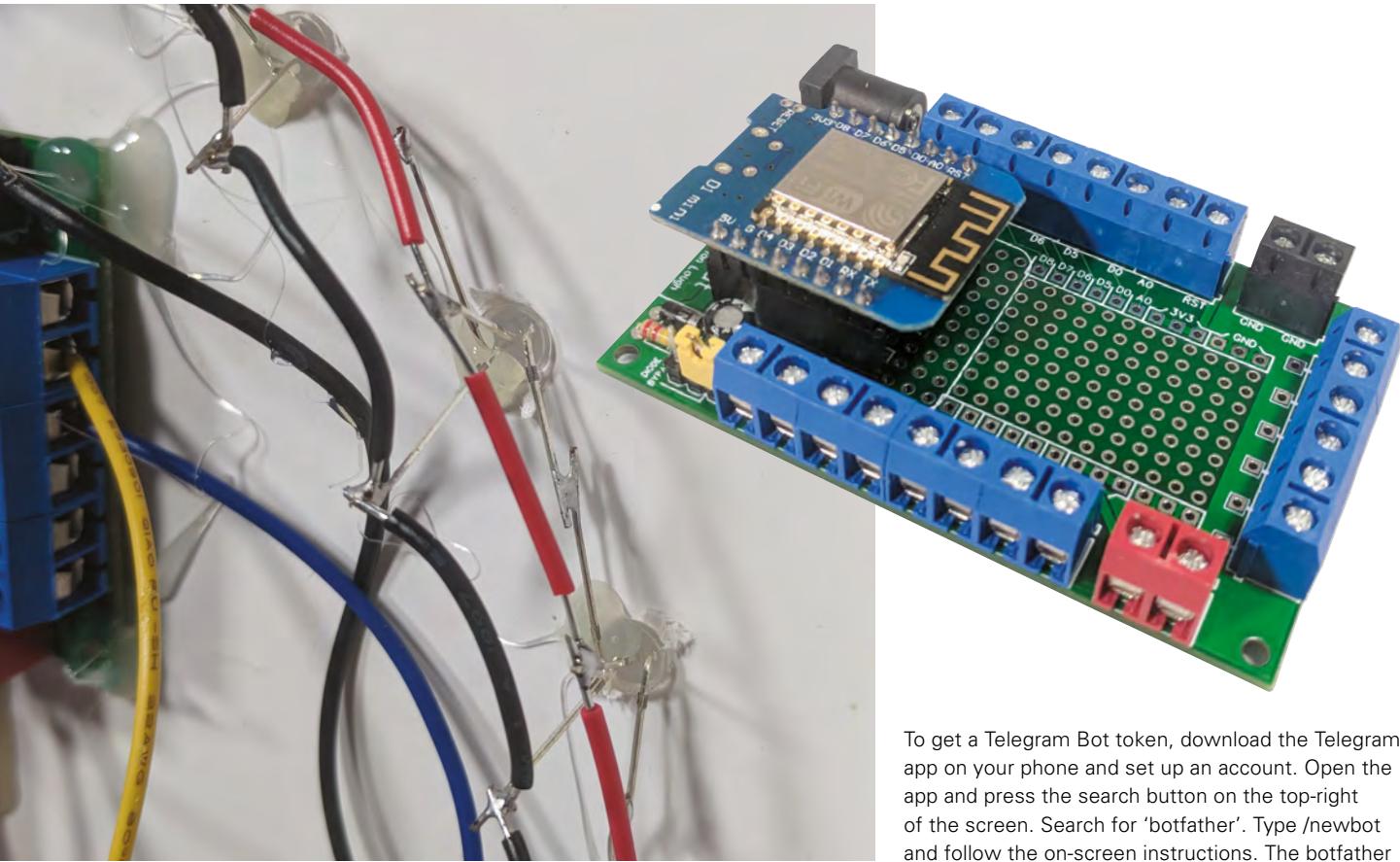
Upload this to your ESP8266 and you should see an LED blink on and off. If you get an error or don't get a blinking light, make sure you've got everything installed correctly and the ESP8266 is properly connected.

Left ◇
Screw terminals are useful for projects where components are separated from the PCB

YOU'LL NEED

- ◆ **An ESP8266 Wemos D1 mini microcontroller** (or equivalent e.g. Adafruit Feather Huzzah or NodeMCU etc.)
- ◆ **4-in-1 Max7219 dot matrix display**
- ◆ **A small servo** (sg90)
- ◆ **A 3D-printed dial for the servo** (optional, could be made from anything!)
- ◆ **11 x through-hole NeoPixel** (I used PL9823 LEDs)
- ◆ **220 pF capacitor**
- ◆ **Passive buzzer**
- ◆ **1 kΩ resistor**
- ◆ **NPN transistor**
- ◆ **Protoboard** (I used a prototype PCB I designed, but the project can easily be built with standard protoboard)
- ◆ **Screw terminals** (optional)
- ◆ **IKEA RIBBA frame**
- ◆ **A3 piece of 3 mm foam board**
- ◆ **Hot glue gun**
- ◆ **4 mm wood drill bit**
- ◆ **Sharp knife**
- ◆ **A metal ruler**
- ◆ **A compass and a protractor**
- ◆ **Micro USB phone charger** (for powering the project)

TUTORIAL

**Above right ↗**

This is a custom PCB that breaks out all the pins of the D1 Mini to screw terminals, but it can be easily recreated with a standard protoboard

Above ♦

The address pins of the LEDs should be able to reach each other without the need for extra wire. The power pins need to be joined by wire

After installing these libraries, you should click the 'verify' button (shaped like a tick) on the WayHomeMeter sketch to make sure that everything compiles fine.

SOME CONFIGURATION REQUIRED

You will need to make some configurations to get this sketch to work for you, but you will first need to get:

- Telegram Bot token
- Google Maps API token
- GPS coordinates of your home

GOOGLE BILLING

Google maps gives a free monthly allowance of credit – equivalent to 20 000 requests. That is just under what's required to send a request every two minutes in a month (about 22 000). This device only makes the request every two minutes that it is actively monitoring someone's home journey, so should stay under the limit if used occasionally. It's possible this limit will change in the future. How often it checks can be configured in the sketch by changing `delayBetweenGoogleMapsChecks`.

To get a Telegram Bot token, download the Telegram app on your phone and set up an account. Open the app and press the search button on the top-right of the screen. Search for 'botfather'. Type /newbot and follow the on-screen instructions. The botfather will provide you a link to the bot and an access token. The link is for the chat where people will share their location; the access token is used in the sketch to authenticate your ESP8266 as the bot you just created.

Next, you will need to get a Google Maps API key. Start by going to the following URL:

hsmag.cc/mPqFqh.

Check the Routes option and click Continue. You will then be asked to create a project; you can give this any name. You will need to add a billing account, but this device will comfortably operate on the free allowance given by Google. You will then get an API token that can be used in the sketch.

And finally you will need to get your home's GPS location. A simple way of doing this is using Google Maps. Using a web browser (not the app), navigate to your house on Google Maps and right-click and click 'Directions from here'. This will modify the URL, which will now contain the coordinates of your home; copy and paste these from the URL e.g. 51.5546466,-0.2794867.

You now have everything you need to configure the WayHomeMeter. Open up the WayHomeMeter sketch and click on the config.h tab. First thing you will need to enter is your WiFi details so that the ESP8266 is able to connect to your WiFi.

NEOPIXELS WITH A 3.3 V DEVICE

You can often have issues using NeoPixel LEDs with a 3.3V logic level device such as an ESP8266 or Raspberry Pi. You can get around this issue by using a logic level shifter to convert the 3.3V to 5V for the Data In connection for the first LED. However, we've found that it works fine with just a small capacitor between Data In on the first LED and Ground (as seen in this project).

Next, you will need to add your Telegram Bot token, your Google Maps API key, and your home location. Finally, if you are not in the UK or Ireland, you will more than likely need to change your time zone. Uncomment the appropriate time zone and comment out the UK and Ireland time zone.

WHAT MAKES IT TICK

You'll need to wire everything together as shown in **Figure 1** (overleaf). The LEDs are addressable RGB LEDs, so they only require a single GPIO pin of your microcontroller and you can set the colour of each LED individually. The input of the first LED (the one over on the left when looking at it from the front) will be connected to the Wemos, and its output will be connected to the input of the second LED. For all subsequent LEDs, the input of the next

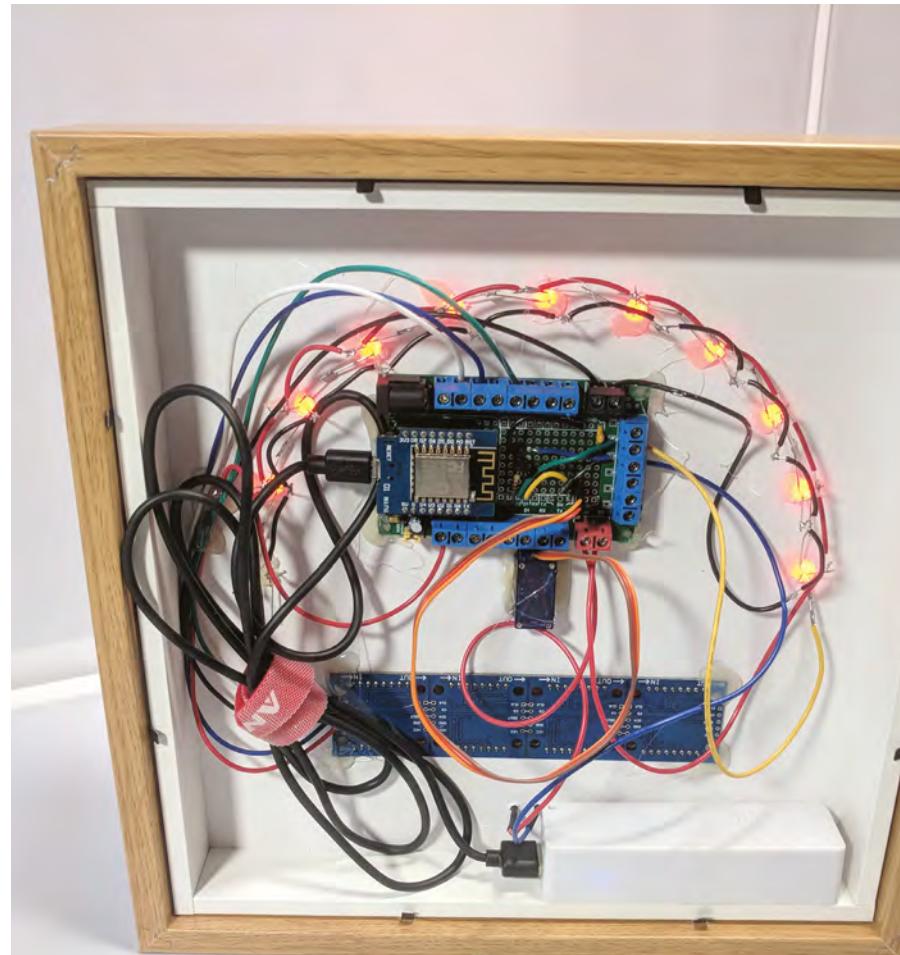
**Once you are happy
everything is working
correctly, secure the
components in place
with some hot glue**

LED is connected to the output of the previous one. The output of the final LED will not be connected to anything. Soldering these LEDs should be left until the final assembly stage.

CREATING THE MOUNT

Take the back panel off the picture frame and use it to trace a square onto your foam board. Using a sharp blade, cut out the square.

Next, you'll need to separate the panels from the display, as you'll be placing the PCB on the back side of the foam and the panels on the front; this will hold the display in place and hide the cuts from view.



Above A peek at what's hiding at the back

Carefully remove each of the dot matrix panels from the dot matrix display. There are markings on the side of each of the dot matrix panels; make a note of what direction they are facing in comparison to the PCB to ensure they are put back in the correct orientation.

If your PCB has header pins attached, desolder and remove them. Replacing these with wire will make the PCB fit flush to the foam board.

Measure the rectangle created by the pins and mark out that shape where you want to place it on the foam board. The objective is to cut out a shape that the pins of the PCB will fit through, but the PCB itself will be too big for.

The LEDs for this project are all on an arc around the centre point of the servo. Mark where you want the centre point of the servo arm to be and, using your compass, draw a semicircle lightly for where you want the LEDs to be. →

SHORT ON TIME?

If you are short on time, or just interested in quickly trying this project out, strip it back to be just the dot matrix display, the Wemos D1 Mini, and use DuPont cables to connect them together. The key piece of functionality, displaying the expected arrival time, uses only the display.

TUTORIAL

QUICK TIP

It's a good idea to practise cutting out the shapes and making the LED holes using scrap pieces of the foam board!

Place your protractor on the centre point and mark every 18 degrees. Then, using a ruler, line up the centre point and these new marks; where this line intersects with the semicircle is where each LED should be placed. Starting on the side that you want to be the front, use the 4mm drill bit by hand (no power drill needed) to create a hole for each of the LEDs where you have marked.

Measure the dimensions of your servo and mark it on around the centre point. Remember that the part of the servo that rotates should be the centre point, so offset the servo shape to suit. Cut the shape out of the foam board and place the servo through from the front.

Finally, you will need to place the buzzer module. You can simply place the pins of the module into the foam to mark where the holes should be and, using a piece of wire, pierce the two holes so they go through the foam board.

ON THE FINAL STRETCH

Place all the LEDs into the foam board from the back. Bend the input pin of each LED back towards the previous LED, and bend the output pin of each

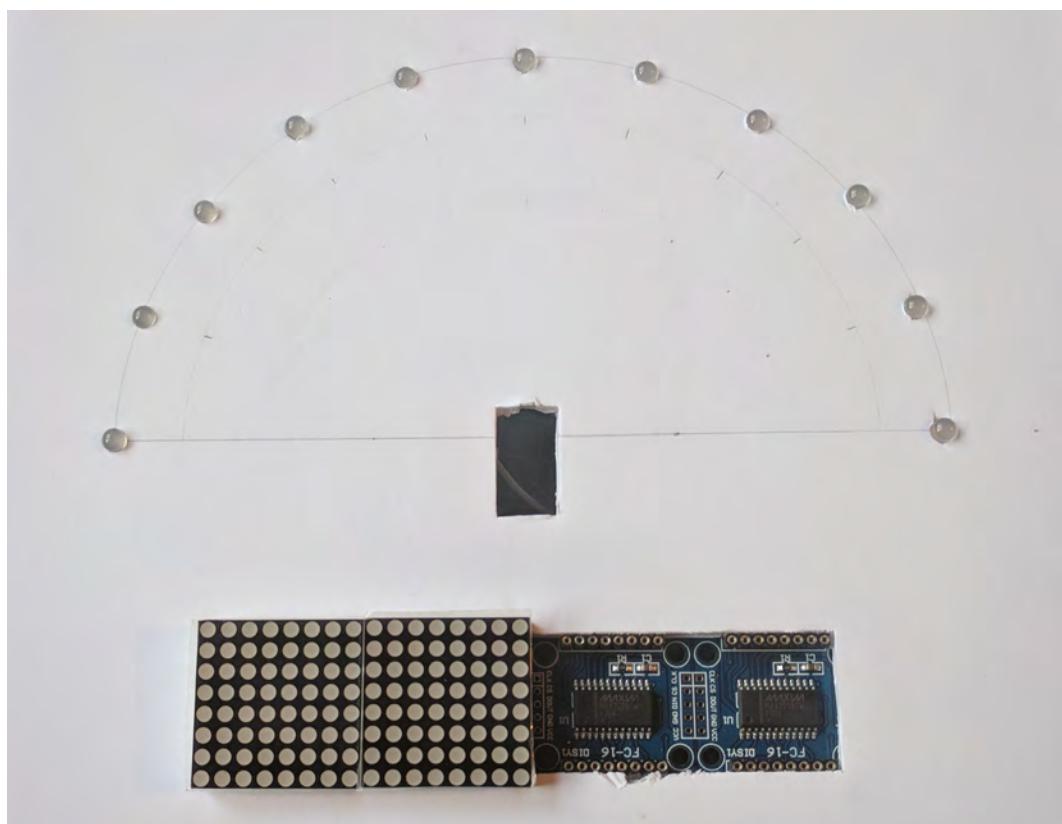
GOING FURTHER

You can take this project and make it differently, depending on your requirements. Here are a few suggestions:

- Add the ability to send the device a location and time, and have it calculate when you need to leave your home to make it on time.
- Add support for multiple people. Currently the device will request the travel time for the person who last sent a coordinate and display the correct name and information for them, but this could be improved to handle multiple people.
- Configurable alarms. Get notified when a person is X number of minutes away. Useful for starting dinner!

towards the input pin of the next LED, and solder them together. Slightly bend all the Ground pins of the LEDs towards the centre of the circle and all VCC pins away from the centre of the circle. Solder wire between all the Ground and VCC pins.

Place the dot matrix PCB in the cut-out, and put all the panels back in place. Pay careful attention



Right ♦

Foam board all cut and LEDs placed. The foam board will be clamped between the display module and the PCB

A POTENTIAL WEASLEY CLOCK?

A lot of people who saw early versions of this project mentioned that it reminded them of the Weasley Clock from Harry Potter, a clock that showed the current location of each of the members of the Weasley family. This Telegram-based solution could be used for a project like that, but it does require each of the users to actively enable the location sharing. A more passive solution might be better.

QUICK TIP

Always be generous with the lengths of wire you use, especially in a project where space is not an issue. If you need to make adjustments, it's easier to shorten them than lengthen them.

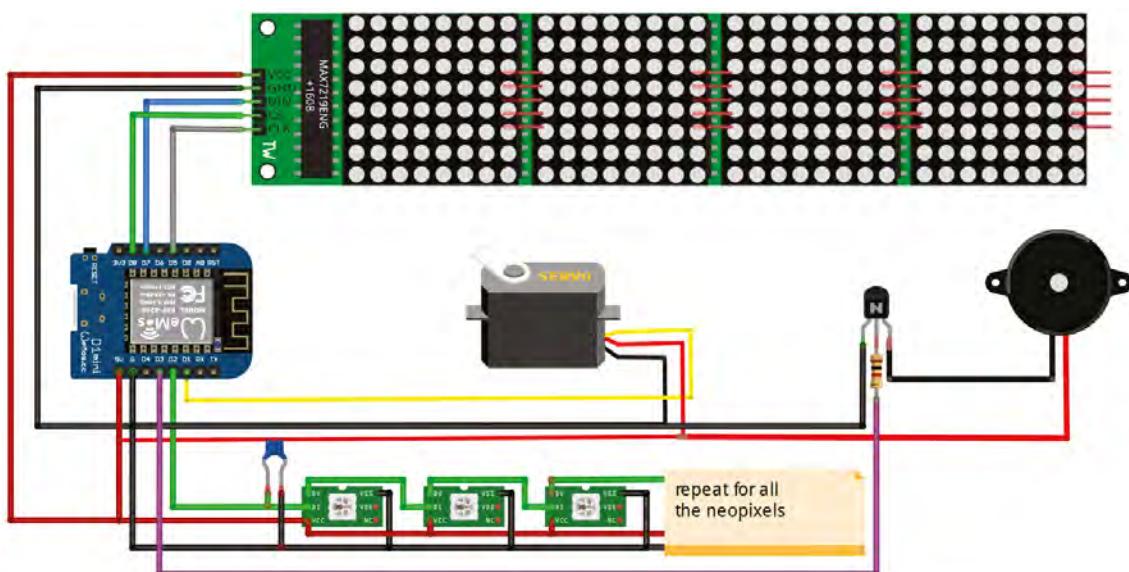


Figure 1 The wiring diagram for our Way Home Meter

Below The name displayed comes from the user's Telegram name

to the orientation of the panels, as it is very difficult to remove these again without damaging the foam board.

You then want to thread the wire of the servo module through the hole for the servo, and then insert the servo. Glue the dial hand onto one of the connectors that comes with the servo, and attach it to the servo when dry. The 3D design used in this project can be downloaded from here: hsmag.cc/iqOPiP. However, you can use anything you want (and a model car could be substituted if you don't have access to a 3D printer).

Finally, solder wire to each pin of the buzzer module and push it through the front of the foam board.

Connect all the modules to the Wemos on the protoboard and test everything out. Once you are happy that everything is working correctly, secure the components in place with some hot glue. You are now ready to have super-accurate home arrival times! ☐



Monkey skull treasure bowl

Anyone who is daring enough to reach for treasure from this bowl will be in for a big surprise



Andy Clark

workshopshed

For the last 10 years Andy has been making and repairing in a shed at the bottom of the garden. You can see more of his exploits at workshopshed.com



The Kuriologist

@strangecurios

A self-taught cognitively disabled 'outsider' artist. To overcome housebound boredom, creates mixed media sculptures inspired by 'cabinets of curiosities' & dead stuff.
kuriology.com

Right ◆
Monkey brains anyone?

We teamed up with artist 'The Kuriologist' to make this solder-free project, inspired by the *Indiana Jones* films. Despite having a hollow head, this papier-mâché skull bowl actually has its own brain, a tiny Adafruit Gemma M0 board. We wired that up to some bejewelled LEDs using crimp connectors. The rim of the bowl hides the wire for a capacitive sensor which triggers our spooky skull.

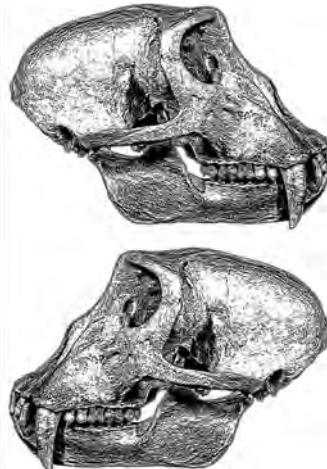
MAKING THE SKULL

The base of the skull is made from a used tape roll. On top of that, place a small box which will be used to house the electronics. The opening part of the

box should be positioned to the back of the skull, use masking tape to secure. A second roll taped on top gives the skull some height and forms the bowl. Scrunch the newspaper into packets of a 'sausage roll' shape.

You can then tape the packets to the rolls to form the rough outline of the skull. The jaw can be cut from thick card and taped to the front of the skull. Cut nasal and eye sockets from card and add with more tape. Keep poking, squeezing, adding more masking tape, and building up areas until it looks like a skull.

Mix up some PVA glue, three parts glue to one part water. The water will help the glue soak into the paper and improve drying times. Rip up white paper





and dip into the glue mix before applying all over the skull. Ensure it is fully covered, and apply two or three layers for strength. It's a good idea to leave the skull 30 minutes to one hour between each layer to dry. Then leave overnight for it to harden.

Once it's dry, we need to decorate the skull. This can be done with a final skin of paper using a printed image of a skull, a technique called découpage. An example skull image is available on Wikimedia Commons: hsmag.cc/eBBMCv. Apply this in the same way as the papier-mâché layers, by ripping the picture into smaller pieces and sticking them onto the skull. Use plain black paper for the eye sockets and red paper for the bowl. An alternative would be to paint or colour these with a permanent marker.

To give the skull a protective skin, it is coated with shellac. The 'button' or 'garnet' colours work best. It dries quickly and gives an antique appearance. One coat should be sufficient. If you use a dedicated brush for shellac, you can give it a quick rinse in thinners before storage, and then a soak before the next use.

GLOWING GEMSTONES

Select some acrylic gemstones of a suitable size to fit into the eye sockets. The stick-on or stitch-on type, that have a flat side, should be easiest to use. The flat side is painted silver and helps reflect the light. So that the light from the LED will shine through, lightly sand the silver paint.

A reflector cone is formed from kitchen foil. To ensure there is some overlap to join the cone, we needed to draw our template slightly oversized. Draw a 15mm diameter circle, and a second circle

of 40mm around that. Cut out the large circle and slit the disk to the centre. Now cut out the inner circle.

Form the foil into a cone and wrap around a 10mm LED. Tape this into place. If your gemstones are not round, you may need to make some adjustments to the shape of the cone. You can do this by folding over the edges, pinching and folding, or by cutting off the excess. Fill the cone carefully with hot glue and stick the gemstone to the top and leave to cool. Repeat with the other LED and gemstone.

WIRES AND CRIMPS

You will need two lengths of different coloured wire for each LED, which are long enough to pass through the skull and out of the back. This will mean you can connect the Gemma before sliding it back into the box. Crimp one wire to each leg of the →



Above

The reflector helps spread the light across the gemstone

Far left
A solid base for the skull and padding to shape

Left
Skull shape formed from tape and card

QUICK TIP

The 38mm masking tape is best, as it is not too wide and not too narrow.

YOU'LL NEED

- ❖ Newspaper
- ❖ Small cardboard box
- ❖ Paper and card
- ❖ Used tape rolls
- ❖ 38 mm masking tape
- ❖ PVA glue
- ❖ French shellac polish
- ❖ 2 × jewels
- ❖ 2 × 10 mm red LED
- ❖ 2 × 220 Ω resistor
- ❖ Kitchen foil
- ❖ Hot glue
- ❖ Adafruit Gemma M0
- ❖ Wire
- ❖ Approx 30 cm of screened wire
- ❖ Crimp connectors
- ❖ 3 × AAA battery box with JST connector
- ❖ 2.5M nuts, bolts, washers

LED. Make a note of which wire connects to the anode of the LED.

Use a bradawl or skewer to make a hole in the back of the eye sockets and into the cardboard box. Pass the wires through the holes and out of the box. Crimp a resistor to one of the LED wires.

YOU'VE GOT THE TOUCH

The sensor for the touch is formed from the screening from the screened wire. Remove the outer plastic cover of the screened wire and slide the braided screen off the wires. Flatten the braid and form into a loop to fit around the rim of the bowl. Twist the ends together and crimp to a wire. Make a hole from the bowl through to the electronics box and pass the wire though. The braid can then be attached to the bowl with hot glue.

Strip the ends of the wire and wrap around an M2.5 bolt, and twist the end to form a loop. Form a loop on the end of each resistor. Bolt one LED anode wire to the D0 connection on the Gemma, and the other anode wire to D2. The cathode wires should be bolted to the 0V connector. Bolt the sensor wire to the A0 connection.

BRAIN POWER

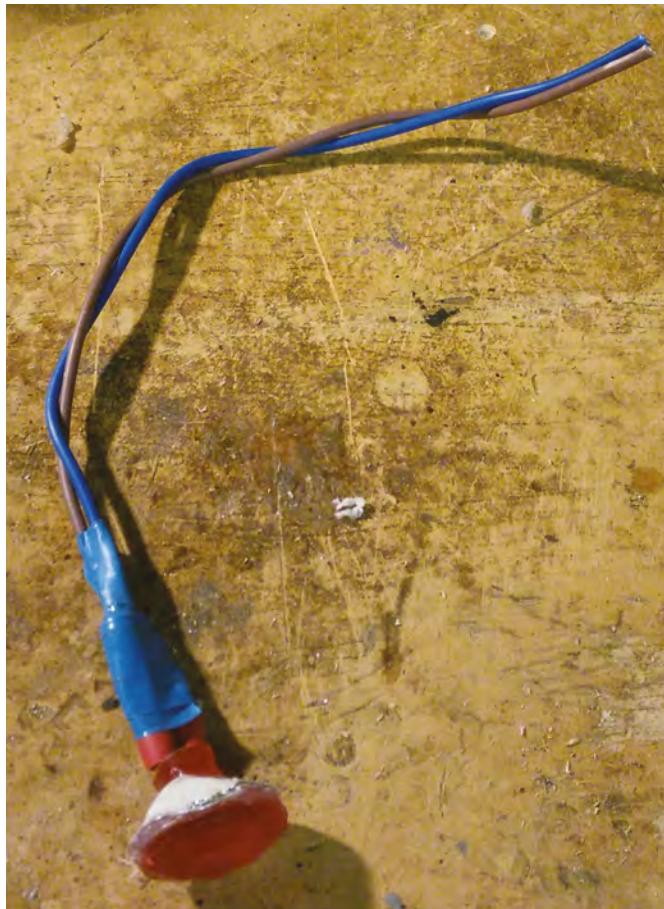
The Gemma M0 uses an ATSAMD21E18 processor, and this is powerful enough to run a subset of the Python language called CircuitPython. Combined with the provided libraries, this means we can build this kind of project with just a few lines of code.

```
import time
import board
import touchio
from digitalio import DigitalInOut, Direction
```

We start with instructions to use the libraries for time, board, and touchio. For the digitalio import instruction, we've just picked the items we need.

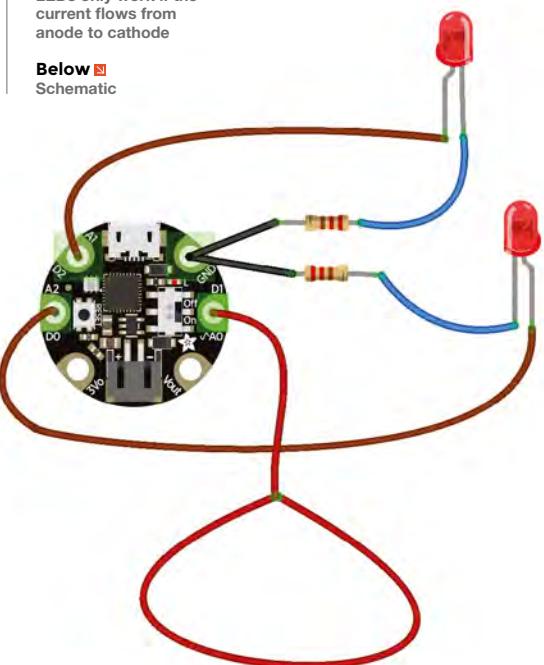
CAPACITIVE SENSING

The capacitive sensors on the Gemma work by charging and discharging the output pins and measuring the voltage. The pins have a small amount of capacitance and the microcontroller is sensitive enough to measure this. When a person is very near to the sensor, they affect the circuit by acting as a second capacitor. This means that when the microcontroller charges and discharges the pin, it will see a change in behaviour and know that it has been touched.



Above LEDs only work if the current flows from anode to cathode

Below Schematic



This can be a useful technique to simplify the code you write later.

```
touch_pad = board.A0
touch = touchio.TouchIn(touch_pad)
led1 = DigitalInOut(board.D0)
led1.direction = Direction.OUTPUT
led2 = DigitalInOut(board.D2)
led2.direction = Direction.OUTPUT
```

The pins need to be configured before we can use them. The A0 pin is configured for touch, D0 and D2 are configured as digital outputs. Note how **Direction** and **DigitalInOut** don't need a prefix as we imported those specifically above.

```
def detected():
    for i in range(3):
        led1.value = True
        led2.value = True
        time.sleep(0.5)
        led1.value = False
        led2.value = False
        time.sleep(0.2)
```

This code defines a function we can call when the skull is triggered. It flashes the LEDs three times, with a short delay between each flash.

```
detected()
while true:
```

```
if touch.value:
    detected()
    time.sleep(0.05)
```

To check that the lights are working, the code calls the **detected** function when the Gemma first powers. The **while** statement creates a loop that will keep running until the power is turned off. It checks the touch sensor to see if it has been touched. If it has, then the code calls the function defined earlier to flash the lights.

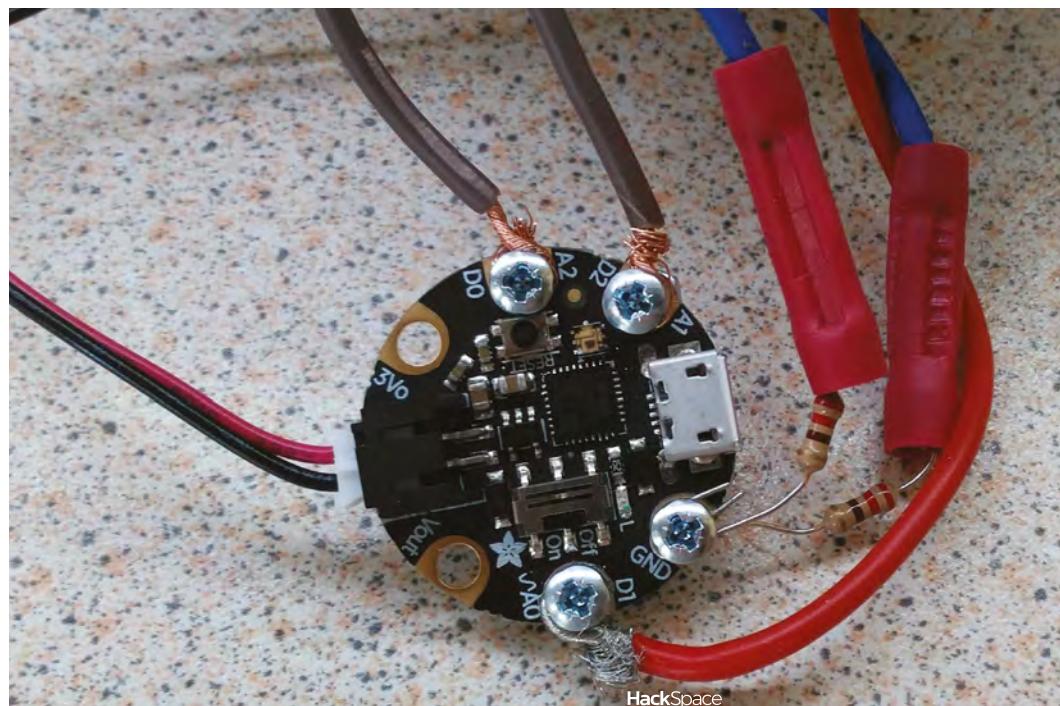
Upload the code to the Gemma and remove the USB connector. Fit the batteries to the battery pack, and plug into the Gemma. Test everything is working, then fit the electronics into the back of the skull. Add some treasure to the bowl, e.g. sweets. Your monkey skull bowl is now ready for use. □

QUICK TIP

On the Gemma M0, like many microcontroller boards, the pins are shared and can do different tasks.

OTHER IDEAS

- Paint your skull in candy skull style for a 'Day of the Dead' look.
- Use PWM to make the eyes fade up and down, rather than flash.
- Use servos to make your skull move.
- Swap out the Gemma with a Circuit Playground Express board, and add some audio effects to your project.



Left ◇
Bolted connections
using washers and
2.5 mm bolts

Copper pipe lamp

Steampunk lighting that you can make at home



Ben Everard

@ben_everard

Ben loves cutting stuff, any stuff. There's no longer a shelf to store these tools on (it's now two shelves), and the door's in danger.

Right
The completed lamp doesn't have to illuminate your wine rack, but it can

Electric lights are perhaps the most basic electrical item. At their most basic, they contain just two components – a bulb and a switch. This simplicity gives us plenty of scope to put our own artistic twist on the design. We've gone down a common route of using copper pipe to create a custom frame for our design. This pipe is both easy to work with and looks great. A yellow-orange light accentuates the colour of the copper, resulting in a visually pleasing lamp.

When working with mains electricity, we obviously need to make sure that we're kept safe from live current. When building appliances that are hopefully going to be used for many years, we don't just need to make sure that they're safe at the point we build them, but that they'll remain safe. We'll do this by both reducing the chances of something going wrong, and putting a mitigation in place in case it does. We've built a lamp that we're happy with, but ultimately it's your responsibility to understand the risks inherent in working with mains voltage electricity, and make sure your lamp is safe. It's beyond the scope of a magazine article to go through all the risks of working with mains electricity.

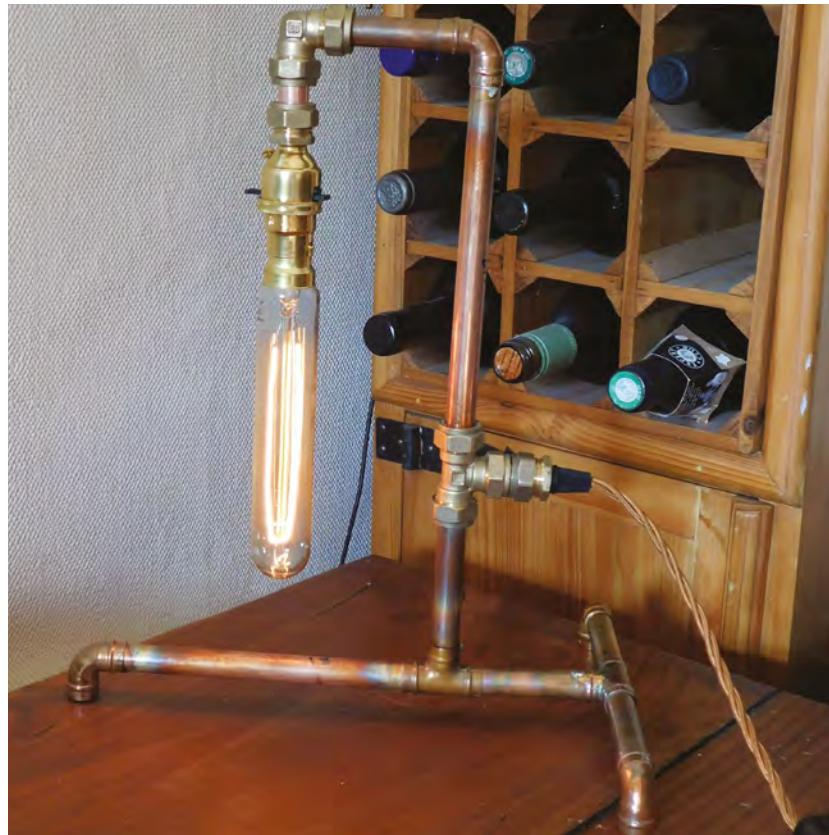
The obvious risk with a copper pipe lamp is that, somehow, the live wire gets in contact with the copper pipe. If someone then touched the pipe, they could have a serious shock. Because copper

WARNING

Mains electricity is dangerous – This project involves working with mains electricity. We have endeavoured to provide accurate advice. However, it is ultimately your responsibility to make sure that any lamp you make is safe. Make sure you understand the risks of working with mains electricity before embarking on any project that uses these voltages, and seek advice from someone experienced with the risks if you're unsure. You are responsible for your own safety. Take that responsibility seriously.

is conductive, it's inherently more risky than making a lamp out of non-conductive material, so we have to be more careful to ensure this doesn't happen.

The lamp could get live in two ways: either the cable could be pulled out of the bulb holder, or the flex could become damaged inside the pipe. To reduce the risk of these happening, we'll make sure that the cable is secured at the point it exits the lamp, so tugging on the cord doesn't put force on the connection between the flex and the bulb holder,





Left ♦
Our copper pipe ready for assembly. Getting the cuts as smooth as possible will help make sure everything fits together properly

SWITCHES AND DIMMERS

Our bulb holder has an integrated switch, but there are a few other options. You can have a torpedo switch on the flex cable (just make sure that your switch has earth pass-through, so the effort to protect your lamp isn't wasted).

Edison-style filament bulbs look particularly good with the copper, and these dim particularly well. You can get in-line dimmers that can be wired into the flex, just as a torpedo switch could be. We find the orange-red light of a dimmed filament bulb very soothing in the evening.

and this will also reduce the movement of the flex inside the lamp, which will also minimise the risk of it being damaged inside the lamp.

To mitigate the risk, should the flex insulation somehow fail, we'll ensure that the whole body of the lamp is earthed. This means that if there is a contact between the live part and the body of the lamp, it'll get harmlessly diverted to ground, and this should either blow a fuse or trip a circuit breaker.

We'll look at how to put these two protections in place as we build the lamp.

SNIP SNIP

Copper pipes are soft and easy to work with. You can get specialist pipe cutters, but most metal saws

will make short work of it. Whichever you use, you need to make sure that the cuts are clean before assembling your lamp. If there are any burrs, they could stop the joints fitting together, or they could dig into the cable, which could have disastrous consequences. You'll need a file that fits inside the pipe, and give it a good scrubbing until there's nothing protruding from the cut.

Let's take a look at our options for joining the pipe together. The most elegant method is to use soldered joints. These are pre-formed joints and bends that slot around the pipe and can be heated and soldered together. Some already contain solder, while others need to have solder added. Whichever version you use, you need to add flux to the pipes first to ensure a good joint. They should fit quite snugly, so you can push the pipes into place and line everything up, then go around with a blow-torch and solder every joint. The big disadvantage of these joints is that you have to seal them without the cable inside, and this makes it hard to thread the cable through more complex shapes. We found that we could thread the flex round a 90 degree bend, but any more than this proved challenging.

The second joint option is compression joints. These work by having three parts: the joint itself that you press the pipes into, then an olive that looks like a ring of brass or copper, and finally a threaded end. These work by screwing the threaded end →

YOU'LL NEED

- ♦ **15 mm copper pipe** (about 1 metre)
- ♦ **Earthed bulb holder**
- ♦ **Assorted pipe joints** (elbow and 'T')
- ♦ **2 x pipe end stop, with M10 threaded holes**
- ♦ **Hollow M10 threaded rod** (about 30 mm long)
- ♦ **M10 cord gripper**
- ♦ **3 m of flex**
- ♦ **Plug**
- ♦ **Blow-torch** (optional)
- ♦ **In-line dimmer** (optional)

Copper pipe lamp

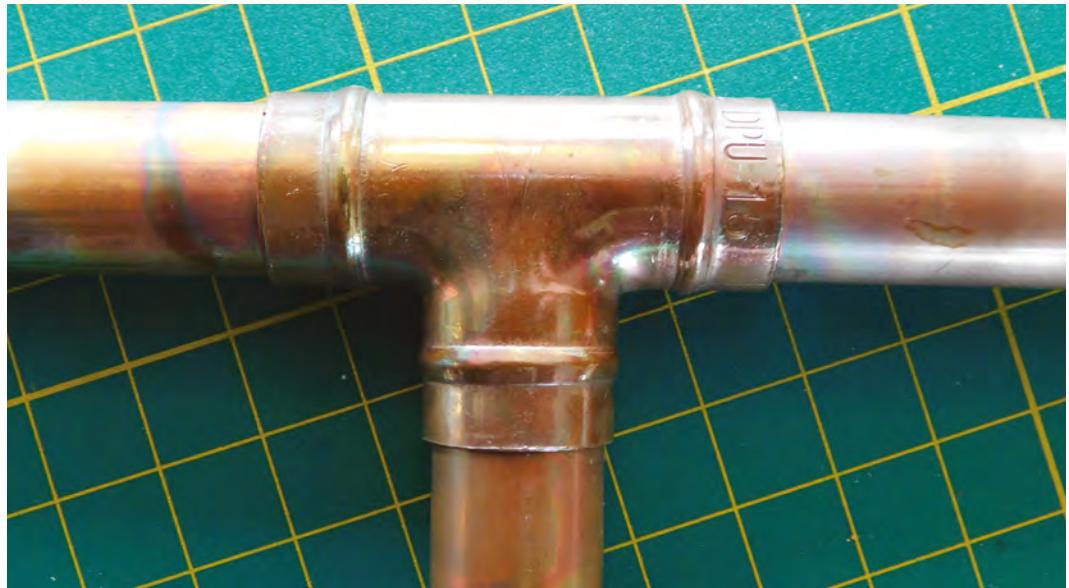
TUTORIAL

Right

A soldered copper T-joint. The ridges around each edge hold the solder that's heated to seal the joint

Below

Our bulb holder with the live and neutral going into the fixture, and the earth secured in the base



onto the joint with the olive between. As the two parts are forced together, there's enough pressure on the olive to deform it and create a solid joint. These are bulkier and this affects the look of the lamp, but they have two advantages: they don't require a blow-torch, and they can be done after the flex is threaded through the lamp, so there's not a problem creating complex shapes.

There are a couple of other options that would join the pipes together but that we can't recommend. There are plastic pressure-fit connectors and you can secure the solderable joints with glue rather than solder. The reason we can't recommend these is because they don't make electrically conductive joints. This means that you can't easily earth the entire metal body of the lamp, so if the live cable frayed in an unearthing part of the lamp, you could have a dangerous object.

You can bend copper pipes to create interesting shapes. The challenge is to create a bend without

the pipe collapsing, and there are a few tools out there for the job, but that's beyond the scope of this article.

GET INSPIRED

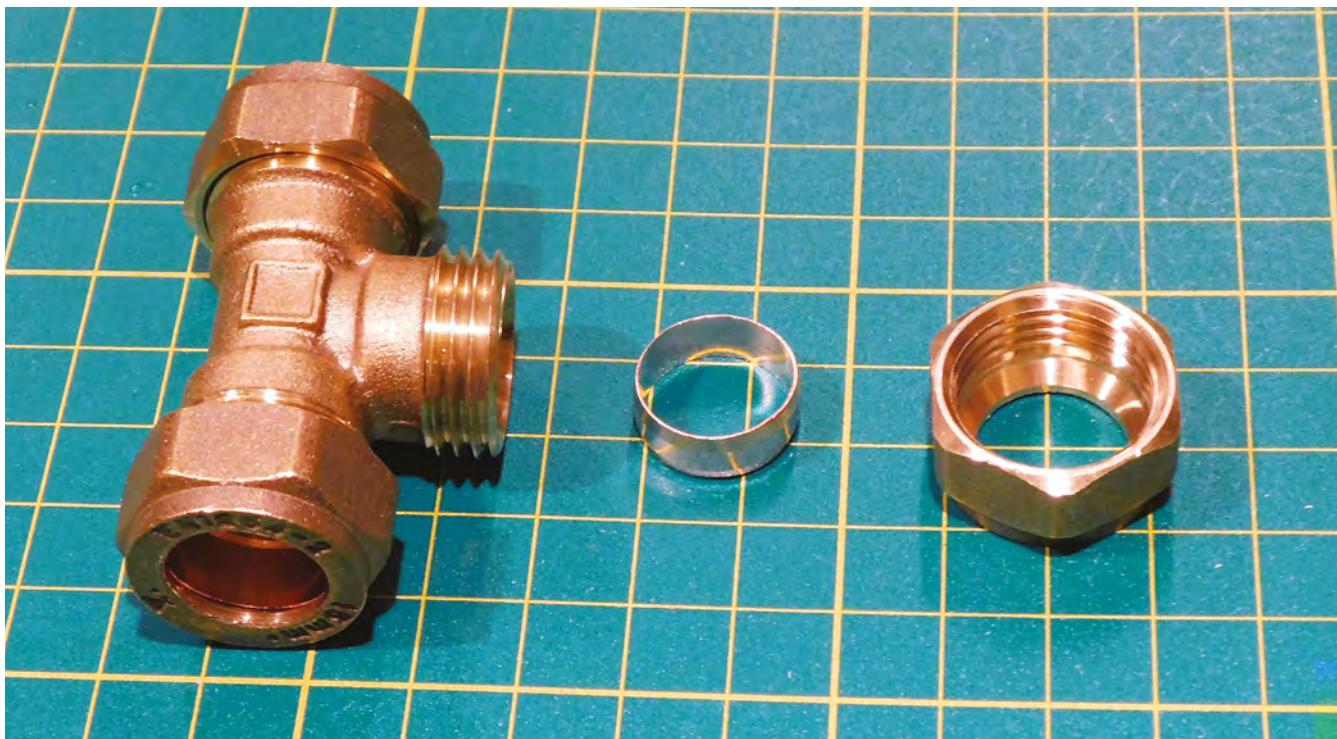
You can get creative with the design – the only restriction is that you can actually thread the flex through the design.

We've gone for a three-pointed base, for the simple reason that three legs can't wobble, but four or more can. Copper is a bit forgiving though. If you do opt for more legs and there's a little wobble, you can bend copper slightly by hand to correct any misalignment.

Above this, we've got a vertical pipe with two 90 degree bends that end up hanging the bulb down. Originally, we had both of these corners as soldered joints, but we couldn't thread the flex, so we had to cut one out and replace it with a compression joint. We could have replaced either one, but elected to do the outside one. We've also got a compression T-joint on the upright which gives us a place for our cable to escape.

The exact measurements don't have to be precise – ours is about 30 cm high and 20 cm wide, but we didn't even use a tape-measure, and just cut it by eye where it looked about right. It's useful to have a bulb with you when cutting the lengths out so you can picture how it will look when it's all fitted together.

Typically, bulb holders end in a 10 mm thread, and you'll also need a pipe end stop with a 10 mm threaded hole that you can attach to the end of your lamp. You can join these two fittings with a 10 mm hollow threaded rod, and through this the cable will get from the bulb holder into the body of the lamp.



All these bits you can get from a lamp maker store such as lampspares.co.uk.

Make sure that your bulb holder is either metal and earthed (which will earth the whole lamp provided all the joints are conductive), or you find another way to safely earth your lamp body.

At the other end, the flex needs to be secured as it leaves the pipe. Again, we'll use a 10mm threaded end stop, but this time we'll need a cord gripper with a 10mm thread. These are most commonly used to secure lamps hanging by their cord from the



wire up the bulb holder and plug, according to the manufacturer's instructions. This could be an easy build or it could be complex – it completely comes down to your design.

We're basically there now, but there's a couple of safety checks before getting started. If you've got a PAT tester, then this is ideal, but if not, we can check some things with a multimeter. First, check that the lamp's properly earthed by ensuring that the resistance between the earth pin on the plug and any exposed metal is low enough (less than a few ohms). Secondly, make sure that the resistance between the live and earth pins is very, very high (this ensures that there's not a leakage already). Remember that, just because your lamp is properly assembled now, it is no guarantee that it will remain that way, and you can repeat this test if you're ever unsure about the safety of the lamp.

That's all there is to it. Your lamp is now ready for use. □

Above The three parts of a compression joint. In the middle is the olive that is deformed when the joint is secured

Left Securing the flex as it exits the lamp helps make sure your lamp will continue to work for years to come

**The exact measurements
don't have to be
precise – we didn't
even use a tape -measure,
and just cut it by eye**

ceiling, but also work to secure the flex in this case. An alternative option is to use a rubber grommet to protect the flex, and add a flex strain relief to prevent the cable pulling through this hole. We found that this doesn't fit in a 15mm pipe, but you might be able to use this method if you're using thicker pipe.

That's all the pieces you'll need – let's put it all together. First, solder any joints you'll be soldering, then thread the flex through all the parts, including compression joints. Join everything together and

Build your own ANPR home automation setup

Automatically switch on the lights when your car is recognised parking outside your house



Paul
Freeman-Powell

@paulfp

Paul is a technology enthusiast, geek, video producer, drummer, cyclist, husband, father to three children, including twins, and host of the award-winning Switched On Network YouTube channel. You can find his website at switchedonnetwork.com

A

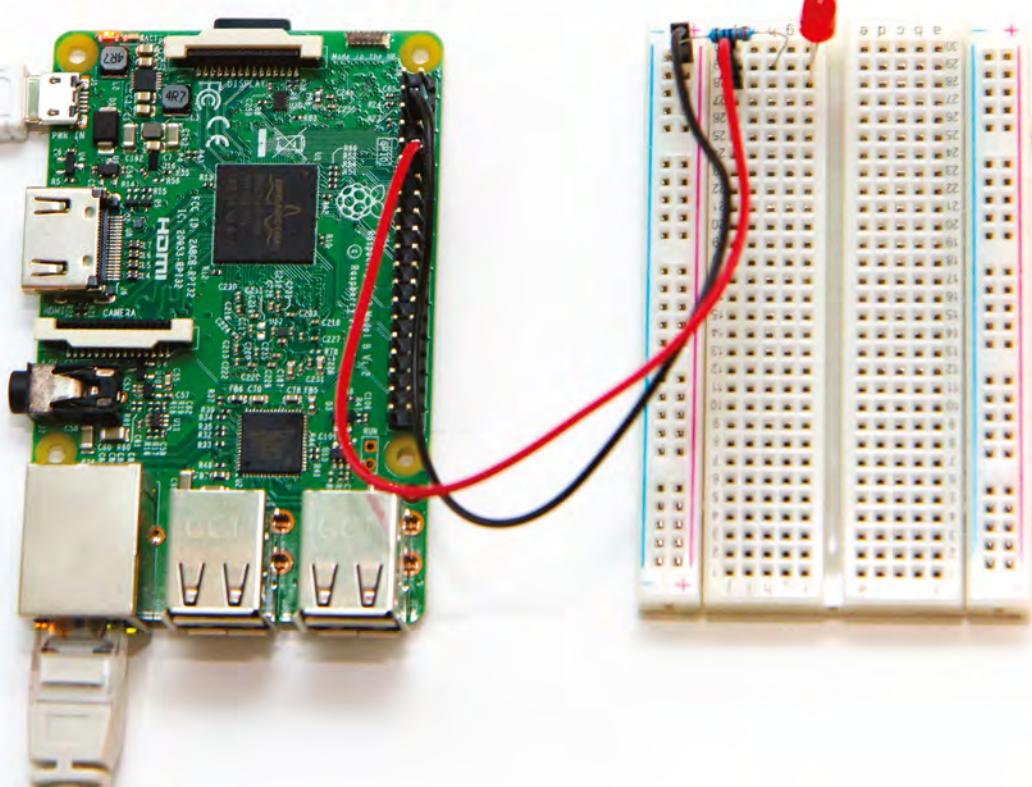
utomatic number-plate recognition systems (ANPR, or ALPR / 'License Plate' for our American friends) are used by the police to spot cars of interest, and by some car parks to record when

you enter/leave. We've always fancied building one but, as we run neither a car park nor an international crime-busting agency, there's been little point... up until now. In recent years, with the explosion of the Internet of Things and the IoT's integration with cloud-based services for such devices, a world of possibilities has opened up, and the price of components has also become much more accessible for the hobbyist, so that got my imagination ticking.

This project works by setting up an IP security camera to point at your driveway, or the street outside

your house, at the spot in which you park your car. A Raspberry Pi, connected to your home network, then monitors the network stream from the camera and passes the video through some free software called OpenALPR, which spots and reads car number-plates. A Python script then processes the results and any number-plates found and, when it spots your car number-plate arriving home, it communicates with a smart WiFi LED bulb, via the IFTTT service, to turn the hallway light on. When you drive your car away, the reverse happens and the light is switched off again. See the diagram overleaf to understand the program flow and sequence of events.

There are a number of ways you can approach this project. OpenALPR offers a cloud-based API where your program can upload still images and return results found. You can also choose to use their cloud



Right ↗

Pi connects to your home network with an ethernet cable. The indicator LED connected via a breadboard is optional, but you can use this same concept to drive a relay switch to do other things, if you like

streaming service to have a stream from your camera setup monitored by their servers. However, for this build, I'll be installing the software locally (which is free) and running it as a daemon on the Raspberry Pi to continually monitor the IP camera.

I'VE GOT THE POWER

Since processing a video feed and continually searching for number-plates within it is a fairly intensive operation, we'd recommend using the most powerful of Pis, which (at the time of writing) is the Raspberry Pi 3B+. For more power, you could install it on something like an Intel NUC Mini PC, and if you need really fast and accurate results you can even configure OpenALPR to use NVIDIA GPUs.

You'll need a Raspberry Pi set up with the latest version of Raspbian Lite.

It's also useful to connect up an LED to the Pi's GPIO ports to allow you to view at-a-glance whether

If you need really fast and accurate results you can even configure OpenALPR to use NVIDIA GPUs

the system thinks your car is currently home or away. This is completely optional, but it helps with debugging and could also form part of a permanent installation if you like. Using a breadboard and a couple of Female-Male DuPont cables, connect the LED and a 330Ω resistor to the GPIO 18 pin, as shown in the picture on the previous page.

Finally, connect the Pi to your network – ideally via a network cable rather than WiFi, as you want the best possible connection to your camera. We'll use SSH to access the Pi, which you can set up using the steps listed here: hsmag.cc/PPbxkq.

Determine the Pi's IP address on your home network by opening a Terminal and typing the command `hostname -I`.

You can now disconnect the keyboard and monitor from your Pi, and instead connect to your Pi via PuTTY from another computer on your network.

INSTALLING THE SOFTWARE

The first step is to install OpenALPR on your Raspberry Pi, using the precompiled Linux binaries as described at hsmag.cc/GQGQGs.

SSH FOR WINDOWS

WinSCP and PuTTY are two very useful utilities for connecting to your Raspberry Pi (or any Linux machine). WinSCP for Windows gives you an easy tool for transferring files between your Windows machine and the Pi, and PuTTY allows you to run commands on it remotely, as if you were connected to the machine directly with a screen and keyboard. You can install the latest version of each from ninite.com – look under Developer Tools.

To get started, type the following commands:

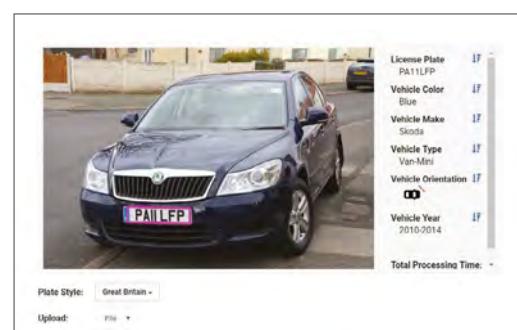
```
wget -O - http://deb.openalpr.com/openalpr.gpg.key | sudo apt-key add -
echo "deb http://deb.openalpr.com/master/trusty main" | sudo tee /etc/apt/sources.list.d/openalpr.list
sudo apt-get update
sudo apt-get install openalpr openalpr-daemon openalpr-utils libopenalpr-dev
```

Once that's installed, it is time to test that it's working, so go outside and take a photograph of your car, making sure that the number-plate is clearly visible. For this initial test, try to take the photograph in as close to perfect conditions as you can, i.e. looking straight-on at the car's front or back, with the number-plate clearly visible and large in the frame.

Use WinSCP to transfer the photo of your car to the `pi` user's home directory (`/home/pi`), and then run this command to test out the automatic number-plate recognition:

```
alpr -c eu mycar.jpg
```

The `-c` switch in that command allows you to specify the country of origin of the number-plate. It is not essential, but it gives the software a little helping hand if it knows what style of plate it is looking at. →



YOU'LL NEED

- ◆ **Raspberry Pi 3B+** (or a more powerful computer if you have one spare)
- ◆ **A network/IP camera** (must output a network stream via RTSP, and not restrict you to using the manufacturer's own app)
- ◆ **A smart WiFi LED light bulb** (like this one: hsmag.cc/jPFBp1)
- ◆ **2 x DuPont cable** (female/male)
- ◆ **1 x LED**
- ◆ **1 x 330Ω resistor**
- ◆ **1 x breadboard** (or soldering iron for more permanent setup)
- ◆ **An IFTTT account** (free)
- ◆ **A Kasa account** (free)
- ◆ **An internet connection and router or switch**
- ◆ **Lengths of Cat5e or Cat6 cable** (to connect your IP camera and Raspberry Pi to your router/switch)
- ◆ **Optional:** a PoE router, if you choose an IP camera which can be powered over Ethernet

Left ◆

The OpenALPR software installed on your Raspberry Pi scans the video feed from your IP camera, looking for and reading car number-plates. You can test it out with still photos, to help with your camera placement, at: hsmag.cc/RjUeyH

TUTORIAL

QUICK TIP

It's a good idea to give your Raspberry Pi a static IP on your network (outside of your router's DHCP range) so that weeks later and after several reboots, it will still be accessible via the same IP address.

As I'm in Europe, OpenALPR had a good go at reading my plate when I specify 'eu'. However, depending on the clarity of the image and angle, etc. it sometimes mistook the number '1' for an 'I' for indigo, so PA11LFP became PAAILFP, or even PAULFP. Since the sovereign states of the EU are all able to set their own independent standards and systems for car number-plates, OpenALPR allows you to specify which one, and I found that putting `-c gb` increased the accuracy when dealing with 1s and capital Is. See

the 'Accuracy Improvement' boxout for two more top tips for increasing accuracy of plate reading.

NETWORK ACCESS

When choosing an IP camera, you need to ensure that it supports a resolution of at least 720p and ideally has an automatic night-time mode, too. It also needs to support providing a network stream via RTSP (real-time streaming protocol) and must not be restricted to accessing via the manufacturer's own app. The one I'm using cost about £30 / €30 from a well-known internet auction website.

Connect your IP camera to the same network as your Raspberry Pi, give it a static IP address, and set a strong username/password combination. Some cameras display the date/time at the top of the picture and/or allow you to display a label such as 'Front Driveway'. For simplicity's sake, turn all options like this off. If it's a 1080p camera, turn the resolution down to 720p, and you can also drop the frame rate to 15fps – these settings will be perfectly sufficient to capture your car's number-plate, whilst significantly reducing the bandwidth of your stream so as not to overload the Raspberry Pi.

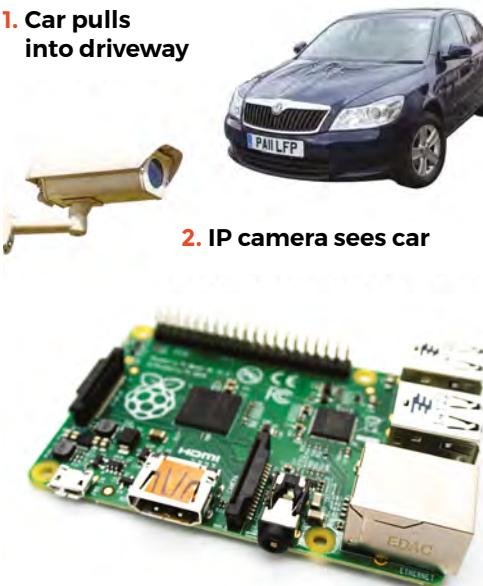
MONITOR YOUR DRIVEWAY

Once you're happy that OpenALPR is working on your Pi with images from your camera, you can now set it to monitor the camera's feed.

First, we need to test that we can view the network stream from the camera. On another computer on your network (not the Raspberry Pi), open up VLC (which you can download from videolan.org) and click Media, then Open Network Stream. Here, you need to type the address of your IP camera's stream which will incorporate the username/password you set as well as its IP address etc. This will vary depending on the make and model of your camera; for my camera, the stream address is:

`rtp://username:password@192.168.0.xxx/11`

To determine the correct stream for your camera, there are a couple of places you can try.



3. Raspberry Pi, running OpenALPR, monitors camera feed and scans for number-plates



4. Raspberry Pi triggers web request to IFTTT when car spotted arriving

5. IFTTT instructs Kasa to switch on the light bulb named 'Hallway Light'



Right ♦
The sequence of events and components/services involved. As you park your car on the driveway, it is identified as yours and your hallway light automatically turns on

IF THIS THEN WHAT?

IFTTT (If This, Then That) is a free online platform that lets you join together your various apps and IoT devices. It can talk to lots of online services and smart devices as both trigger and action, and as the name suggests it allows you to specify that 'if this happens, then do that...'. Have a play, and let your imagination run wild at ifttt.com.

Firstly, OpenALPR installs a file on your system which might point you in the right direction. To view this file, type this command:

```
sudo nano /usr/share/openalpr/runtime_data/
cameras.yaml
```

If you don't find what you're after in there, iSpyConnect have a pretty big Camera Connection Database at: hsmag.cc/nRgTLB, so take a look, click on the manufacturer of your camera, and try the various options in VLC until you find the one that works best for you.

Once that's working and you have constructed the URL of your camera's network stream, this needs to be input into the config file for the OpenALPR daemon. First, open up the defaults file like this:

```
sudo nano /usr/share/openalpr/config/alprd.
defaults.conf
```

Select and copy the contents of that file, then open the empty config file to paste it in ready to be adjusted:

```
sudo nano /etc/openalpr/alprd.conf
```

Find the section entitled 'Example stream config:'. Here, you can list camera streams and paste the full URL on a line starting 'stream = '. This file also allows you to specify the country code of the plates you'll be looking at, so in my case, as my car has a British plate, I've set this to 'country = gb'. Once you've made those

The OpenALPR daemon will now be monitoring your camera and placing any results into a queuing system called Beanstalk

changes, press **CTRL+X**, followed by **Y**, and **ENTER** to save and exit.

As you'll see from the daemon config file, you can build a pretty big system monitoring several camera streams across multiple sites, if you want. However, you'd certainly need something bigger and more powerful than a humble Raspberry Pi for that, although the basic concepts all remain the same.

Once you've set your configuration for the daemon, run the following command to restart it using those settings:

```
sudo service openalpr-daemon restart
```

```
login as: pi
pi@192.168.0.123's password:
Linux raspberrypi 4.14.62-v7+ #1134 SMP Tue Aug 14 17:10:10 BST 2018 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Sep 10 13:28:37 2018 from 192.168.0.10
pi@raspberrypi:~ $ alpr -c gb skoda.jpg
- PA11LFP    confidence: 83.9771
- PA10LFP    confidence: 81.1242
- FA11LFP    confidence: 71.6121
- PA11LFF    confidence: 71.3341
- FA10LFP    confidence: 68.7593
- PA10LFF    confidence: 68.4813
pi@raspberrypi:~ $
```

HARVESTING THE BEANSTALK

The OpenALPR daemon will now be monitoring your camera and placing any results into a queuing system called Beanstalk, and we'll be using a Python script to read the results out of that queue and act upon them.

The Python script that you'll install later uses the beanstalkc library to read the Beanstalk queue, so install the pip tool and then use it to install beanstalkc and PyYAML with these three lines of code:

```
sudo apt-get install python-pip
pip install PyYAML
pip install beanstalkc
```

The requests library is used to make the POST request to talk to IFTTT, and can be installed like this:

```
sudo apt-get install python-requests
```

Above Test out your on-premises ANPR solution by taking a still photo of your car and then running it through the OpenALPR software at your Raspberry Pi's command line

BIG DATA MEETS LITTLE DATA

The Python script makes use of a small database in which data such as your car number-plate and its current 'home or away' status is stored. To keep resource usage to an absolute minimum, we'll use SQLite for this, which can be installed with the following command:

```
sudo apt-get install sqlite3
```

To create a database called 'anpr.db', type this:

```
sqlite3 anpr.db
```

We can now create a table in which to store our data, like so: →

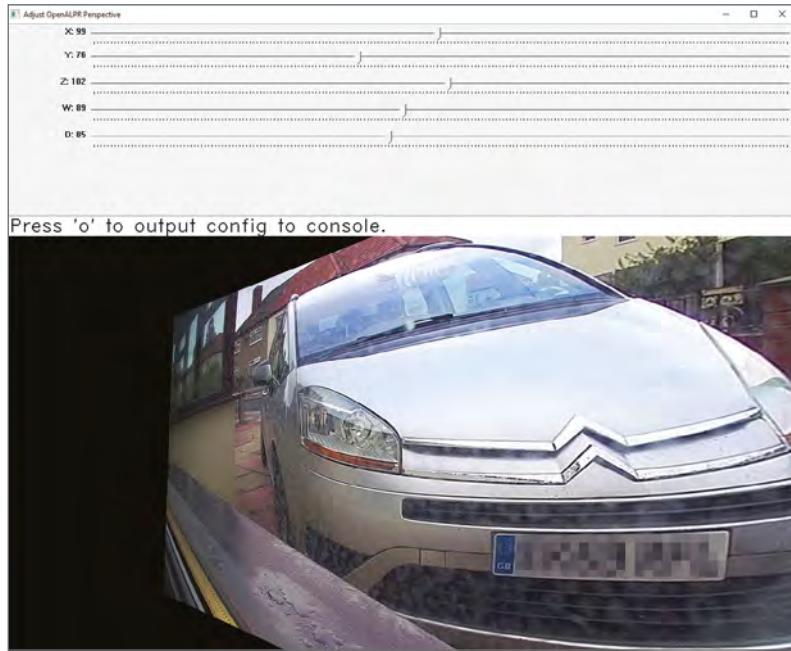
ACCURACY IMPROVEMENT

If your camera is unavoidably placed at an awkward angle, don't worry! You can calibrate the software to warp the image (a bit like keystone adjustment on a projector) so that the car appears as close to straight-on as possible before being scanned for number-plates.

There's loads of information about how to do this on OpenALPR's website:
hsmag.cc/GKEwaX.

Build your own ANPR home automation setup

TUTORIAL



You're nearly ready to run the script, but first it's time to set up the action(s) you want to happen each time you park your car on your driveway

Above Using the calibration utility, you can artificially adjust the perspective of the image which is given to OpenALPR, to make up for less-than-perfect camera angles

```
sqlite> BEGIN;
sqlite> CREATE TABLE anpr_cars(id INTEGER PRIMARY KEY AUTOINCREMENT, car TEXT, isHome INTEGER, lastChangeDate DATE, lastChangeTime TIME);
sqlite> COMMIT;
```

Finally, write an INSERT query to populate the table with the data you'll be using:

```
sqlite> BEGIN;
sqlite> INSERT INTO anpr_cars(car, isHome, lastChangeDate, lastChangeTime) values("PA11LFP", 0, date('now'), time('now'));
sqlite> COMMIT;
sqlite> .quit
```

IF THIS, THEN THAT...

You're nearly ready to run the script, but first it's time to set up the action(s) you want to happen each time you park your car on your driveway. In my case, it's going to turn on the hallway light by my front door, which is a TP-link WiFi-enabled smart light bulb controlled by the Kasa app. Kasa is supported by the IFTTT service, which will act as a bridge between the Raspberry Pi and the light bulb by using webhooks.

To configure webhooks on your IFTTT account, log in and click on My Applets, then Services. Type

'webhooks' in the search box, and then click on the Webhooks icon that comes up. Click Settings in the top-right corner and click Connect. This will generate a unique URL for your account with a key which authorises your script to use your IFTTT account. Copy and paste the URL into your address bar to bring up the instructions page, which briefly explains how to use the webhooks as various triggers in Applets you can create. Essentially the Python script will make a POST request to the IFTTT servers, and specify an {event} followed by the key which both associates the action with your account and authorises the request.

Make a note of the key from this page, as we'll need it in a moment, but first we need to create the Applets – one to turn the light on when the car arrives, and one to turn it off when the car leaves.

Go back to your My Applets page and click on New Applet, then click on the big '+this' to add your 'if this happens...' scenario. Type 'webhook' in the search field and click on the Webhooks icon that comes up. Click on 'Receive a web request' for the trigger and then type your event name. This must match the event name in the script exactly, so type in 'car_arrived' (without the quotes) then click create trigger.

Next, click on the '+that' and search for 'Kasa'. If you've not already connected your TP-Link Kasa account to IFTTT, then you'll be asked to do so. For the action, choose 'Turn on' and all your Kasa devices will be available in a drop-down; select the device you want to turn on – in my case it's Hallway Light. Click Create Action and then Finish. Then, do exactly the same and create an Applet with the event name 'car_gone' and set this to Turn off Hallway Light instead.

Once that's done, install the main Python script:

```
sudo apt-get install git
git clone https://github.com/paulfp/ANPR-Home-Automation.git
```

This will download the script into a new directory called 'ANPR-Home-Automation' within the **pi** user's home directory. Once that's done, open up the Python script with the following command:

```
cd ~/ANPR-Home-Automation
nano anpr.py
```

Look for the line which sets the `ifttt_webhook_key` variable, and paste in the key you noted previously:

```
ifttt_webhook_key = "__YOURIFTTTKEY__"
```

Save and exit the file. You can now run the Python script like this:

```
python ~/ANPR-Home-Automation/anpr.py
```



Now, park your car in your driveway and watch as your light switches on, all by itself! Drive away again to see it turn off.

HOW THE SCRIPT WORKS

The script starts by first connecting to the SQLite database to retrieve the number-plate to watch for, as well as the last recorded home/away state. This is saved in a local variable, as well as being represented by switching on or off the LED which is connected to the Pi's GPIO ports. The script then constantly checks the Beanstalk queue for new Jobs which are inserted by the OpenALPR daemon each time it spots car number-plates in the frame. When processing each Job, it compares the plate found to the one it's looking for. If a match occurs when the car's home/away status is currently 'away', the system considers the car to have just arrived home and switches on the LED, updates the local variable, and makes a POST request to the IFTTT servers triggering the 'car_arrived' event. This in turn, switches on the hallway light via the Kasa web service to your smart light bulb.

The Python script then deletes the current job, clears the queue of any others and sleeps for ten seconds – the ten-second sleep introduces a crude rate-limit into the setup, meaning the lights could never flash on/off irritatingly if something goes wrong, and clearing the queue ensures there's not a huge backlog of results to work through once the script wakes up again.

When the car drives away, the Beanstalk queue should pretty quickly become – and stay – empty, because there's nothing in frame for OpenALPR to find a number-plate in. This is handled in the section of code which starts with the line `if job is None:`: where if the current home/away status is set to home, the system considers the car to have just left and the opposite occurs, eventually triggering the 'car_gone' event via IFTTT which turns the hallway light back off again.

STAYING IN HOUSE

You could, of course, make this completely self-contained without the need for an internet connection, and the use of external services like IFTTT and a Kasa light bulb, if you're comfortable doing a bit more electronics. Rather than using the GPIO ports to turn on the little LED, you could instead turn on/off a relay switch and use that to control lighting or anything else in your house. However, please only attempt this if you're competent and experienced working with mains voltages!

For the sake of simplicity and readability, I've left out important things like logging, proper error handling, and resilience from the script I've shared on GitHub, but if you're building this system for yourself, I would recommend enhancing the code appropriately.

Finally, set the script to auto-run each time your Pi boots. To do this, edit the Pi's autostart configuration file:

```
sudo nano ~/.config/lxsession/LXDE-pi/autostart
```

Add this line at the end of the file:

```
sudo service openalpr-daemon start
python /home/pi/ANPR-Home-Automation/anpr.py
```

The `anpr.py` script will now load each time the Pi boots up, and the ALPR daemon will also be started, just in case it doesn't start automatically.

Now you know how to set up your own automatic number-plate recognition system and tie it in with your own home automation and IoT devices, what are you going to set to happen when you drive your car onto your driveway? We'd love to see what this inspires you to make – get in touch via Twitter to show off your creations! ☺

Above ◆
All you need is a cheap and cheerful IP camera like this one. Connect it to your home network with a cable rather than WiFi, and make sure it's an outdoor one if you're installing it outside

Thinking Pi

Transform your Raspberry Pi into a silicon brain using scikit-learn



Zack Akil

[ZackAkil](#)

Zack is a data scientist focusing on the big problems, such as the proper filming of rugby matches.

YOU'LL NEED

Raspberry Pi

There's a lot of talk flying around about super-powerful artificial intelligence (AI) running on GPU clusters in the cloud. This makes AI sound like it's just for people with their own supercomputers and PhDs.

However, you can easily build your own learning machines on practically any hardware you want. In fact, one of the top machine learning libraries is a simple Python library that can be running on your Raspberry Pi in minutes!

AI can be a bit of a troublesome phrase, not least because no one seems to agree on what intelligence actually is. We're going to focus on one aspect of intelligence: learning. This is the ability to do actions based on what you've observed in the past – or in this case, your Raspberry Pi performing actions based on what input it's previously received.

Building and running machine learning on devices like Raspberry Pis allows you to build clever machines using whatever data you want (public or private), and apply that to applications completely untethered from the internet. A common example is a thermostat that not only lets you control it with a quirky interface, but also learns what temperature to set, based on time of day, humidity, who is in the room, and learnt solely from how you have set the temperature in the past.

Getting into machine learning can seem like a huge undertaking if you have never touched it before. Fortunately, one of the most popular machine learning libraries is founded on the principle of being as simple as possible to use. Scikit-learn is an open-source comprehensive library that has implementations of the most common machine learning algorithms, packaged

in a user-friendly API. Let's install it in onto our Raspberry Pi using pip, (a package installer for Python)!

`pip install -U scikit-learn`

Before we jump into the library, here is a little machine learning intuition.

In the type of machine learning we are going to do (supervised learning), our code is going to learn the pattern that maps a set of inputs to a set of outputs. What does this mean? If you were a botanist (plant expert) who sees a bunch of different flowers that you don't know the name of, and another expert points to a couple of unique flowers and tells you their name, in your head you are learning a mapping between the unique features of the flowers (e.g. colour, shape, smell) and the name of the flower. So, when you see another flower, you can use that 'mapping' to predict the name of the flower, based on its features, without the other expert's help. So the flowers' features (e.g. colour, shape, smell) are the inputs, and the names of the flowers are the outputs. Our machine learning algorithm will be tasked with learning the mapping between them, so that with just inputs (flower features), it can predict the outputs (flower name).

SIGNAL AND NOISE

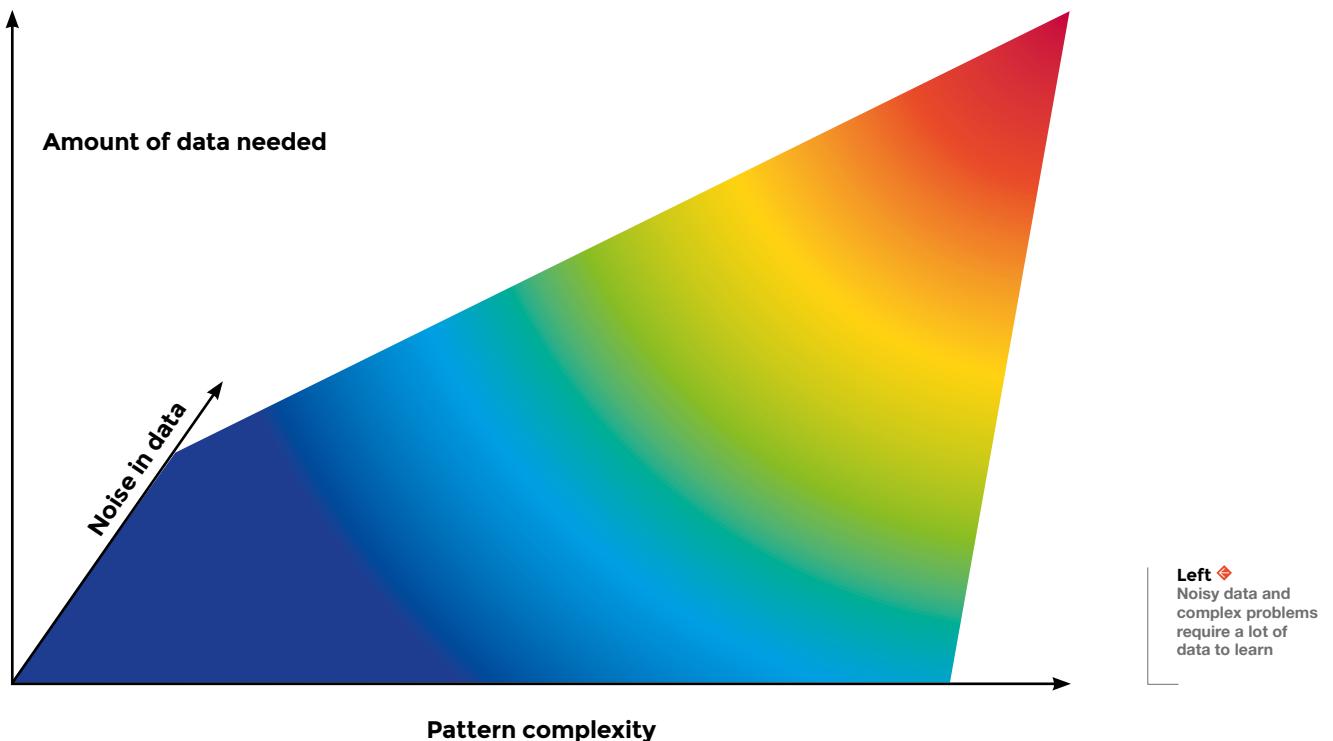
The most important skill in machine learning is in developing a good intuition around 'how much' and 'what kind' of data you need to learn a mapping/pattern. Imagine you have no idea about the rules of Quidditch (maybe you don't have to imagine), and your task is to figure out the rules just by watching it being played. How many matches would it take you? How many matches go to penalty shoot-outs? If you don't see it, you won't know that it's part of the rules, so you'll have to see enough to see how every rule applies to every scenario. What happens if, in some of the matches, the referee is making wrong calls? You will learn the wrong rules, unless you watch enough matches where the referee is making the correct calls to compensate.

Right ♦

Scikit-learn is a hugely powerful library that we'll only just be scratching the surface of. There are many more features if you dig beneath the surface



Machine Learning with Scikit-Learn



In the example of learning the rules of Quidditch based on pure observation, you can imagine that the simpler the rules, the fewer observations you would need in order to learn the rules. The same principle applies to all machine learning: the more complex the problem, the more data you will need to learn it, and vice versa. The other issue of compensating the anti-learning effect of seeing a bad referee by seeing even more good referees, can be mirrored within machine learning as well. It's known as a noisy data problem (signal-to-noise for Nate Silver fans). The noisier the data (i.e. the more examples that don't perfectly match the pattern you are trying to learn), the more data your machine learning model is going to need to compensate for it.

HELLO PETAL

It should come as no surprise that the problem we will be solving using the Raspberry Pi is a simple one with very clean data (and because of that, we won't need very much data). The problem is predicting what type of flower we are looking at, based on different attributes (or 'features' in machine learning lingo) of the flowers, e.g. petal length and width. First, let's get the data!

If you haven't done interactive Python coding before, you're in for a treat! In the Raspberry Pi command line, you open an interactive Python session by typing in and running (pressing **ENTER**) the following:

```
python
```

All commands will now run in Python as they are entered.

Let's import our flower data from scikit-learn. It comes with nice toy datasets to play with.

```
from sklearn.datasets import load_iris
iris = load_iris()
```

`iris.keys()` shows all of the different things stored inside the iris object:

```
iris.keys()
>>> ['target_names', 'data', 'target', 'DESCR',
     'feature_names']
```

`iris['data']` contains the input data (flower features) for 150 different flowers in a matrix, where each row is a different flower, and each column is a different feature. →

Below The first five rows of flower data

sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
5.1	3.5	1.4	0.2
4.9	3	1.4	0.2
4.7	3.2	1.3	0.2
4.6	3.1	1.5	0.2
5	3.6	1.4	0.2

`iris['target']` contains the output flower name (encoded as a 0, 1, or 2) for each flower row of the input data matrix. Run `'iris['target']'` and you can see the first 50 are all 0s, so the first 50 rows from the input data matrix are all the same flower type.

Each number corresponds to the flower names in `iris['target_names']`. 0 is I. setosa, 1 is I. versicolor, and 2 is I. virginica.

“We split the data into training and test data, and also shuffle the data rows so that there is a good mix of each flower in both train and test data”

You can get more information on the dataset by printing out the DESCR of the object:

```
print(iris['DESCR'])
```

OK, so we have our data! The machine learning algorithm that will learn the mapping between the flower features to the flower name will be a decision tree. Decision trees are models that try to build a tree of questions that split the data into the separate classes (flower types).

We'll start by importing the algorithm from scikit-learn.

```
from sklearn.tree import DecisionTreeClassifier
```

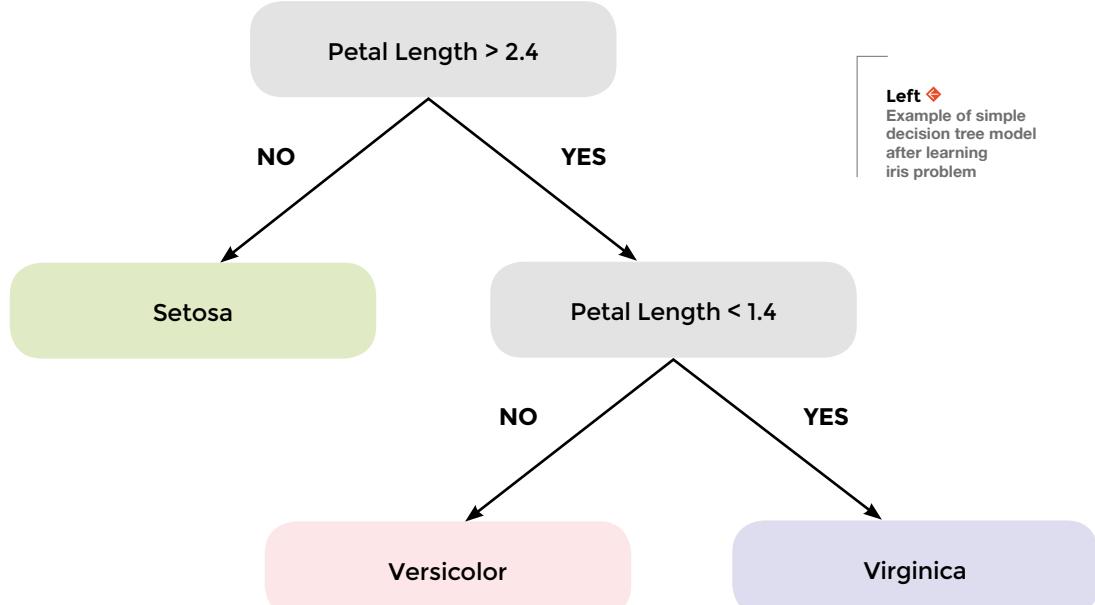
We want to train our model on some of the data, and save a portion of our data for testing. It would not be wise to test a student by just getting them to do an exam they have been practising with and already have the answers for, right? They could just memorise the answers without learning the pattern. So, you give them a different exam that they don't already have the answers to, and compare the answers they 'predict' with the real 'target' answers.

So, we split the data into training and test data, and also shuffle the data rows so that there is a good mix of each flower in both train and test data.

The `train_test_split` function in scikit-learn does all of this for us and, by default, splits the data so that 25% is put into test, and 75% for training.

```
from sklearn.model_selection import train_test_
split
X = iris['data']
y = iris['target']
X_train, X_test, y_train, y_test = train_test_
split(X, y)
```

Now we create a new decision tree model and train/fit it to our training data. This is the training phase, so it gets to look at the inputs (flower features stored in `X_train`) alongside the outputs (flower names stored in `y_train`).



```
model = DecisionTreeClassifier()
model.fit(X_train, y_train)
```

That's it! Now you get predictions on the test data:

```
model.predict(X_test)
```

You can see how many of these it predicts correctly by manually comparing them to the answers stored in `y_test`.

You should find that they're mostly correct, but we can check the model's performance in an easier way. Scikit-learn models have a `.score` function that will tell you what ratio of predictions it gets right for some input data, alongside their answers:

```
model.score(X_test, y_test)
```

We got a result of 0.9473684, which means we were 94% correct. It's not perfect, but pretty good for a few lines of code.

Now, because we are on a Raspberry Pi using Python, we can control anything we want with the output. Some LEDs perhaps? We can import the `gpiozero` library, which will allow us to easily control hardware using Python.

```
from gpiozero import LED
# connect an LED to pin 17 on the Raspberry Pi
led = LED(17)
```

We can create a function that will turn on the LEDs if we predict a flower to be of type setosa (output label of 0).

IRIS DATASET

The iris dataset is one of the most commonly used in machine learning. It was first presented by Ronald Fisher in the paper, *The use of multiple measurements in taxonomic problems*, released in 1936. The four attributes (Sepal Length, Sepal Width, Petal Length, and Petal Width) combine to determine the species, though no one can do it by itself.

It's a great dataset to get started with. If you want to try your new-found machine learning skills with more inputs, there are some other datasets at hsmag.cc/HGGFOa that you can download and use (though some of them may need a little manipulation before they're in a suitable format).



```
def led_on_if_setosa(input_data):
    # input data is a list with the following
    # values:
    #[sepal length (cm), sepal width (cm), petal
    #length (cm), petal width (cm)]
    prediction = model.predict([input_data])[0]

    if prediction == 0:
        led.on()
    else:
        led.off()
```

Let's see if we found a setosa flower:

```
led_on_if_setosa([ 4.7,  3.2,  1.3,  0.2])
```

Hopefully your LED has turned on.

This is the basic structure for code to bring machine learning to physical computing projects. You need some training data, a learning algorithm, and a way of performing actions depending on predictions. If an automatic flower identifying kit isn't what you're after, then you can send almost any type of data into this system (provided it's not too noisy). Temperature and other environmental sensors can work really well, but it depends on what you want to control. ☐

Above ♦
Iris versicolor, also known as the Blue Flag or Purple Iris, is the official flower of Quebec

Credit
Danielle Langlois
CC-BY-SA

DON'T MISS THE BRAND NEW ISSUE!



SUBSCRIBE FROM JUST £5

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



THE GREATEST
RASPBERRY PI
PROJECTS

FREE PI ZERO W
With your 12-month subscription to the print magazine
magpi.cc/Subs1

PLUS! FREE
CASE, THREE
COVERS &
CABLES



Buy online: store.rpipress.cc

FIELD TEST

HACK | MAKE | BUILD | CREATE

Hacker gear poked, prodded, taken apart, and investigated

PG
114

DIRECT FROM SHENZHEN: **MULTIMETERS**

Measuring electricity without
breaking the bank

PG
122

CAN I HACK IT?

A musical multicoloured bulb
becomes the parts for our
next project

PG
116

BEST OF BREED



Lots of LEDs! If you're drawn like a
moth to a flame, these are the kits and
components you need

REVIEWS

124 Anet A8

A budget printer for tinkerers

124 SparkFun Toolkit

All you need to build electronics

128 Retro WiFi

Adding WiFi to old computers



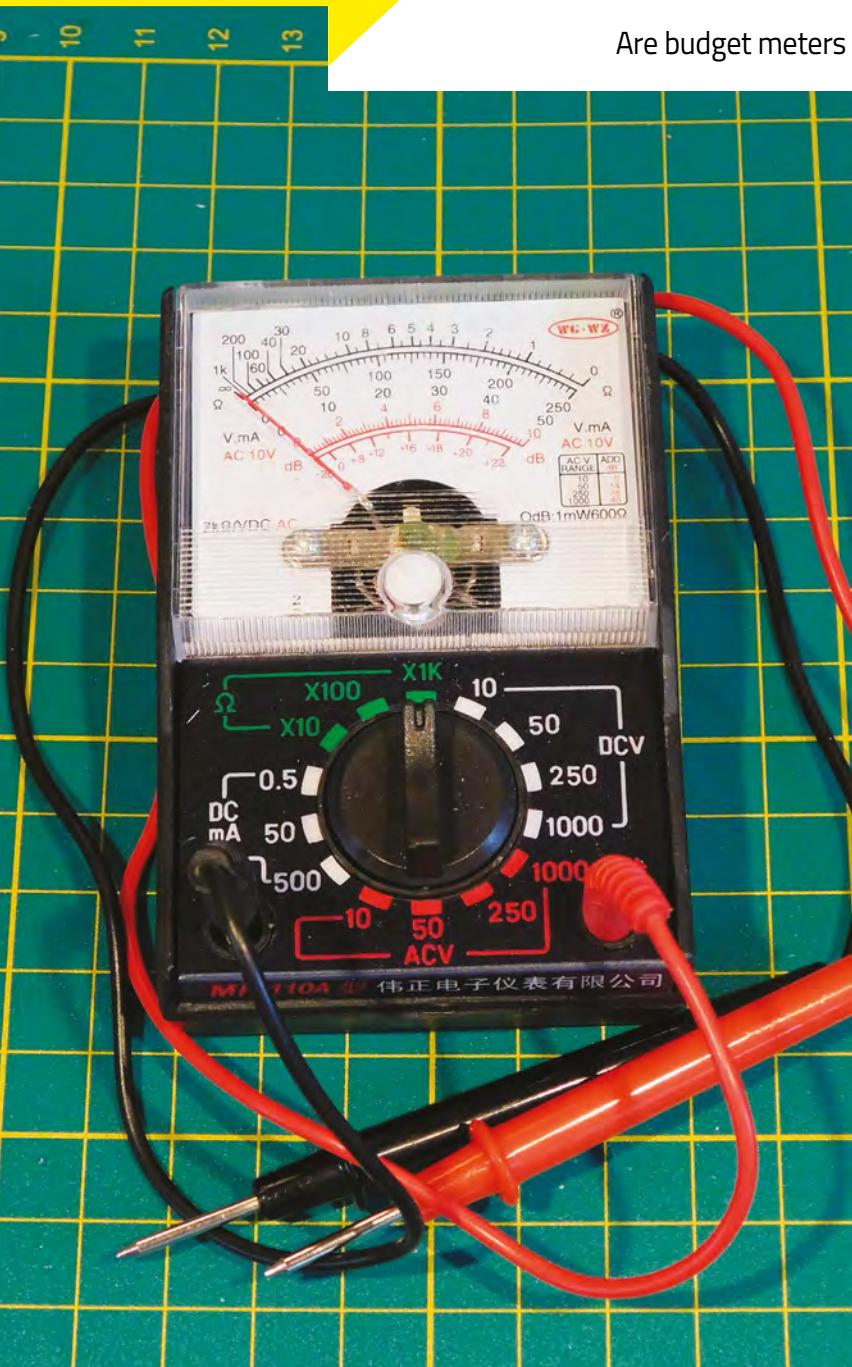
129 Woodworker's Manual

Build things the old-fashioned way

DIRECT FROM
SHENZHEN

Cheap multimeters

Are budget meters good value for money?



By Ben Everard

 @ben_everard

If there's one tool that's almost indispensable for electronics, it's a **multimeter**. These electronic Swiss Army knives are often the first thing we reach for when we have a problem with a circuit. Without one, it's hard to find non-conducting

soldered joints, dodgy power supplies, or any of a myriad of gremlins that can work their way into an electrical circuit. Working without a multimeter would be like trying to program without print statements (or a debugger) – it's not impossible, and if you're determined enough you can do it, but you're going to have a lot of unnecessary headaches along the way.

There are no set features a multimeter has to have, but generally, the absolute minimum is a voltmeter, an ammeter, and an ohmmeter. Beyond this, there's a wide range of features that can find their way into multimeters, including capacitance measurement, frequency measuring, and auto-ranging. Multimeters can cost thousands of pounds, but hobbyist-level ones are quite a bit cheaper. For example, the best-selling multimeter on Amazon UK costs £28.99, while the cheapest one was £7, at the time of writing.

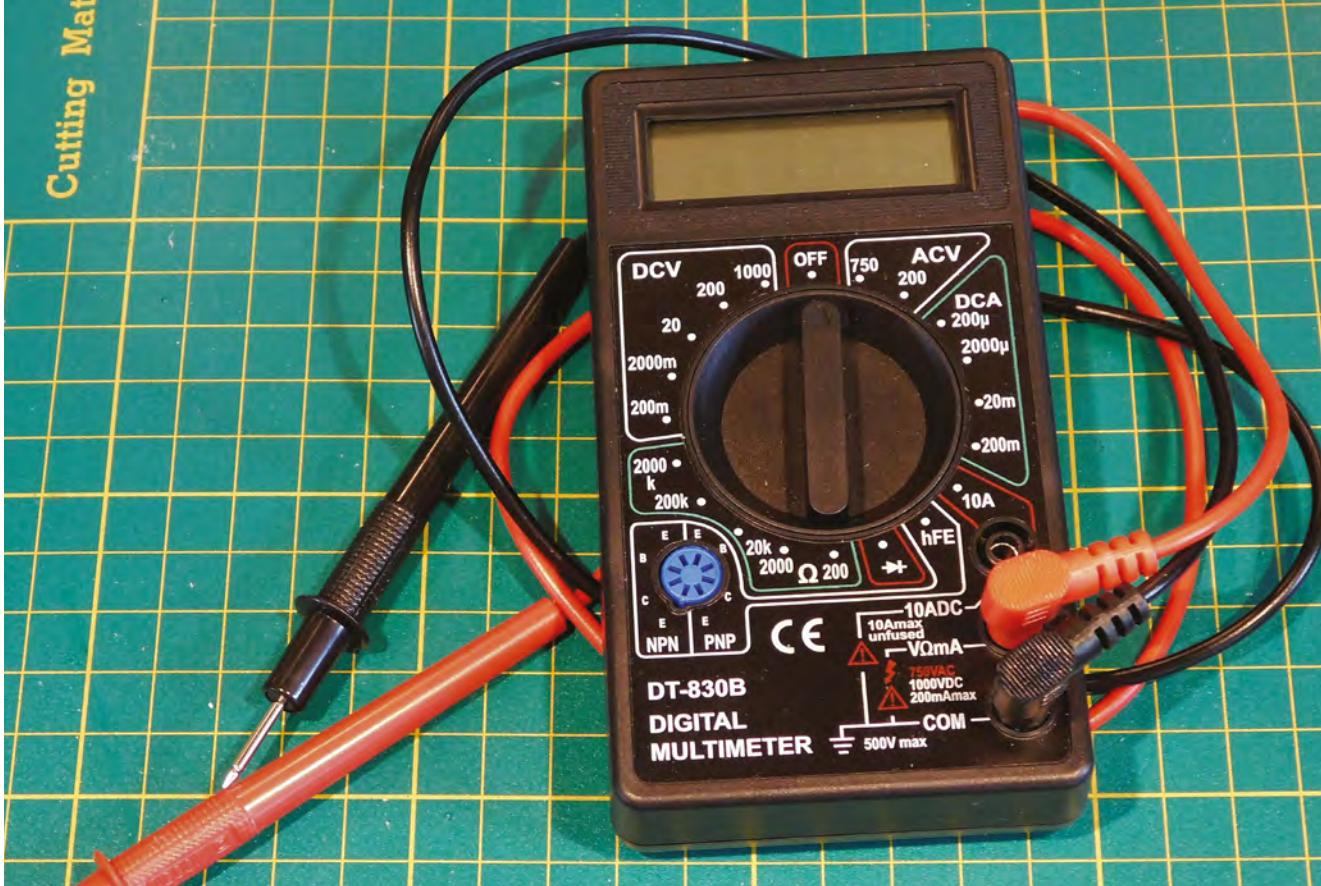
You can get cheaper meters on direct-from-China websites, and we set out to try the two cheapest options we could find.

PICK OF THE BUNCH

We bought two from AliExpress: an analogue multimeter from NightStar Store for £2.09, and a digital multimeter from ATTOKO Store for £2.29 (both including delivery to the UK). Can you really get a working meter for this little?

Left

Much as we love the look of analogue meters, they're just not as easy to use as their digital counterparts



The analogue meter has the three basic functions that we'd consider essential for a multimeter – a voltmeter, ammeter, and ohmmeter – but nothing else. The digital multimeter also has diode and transistor checkers.

The readings aren't perfectly accurate on either meter, but should be good enough for basic hobby usage. It'll tell you how much power is being supplied, what value a resistor is (if you're not sure on your bands or are colour-blind), and whether or not there's a good connection between two points (using the ohmmeter, as there's no continuity checker on these meters). The biggest problem on the analogue meter is reading the display. The size of the needle on the small display means that it's often hard to read it to within about 5% accuracy. That's not the only problem with the analogue meter, though: it's also got quite a low limit on the current it can read – 500mA. This isn't necessarily a deal breaker because the current limits are several times higher than the amount of current you can use with the I/O pins on a Raspberry Pi or Arduino, but at the same time, they're much lower than you'd need to drive a motor or a large group of NeoPixels. The digital meter has a fused input up to 200mA, but the unfused input claims to go up to 10A. We tested this to our power supply's limit of 3 amps, and it worked without problems, but we wouldn't be confident pushing it to the top of this range.

COST SENSITIVE

There's no getting around the fact that these meters are at the budget end. They're both small and flimsy, and the leads don't give us much confidence that they'd last years of heavy use. That said, they did

last through our tests and, if they're not roughly treated, they should survive home use (we wouldn't recommend for a heavily used workshop, though).

According to the voltage ratings, the analogue meter can work with voltages up to 1000 volts (both AC and DC). On the digital meter, the voltages can go up to 1000V DC and 750V AC. We don't have a voltage supply high enough to test this, but we're extremely suspicious about the claimed voltage range. If you're working with voltages high enough to shock you, you need equipment that's well built enough to protect you. These are both flimsy plastic, with thin wires, and not devices we'd be willing to use anywhere near this range. At the low voltages and currents of hobbyist electronics, though, there's no risk here. As a basic rule of thumb – if it's safe enough to plug into a breadboard, it's safe enough to use with an untrusted multimeter.

There's a certain charm to the analogue meter but, ultimately, it's just not a particularly practical measurement tool. The world has moved to digital meters with good reason. If you want something to add a bit of interest to a steampunk build, then it's got plenty of potential, but as a measurement tool, we can't recommend it.

If you're just getting started in electronics and you're after a cheap meter, then the digital option here is a good choice. There are plenty of features it's missing – such as a capacitance meter, auto-ranging, and continuity buzzer – but none of these are essential and, if you're on a budget, it might be better to spend the extra money elsewhere. You'll almost certainly outgrow this meter at some point, but that's fine – you'll probably outgrow your first multimeter anyway, and at this price it feels a lot less wasteful. □

Above
When first starting out, simple cost-effective equipment is often the best option

DIRECT FROM SHENZHEN

**ONLY THE
BEST**

LEDs, lots, and lots of LEDs

Everyone's favourite electroluminescent phenomenon

By Marc de Vinck

 @devinck

From its humble beginnings in the early 1960s, the light-emitting diode, or LED, has become a favourite component for electrical engineers, artists, and even kids. It's hard to imagine a piece of electronics that doesn't have at least one LED indicating its power. What's even more surprising is that this electroluminescence phenomenon was discovered over 100 years ago, in the early 1900s.

It's no wonder the LED has become so popular, from the fact they exponentially outlast incandescent light bulbs, come in a variety of colours, including smart LEDs that can be programmed to represent almost any colour of the rainbow, and they are incredibly inexpensive. What's not to love?

In this Best of Breed, We'll be looking at some of our favourite products that contain lots of LEDs (LoL). Some have hundreds of LEDs, and others only feature a handful, but all of them are reliant on the usefulness and beauty of the humble LED.

FUN FACT

Everyone knows that the LED emits light, but did you know that it can detect light too? Through the use of a microcontroller and carefully controlling the power, creating a reverse bias of the circuit, you can actually sense light and dark. It's not perfect, but it's still a fun hack of a standard LED. There's lots of information online about how to build the circuit – just don't expect it to replace a photodiode, as its sensitivity is not as good.



Left ♣
You can never have too many LEDs

LED Charlieplexed Matrix vs SparkFun LED Array – 8x7

Making LED grids easy to control

CHARLIEPLEXED \$6.95 | adafruit.com

SPARKFUN \$8.95 | sparkfun.com

Adding lots of LEDs in a Charlieplexed arrangement is fairly simple, though a little tedious, when you are using through-hole components like 5mm LEDs.

But what happens when you want those LEDs to be really small and packed together so you can create a higher-resolution display? That's where the LED Charlieplexed Matrix comes in handy. We can't imagine soldering all those LEDs in place, especially when they are that close together.

To control this many LEDs effectively, you need a backpack, or a breakout board like Adafruit's 16x9 Charlieplexed PWM LED Matrix Driver. Although this matrix needs another driver board to work, it's well worth the additional hardware. Once connected, you have complete control over each individual LED, including its brightness, which is not something you see that often in a Charlieplexing array. This allows for some smooth animations and effects. The LED array is available in six colours, and Adafruit has a great tutorial including code examples and more! ☐



The SparkFun LED Array is a set of 56 red LEDs arranged in a nice 8x7 grid. This little board requires eight pins, while the library supports any ATmega168- or 328-based Arduino device. The SparkFun LED Array

was developed to be used in conjunction with the BadgerStick as a fun and interactive way for people to learn about soldering, and engage with SparkFun products at the multiple trade shows they appeared at. Now this array is available to everyone!

The SparkFun LED Array relies on the Charlieplexing to control individual LEDs. This is a very popular technique that allows you to control all 56 LEDs with just eight pins of your microcontroller. The layout of the board makes it a perfect addition to the BadgerStick, now known as the SparkFun RedStick. As mentioned, SparkFun has developed a library for the array, supporting both ATmega168- and 328-based Arduino platforms, making it easy to implement into your next project. ☐

Left 56 LEDs neatly organized in an 8x7 grid

Below 144 LEDs. What's not to love?

VERDICT

LED Charlieplexed Matrix

Great price, but be aware that it needs a driver board.

9/10

SparkFun LED Array

You can control all 56 LEDs with only eight pins.

8/10

BEST OF BREED

Bulbdial Clock – monochrome kit

A sundial that works indoors

BULBDIAL CLOCK ⬤ \$65 | shop.evilmadscientist.com

Evil Mad Scientist wants to make telling time a unique experience, and it has definitely achieved that goal with its **Bulbdial Clock kit**. The original concept for the clock was by David Friedman of Ironic Sans. It was an idea of an electric sundial clock, where the LEDs replace the sun and can accurately display the time of day. No need for hands, just shadows cast from a centre peg.

Dreaming up such a clock is one thing; making it a manufacturable product is a whole other exercise in product development. That's where Evil Mad Scientist steps in and makes the concept a reality. In typical Evil Mad Scientist form, the kit is exquisite in execution. You have a choice of buying just the clock

components in kit form, or you can add one of several different cases. If accuracy is really your thing, there is also the option to add an external higher-accuracy quartz crystal oscillator, which has the added benefit of being a battery backup. □

Dreaming up such a clock is one thing; making it a manufacturable product is a whole other exercise in product development



Left ⬤
This clock has no physical hands, just shadows cast from LEDs

VERDICT
A truly unique timekeeping experience.

9 /10

SpikenzieLabs Dice Kit

Chance your luck

SPIKENZIELABS \$19.95 | spikenzielabs.com

Simple analogue dice work when playing your favourite board game, but when you really want to impress your friends, take out the LED Dice Kit from SpikenzieLabs and get playing. Most beginner soldering kits come supplied with a circuit board and a bunch of parts which once soldered together can be fairly fragile, and honestly, a bit unappealing. Not this kit! SpikenzieLabs has designed a really nice acrylic base which allows you to proudly display your kit, as well as actually use it.

With a simple tap on the table, the LEDs will light up depicting the 'roll' of the dice. And those aren't



Left Yes, even dice need LEDs! Big LEDs!

VERDICT
Seven large LEDs and a fun user interface make this a great first soldering kit.

L 10/10

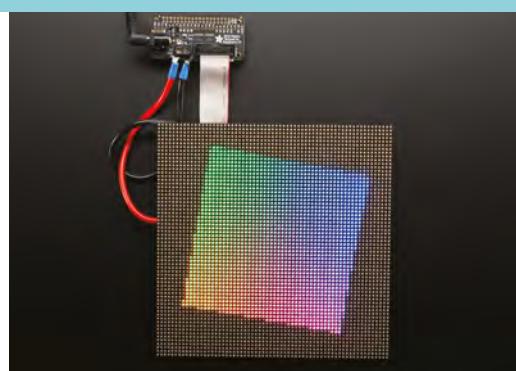
Adafruit 64x64 RGB LED Matrix

Four thousand lights

ADAFRUIT \$74.95 | adafruit.com

When you need a lot of LEDs, the Adafruit 64x64 RGB LED Matrix, with its 4096 LEDs, aims to please. Because of the fine pitch, or 2.5mm space between LEDs, and fast refresh

rate, you essentially have a low-resolution full-colour TV screen. No, it's not 8K or even 4K, but it's totally retro cool! It can easily display full-colour animations when controlled by your Raspberry Pi. Unfortunately, we can't show you just how beautiful this LED panel is in a print magazine, but for now you'll just have to trust us, or head over to the Adafruit website where they have a video demo.



Left Imagine being able to control 4096 RGB LEDs!

VERDICT
When you need a lot of RGB LEDs, this is it!

L 9/10

BEST OF BREED

Unicorn HAT HD

Make your Raspberry Pi glow

UNICORN  \$42 | shop.pimoroni.com

We were big fans of the original Unicorn HAT with its 64 RGB LEDs, but it pales in comparison to the new and improved Unicorn HAT HD, boasting 256 tightly packed

RGB LEDs. With four times as many pixels as the original Unicorn HAT, you are able to generate much more complex and smooth animations.

The board features an ARM STM32F and three LED drivers. These components take care of all the fast processing needed to control all 256 LEDs. This allows your Raspberry Pi to do other things, rather than worry about the refresh rate of a bunch of



Left  256 LEDs all packed into this HAT

VERDICT

More LEDs than we thought was possible to pack into a HAT form factor board.

10/10

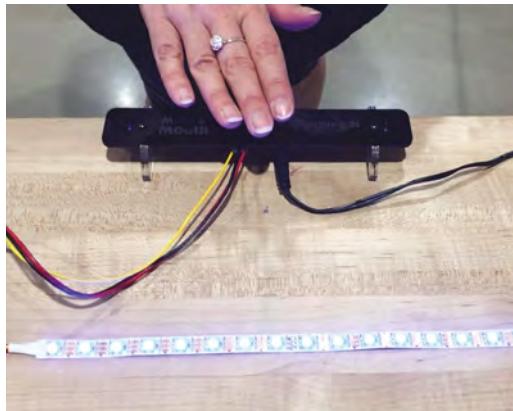
Rainbow Light Show

Dancing lights

RAINBOW LIGHT SHOW  \$29.95 | spikenzielabs.com

The Rainbow Light Show from SpikenzieLabs is a fun kit that allows you to control a strip of RGB LEDs with a simple wave of your hand.

The kit is available in a traditional bring-your-own-soldering iron version, or you can pick it up in a simple no-solder version too. Once assembled, both kits allow you to adjust the red, green, and blue colour mixing of the LED strip. The kit comes with a strip of 16 addressable LEDs, but it's expandable up to 64 in total. That's enough to light up a room! ☐



Below  Make your own light show with a wave of your hand

VERDICT

A fun way to create a light show with the swipe of your hands.

8/10

5x5x5 – Cube:Bit Magical RGB Cubes of Awesome

Lighting up three dimensions

CUBE:BIT MAGICAL ◇ \$27–\$117 | shop.pimoroni.com

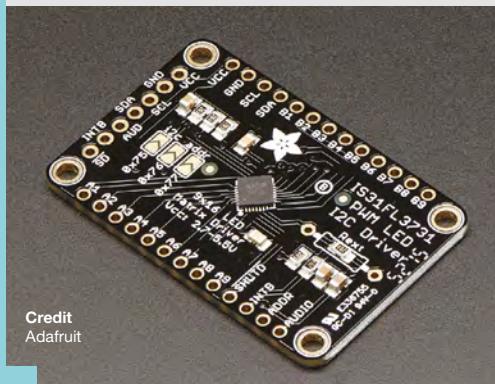
If you are unfamiliar with the micro:bit platform, you should really check it out. It's a pocket-sized device powered by an ARM Cortex-M0 processor. You can do a lot with just a micro:bit, but it really shines once you add accessories. One great example is the 5x5x5 – Cube:Bit Magical RGB Cubes of Awesome.

Inspired by the work of Lorraine Underwood we featured in issue 9, this kit allows you to build a 3D array of RGB LEDs and control them from the

ADAFRUIT 16×9 CHARLIEPLEXED PWM LED MATRIX DRIVER

\$5.95 | adafruit.com

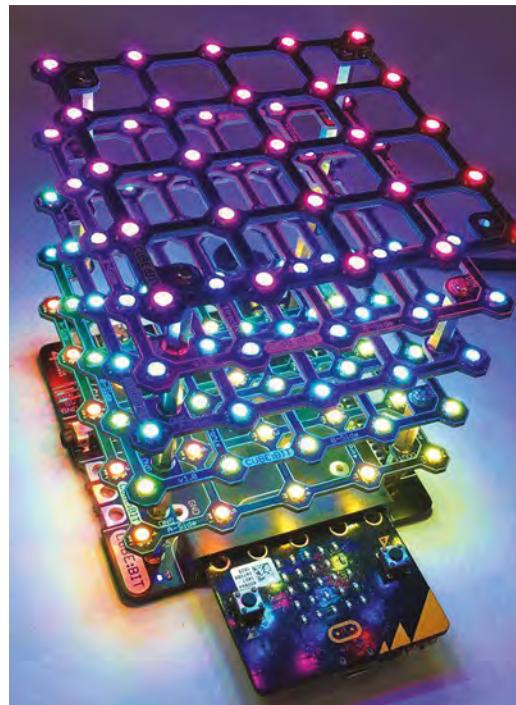
Controlling an LED matrix isn't too difficult if it's only a few LEDs. As the matrix grows, you soon run out of pins or the code just gets really complicated. That's where the Adafruit 16×9 Charlieplexed PWM LED Matrix Driver really shines. This board allows you to connect a 16×9 LED matrix and easily control each individual LED, including its brightness. Try doing that with just your microcontroller!



Credit
Adafruit

micro:bit. The kit is available in a 3×3×3, 4×4×4, or enormous 5×5×5 configuration. Regardless of which size you choose, the cube will have RGB LEDs on both sides, creating an amazing light show that is easily controlled through a drag-and-drop interface of MakeCode. □

“ Regardless of which size you choose, the cube will have RGB LEDs on both sides, creating an amazing light show that is easily controlled through a drag-and-drop interface of MakeCode **”**



Left ◇
This breakout board is perfect for working with LED matrices

VERDICT
A variety of sizes are available depending on your budget.

8/10

CAN I HACK IT? ■

Can I Hack It?

A Bluetooth RGB LED light bulb?

Can we harvest a smart bulb for parts?



Les Pounder

@biglesp

Les Pounder loves taking things to pieces and seeing how they work. He teaches others as part of the Raspberry Pi Foundation's Picademy event. He blogs over at biglesp.com

YOU'LL NEED

❖ Elelight LED E27 12 W 220 V Music Bulb

❖ COST £12.99

❖ WHERE hsmag.cc/YqWGpt

Right ❖ Inside this innocent bulb there are five components that we can build into much greater projects

W

hat can you get for £13? Well, it seems quite a lot if you are prepared to tear something apart. This issue we take a look at a cheap Bluetooth speaker embedded in an RGB LED bulb.

Before we get into the teardown, a word of warning. If you plug in this bulb *before* the teardown, there is a good chance that the capacitors used in the conversion of AC to DC power on the adapter board will still have a charge and could give you an electric shock. So, do not plug it in before taking them apart unless you fully understand the appropriate ways of working with mains electrics and high voltage capacitors (you should still take every precaution during the teardown). If you have never handled components at mains power, ask someone who has been properly trained to help you. With the warning over, let's take it apart...

GENERAL CONSTRUCTION

The bulb is made from strong plastic that is hard to work with using hand tools. The LED light diffuser is a softer plastic, but not easy to work with. Removing the diffuser requires turning the bulb until a click is heard. This releases the diffuser, but you may need to use 'spudgers' to lever the plastic apart, as our bulb also had a sealing layer of glue.

POWER

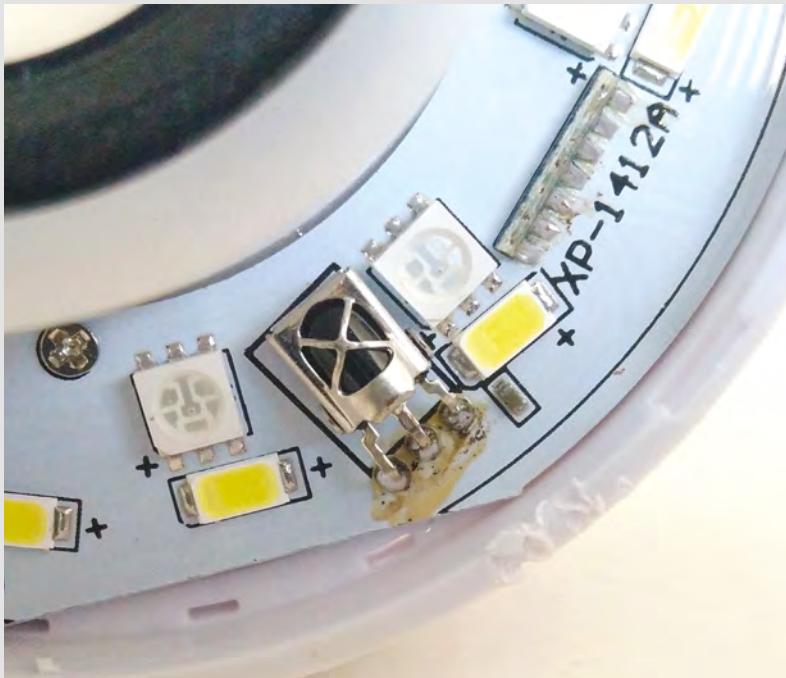
As this is a UK bulb, it is powered from 240V AC mains power. This is via a step down buck power converter board, which converts 240V AC to 5V DC. So we can desolder the board and put it safely in the bits box for a future project. (Again, don't plug this into the mains before the teardown

as the capacitors will hold a charge.) We can safely power the lights and the Bluetooth speaker using just a 5V 1.5A supply. In fact if we removed the lights, we can power just the Bluetooth audio and speaker with 400mA, which can be easily supplied from a standard USB port. But if we want to keep the LEDs, then we really need to use a higher capacity power supply as the white LED ring pulls around 900mA of current, red 530mA, green 530mA, and blue 510mA. This is as well as the 400mA pulled for the Bluetooth audio.

ELECTRONICS

We are very lucky as this bulb is a goldmine for parts! In the bulb we have an infrared-controlled RGB LED ring, a configured Bluetooth audio device that works





with phones, computers, and your Raspberry Pi. We have the buck voltage converter that we can safely remove and use in another project. Oh, and if we take the bulb apart carefully, we can reuse that in a low-power project.

So, we can desolder the light ring from the Bluetooth / control board and embed the speaker and board inside our own homemade speaker case. The solder on the pins is rather crusty, so use some fresh solder to refresh the old. Using a LiPo charger and battery, we can enable the speaker to be portable. You could repurpose a Poundland £1 USB battery for this task, but be aware that it has a max output of 1 A, so using the LEDs along with Bluetooth is a big no! Adafruit also has its PowerBoost 1000 charger which can charge a LiPo battery and provide power to the device, but again this has a 1 A max output.

The light ring operates at 5 V and we have two rings to control. The first is a simple ring of white LEDs that have direct connections to the pins that connect the Bluetooth control board to the ring. The second is an RGB ring which we can easily control once we work out the pins, nothing a little multimeter work won't solve! The LEDs also get warm, and use the aluminium sheet they are attached to as a heatsink. So be careful holding the light, or integrating into any projects, as it will get scorching hot when the white LEDs are used for a long period.

The 4 Ω 3 W speaker is easily replaced and comes with a two-pin JST connector, so we can swap without soldering. The supplied speaker is decent, but it didn't cope very well with our 1990s dance playlist. Spoken word is clear and we can really up

the volume. Oh, and volume control is via the device playing the audio – there is no audio control on the board.

You get a lot of tech for your money. Something to convert power, two LED light rings, infrared control, and Bluetooth audio. The preconfigured Bluetooth audio is exceptionally handy for makers who want to drop that feature into their projects. The LED light rings are not WS2812 NeoPixels, but they are bright and easy to use, thanks to the infrared controls, so this can be easily used in your projects.

We love this little bulb – a really nice surprise that we can power it from a safe 5 V supply and it gives us so much potential for projects, for so little money. □

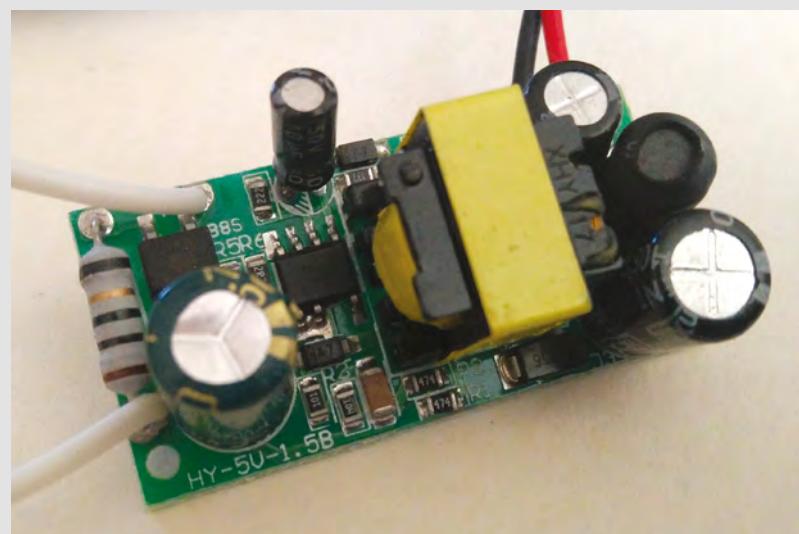
HOLDING CHARGE

The bulb in this teardown was designed for use with the UK mains supply, 240 V, and this voltage is dangerous if not handled correctly. If the bulb were to be plugged in before our teardown, then the capacitors on the buck converter may still hold a charge that could cause you harm. To mitigate this risk, we advise that you do not connect the bulb to the mains, and that the teardown is the first thing to be done! We can't stress enough that you should treat the capacitors as holding a charge and take care when removing and discharging them. If you have never worked with high voltage components, then perform this teardown with someone who is suitably experienced. Mains voltage is not to be played with, and capacitors can store a charge over time and even if they have a resistor connected to the AC input voltage.

Here at HackSpace magazine, we want you to hack safely!

Above The outer white LEDs and the inner RGB LEDs are both powered from a 5 V supply. But, they cannot be used together

Below This simple-looking board converts 240 V AC to 5 V DC, and it can be used to power your next project



Anet Prusa A8

3D printing that doesn't break the bank

ANET ♦ AROUND £100 | anet3d.com

By Glenn Horan

@batglen13

The RepRap (replicating rapid prototyper) movement was started in 2005, at the University of Bath, as a means of fulfilling what sounds like something out of the realms of science fiction – to design and build a 3D printer that can print its own parts, thus being able to replicate itself. Undoubtedly, the most popular example of this paradigm is the Prusa-style design, epitomised by the 'Original Prusa i3', currently on its third iteration. The Anet Prusa A8 is a printer built on the same open-source design, but at approximately one seventh of the price.

This is a printer kit with a ~10-hour build-time. If you are looking for something that works out of the box, the A8 is not for you. All the tools you

It comes with a heated bed out-of-the-box, which improves bed adhesion and allows users to print in ABS and other materials that are prone to warping

need for the build are included in the box, and there are both written instructions and a link to a set of YouTube videos, where you can follow along with an instructional build of the printer. We found building the printer an enjoyable process, and it forces you to acquire knowledge regarding the intimate workings of your new toy.

While the A8 doesn't have all the bells and whistles of the 'Original Prusa', it has all the important features that a decent 3D printer should have. The print volume



is 22cm x 22cm x 24cm, which is enough for a wide range of hobbyist needs. It comes with a heated bed out of the box, which improves bed adhesion and allows users to print in ABS and other materials that are prone to warping. The A8 also supports printing from an SD card, meaning you can add your files from your software of choice to the SD card, plug it into your printer, then select them using the LCD screen. While this doesn't sound wildly exciting, the alternative is to have your machine plugged into a computer at all times. This feature gives you the freedom to place your printer anywhere in the house (pending partner/housemate approval of course).

A QUALITY PRINT

Getting down to brass tacks, the print quality from this kit is not just good for the price, it's plain good. You can see the quality it's capable of in our first ever print, and we've never had any major issues with print quality that haven't been fixed with some form of calibration, or by replacing cheap parts. You will quickly learn there are a plethora of factors that affect print quality that could be the subject of a series of articles in

Above The Anet Prusa A8 is a basic, but capable, 3D printer for people who like to tinker

themselves, but none of these are unique to the A8 and they affect virtually all 3D printers, regardless of the price. We've had no issues printing in PLA, and we've had good results printing in ABS, with a little bit of tinkering in my slicer settings.

This brings me back to a point we made earlier; because this is a kit that has been built from scratch, you will know your way around this printer when it comes to tinkering, upgrades, and (almost inevitably) troubleshooting. Belt not tight enough? Easy – clip this cable tie, tighten the belt, put on a new cable tie. Clogged extruder? No problem – take it apart again and remove the clog.

COMMUNITY SUPPORT

If you do decide to delve into the world of budget 3D printing with the A8, you're absolutely not going to be on your own. This model has a huge and active community of people happy to give you advice. The reddit page for this model alone has 1800 subscribers, Facebook has a number of groups, the biggest with 5800 members actively sharing their latest prints, tips, and upgrades, as well as troubleshooting any and all issues that could crop up.

Regarding upgrades, there are literally hundreds of free designs online that you can print to improve the A8. One of our favourites was the noise-dampening feet that sit under our printer (so effective that they

used to trick the author into thinking the printer wasn't running when he was in the next room). For those who wish to dive a little deeper into the firmware, you can buy a third-party auto-levelling probe and set it up to work with this printer, along with a custom-designed mount, saving both time and resulting in the perfect first layer every print.

This review wouldn't be complete without mentioning a key safety issue of this printer as supplied: the heating rod for the extruder block is not securely held in place. If it falls out, it will keep heating, causing an obvious fire hazard. As bad as this sounds, it is easily fixed either by updating the firmware on the board to introduce thermal runaway protection (a feature added in a more recent edition of the firmware), or by taping the heat block into place using heat-resistant tape (you could also easily and cheaply replace the extruder block altogether).

This printer is a fantastic first 3D printer for people who like to tinker. It does have its issues out of the box, and you will be endlessly tinkering with it to try to squeeze the very best out of it. What you save in money, you'll pay in time. If you want to print perfect prints every time with minimal input, you will be better served by a different machine. If you want to learn about 3D printing, and have something you can build and customise that produces surprisingly good results, the A8 should be on your shopping list. □



Above The author's first ever print – a low-polygon, but high-resolution fox

Design
slavikk

Left A full-size Master Chief helmet printed by the Anet A8 in around 16 different pieces – sanded, primed, and ready for painting

Design
big_red_frog

VERDICT
A great hacker's printer at an almost too good to be true price.

L **8** /10

SparkFun Deluxe Tool Kit

A workshop in a box

SPARKFUN ♦ \$224.95 | sparkfun.com

By Ben Everard

 @ben_everard

If you're kitting out a workshop for electronics, what do you need? Well, a soldering iron – that's a must of course – then there's tweezers and diagonal cutters.

Oh, a pad can be useful. There's that thing of course, you know the one, it does the gripping in different angles – what's it called? Are we forgetting anything? Wouldn't it be useful if someone just put everything together into one place? That's exactly what SparkFun has done with its two tool-kits: the Beginner's Tool Kit (\$49.95) and the Deluxe Tool Kit (\$224.95). We took a look at the Deluxe version, which includes a Weller soldering station, heat gun, multimeter, and 20 other little bits and pieces that you need to make great electronics.

Keeping a kit like this affordable, while complete, is a balancing act. The hardware needs to be good enough to provide users with a pleasant experience, but at the same time, keep the cost manageable. Everything in this kit works and is perfectly good enough to get started with, but there's not many bells and whistles, and at this price, that's what we'd expect – after all, you could spend more than the price of this kit on just a multimeter. Looking at this multimeter, for example, it has all the basic features that you need for electronics, but if you're using it regularly, getting an auto-ranging meter can be useful. The soldering iron is a decent iron with adjustable power, but for trickier jobs, better temperature control may be

Right ♦
Two spools of solder wick are included because we all make mistakes a little more frequently than we'd like



more useful. Conversely, the third hand, for instance, is much nicer than the cheaper options, with hinged metal joints.

MAKING COMPROMISES

Knowing where to draw the line on price and functionality is a difficult skill. This kit has clearly been put together by people who are (or have been) hobbyists, and know what you need. You could get this and never need another bit of kit for building electronics, other than perhaps replacing the consumables like wire and solder – well, the wire anyway, the 113g (or quarter of a pound if you prefer) of special-blend lead-free solder should last you quite a while.

There are a few big-ticket items that more advanced hobbyists might use that aren't included (such as a bench power supply and an oscilloscope), but these aren't needed for getting started, and would add quite a lot of cost to the kit. These are also things that we got quite far without having, and certainly aren't essential to hobby electronics.

The big advantage of buying things in a kit is that you get the bits you need even if you don't know they're the bits you need, and this kit really delivers in this area. It's nice to see little things like the screwdriver or craft knife that could easily be missed out of a soldering kit, but at the same time, they're things you need. Sure, you might have them at home, but if you're kitting out a workshop, having them all together in one package is ideal. There's even a small water bottle included, for keeping water to top up your soldering iron cleaning pad. Another slightly surprising, but useful, entry is the heat gun. Again, this isn't something you'll typically see in electronics kits, but is actually a useful bit of hardware for a couple of tasks: heat-shrink tubing, and hot-air rework. The former of these is basically an upgrade to electrical tape for keeping your connections insulated, and the latter helps for fixing mistakes or breaks in PCBs, particularly with surface-mount components (it's not a full-on hot-air rework station, but can just about be pressed into service for simple jobs).

HIGH VOLTAGE

People outside of the USA, be warned, the soldering iron and heat gun are only rated to 110V, so you'll need a transformer, as well as a socket adapter, if using it somewhere with different voltages.

You can buy most of the parts from SparkFun separately (only the water bottle isn't available), but



it is (by our count) about 10% cheaper to get the kit rather than buying the individual components, so even if there are one or two bits you don't need, it may still be cheaper to get the kit and then lend these to a friend in need.

This kit only contains the bits for putting together electronics, and not the bits for electronics themselves. There are no resistors, Arduinos, cables, breadboards, or anything like that. This kit

Above The Beginner's Tool Kit has everything you need to get you started

“ **The big advantage of buying things in a kit is that you get the bits you need even if you don't know they're the bits you need, and this kit really delivers in this area**

is for putting electronics together, not designing or prototyping them, so it's the perfect companion to soldering kits or prototyping kits (such as the SparkFun Inventor's Kit).

Many people will prefer to build up their toolset bit by bit, and this often makes sense if you're just starting out with something new (you might want to take a look at the SparkFun Beginner's Kit). As you continue in electronics, the Deluxe kit is a good point of reference for extra bits and pieces that you might want to expand your fledgling workshop. However, if you're looking to kit out a workshop with a cost-effective electronics kit, this kit should give you everything you need to work through most basic and intermediate electronics projects. □

VERDICT
Everything you need to get started building electronics.

L 9 /10

Simulant Retro WiFi SI

Re-live those misspent nights of bulletin board chats

SIMULANT £32.99 | simulant.uk

By Graham Morrison | [@degville](https://twitter.com/degville)

t's a dark and rainy night in the mid-1980s. The CRT TV is the only light source, casting its blue rays across a Commodore 64. Someone types 'ATDT' into the keyboard, followed by a telephone number. A hidden modem springs to life, talking digitised squabble to another computer at the end of the telephone line. Seconds later, the C64 is connected to some remote BBS, and eight-colour ASCII art scrolls across the display, presenting a menu for news, message boards, and games. This is 1980s cutting-edge connectivity, the internet of its day.

Decades later, very few dial-up bulletin board systems (BBSs) remain. But there is a growing trend to resurrect old BBSs from their last backups and make them accessible again – and the only way to really experience them is from one of those old computers. But those old computers are barely capable of connecting to anything, let alone running TCP/IP for internet access. Simulant's Retro WiFi SI solves this problem in a 3D-printed case embedding a stack of wires and an ESP8266 chip. It connects to an old serial port, either 9-pin or 25-pin (with an adapter), takes 5V of micro USB power, and runs an open-source firmware that operates exactly like an old-fashioned modem. This lets you use original terminal software to send the same 'Hayes AT' commands to make connections. But there's one big difference: instead of negotiating a connection



across a telephone line, this little box negotiates connections via your WiFi network.

There's nothing quite like the feeling of firing up your old terminal emulator, perhaps for the first time in two decades, and tentatively setting the serial port to 1200 baud. Yes, that's 1200 bits per second. Then typing 'AT...' if all is well, 'OK' appears, and you've been instantly transported back in time. Type 'AT+CONFIG' to access a menu system, allowing you to connect to the wireless network and save common phone numbers, sorry, domain names and ports. There are plenty of additional AT commands too, including comprehensive control over carriage returns, updating firmware, and hardware flow control. You change the serial speed with ATB, and we had great success on an Amiga 4000 running at 115200 bps. You can then connect directly to any Telnet server, including those hosting an old BBS, or servers on your own machines. It works perfectly, and while we question the 'no returns' warranty and the lack of a bundled power supply and USB cable, the simple joy and utility the Simulant Retro WiFi SI brings to old computers makes it worth the asking price. □

Left Access many Telnet-based servers and BBSs, and even run your own, from the original hardware via WiFi

Left The Retro WiFi's case is 3D-printed, takes micro USB power, and includes a switch to reset all its settings

VERDICT
A brilliant way to give your retro computer a little online company.

9/10

Collins Complete Woodworker's Manual

Albert Jackson and David Day ◇ £30 | harpercollins.co.uk

By Richard Smedley

 @RichardSmedley

Personally, one of this author's favourite things about the maker movement is not all of the soldering, coding, and robots – much as I love them – but the revived interest in traditional crafts.

Fabrics, etched glass, wood. Particularly wood.

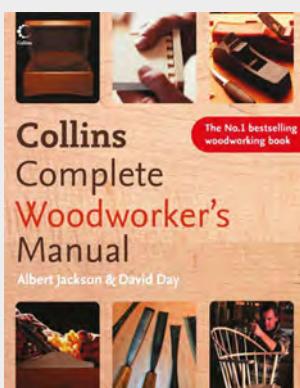
Wood is a material that brings uniqueness – through grain, knots, and finish – to the appearance (and feel) of each object made. It carries history too – whether from a conifer planted during our lifetime for low-cost softwood supplies, an oak planted long before Nelson won at Trafalgar, or a tropical hardwood recovered from a demolished Victorian building.

A Raspberry Pi-controlled CNC machine (see issue 38 of our sister magazine, *The MagPi*) may bring your workshop up to date, but most projects can be completed with simple hand tools and near timeless techniques. Power tools and machine tools will speed up the job, but you also need an understanding of the basics.

Collins Complete Woodworker's Manual gives you the lot – bar the Pi-controlled CNC machine – and does it extremely well. Before the World Wide Web (the first edition dates from 1989), weighty tomes like these were a great way of collecting knowledge on a topic. Here, design, tools, and techniques are taught, with little presumption of prior knowledge. The book is not project-led, yet the principles and practice of designing and building projects like bookshelves, cupboards, chairs, and tables are covered.

For some, this will be enough to get started; for others a project-based book will be a good companion to this manual, to build the practical experience and confidence to tackle the design and construction of the cabinets you dream of for your projects. In either case, Jackson and Day, aided by Simon Jennings' wonderfully clear layouts, bring you knowledge far more clearly and concisely than most YouTube videos.

Easily picked up for a pound or two at charity shops and car boot sales, the first and second editions (the latter owned by your reviewer) are astonishing bargains. But the later edition from 2005, bought new, is cheaper than most programming books – yet the skills unlocked within will not go out of date. An essential for beginners, and you'll refer to this for decades to come. □



VERDICT

Authoritative, clear, concise guide to all aspects of working with wood.

9/10

next mag

issue
#13

ON SALE
22 NOVEMBER

FEATURING BECOME A **PROFESSIONAL MAKER**

ALSO

- ROBOTS
 - ARDUINO
 - 3D PRINTING
 - PLAY DOUGH
 - THE BEST PROJECTS FROM THE MAKERSPHERE
-

DON'T MISS OUT

hsmag.cc/subscribe

HackSpace

TECHNOLOGY IN YOUR HANDS



MEASURE TWICE, CUT ONCE

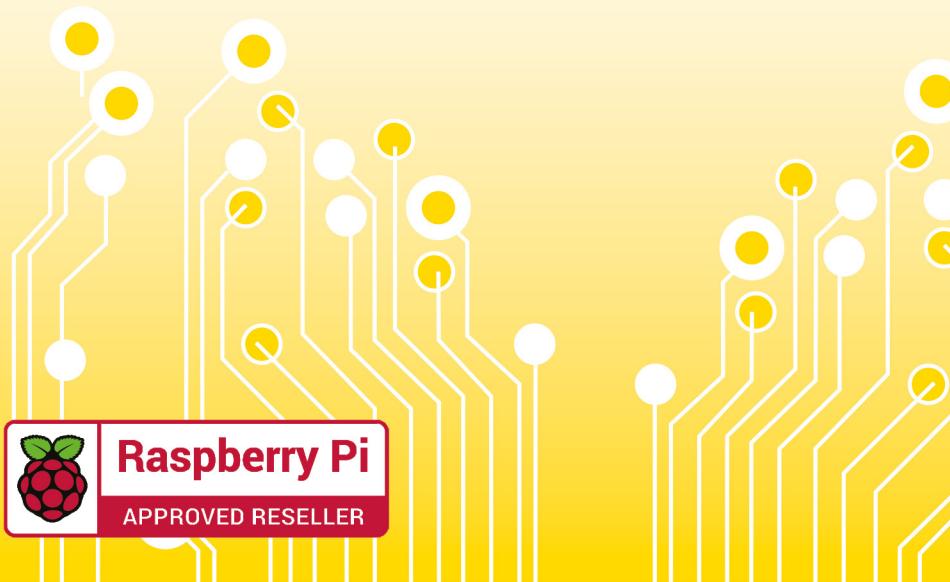
— hsmag.cc —

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*



Raspberry Pi

APPROVED RESELLER

Available for worldwide shipping at:

WWW.CANAKIT.COM

\$74^{.99}
US DOLLARS

£59^{.99}
EXCLUDING VAT

€64^{.99}
EXCLUDING VAT

Raspberry Pi is a registered trademark of the Raspberry Pi Foundation.
CanaKit is a registered trademark of Cana Kit Corporation.