The MagPi

Issue 95 | July 2020 | magpi.cc

The official Raspberry Pi magazine

BUILD A RASPBERRY PI 4

GAMES CONSOLE

Play classic arcade games with RetroPie

Upcycle a vintage radio

Make a DOS Emulation System

Complete Guide!

\+ TOP RASPBERRY PI CASES | LEARN CODING & COMPUTING | HACK E-INK DISPLAYS

44 PAGES OF PROJECTS & TUTORIALS

0800 587 0991
DIGIKEY.CO.UK

Digi-Key ELECTRONICS

9 MILLION+ PRODUCTS ONLINE | 1,000+ INDUSTRY-LEADING SUPPLIERS | 100% FRANCHISED DISTRIBUTOR

*A shipping charge of £12.00 will be billed on all orders of less than £33.00. A shipping charge of $18.00 USD will be billed on all orders of less than $50.00 USD. All orders are shipped via UPS, Federal Express, or DHL for delivery within 1-3 days (dependent on final destination). No handling fees. All prices are in British pound sterling or United States dollar. 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.

ECIA MEMBER
Supporting The Authorized Channel

WELCOME
to *The MagPi* 95

EDITOR
Lucy Hattersley

Lucy is editor of *The MagPi* and is currently building Tetris from scratch. She'll let you know when the blocks start making sense.

@LucyHattersley

It won't surprise you to hear that many folks on *The MagPi* have been avid gamers for decades. And like many gamers, we love classic consoles and retro games. Partly it's the thrill of rediscovering lost gems, and we also love the physical process of building consoles.

The recent release of Retro Pie 4.6 had us all scrambling to build a new games console with Raspberry Pi 4 (page 30).

You can play games made by other people, but it's more fun to make your own. This month we show you how to create a retro turn-based combat game (page 58), and how to build a DOS emulation system (page 44).

If all this gaming is a bit much, but you still love the classics, then PJ shows us how to upcycle a vintage radio (page 48).

I like to think that *The MagPi* magazine blends the best of the classics with the finest new technology. Every issue has a range of perfect projects from makers of all ages.

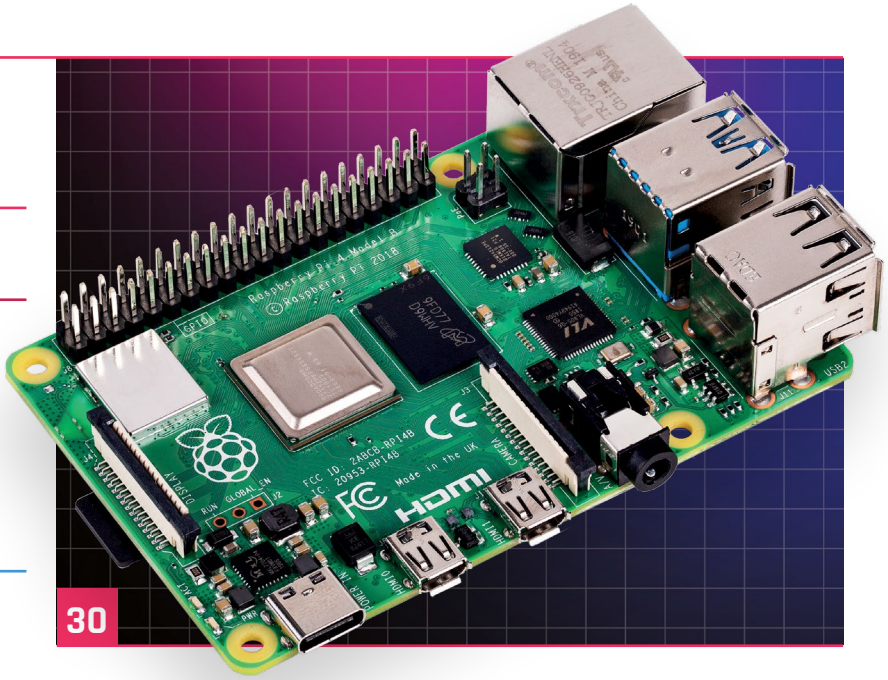
Lucy Hattersley Editor

GET A RASPBERRY ZERO W KIT WITH A SUBSCRIPTION! PAGE 28

Contents

Issue 95 July 2020

Cover Feature



Regulars

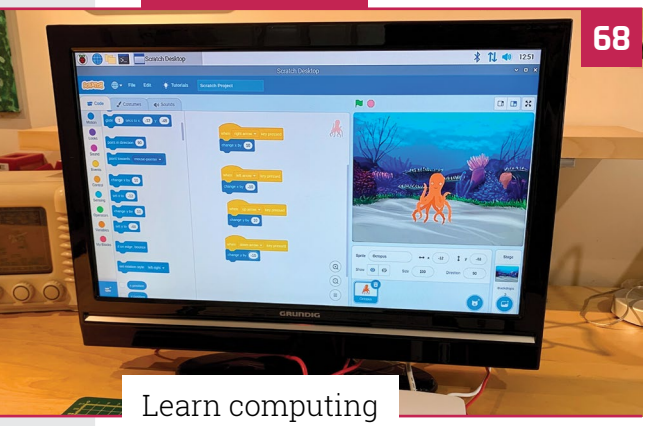
Project Showcases

Whidbey Island Distillery

Pi Commander

Tutorials

The Big Feature



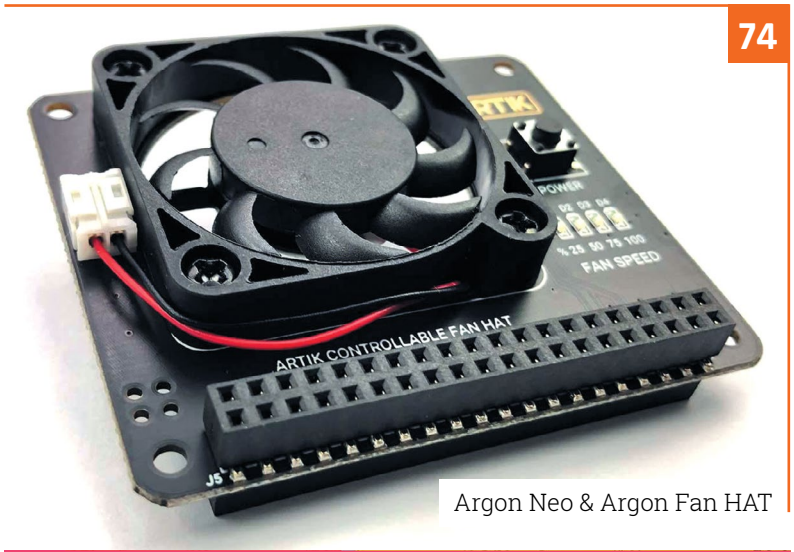
Learn computing

Reviews

Community

Upcycle a vintage radio

Inky wHAT



Argon Neo & Argon Fan HAT

Helene Virolan interview

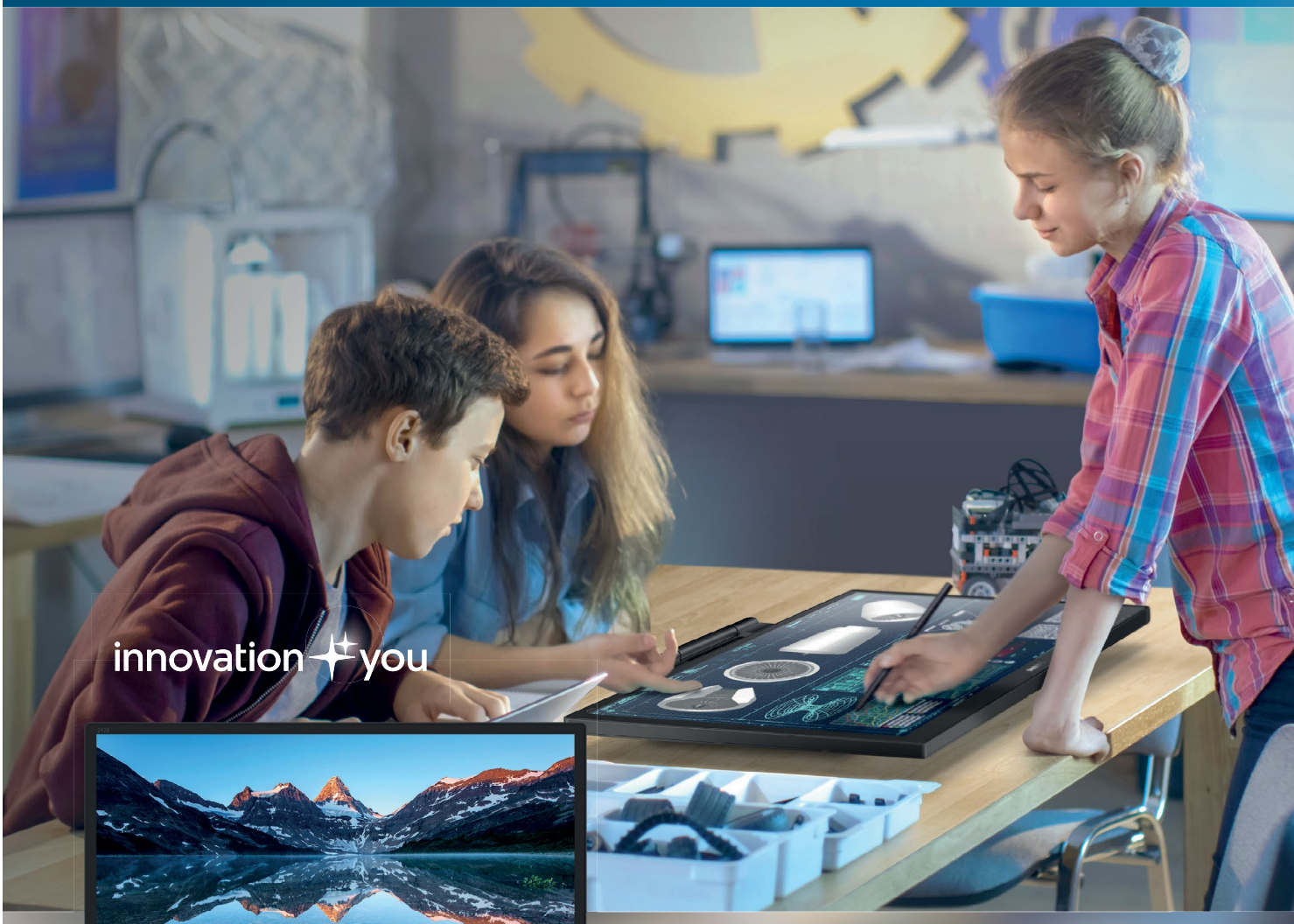
WIN 1 OF 5 8GB RASPBERRY PI 4 **95**

PHILIPS

Monitors

Brilliant interactive display

with Smooth Touch



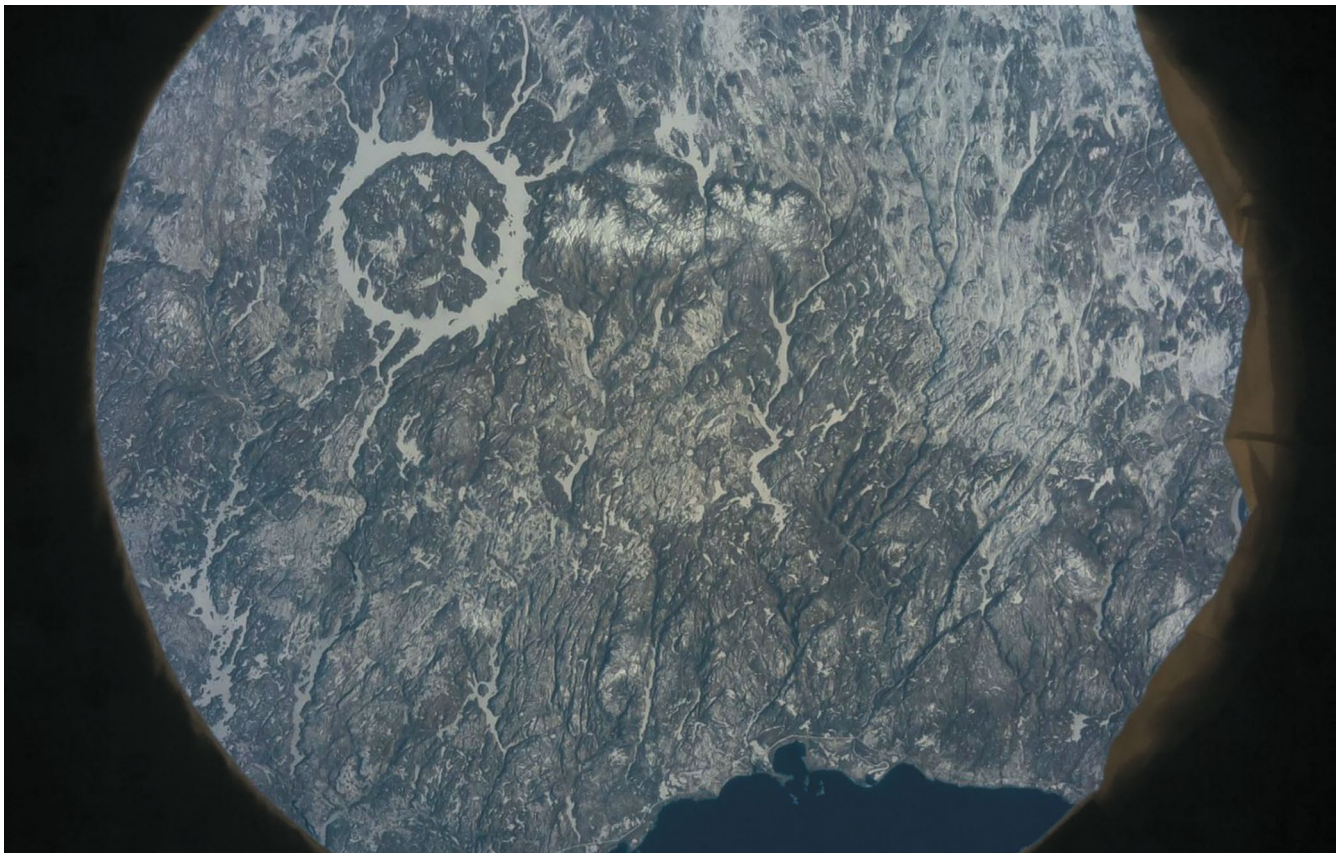
B line touch monitor

Water and Dust Resistant

SmartContrast

LowBlue Mode

Available at: Softcat Insight stone XMA



Astro Pi runs 6564 experiments aboard ISS

Astro Pi mission attracts record number of participants, reports **Rosie Hattersley**

▲ Mission Space Lab investigates life on Earth from the perspective of space

The International Space Station has run more Astro Pi experiments this year than any other. To date, it has run 6564 programs from Mission Zero and Mission Space Lab combined.

The experiments were submitted by students, Code Clubs, and CodeDojos in 25 different countries. A total of 6350 Mission Zero teams with 14,851 participants took part. Each displayed a message to the astronauts on board.

A further 539 teams (with 2147 participants) took part in Mission Space Lab, a more complex challenge which allows young people to design and program an experiment to be run on an Astro Pi computer. Of these, 214 teams have made it through to Phase 4 of the challenge and had their experiments run on the ISS.

The experiments aboard the ISS were overseen by astronaut Chris Cassidy.

Rose-tinted view

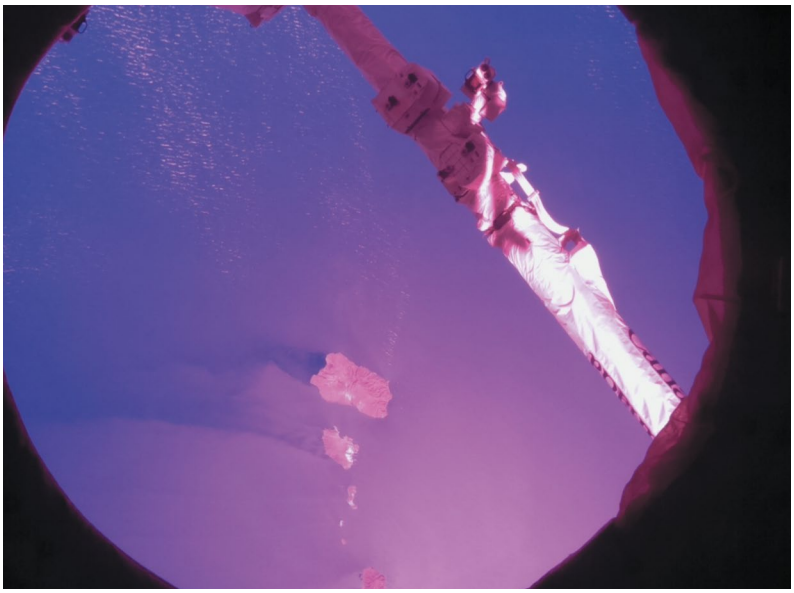
Some of the 208 additional experiments performed by Astro Pi's Izzy took on an odd hue due to some incorrect coding on Izzy and an

> There was an unexpected photobomb by the Canadarm2 robotic arm

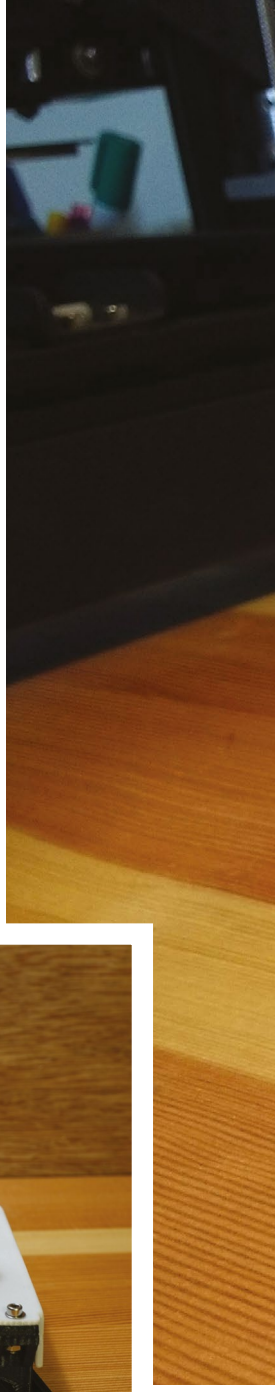
unexpected photobomb by the Canadarm2 robotic arm. ESA had to ask NASA to remove the arm and rerun the 'Life on Earth' experiments so the images containing the arm wouldn't negatively affect experiments. But the unexpected pictures were stunning.

The 2019/2020 Mission Space Lab winners will be announced once competitors' reports have been submitted.

Read more about Astro Pi on the Raspberry Pi blog: **magpi.cc/astro2020**.



▲ The ISS's Canadarm2 robotic arm was unexpectedly caught on camera



BB the robot

Robots come in many forms, like BB, a robot that reacts to people's faces. **Rob Zwetsloot** puts on his best smile and takes a look

MAKER

Sean Glendinning

A 17-year-old student from Aberdeenshire, studying physics, maths, and mechanics. He enjoys building and programming robots and other electronic projects.

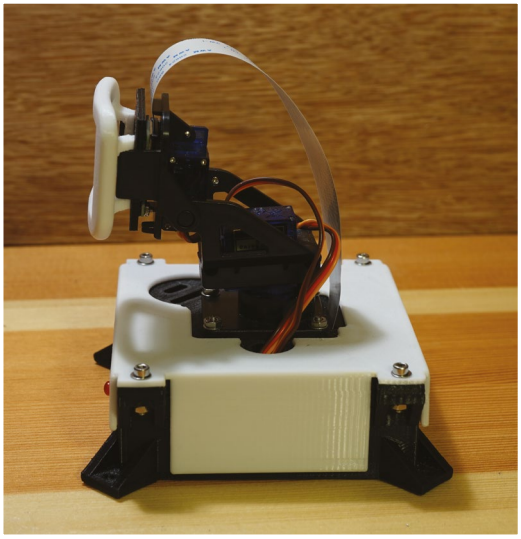
Computer vision in robots, especially Raspberry Pi ones, is a great way to improve their automation and functionality. It makes your robot more than a glorified, customisable, DIY RC car. Instead of adding computer vision to a robot kit he already had, Sean Glendinning created a robot to experiment and test with it. This ended up in the creation of BB.

"BB is a small, wall-powered robot designed to test out offline computer vision, detecting people and faces," says Sean. "It searches a room for people, then tries to recognise any faces if it detects a person. If it comes across a person in its memory, it will respond appropriately, e.g. 'Hello Sean', in a squeaky voice. It tracks any people it detects, following them around the room with its pan-tilt mounted camera."

> "It searches a room for people, then tries to recognise any faces if it detects a person"

Seeing robot

Like a lot of Raspberry Pi robots, BB uses OpenCV. It's a Python library that we've covered in the magazine before that allows people to add machine learning to image processing. Computer vision works fine on a Raspberry Pi – OpenCV has been used for the Formula Pi racing league, and face unlock Raspberry Pi tutorials have been around for as long as Raspberry Pi has.



▲ From this angle you can get a better look at the servos that control the camera

Sean tells us he has a lot of experience using a Raspberry Pi: "I've found it a great platform for building robots with. It is more powerful than Arduino and other microcontrollers, yet smaller and cheaper than full desktops."

A Raspberry Pi 3A+ is the main component, but extra circuitry is included. "A DIY power management circuit is used to turn Raspberry Pi

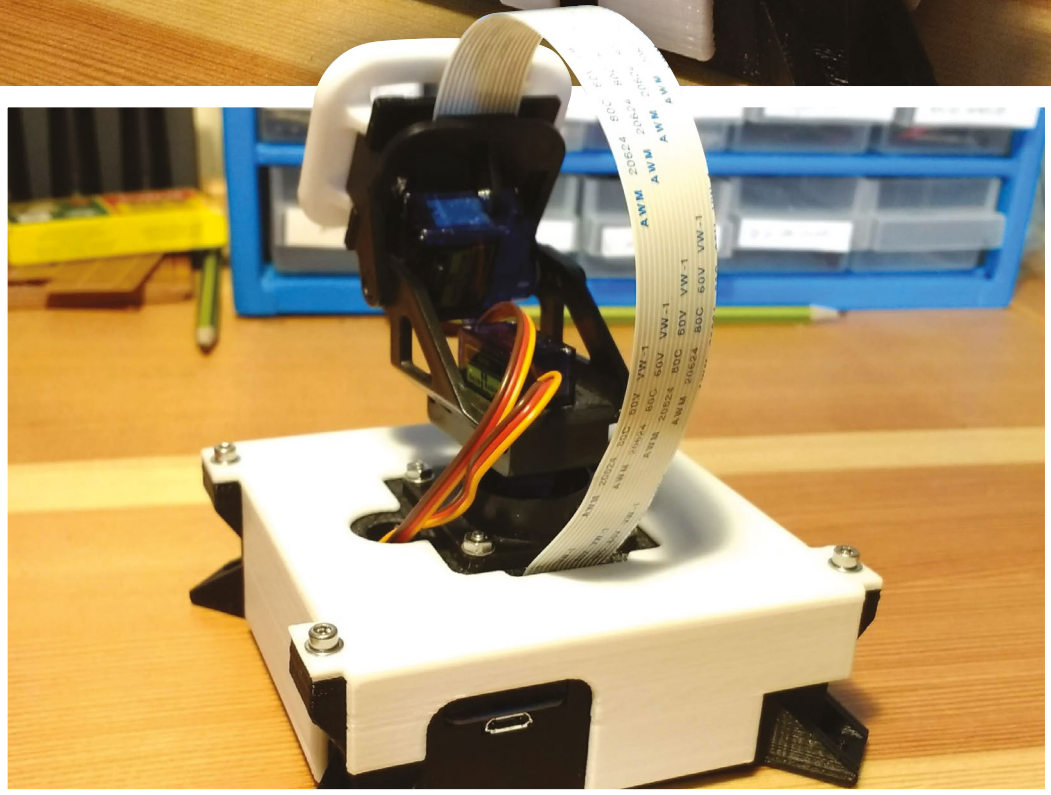
Using a standard Raspberry Pi Camera Module is enough to recognise human faces

This 3D-printed base includes a speaker, activation LED, and an on/off switch, although it can't move otherwise

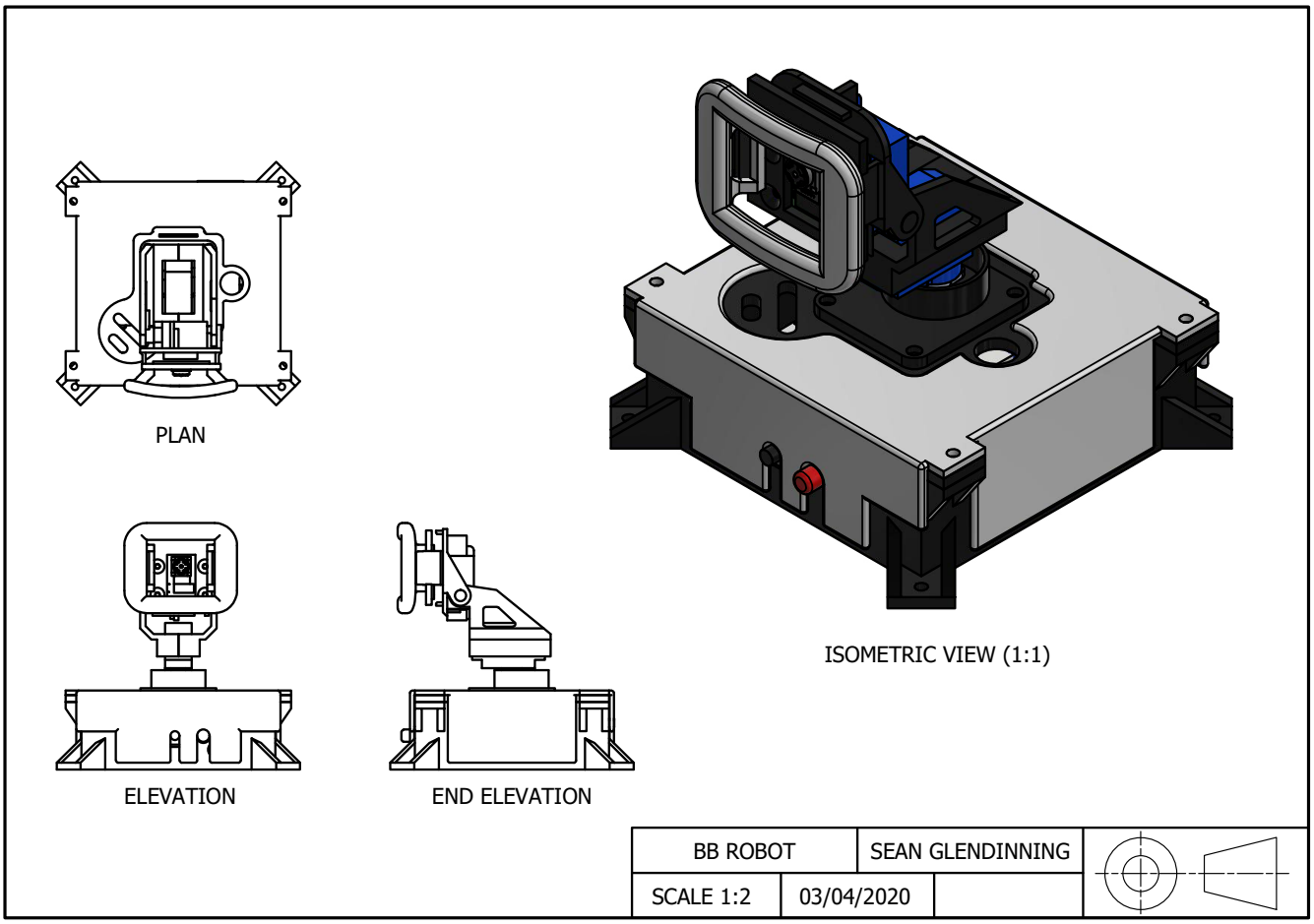
BB will look around to find a face, thanks to pan and tilt servos attached to the base

Quick FACTS

- BB has an 180-degree viewing angle
- Like PiArm and other arm robots, BB cannot move around
- There's an inner case that holds all the electronics
- Sean designed a relay circuit based on this one: **magpi.cc/roborelay**
- The high-pitched voice is supposed to replicate the turrets from Portal



BB has an 180-degree viewing angle, so it can't look behind itself



▲ A 3D model for BB, one step towards bigger and better robots

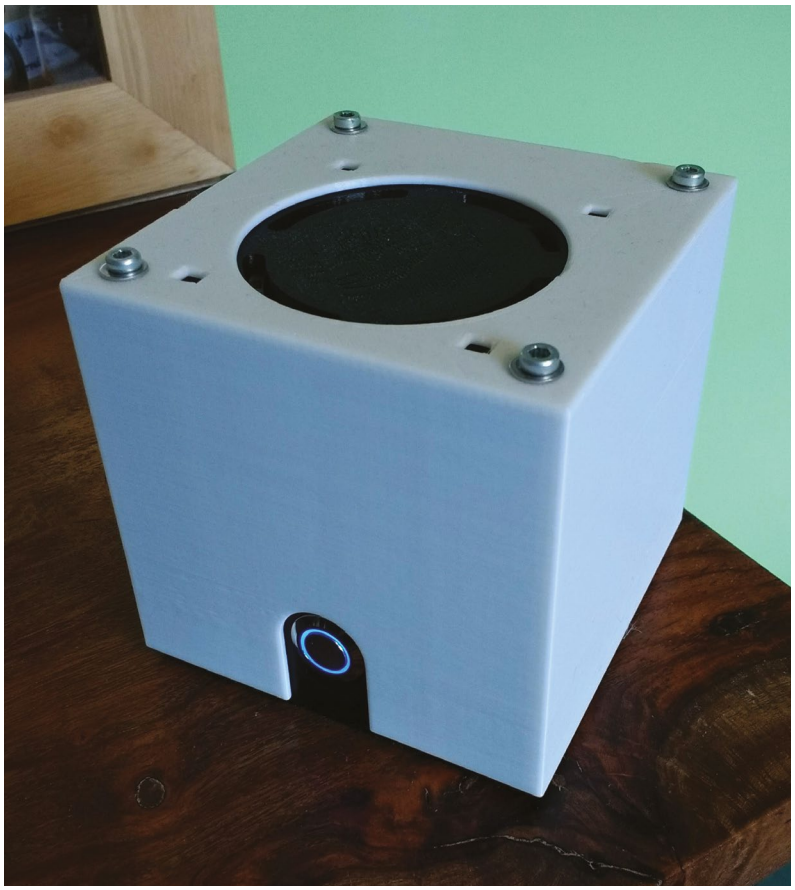
▶ It uses a standard Raspberry Pi Camera Module to detect faces

on and off with a button, with a power indicator LED," he explains. "An amplifier is included to power a small speaker, as well as connectors for the two servos. The camera is a standard Raspberry Pi [Camera Module], and the whole robot is held together with a black 3D-printed case, with a white shell to cover unsightly bolts."

> "An amplifier is included to power a small speaker, as well as connectors for the two servos"

First steps

Creating BB was just the start of what Sean plans to do, with bigger and better robots already in development.

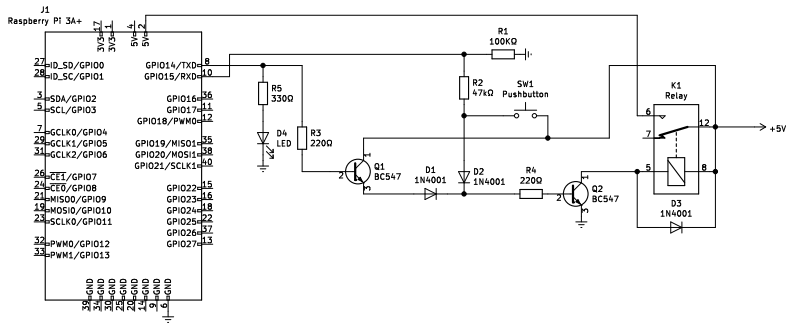


▲ Sean's next robot is capable of more advanced speech synthesis, as well as voice recognition

"I wanted to make a small 3D-printed robot, since I just acquired a 3D printer and wanted to make something cool," he says. "I also wanted a platform to test offline computer vision on. My long-term goal is to create a Raspberry Pi-powered robot, capable of computer vision, speech recognition, and speech synthesis. I've already finished building my new platform to recognise speech... It is cube-shaped, with a similar colour scheme to BB. It features four microphones, as well as a speaker and LED ring. I'm hoping to combine the two robots to create a sophisticated machine, as if from a movie."

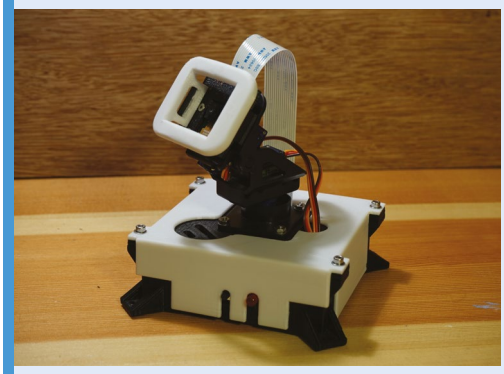
We look forward to seeing this sophisticated machine in the future.

▼ A schematic for BB's circuit design

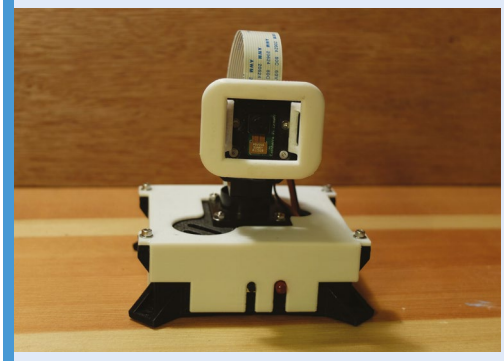


Find a face

01 BB works by using OpenCV Python software (**opencv.org**) to detect people, and looks at the first person in the image from the camera, using two 5V micro servos for pan and tilt.



02 Then, only if a person is detected, it will try to recognise any faces that are in the image (**magpi.cc/facerec**), and respond appropriately if it recognises that person.



03 It has a database of people's faces in it, including Sean and his friends (with their consent). It also includes a real-time clock module, so it can wish a person happy birthday if the current date matches their stored birthday.

Raspberry Pi Smart Scale

Eating a raspberry pie could pile on the pounds, so let a Raspberry Pi help you lose them. **David Crookes** explains

MAKER

Jamie Bailey

Jamie is the CEO and founder of the data streaming and visualisation platform Initial State, as well as a self-confessed nerd.

github.com/initialstate

Content Warning!

This article discusses weight and weight loss issues that may not sit well with everyone.

It's not always easy to find the time or motivation to keep fit, but it's certainly important to look after your health. One way of monitoring fitness is to keep an eye on your weight and BMI, which is where the Raspberry Pi can assist, particularly when teamed up with an abandoned Wii Balance Board.

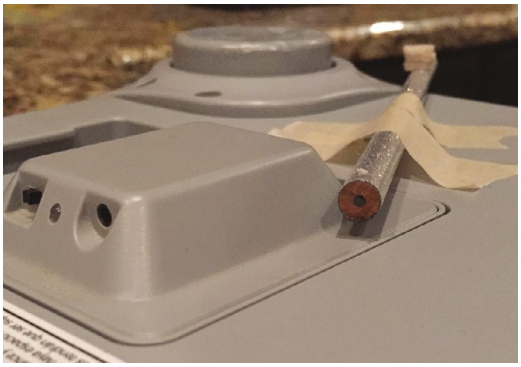
"I wanted a scale to automatically log my weight for tracking," says Smart Scale creator Jamie Bailey. "I also wanted to give it a personality and to text me like a personal trainer might do."

To achieve this, Jamie paired a Raspberry Pi 3 with a Wii Balance Board – a peripheral originally designed for the Nintendo Wii console. This was easily done via Bluetooth, but permanent pairing proved more tricky, so Jamie worked on a system that would connect the two when a Python script was run.

Power-to-weight

He did so by modifying an existing script called Gr8W8Upd8M8.py created by Stavros Korokithakis. "The script made reading the Wii Balance Board pretty straightforward and it did the heavy lifting for the raw communication between the device and Raspberry Pi," Jamie says.

▼ Rather than flip over the Wii Balance Board to locate the sync button, Jamie taped a pencil pivoted by felt pads to the underside as a makeshift lever



▲ Power adapters are available from third-party companies which get around the need to constantly replace or recharge the four AA batteries

Indeed, when the script starts, it searches for a Wii Balance Board with which to pair. "Once you press the setup button on the bottom of the Wii Balance Board, it broadcasts the Bluetooth address and the connection is established," Jamie continues. "As long as neither Raspberry Pi or the Wii Balance Board loses power or the script is killed, that connection will remain."

This, however, requires pressing a small button on the underside of the Balance Board whenever the Raspberry Pi is powered up. For convenience, Jamie taped a pencil to the underside of the Board, laying it over the button and allowing it to poke out from the front. A tap of the pencil hits the button. It's a low-tech solution, but works perfectly.

Weight watching

So what happens when you stand on the Wii Balance Board? First, a weight greater than zero is detected, kicking off the measuring mode which collects 250 measurements and averages the results. "This is necessary because scales are noisy circuits," Jamie explains. "The more measurements you take, the longer the final

result takes to calculate – 250 measurements takes about five seconds, providing a good balance of time and accuracy."

With a valid weight calculated, the measurement is stored in a variable. This is streamed to the Internet of Things service, Initial State, which not only grants access to a mobile-friendly dashboard detailing your weight history at-a-glance but allows real-time triggers to be sent via text message. Crucially, it gave Jamie scope to inject some motivational fun.

"I could include a motivational quote based on whether weight was lost, gained, or maintained," he says, having come up with a host of encouraging statements which he tapped into the script. "In each case, a message is returned and streamed to Initial State. This gets sent via text message to your phone and the idea is that it keeps spurring you on to achieve your goal."

Jamie gets told that every step is progress, no matter how small, and that there is no such thing as failure. But has it done the trick? "Well, my scale has just texted me to 'Keep going!'. I think that means I need go for a run now."

> A mobile-friendly dashboard details your weight history

▲ SMS text messages are sent to a user's mobile phone with a quote and their current weight

Quick FACTS

- Used Wii Balance Boards are relatively inexpensive
- You can add your own motivational quotes
- The number of weight measurements can be adjusted
- Initial State is for UK and US students
- The quotes are contained in a file on GitHub: **magpi.cc/scalepy**

Give it a go!

Jamie showed us how to build this in issue 46 of *The MagPi* – give it a look if you want to make your own! **magpi.cc/46**

Piano-Playing Robot

This robot can play scales, chords, arpeggios, or totally new musical scores. **Phil King** listens in

MAKER

Étienne Allaire

Étienne is a fourth year undergraduate student in Electrical Engineering at Polytechnique Montréal and plans to pursue a master's degree in Artificial Intelligence.

magpi.cc/pianorobot

Looking at the piano in his living room, Étienne Allaire felt a little regretful. "I realised that we played very rarely and found it a pity that such a beautiful instrument remained unused for so long!" he recalls. "That is when I had the idea of a robot that could play all by itself to add to the ambience of the home. It is also a combination of several fields that I love: music, engineering, and artificial intelligence."

It took him around four months, on and off, to develop his Piano-Playing Robot. As shown in his YouTube project video (**magpi.cc/pianorobot**), it features 15 solenoids mounted on a wooden frame placed on top of an electronic piano keyboard.

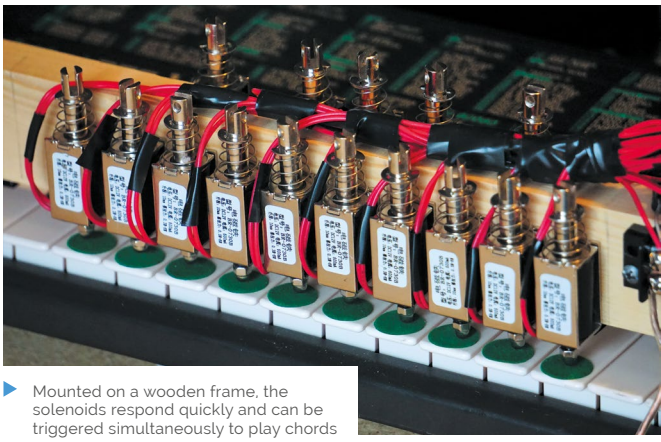
"Basically, the system is quite simple," reveals Étienne. "Each solenoid pushes on a piano key and is controlled by a specific GPIO pin of a Raspberry Pi 3 Model B." Since the latter doesn't have enough electrical power to activate the solenoids directly, an electromechanical relay module acts as a buffer between the two."

> "A camera was added to read and decode new musical scores that had never been shown to the robot before"

Fast 'fingers'

The system can read a MIDI file, convert it into on and off signals for each note, and play it. A basic UI allows the user to choose between scales, arpeggios, or melodies.

Some of the solenoids are positioned over black notes, enabling the robot to play sharps and flats in various scales and melodies



Mounted on a wooden frame, the solenoids respond quickly and can be triggered simultaneously to play chords

"As of now, the robot can play a full scale in just under two seconds," reveals Étienne. "That is, at the fastest rate, it plays about eight keys per second." This impressive speed makes it possible to get the timing of the notes just right.

"However, the system is limited by the push and pull times of the solenoids, which add up to a few tens of milliseconds," he adds. "At a first glance, it may not seem like much, but in a fast melody it can make a difference. Our robot still has a long way to go to overtake human-beings: the fastest pianist in the world claims to be able to reach up to almost 20 keys per second!"

Sight reading

So far, he has programmed the robot to play six scales, six arpeggios, and five different melodies. The most impressive feature, however, is its ability to read previously unseen sheet music – using a Raspberry Pi Camera Module – and play it.

"A camera was added to the system to read and decode new musical scores that had never been shown to the robot before," says Étienne. "It then scans, decodes, and plays them."

To do this, it uses a free open-source tool called Audiveris. "It is a complex program for optical music recognition (OMR), which is recognition of musical symbols. Unlike optical character recognition (OCR), this task requires decoding the

Mounted on a block of wood, a Raspberry Pi Camera Module reads sheet music, which is analysed using OMR

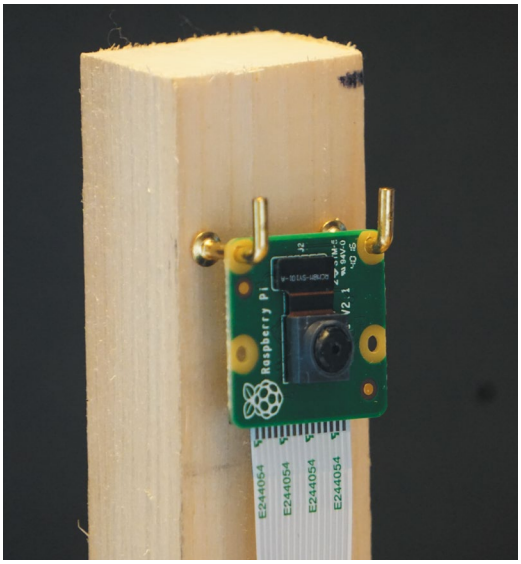
When activated, each spring-loaded solenoid moves downward to play its note

Raspberry Pi triggers relay switches to control the 15 solenoids, enabling enough electrical current to be sent to them

configuration of symbols in two dimensions. In addition, the order of magnitude of the symbols is extremely wide (sometimes ranging from tiny dots above musical notes to symbols that are drawn across a large part of the page).”

In brief, the recognition process is done with classifiers and neural networks that make matches with symbols previously learned. “For my project, the decoding part was done on a laptop since Audiveris was more easily compatible with Windows,” he explains. “On average, it takes around 30 seconds for the robot to decode a sheet of music containing a few staves.”

Étienne is hoping to add extra features to the robot. “In particular, I would like to extend the concept to 88 solenoids. This way, the robot would theoretically be able to play all the existing melodies on the piano! Obviously, there is a lot of development to do in terms of Optical Music Recognition, but I like to imagine a piano concert performed entirely by a robot!”



▲ A Raspberry Pi Camera Module is used to read sheet music, its images being analysed using optical musical recognition

Quick FACTS

- This is Étienne’s first serious project with a Raspberry Pi
- The robot took around four months to develop
- He burned a few transistors at the start of his experiments
- When activated simultaneously, the solenoids require a lot of electrical current…
- …At first, he had trouble supplying them with enough power

Singing Toilet

A trip to the washroom need no longer be boring – meet the Singing Toilet! **Nicola King** seeks out the heavenly harmonies

MAKER

Max Björverud

A sound and code artist based in Stockholm, Sweden, Max's work involves musical instruments, museum exhibits, public art, interactive installations, and more.

bjorverud.com

It may be the ultimate bathroom experience: wall-mounted toilet rolls that 'sing' various notes when the paper is pulled, thus creating an extremely pleasant and calming trip to the smallest room. But how did this very novel idea come to fruition?

Maker and artist Max Björverud explains that this custom installation was born from a request from a client: "Fredrik Öst at Snask (a creative agency) here in Stockholm wanted something for the toilet. I don't remember how our conversation really panned out, but a week later I presented the first prototype. The build itself was quite quick, but it needed some iterations to find the best holder and roller. The most difficult part was to make it usable. They need to be able to switch toilet paper without a technician."

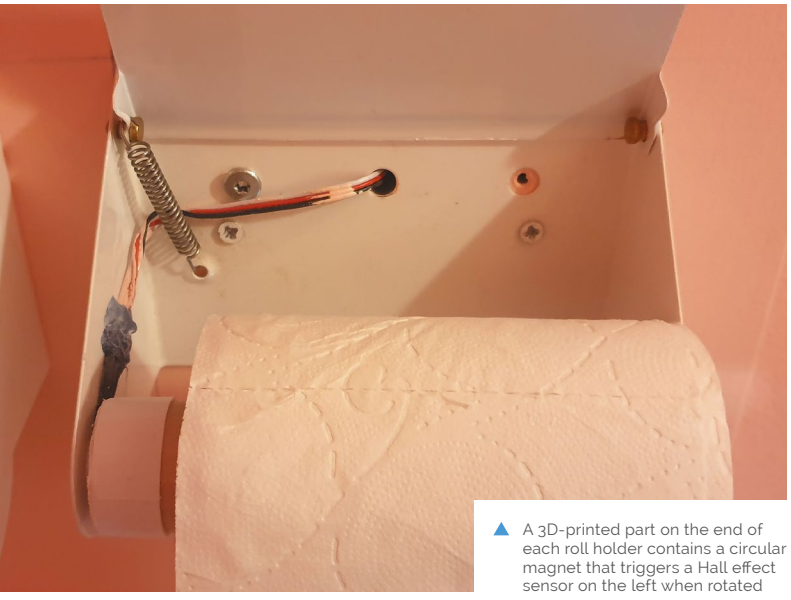
You gotta roll with it

The system is built around a Raspberry Pi, a HiFiBerry AMP to output high-quality sound to a hidden speaker, and an Arduino Mega to take readings from eight Hall effect sensors. In each toilet roll holder, a 3D-printed part contains a circular four-pole magnet that triggers the Hall effect sensor when rotated. Whenever it receives data from the sensors via the Arduino, Raspberry Pi outputs the respective sound sample for each moving toilet roll.

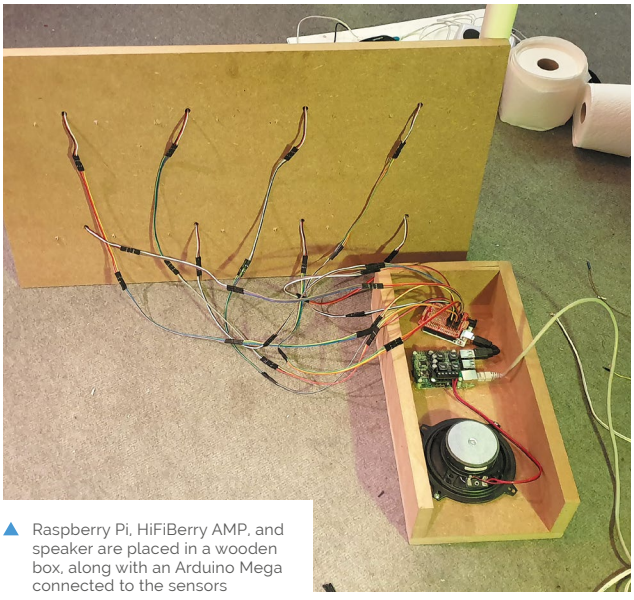
"As long as a roll is rolling, the Arduino will send a continuous message telling the Pure Data patch to play that specific sample," says Max. "It's built in such a way so that the voice will sing as long as the roll is rolling."

A few technical issues were encountered in the build process, including random instances of failure to work: "My audio patch (program) consumed 100% CPU and clogged the whole Raspberry Pi," recalls Max. "That version used windowing functions to create seamless pitch shifting of the samples I used for audio playback. Had to abandon that…"

Max also had to fiddle with the Hall effect sensors for quite a while before he was happy with their placement, and a colleague assisted with the 3D prints, but all their efforts proved worthwhile.



▲ A 3D-printed part on the end of each roll holder contains a circular magnet that triggers a Hall effect sensor on the left when rotated



▲ Raspberry Pi, HiFiBerry AMP, and speaker are placed in a wooden box, along with an Arduino Mega connected to the sensors

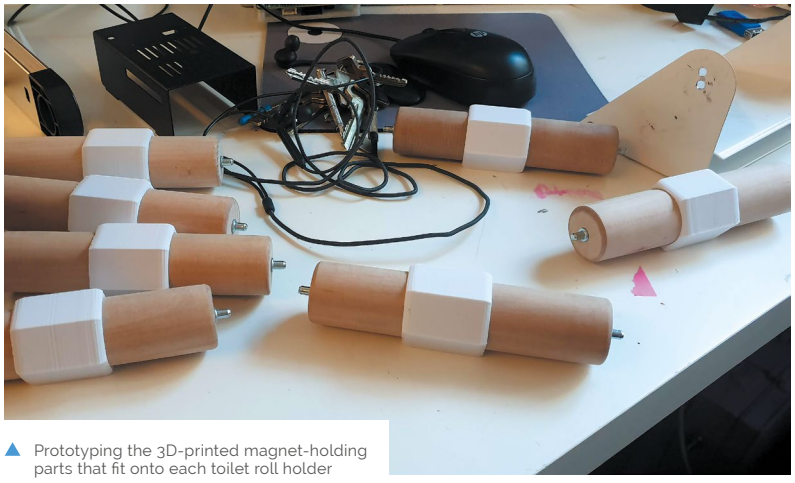
When paper is pulled from one of the toilet roll holders, its Hall effect sensor is triggered and Raspberry Pi plays the choir sample note for that holder

Raspberry Pi and the sound speaker are located inside a wooden enclosure, "painted pink just like everything else so you won't see it"

Quick FACTS

- The project took around six months to develop
- Watch the project in action: **magpi.cc/singingtoilet**
- Max has created other installations for Snask...
- ...A window that compliments you, and a modified doorbell
- Read more on Max's work here: **bjorverud.com**

> "It's built in such a way so that the voice will sing as long as the roll is rolling"



▲ Prototyping the 3D-printed magnet-holding parts that fit onto each toilet roll holder

When it comes to changing the rolls, this is just done normally and the only thing that people need to remember is to put the roll back in the correct direction: "The magnets inside the 3D print must face the Hall effect sensor on the left."

Flushed with success

While Max has had great feedback on his creation, "there hasn't been much interaction since people are out of office nowadays. I'm currently learning how to play it so I can hold toilet concerts."

Integral to the Singing Toilet, and to all his work, has been Raspberry Pi. "[It's] my go-to tool for interactive audio, mainly because I'm so used to it, but also because there's such a great community where any help needed is out there to be found," Max tells us. "My past projects with Raspberry Pi include a huge dance carpet, a synth made of steel, museum exhibits, and various interactive audio projects."

Whidbey Island Distillery

A family-run distillery focused on ingredients first and automation second, resulting in an award-winning success story. **Rosie Hattersley** explains

MAKER

Jim and Steve Heising

Jim and his father Steve have backgrounds in software engineering, physics, and the aerospace industry. Jim learnt about computing hardware in order to manage Whidbey Island's distillery setup.

magpi.cc/whidbey

Some great business ideas are hastily written down on a napkin. This one emerged from a 'jokey' family lunch. Appropriately enough, it was Easter Sunday when the Heising family light-heartedly hashed out how they might all contribute to an exciting new chapter. A few months later, Bev and Steve Heising bought several acres of land and the family began refurbishing the existing 'bunker' buildings to live in and as their business premises. The Heisings threw open the doors to Whidbey Island Distillery a year later. A decade on and it is now an award-winning distillery open every day for visits and tours.

"Anyone can come by and see the stills run and learn about them in more detail," says son Jim, cheerfully. Given its location on a rather pretty island just north of Seattle, Washington, and the quality of the liqueurs and whiskeys produced there, it's little wonder the Heisings have plenty of visitors. Their blackberry liqueur, distilled from one of Bev's recipes, is the highest-rated of all spirits in North America by **Tastings.com**. The other special ingredient in the business's success is Raspberry Pi.

Refreshing approach

Although he's responsible for the monitoring equipment and tech that keeps it running, Jim says it's really a 'mom and pop' business. His mum comes up with all the recipes while "dad figures out how to adapt our stills and manufacturing process to produce them in large quantities." Steve is a former aerospace engineer, while Jim has more than 20 years' experience as a self-taught software engineer and startup founder. He built most of the hardware and software for the still, which dad Steve designed. "Hardware is relatively new for me, but I learned enough to get by with tinkering and asking a lot of questions," reveals Jim.

Distillation doesn't require much technology to run, Jim explains, but it's very labour-intensive and using technology can make a big difference to what a distillery can do. "Many smaller distilleries struggle because they spend so much time distilling that they have little time for anything else," he says. Often, they will modify alcohol bought in from a larger concern and put their own label on it. The Heisings wanted to make their own. Raspberry Pi enables their stills to run completely automated 95% of the time. Jim explains that its

> "Raspberry Pi enables their stills to run completely automated 95% of the time"

CPU is powerful enough and has plenty of RAM (he'd love direct 12 V DC power via the header pin too, though). Other setups he's used couldn't. "We have 15+ temperature sensors, a barometer, multiple D2As (digital-to-analogue converters) connected to high-voltage variable SSRs (solid-state relays), multiple low-voltage D2As for pumps, and multiple relays for solenoids."

Whidbey Island liqueurs and whiskeys have received multiple accolades from Tastings.com

Warning! High Voltage

This project features a high voltage controlled by a solid-state relay (SSR). Be careful when working with high voltages.

magpi.cc/electricalsafety

▶ The laptop screen is a convenient way of viewing readings from the still's sensors

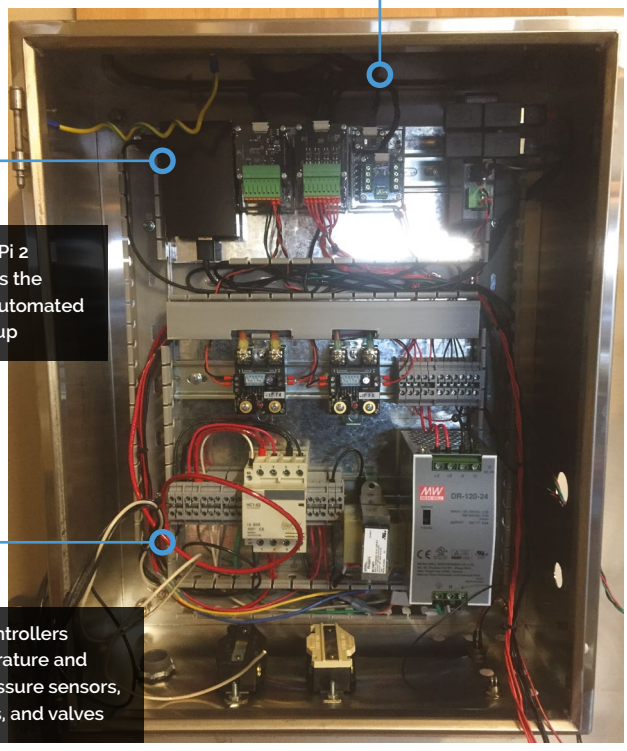
▲ Bev Heising develops the recipes while former aerospace engineer Steve works out how to produce them in volume

"Having I/O systems we could communicate with through USB was a bit of a game-changer," he says, "because we could run and debug the whole control system from a laptop, then simply unplug the USB cable and plug it into Raspberry Pi to let it run completely hands-off." Jim credits "amazing" I/O products from Tinkerforge (**tinkerforge.com**). "We connect Raspberry Pi via USB to Tinkerforge's products, import a simple SDK, then we have an enormous set of options for I/O at our fingertips." You can see the source code and wiring diagrams at **github.com/bunker-stills**.

Ever evolving

The Whidbey Island setup is still evolving, says Jim. "Every day we're making changes and we continue to build new stills for our distillery. At some point in the future, we may even offer kits for other distillers to make their own, but only if we can find some time (which is always hard)."

Raspberry Pi has enough RAM and a powerful enough processing power to handle the controllers' demands

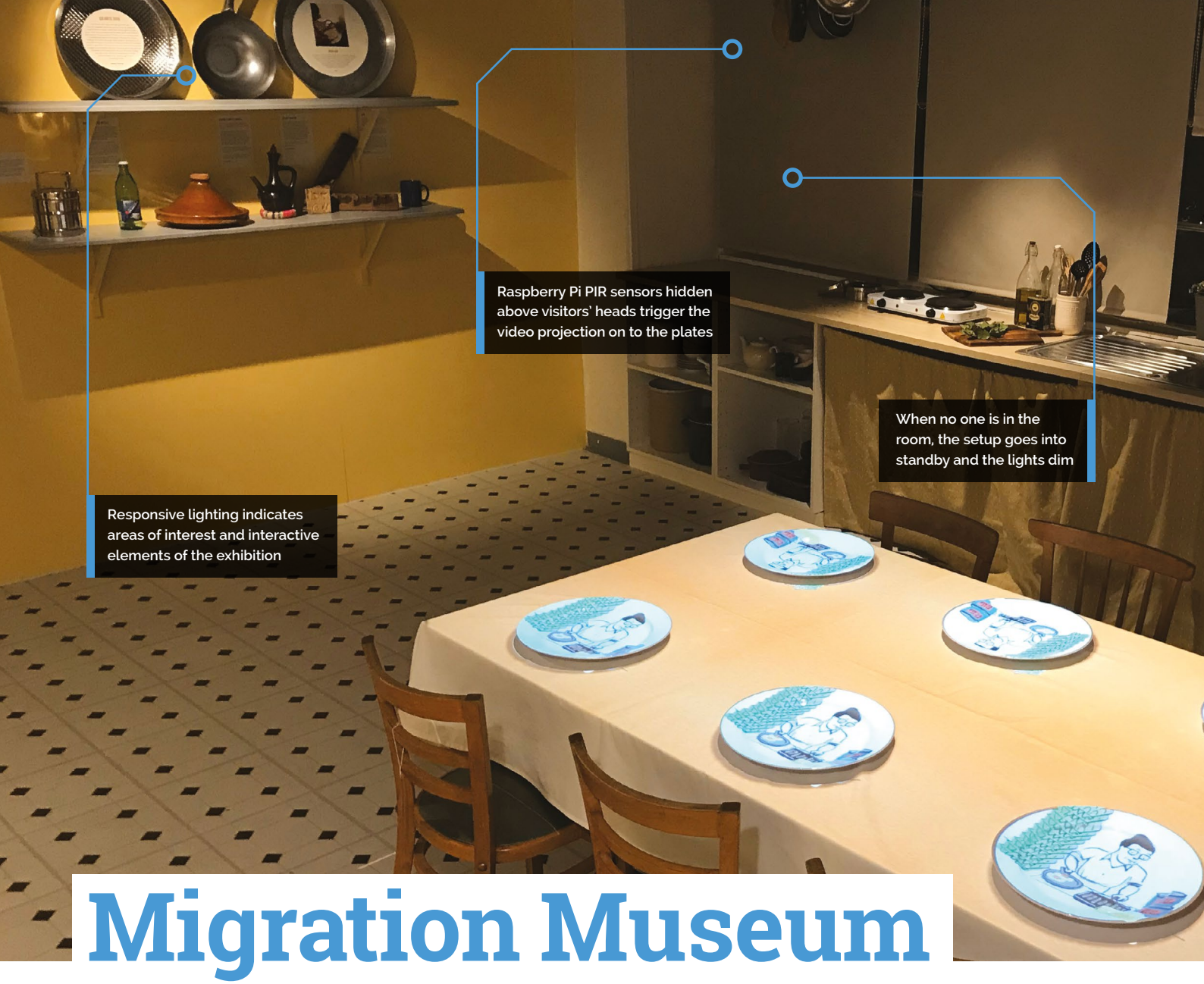


The Raspberry Pi 2 inside this box is the 'brains' of the automated still control setup

Tinkerforge controllers monitor temperature and barometric pressure sensors, heaters, pumps, and valves

Quick FACTS

- Bev Heising devises the liqueur recipes
- Project 'mastermind' Steve works out how to make them in large quantities
- Jim designed the original circuit board for the automated still
- It is now on display in his garage
- Whidbey Island's blackberry liqueur is rated No.1 in the US



Migration Museum

A museum celebrating the diversity of London's residents comes to life when visitors linger near its artefacts. **Rosie Hattersley** discovers more

MAKER

Chris Owens

Chris Owens is one of the founders of Clay Interactive, a company that specialises in creating interactive exhibits for museums.

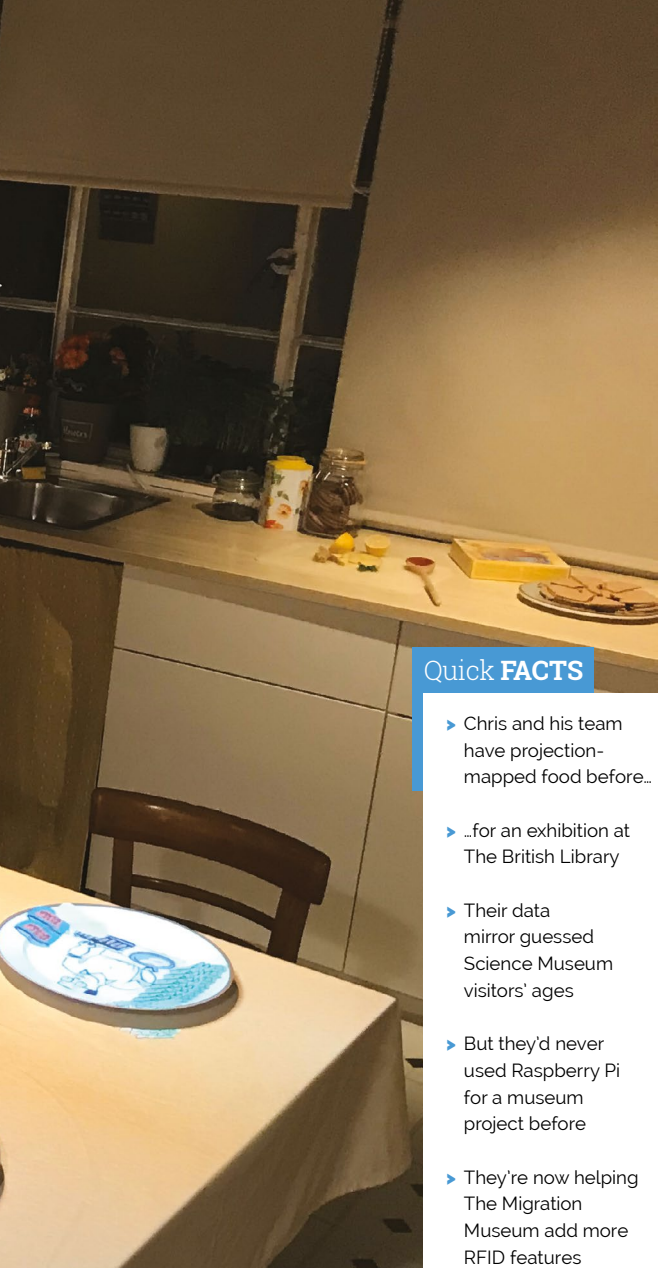
clayinteractive.co.uk

A few weeks before this spring's lockdown there was a buzz of excitement among visitors to Lewisham Shopping Centre in south-east London. Slabs from the Berlin Wall emblazoned with brand-new street art greeted visitors at the entrance to the former H&M store, while the storefront display is one of assorted luggage with tags from around the world and many generations.

Thoughtfully curated bookshelves in the gift shop contain tales and memoirs from Windrush Generation writers and set the tone for the rest of the exhibition. The reconfigured former retail unit now contains TV set-like areas with a classroom, kitchen, bedroom, and a hairdresser's shop. Peeking in the bedside drawer or rummaging among the clutter on the mantelpiece, the curious visitor finds postcards, paintings, maps, and photos revealing the lives of those who made London their home. Taking a seat in the hairdresser's, you're invited to don headphones and listen to the stories recounted by the figure who appears in the mirrored screen. Gazing at the dressed dining table prompts a light to beam images onto the bare dinner plates, with recipes and muttered memoirs of favourite dishes.

Visual storytelling

The Room To Breathe interactive displays are the handiwork of Chris Owens and his colleagues from Clay Interactive. Commissioned by Aditi Anand, The Migration Museum's head of creative content, they created interactive sets created from artefacts that help tell immigrants' tales.



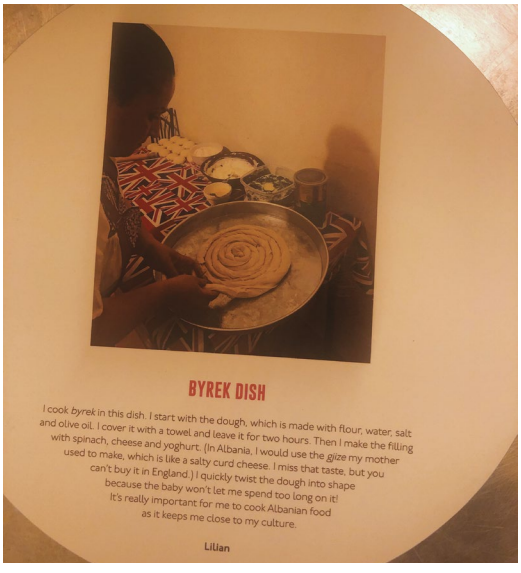
Quick **FACTS**

- Chris and his team have projection-mapped food before…
- …for an exhibition at The British Library
- Their data mirror guessed Science Museum visitors' ages
- But they'd never used Raspberry Pi for a museum project before
- They're now helping The Migration Museum add more RFID features

> "The technology had to work unattended all day, every day"

"Our role was to try and make the various rooms in the exhibition as interactive and engaging as possible within their very tight budget," explains Chris. "We knew we needed to have perfect full HD playback and the technology had to work unattended all day, every day, so reliability was also key."

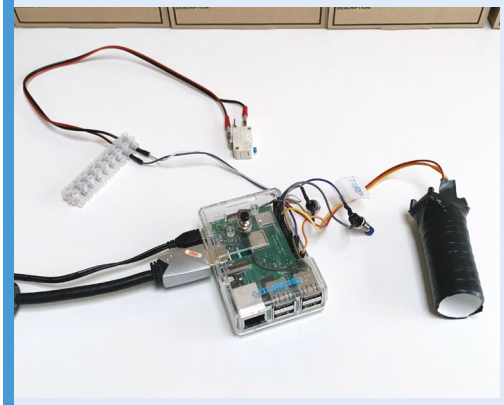
Chris says he needed a 'makers' mindset' as off-the-shelf hardware setups typically used by museums were simply too expensive. "Having used Raspberry Pi for a home automation project, I knew a bit about the platform and saw the benefits of the built-in GPIO combined with a compact low-power platform that could support pretty much any programming language."



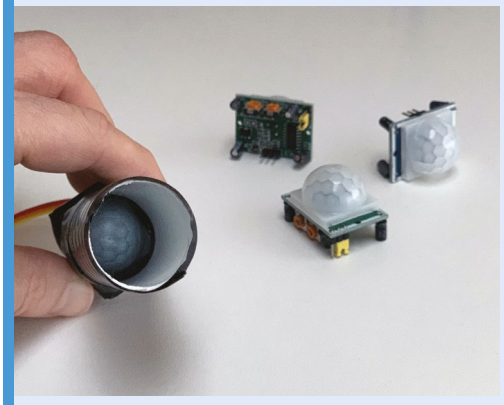
▲ Framed pictures in the living room become videos of people sharing their memories

◀ A dish for making byrek, a traditional Albanian filled pastry

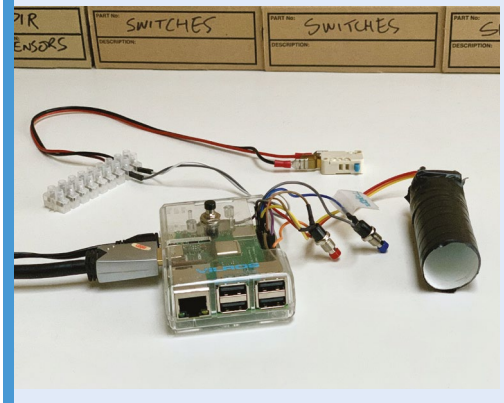
Interactive display details



01 A PIR sensor is used to trigger an object to respond to someone's presence.



02 The PIR is enclosed in a tube to limit its sensor range, meaning it only triggers when someone is close by.



03 Discreetly placed Raspberry Pi-controlled switches trigger the audio when the headphones are picked up.

▲ Interactive rooms, such as the Golden Scissors barber's, bring personal stories from new arrivals to Britain to life in unexpected ways

They created the digital signage themselves using hardware from The Pi Hut, as well as Amazon. "It was really easy for us to get a prototype of the hardware up and running as the GPIO pins on Raspberry Pi make it easy to connect up and test switches and sensors," says Chris.

The broader picture

"The PIR (passive infrared sensor) has a very wide viewing angle as it's designed to look at a whole room," says Chris. "We wanted only small areas to trigger the media, such as sitting on a chair or standing very close to something." To reduce the PIR's field of view, they simply put it inside a tube so it could only 'see' what they wanted it to see. Occasionally, the PIR would be triggered by

▼ Like the other items, this case bears a label telling the story of how its owner came to Britain

The Migration Museum, housed in Lewisham Shopping Centre, features pieces of the Berlin Wall

Postcards, letters, and spoken stories tell children's stories about coming to live in the UK

WiFi activity, but they fixed this by switching off the wireless LAN adapter on Raspberry Pi since it wasn't used for this project.

The team used switches to trigger audio playback when visitors picked up headphones. Other switches were added to restart an audio or video feed, or to shut down Raspberry Pi.

> "The software is mainly just keeping track of whether someone is still present"

Achieving smooth 1920×1080 video playback was one of the trickier elements. Chris and his colleagues decided to use omxplayer (**magpi.cc/omxplayer**) because it can fade in and out. This was controlled using the omxplayer-controll package (**magpi.cc/omxcontroll**).

The main program was written in Node.js as that's where their programmer's expertise lay, and also because it offered several useful libraries. Clay's expertise is in creating exhibits with projection mapping, presence detection – ideal for the PIR motion sensors – and two-way mirrors.

"After being triggered and starting to play media, the software is mainly just keeping track of whether someone is still present," says Chris.

Once there's nobody around, it will stop playing media. In the case of the barber's, the mirror becomes a normal mirror again and, in the kitchen, the animated plates and tablecloth will return to normal.

"Raspberry Pi is a really interesting platform for museum work as it's small, flexible, and cost-effective. Its popularity also means it's easy to get hold of. We are definitely thinking of it as a possible solution to all sorts of needs."

The Migration Museum is planning a phased reopening in late summer / early autumn. For more details, visit **migrationmuseum.org**.

Pi Commander

Adrien Castel's idea of converting an old electronic toy into a retro games machine was no flight of fancy, as **David Crookes** discovers

MAKER

Adrien Castel

Adrien is an IT technician and graphic designer specialising in 3D and VFX. He lives in France and loves building stuff that involves electronics.

magpi.cc/picommander

The 1980s was a golden era for imaginative electronic toys. Children would pester their parents for a Tomytronic 3D or a Nintendo Game & Watch. And they would enviously eye anyone who had a Tomy Turnin' Turbo Dashboard with its promise of replicating the thrill of driving (albeit without the traffic jams).

Two years ago, maker Matt Brailsford turned that amazing toy into a fully working Out Run arcade machine and Adrien Castel was smitten. "I loved the fact that he'd upcycled an old toy and created something that could be enjoyed as a grown-up," he says. "But I wanted to push the simulation a bit further and I thought a flying sim could do the trick."

Ideas began flying around Adrien's mind. "I knew what I wanted to achieve so I made an overall plan in my head," he recalls. First he found the perfect toy: a battery-powered Sky Fighter F-16 tabletop game made by Dival. He then decided to base his build around a Raspberry Pi 3A+. "It's the perfect hardware for projects like this because of its flexibility," Adrien says.

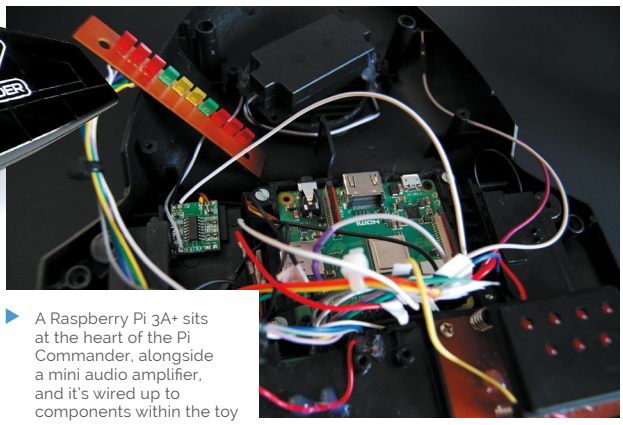
Taking off

The toy needed some work. Its original bright red joystick was missing and Adrien knew he'd have to replace the original screen with a TFT LCD. To do this, he 3D-printed a frame to fit the TFT display and he created a smaller base for the replacement joystick. Adrien also changed the microswitches for greater sensitivity but he didn't go overboard with the changes.

"I knew I would have to adapt some parts for the joystick and for the screen, but I didn't want to modify the look of the toy," Adrien explains. "To be honest, modifying the toy would have involved some sanding and painting and I was worried that it would ruin the overall effect of the project if it was badly executed."

As such, a challenge was set. "I had to keep most of the original parts such as throttle levers and LEDs and adapt them to the new build," he says. "This meant getting them to work together with the system and it also meant using the original PCB, getting rid of the components and re-routing the electronics to plug on the GPIOs."

The games can make use of the full screen. Adrien would have liked a larger screen, but the original ratio oddly lay between 4:3 and 16:9, making a bigger display harder to find



A Raspberry Pi 3A+ sits at the heart of the Pi Commander, alongside a mini audio amplifier, and it's wired up to components within the toy

Quick FACTS

- Dival's Sky Fighter F-16 toy is repurposed
- Its internal components were desoldered
- A four-inch 800×480 HDMI TFT screen is used
- Adrien suggests trying an analogue joystick
- Games are chosen using a Select key

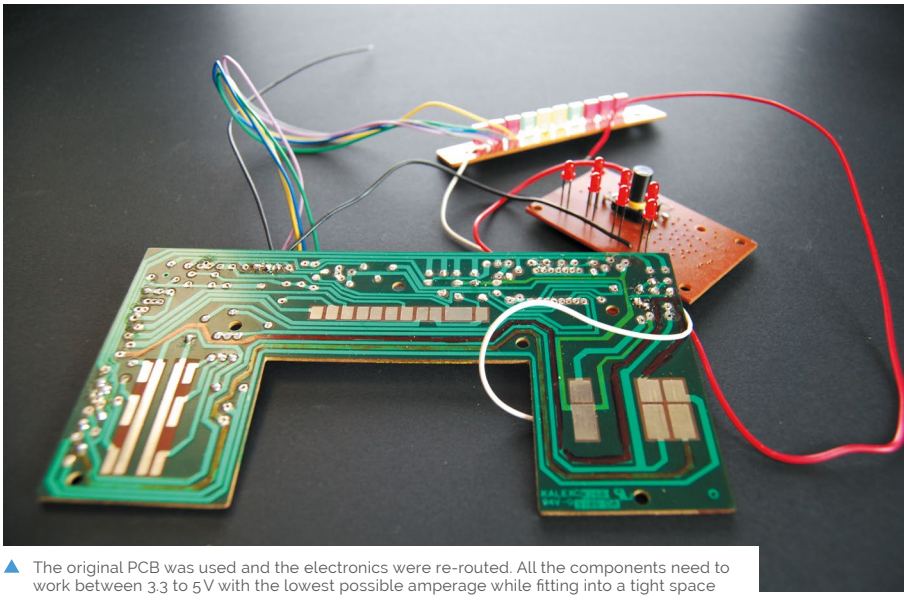
There were some enhancements. Adrien soldered a PAM8403 3W class-D audio amplifier to Raspberry Pi and this allowed a basic speaker to replace the original for better sound. But there were some compromises too.

"At first I thought the screen could be bigger than the one I used, but the round shape of the cockpit didn't give much space to fit a screen larger than four inches." He also believes the project could be improved with a better joystick: "The one I've used is a simple two-button arcade stick with a jet fighter look."

> "I didn't want to modify the look of the toy"

Flying high

By using the retro gaming OS Recalbox (based on EmulationStation and RetroArch), however, he's been able to perfect the overall feel. "Recalbox allowed me to create a custom front end that matches the look of a jet fighter," he explains. It also means the Pi Commander plays shoot-'em-up games alongside open-source simulators like FlightGear (**flightgear.org**). "It's a lot of fun."



▲ The original PCB was used and the electronics were re-routed. All the components need to work between 3.3 to 5V with the lowest possible amperage while fitting into a tight space

SUBSCRIBE TODAY FROM ONLY £5

SAVE UP TO 35%

Subscriber Benefits

- **FREE Delivery**
  Get it fast and for FREE
- **Exclusive Offers**
  Great gifts, offers, and discounts
- **Great Savings**
  Save up to 35% compared to stores

Rolling Monthly Subscription

- Low monthly cost (from £5)
- Cancel at any time
- Free delivery to your door
- Available worldwide

Subscribe for 12 Months

£55 (UK) £90 (USA)
£80 (EU) £90 (Rest of World)

Free Raspberry Pi Zero W Kit with 12 Month upfront subscription only (no Raspberry Pi Zero W Kit with Rolling Monthly Subscription)

Subscribe by phone: **01293 312193**

Subscribe online: **magpi.cc/subscribe**

Email: **magpi@subscriptionhelpline.co.uk**

BUILD A RASPBERRY PI 4

GAMES CONSOLE

Retro gaming perfected with RetroPie on Raspberry Pi 4

It's taken a while to get it working, but it's finally here: RetroPie is now compatible with Raspberry Pi 4. For those new to Raspberry Pi and retro gaming on it, RetroPie is an operating system for Raspberry Pi that includes a number of console emulators. With an easy setup procedure, it's one of the nicest and simplest ways to do retro gaming on a Raspberry Pi.

In the past we've shown you how to build big arcade machines and tiny handhelds, but for this new release we thought we'd do something a bit simpler. Create a Raspberry Pi 4 plug-and-play games console. Get ready.

GEAR UP

Cases and other accessories for your Raspberry Pi-powered retro consoles

SLEAK AND MODERN CASES

Official Raspberry Pi 4 case

magpi.cc/pi4case

Argon ONE

magpi.cc/argonone

FLIRC

magpi.cc/flirc

CLASSIC ENCLOSURES

SUPERPi

magpi.cc/superpi

NESPi magpi.cc/nespicase

MEGAPi

magpi.cc/megapi

ACCESSORIES

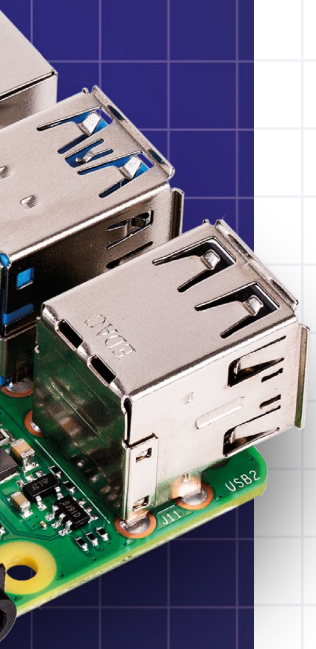
USB controller – SNES style

magpi.cc/usbpad

USB controller – Mega Drive style

magpi.cc/usbsixpad

USB storage



INSTALL RETROPIE ON RASPBERRY PI 4

Use RetroPie software to emulate classic consoles and play retro video games on Raspberry Pi 4

YOU'LL NEED

- Raspberry Pi 4
- Raspberry Pi Imager
- RetroPie 4.6 or later **retropie.org.uk/download**

RetroPie is software you can use to play retro video games on a Raspberry Pi. It's not the only option, but it is popular and extremely easy to use. Until recently RetroPie has only worked on Raspberry Pi 3 and earlier, but the RetroPie 4.6 update introduced Raspberry Pi 4 compatibility.

Video games are stored as ROM files, and instead of loading tapes, discs, and cartridges, you transfer ROM files to your Raspberry Pi storage and load them directly from RetroPie.

In this tutorial, we will install RetroPie 4.6 on a Raspberry Pi 4 and set up our controller.

Warning!

It is illegal to download copyrighted game ROMs from the internet. Please respect the original maker and seek a legal source for retro gaming instead. We use homebrew ROMs made by modern makers for classic systems.

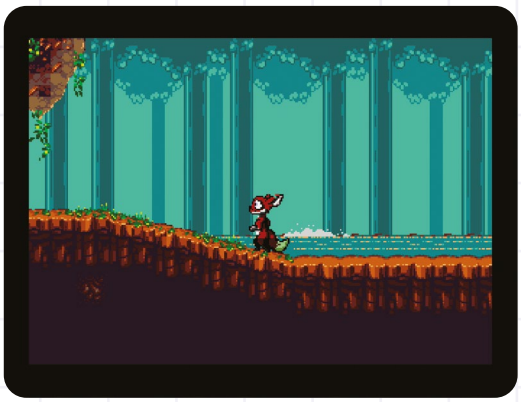
See **magpi.cc/legalroms** for examples and links.

01 Raspberry Pi Imager

Start by installing Raspberry Pi Imager on your Mac, Windows, or Ubuntu computer (**magpi.cc/downloads**).

Raspberry Pi Imager offers a selection of different OS installations, but RetroPie isn't currently one of them. So download the RetroPie 4.6 image file for Raspberry Pi 4 from **retropie.org.uk/download**.

Attach your microSD to the computer and open Raspberry Pi Imager. Click on Choose OS and scroll down the list to 'Use custom'. Click it and choose the **retropie-buster-4.6-rpi4.img.gz** file that you just downloaded, then click Open. Now click on Choose SD Card and ensure that the microSD card you inserted is selected, then click on Write. When prompted, remove the microSD card.



▲ Here we see a modern game called Tanglewood up and running on our emulated system

02 Connect the controller

Connect your chosen game controller to a USB port on your Raspberry Pi 4. It is possible to use RetroPie with a keyboard, but it isn't much fun.

Insert the microSD card into Raspberry Pi 4 and power it up. The welcome screen will appear.

03 Configure the controller

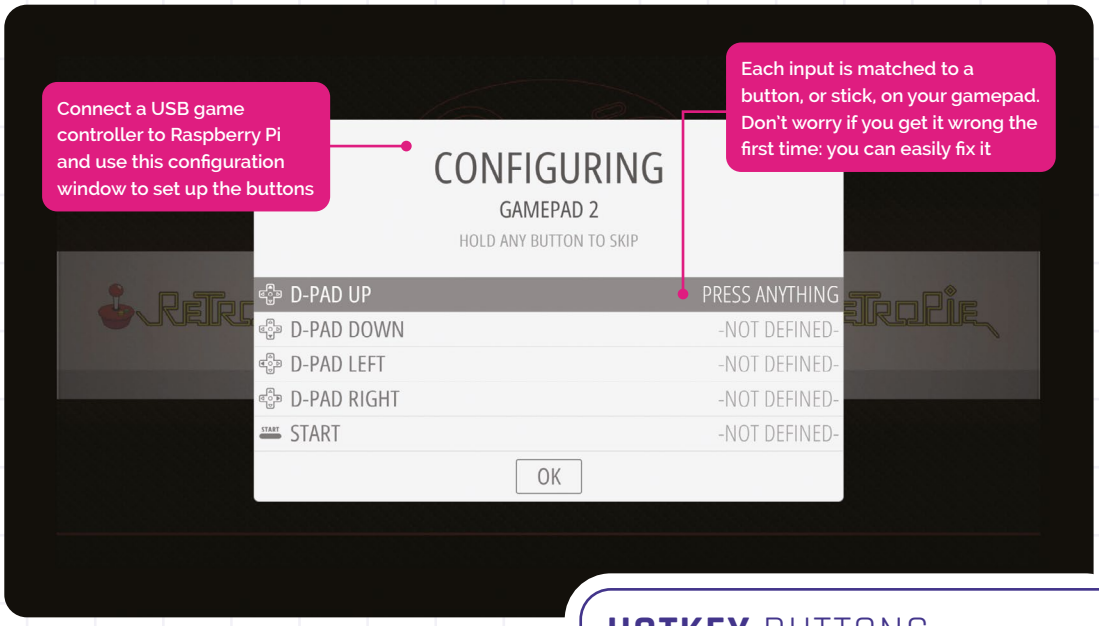
The first thing you need to do is configure the controller. The welcome screen will display 'No gamepads detected'. Don't worry: you just need to push one of the buttons on your gamepad.

The Configuring window will appear. Now you need to push each button on the gamepad in the order of the list. Start by pushing the D-pad up, then D-pad down, and work through the remaining buttons (see this guide for reference: **magpi.cc/retropiecontrollers**). Don't worry if you get a button wrong: when you get to the end of the list, you can use the D-pad and the A, B buttons to reselect and reconfigure the other buttons.

The Hotkey button is used in combination with other buttons to perform actions in a game (such as return to the main menu). It's recommended that you press the Select button for this option. See the 'Hotkey buttons' box for more commands.

If you see 'button already taken', hold down the A button to clear the selection.

When you're happy with all the buttons, use the D-pad and A button to select OK.



04 Where are the emulators?

We are going to use a separate USB thumb drive to store our games. This can be easily transferred back and forth from our computer and RetroPie games console. The USB drive needs to be formatted to FAT32 or NTFS.

Right-click the drive in Windows File Explorer and choose Format. Select FAT32 (Default) as the File System and click Start. On a Mac, use Disk Utility to format the USB stick.

Open the formatted drive and create an empty folder called **retropie**. Remove the USB drive from your computer and connect it to your Raspberry Pi. Give it a few minutes (if it has a LED on it, wait for it to stop blinking).

Remove the USB drive from Raspberry Pi and reconnect it to your computer. Open the File Explorer and take a look inside the **retropie** folder; you'll see a file structure has been created, with three directories: **BIOS**, **configs**, **ROMs**.

Open the **ROMs** folder and you'll see more folders, for many retro gaming systems. You will put ROM files into these folders.

05 Find a game

Now let's find a game to play on our console. We have a collection of legal ROMs you can download for your RetroPie console at **magpi.cc/legalroms**. We're going to use a homebrew game called Tanglewood (**magpi.cc/tanglewood**). This game was designed for the Sega Mega Drive/Genesis system.

HOTKEY BUTTONS

Use the Hotkey and the following buttons to perform in-game actions:

Hotkey Combination	Action
Hotkey + Start	Exit
Hotkey + Right Shoulder	Save
Hotkey + Left Shoulder	Load
Hotkey + Right	Input State Slot Increase
Hotkey + Left	Input State Slot Decrease
Hotkey + X	RGUI Menu
Hotkey + B	Reset

Click 'Download the Demo' and save the ROM file to your computer. Now copy the **TANGLEWD.BIN** file to the USB thumb drive, placing it in the **retropie/roms/megadrive** folder.

Disconnect the thumb drive and connect it to the Raspberry Pi.

Now press the Start button on your controller and choose Quit > Restart EmulationStation > Yes.

06 Play the game

When EmulationStation restarts, you will see a Sega Mega Drive logo alongside the RetroPie logo (and it will display '1 Games Available'). Press A on your controller to select the Mega Drive and then pretty A again to start the Tanglewood demo.

Enjoy playing your modern game on a classic console. When you're done, press Hotkey + X to access the menu (if you followed our advice that will be Select + X). Choose Close Content, then Quit RetroArch to return to EmulationStation.

TOP TIP

Quick save

Be sure to use Hotkey + Right Thumb to save your game and Hotkey + Left Thumb to quickly reload.

TOP TIP

Don't unzip

Our Tanglewood demo is a .bin file. Many ROMs are distributed as zip files. You don't need to extract these – just copy the zip file into the corresponding folder to play the game.

CONFIGURING RETROPIE

Getting RetroPie to play a game is just the starting point. There's a wealth of settings for you to perfect your games console

YOU'LL NEED

- Raspberry Pi running RetroPie
- Wireless LAN network

Now that you have a RetroPie console playing a game, it's time to get it ready for use. Like most video game consoles, you'll most likely want to put it underneath your television in the front room.

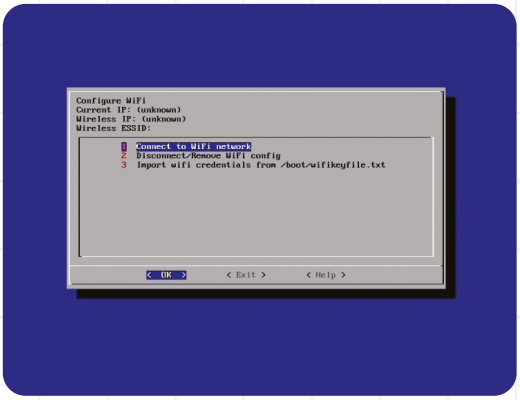
You already have a controller to navigate the EmulationStation interface and play games. But there are a few other things you can do to perfect your retro games console. In this tutorial, we will dive into the settings and configuration window, and set everything up.

01 Connect to a network

Transferring ROMs via the USB stick is practical, but wirelessly beaming ROMs over your network is much easier. If you have an Ethernet cable, connect Raspberry Pi to your modem/router and it will use DHCP to connect.

Otherwise, you can connect RetroPie to a wireless network. From the EmulationStation interface, switch to the RetroPie icon (it will say 'Configuration' beneath).

Use the gamepad to choose the WiFi option at the bottom of the RetroPie Configuration menu and press A to select it.



▲ Connect your retro games console to a wireless (or wired) network to transfer ROM files directly to it

The screen will go blue and you will see a message saying 'You don't currently have your WiFi country set'. Choose Yes to open raspi-config.

02 raspi-config

At this point, you will find it easier to use a keyboard (if you have one attached). Use the arrow keys and enter to choose Localization Options > Change Wi-fi country and select GB Britain (UK) – or your local area. Press **ENTER** and then **ENTER** again at the OK message. Choose Finish.

Choose WiFi again and click 'Connect to WiFi network'. Choose your local wireless LAN network from the list and enter the WiFi key/password. Choose OK.

Insert the microSD card into Raspberry Pi 4 and power it up. The welcome screen will appear.

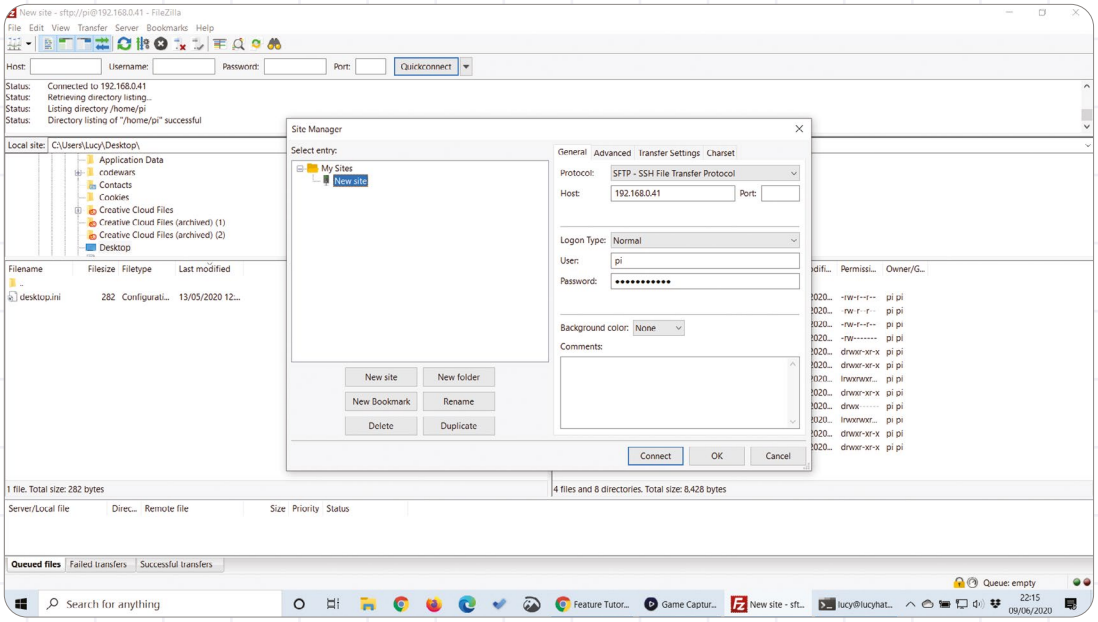
> "You must change the default password to something personal and secure"

03 Change password

We are going to open up RetroPie to work across our network. But first, you must change the default password to something personal and secure.

While still in raspi-config, choose Change User Password. The blue interface will display 'You will now be asked to enter a new password'. Press **ENTER** and a 'New password' prompt appears right at the bottom of the screen (it is easy to miss).

Enter your new password and press **ENTER**; retype it and press **ENTER** again and you will see 'Password changed successfully'. Do not skip this step.



FileZilla is used to transfer files from one location to another

Access raspi-config from within RetroPie to fine-tune the underlying interface options

04 Secure shell

Still in raspi-config, choose Interfacing Options > SSH. Choose Yes and press **ENTER**. You will see 'The SSH server is enabled'. Press **ENTER**. Press OK and choose Finish to head back into RetroPie.

Back in the RetroPie Configuration menu, select Show IP Address. Make a note of the four-digit number (you'll need this to connect remotely to Raspberry Pi and share files).

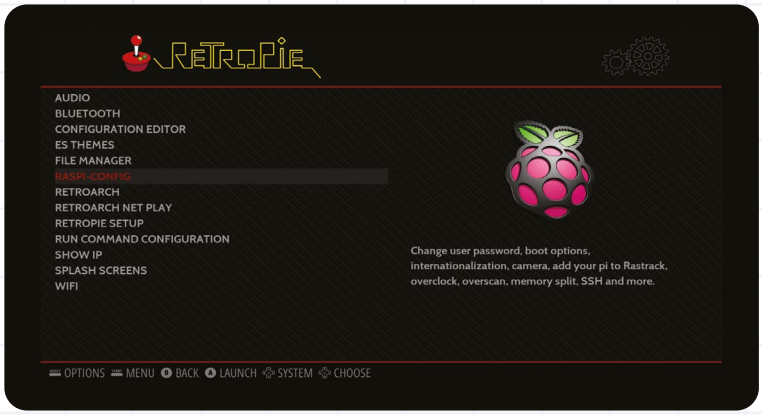
05 Get an FTP program

To transfer files ROMs wirelessly from your computer to Raspberry Pi, you should install an FTP (File Transfer Protocol) program on the computer, although you can do it from the command line using SCP (Secure Copy Protocol).

We recommend using FileZilla, which you can download from Mozilla (**filezilla-project.org**). Open FileZilla on your computer and choose File > Site Manager.

Choose SFTP as the protocol and enter the IP address from Step 04. The username field is 'pi'; enter the password you chose in raspi-config.

Click Connect and you will be able to wirelessly connect to the Raspberry Pi. On the left are your local files; on the right is the file system of RetroPie. Drag and drop ROM files from your local computer to RetroPie on your network.



06 Runcommand

The final configuration tool you should be aware of is runcommand. This menu is accessed when launching a game by pressing the A button as it loads.

Here you can choose which emulator to use for that type of console, or pick a specific emulator to use when launching the ROM file (this can be handy if the emulated game does not work). You can also set the default video mode, and launch with verbose mode to debug games.

As you use RetroPie, you will undoubtedly come across ROM files that do not work straight away. The runcommand interface can help. Enjoy the wide variety of games available to you on your retro games console.

TOP TIP

Overclocking

Overclocking Raspberry Pi will enable it to run faster, at risk of crashing and overheating the hardware. Take a look at RetroPie's documentation to find out more. **magpi.cc/retropieoverclock**

HANDHELD CONSOLES

Want something more portable? Check out these projects and kits for making your own handheld

PIGRRL

The PiGRRL2 is an amazing project from the Ruiz brothers that takes a Raspberry Pi and turns it into a fully functional Game Boy-esque device, albeit with the ability to play many more games. You can buy a kit and get the 3D files to make one yourself, although if you plan to use a Raspberry Pi 4 you'll need to make some modifications.

magpi.cc/pigrrl

RETROFLAG GPI

If you don't fancy 3D printing, this handheld kit has you covered. It also has a gimmick that PiGRRL doesn't: the 'cartridge' is removable, and it can house a Raspberry Pi Zero inside it so you actually have to insert the cartridge to turn the system on. Clever! It also means it makes updating and adding software a cinch.

magpi.cc/gpi

PISWITCH

Yes, this is a handheld retro console that looks and acts like a Nintendo Switch. It even makes use of real Switch Joy-Con controllers which can be slid in and out. While it doesn't dock to a TV the same way, you still get to use the removable controller feature of the original.

magpi.cc/piswitch

PLUG 'N' PLAY ARCADE STICKS

If you'd prefer to replicate the arcade experience, here are some ideas that won't require too many tokens

PICADE CONSOLE

The smaller version of the excellent Picade allows you to simply plug into any HDMI television to start your gaming. It's also a great starter kit, coming with the powerful Picade X HAT which can be used to create any size and type of Raspberry Pi arcade machine you wish to build.

magpi.cc/picadecons

MONSTER JOYSTICK DELUXE

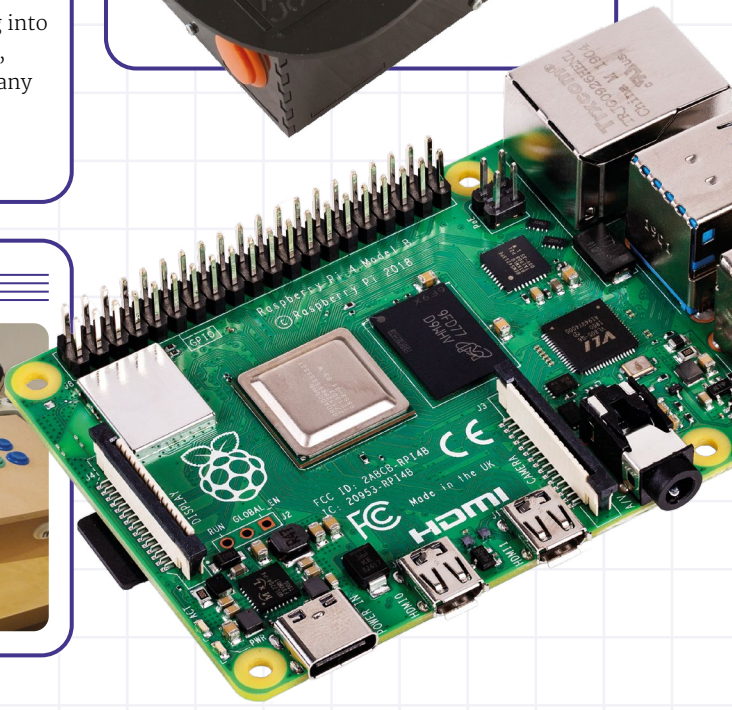
This joystick may not look that special, but it's deluxe for a reason: it contains official Sanwa arcade parts. These are very high-quality buttons and sticks, so will survive the most intense Street Fighter II matches, or the most aggressive Tetris player.

magpi.cc/monster

SELF-CONTAINED ARCADE

While maybe not that easy to move around, this custom two-player arcade stick does have its own charm. Sure, you can plug a USB controller into other sticks, but to get the true arcade co-op or competitive experience, building a custom two-player machine is a great idea.

magpi.cc/2parcade




FROM THE MAKERS OF The MagPi THE OFFICIAL RASPBERRY PI MAGAZINE
RETRO GAMING WITH RASPBERRY PI
RETRO GAMING
WITH
RASPBERRY PI
164 PAGES OF
VIDEO GAME PROJECTS
+ BUILD AN ARCADE MACHINE
PLAY & CODE GAMES!
AMAZE BALLS!

Part 03

High Quality Camera: Control the camera from Python

Use the picamera library to access the camera in Python programs

So far in this series, we've looked at using the HQ Camera (or standard Camera Module) from the command line. But what if you want to control it from a Python program? This is where the picamera library comes in, enabling you to access all the camera's features in Python. In this month's instalment, we'll take a look at how to use it to take stills, shoot videos, alter settings, and add a range of effects.

01 Getting started

The picamera library comes pre-installed in the most recent versions of Raspberry Pi OS. If it's not present already, you can install it manually. In a Terminal window, enter:

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

With your camera already connected and enabled in Raspberry Pi Configuration, open Programming > Thonny from the Raspberry Pi OS desktop menu. Create a new file by clicking File > New file. Save it with File > Save, naming it **listing1.py**. Note: Never name a file 'picamera.py', as this is the file name for the picamera library itself!

Now enter the code from **listing1.py** (overleaf). Save it with **CTRL+S** and run with **F5**. The full-screen camera preview should be shown for ten seconds, and then close. Note: To be able to see the preview when using VNC for remote access from another computer, open the VNC Server menu and go to Options > Troubleshooting, then select 'Enable direct capture mode'.

If the preview appears upside-down, add the line `camera.rotation = 180` just above `camera.start_preview()`. Other possible rotation values are 90 and 270.

You can alter the transparency level of the preview by entering an alpha value – from 0 to 255 – within the latter command's brackets; e.g. `camera.start_preview(alpha=200)`.

It's also possible to change the position and size of the preview. For example, to place its top corner 50 pixels right and 150 down, and resize it to 1024 × 576:

```
camera.start_preview(fullscreen=False,
window = (50,150,1024,576))
```

02 Take a photo

Now let's take a still photo. We can do this by adding the line:

```
camera.capture('/home/pi/Desktop/image.jpg')
```

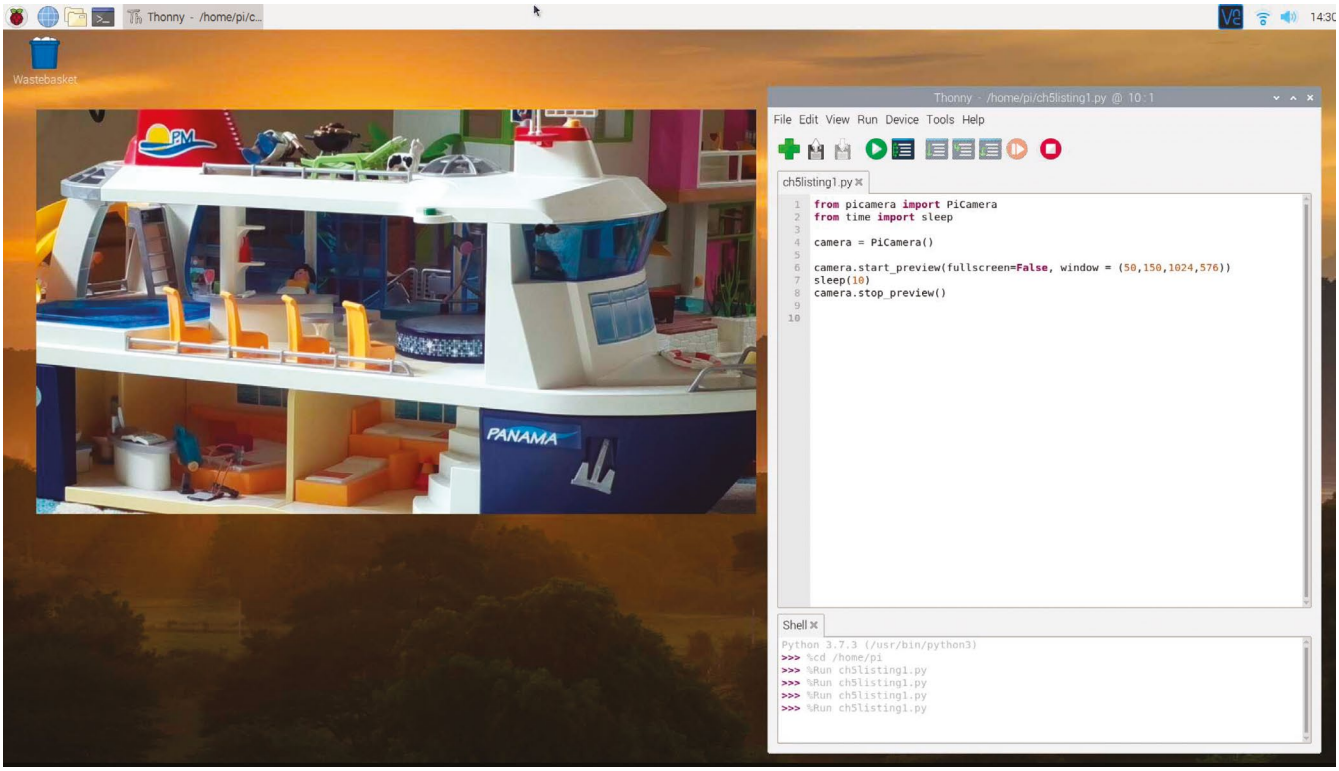
...just after the `sleep` in our code, so it looks like **listing2.py**. Run the code and after a preview of five seconds (as set by `sleep`), it'll capture a photo as **image.jpg**. You may see the preview adjust to a different resolution momentarily as the picture is taken. In this example, the resulting image file will appear on the desktop; double-click its icon to open it.

You can alter the file name and directory path in the code, along with the `sleep` time. Remember, though, that it should be at least five seconds, to give the camera sensor enough time to adjust its light levels.

Top Tip

Video resolution

Note that the maximum video resolution for the High Quality Camera – and both versions of the Camera Module – is 1080p (1920×1080 pixels).



▲ The camera preview can be resized and positioned to your liking

> "The picamera library makes it easy to use a loop to take a sequence of photos"

03 Make a loop

The great thing about using Python with the picamera library is that it makes it easy to use a loop to take a sequence of photos. In Thonny, create a new file and enter the code from **listing3.py**.

After initiating the camera preview, we add a `for` loop with a range of 5, so it will run five times to take five photos. The `sleep` command sets the time between shots, captured using the line:

```
camera.capture('/home/pi/Desktop/image%s.jpg' % i)
```

Here, the `%s` token is replaced by whatever we add after the `%` following the file name – in this case, the variable `i` set by our `for` loop. Note that `i` will range from 0 to 4, so the images will be saved as **image0.jpg**, **image1.jpg**, and so on. Once they're all taken, the preview will close. In this example, you'll see the five files on your desktop; double-click to open them.

You can also use a `for` loop to alter camera setting levels, such as brightness over time. For more details, see Step 04.

04 Control camera settings

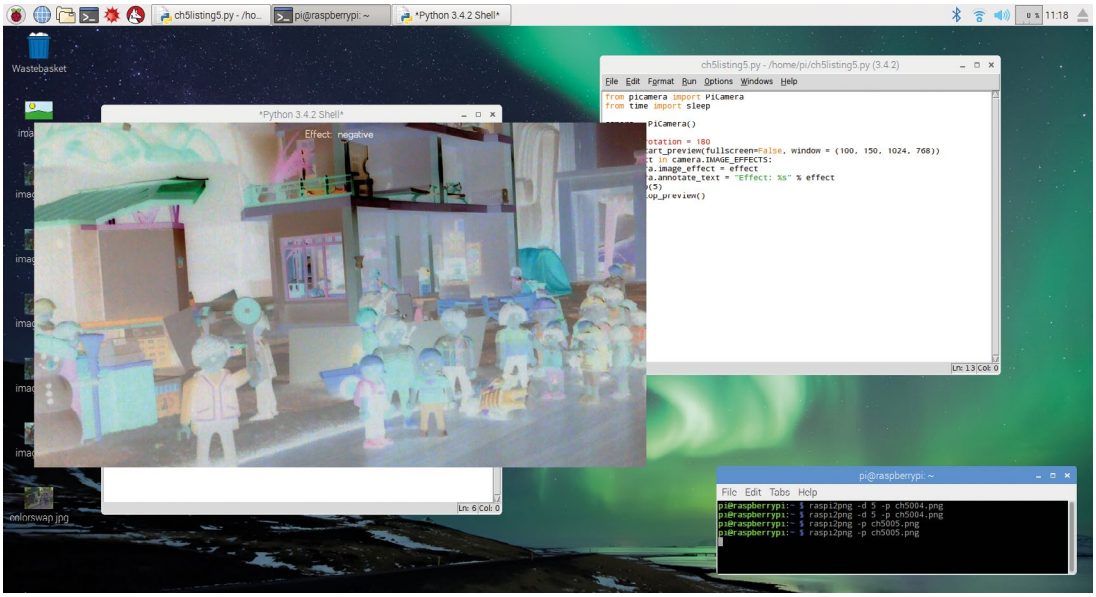
Brightness is just one of numerous settings available for the camera. Here's a list of the main options, along with their default values (and ranges where applicable):

```
camera.brightness = 50 (0 to 100)
camera.sharpness = 0 (-100 to 100)
camera.contrast = 0 (-100 to 100)
camera.saturation = 0 (-100 to 100)
camera.iso = 0 (automatic) (100 to 800)
camera.exposure_compensation = 0 (-25 to 25)
camera.exposure_mode = 'auto'
camera.meter_mode = 'average'
camera.awb_mode = 'auto'
camera.rotation = 0
camera.hflip = False
camera.vflip = False
camera.crop = (0.0, 0.0, 1.0, 1.0)
```

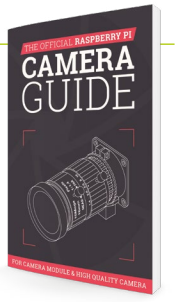
The resolution of the capture is also configurable. For example:

```
camera.resolution = (1024, 768)
```

The maximum resolution for photos is 4056 × 3040 (HQ Camera), 3280 × 2464 (Camera Module v2), or 2592 × 1944 (Camera Module v1). Note: you may need to increase gpu_mem in **/boot/config.txt** to achieve full resolution with the Camera Module v2.



▲ Using the ch5listing5.py code, you can view a loop of all the effects on offer



The Official Raspberry Pi Camera Guide

For further tutorials on how to use your HQ Camera or Camera Module, take a look at our new book, *The Official Raspberry Pi Camera Guide*. Its 132 pages are packed with essential info and a range of exciting projects. **magpi.cc/cameraguide**

05 Add image effects

Just as when you are using the command line, a wide range of effects can be added to the camera in real-time, shown in the preview window. The `camera.image_effect` command is used to apply a particular image effect. The available options are: `none` (the default), `negative`, `solarize`, `sketch`, `denoise`, `emboss`, `oilpaint`, `hatch`, `gpen` (graphite sketch effect), `pastel`, `watercolor`, `film`, `blur`, `saturation`, `colorswap`, `washedout`, `posterise`, `colorpoint`, `colorbalance`, `cartoon`.

For instance, to take an image with a colour swap effect, enter the code from **listing4.py** and run it.

To loop through the various image effects in a preview, run the code from **listing5.py**. Note that this uses the `camera.annotate_text` command to add a text message to the preview; this can also be applied to captured images (when using the sensor's full field of view).

For more details on these effects and other settings, read the official picamera documentation at **picamera.readthedocs.io**.

06 Shooting video

To shoot video footage, we replace the `camera.capture()` command with `camera.start_recording()`, and use `camera.stop_recording()` to stop. Enter the example code from **listing6.py**. When you run the code, it records ten seconds of video before closing the preview. To play the resulting file, open a Terminal window from the desktop and enter:

> A wide range of effects can be added to the camera in real-time

```
omxplayer video.h264
```

(Or you can use VLC instead.) Note that it may well play faster than the original frame rate. It's possible to convert videos to MP4 format and adjust the frame rate using the MP4Box utility (installed with `sudo apt-get install gpac`):

```
MP4Box -add video.h264:fps=30 video.mp4
```

All of the image effects and most of the camera settings can be applied while shooting video. You can also turn on video stabilisation, which compensates for camera motion, by adding the following line to your Python program:

```
camera.video_stabilization = True
```

Note: The built-in video stabilisation only accounts for vertical and horizontal motion, not rotation.

listing1.py

Language: **Python 3**

```
001. from picamera import PiCamera
002. from time import sleep
003.
004. camera = PiCamera()
005.
006. camera.start_preview()
007. sleep(10)
008. camera.stop_preview()
```

listing2.py

Language: **Python 3**

```
001. from picamera import PiCamera
002. from time import sleep
003.
004. camera = PiCamera()
005.
006. camera.start_preview()
007. sleep(5)
008. camera.capture('/home/pi/Desktop/image.jpg')
009. camera.stop_preview()
```

listing3.py

Language: **Python 3**

```
001. from picamera import PiCamera
002. from time import sleep
003.
004. camera = PiCamera()
005.
006. camera.start_preview()
007. for i in range(5):
008.     sleep(5)
009.     camera.capture('/home/pi/Desktop/image%s.
     jpg' % i)
010. camera.stop_preview()
```

listing4.py

DOWNLOAD THE FULL CODE: magpi.cc/github

Language: **Python 3**

```
001. from picamera import PiCamera
002. from time import sleep
003.
004. camera = PiCamera()
005.
006. camera.start_preview()
007. camera.image_effect = 'colorswap'
008. sleep(5)
009. camera.capture('/home/pi/Desktop/colorswap.jpg')
010. camera.stop_preview()
```

listing5.py

Language: **Python 3**

```
001. from picamera import PiCamera
002. from time import sleep
003.
004. camera = PiCamera()
005.
006. camera.start_preview()
007. for effect in camera.IMAGE_EFFECTS:
008.     camera.image_effect = effect
009.     camera.annotate_text = "Effect: %s" % effect
010.     sleep(5)
011. camera.stop_preview()
```

listing6.py

Language: **Python 3**

```
001. from picamera import PiCamera
002. from time import sleep
003.
004. camera = PiCamera()
005.
006. camera.start_preview()
007. camera.start_recording('/home/pi/video.h264')
008. sleep(10)
009. camera.stop_recording()
010. camera.stop_preview()
```

Build a DOS emulation system

Use the powerful DOSBox-X emulator to boot Raspberry Pi to DOS and run anything from Windows 3.11 to classic games

MAKER

KG Orphanides

KG enjoys making new computers pretend to be old ones, creating extremely niche games, and generating uncanny electronic screeching.

@KGOrphanides

The extra oomph of the 4GB or 8GB edition of Raspberry Pi 4 provides plenty of power for emulating classics of the past. That even goes as far as installing and running early versions of Windows. This tutorial will show you how to emulate the demanding software titles released in the early through mid-1990s using DOSBox-X.

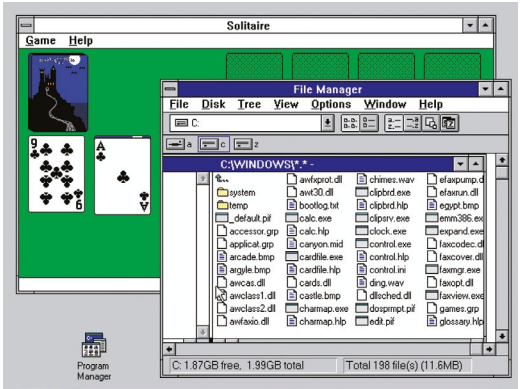
Forked from the original DOSBox emulator, DOSBox-X has more precise hardware emulation, supports a wider range of software, and can effectively run more DOS-related operating systems (up to Windows ME). It also has a sophisticated graphical interface to help you manage tasks such as configuration and virtual disk-swapping.

You'll Need

- DOSBox-X **magpi.cc/dosboxx**
- Windows 3.x (optional) **magpi.cc/vssubs**
- Windows 3.x custom installation guide (optional) **magpi.cc/windows3x**

01 Create your DOS directories

Let's create the directory structure that we'll use to house the software we'll run through DOSBox-X:



▲ Windows 3.11 will cheerfully run either on top of DOSBox-X's default FreeDOS operating system, or installed with DOS 6.22 on a dedicated hard disk image

```
mkdir -p dos/{floppy,cd,games}
```

The **floppy** and **cd** directories will house disk images which we'll be able to switch between in DOSBox-X. This tutorial and our template config files presume you'll keep everything in a **/home/pi/dos/** directory, so be sure to change any paths if you're using a different username or **dos** directory names.

While our generic config file should handle most DOS software well on Raspberry Pi, you can also create separate .conf files for specific programs to better match their requirements and automatically run commands.

02 Tweak your graphics

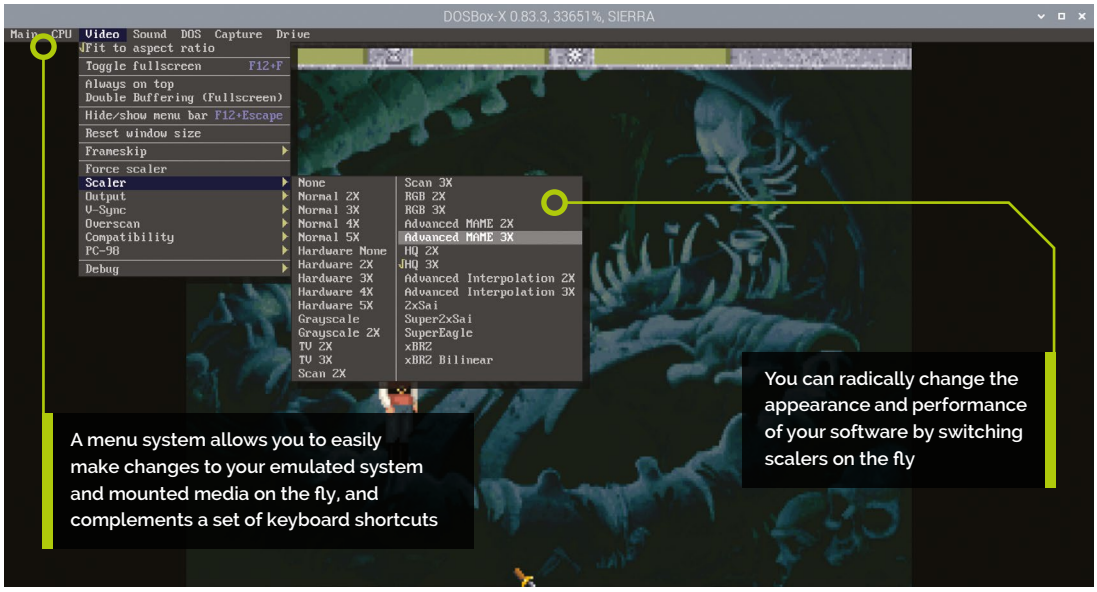
Assuming you're using a standard 1920×1080 display with your Raspberry Pi, you'll find some more demanding DOS software struggling at full resolution, particularly if you have DOSBox-X configured to use OpenGL and aspect ratio correction.

On the desktop, open the main menu, go to Preferences and select Screen Configuration. Right-click on your display – most likely marked HDMI-1 – and from the Resolution menu, select 1280×720. Running your entire GUI at a lower resolution will lighten the load of rendering and upscaling for the emulation and have no adverse effect on games from an era when 800×600 was the norm.

03 Install DOSBox-X

In a Terminal, enter the following:

```
sudo apt install automake libncurses-
dev nasm libsdl-net1.2-dev libpcap-dev
```




```
libfluidsynth-dev ffmpeg libavdevice58
libavformat-* libswscale-* libavcodec-*
 git clone https://github.com/joncampbell123/
dosbox-x.git
 cd dosbox-x
 ./build
 sudo make install
 dosbox-x
```

DOSBox-X should open at its Z: prompt. You can't paste commands into it from the clipboard, but there are some modern convenience features: tab auto-completes, you can scroll through your command history using the up arrow, and you can add startup commands to a config file.

Type `exit` to quit and ensure that the config directory, which we'll need in the next step, is created properly.

04 Export a config file

Restart DOSBox-X and tell it to generate a config file that we can later modify in a text editor, based on the program's default settings and then exit.

```
CONFIG.COM -all -wcd
exit
```

The file we've just made can be found in **/home/pi/.config/dosbox-x** and, at time of writing, is named **dosbox-x-0.83.3.conf**.

> A sophisticated graphical interface helps you manage tasks such as configuration and virtual disk-swapping

As well as being human-readable and conveniently editable in a text editor, you can modify this long and extensively commented file from within DOSBox-X using the configuration GUI in the main menu. This is handy, since DOSBox-X's configuration has more options than that of vanilla DOSBox.

05 Customise your config

For this tutorial, we've created some config files that you can download from **magpi.cc/github**. The code box will run most DOS software. As well as editing your main DOSBox-X config, you can launch DOSBox-X with a specific config file – useful if you wish to easily switch between different OS setups – using the following command-line switch:

```
dosbox-x -conf yourfile.conf
```

We'll take advantage of that later to help install Windows 3.11. Note that your custom config files need only include lines that vary from the defaults. In the following steps, we'll create a config file optimised for playing late-era DOS games on Raspberry Pi 4 with 4GB or 8GB RAM.

Top Tip

A virtual Windows hard disk

For a more authentic Windows 3.x experience – including full networking in 3.11 – you can alternatively use DOSBox-X to create a hard disk image installation of DOS and Windows 3.x.

Top Tip

Take the easy way out

If you don't need DOSBox-X's menus or extra features, the standard version of DOSBox 0.74-3 available in the package repository is a handy alternative. Just `sudo apt install dosbox`. You'll find its config file in **/home/yourusername/.dosbox**.

06 Graphics, scalers, and performance

The default config is already well optimised to run DOS software on most systems, but we should make a few adjustments to improve performance on Raspberry Pi's hardware.

Leave the fullscreen setting as false, as you can enable and disable fullscreen mode using DOSBox-X's menus or the **F12+F** keyboard shortcut; fullresolution should be left as 'desktop'.

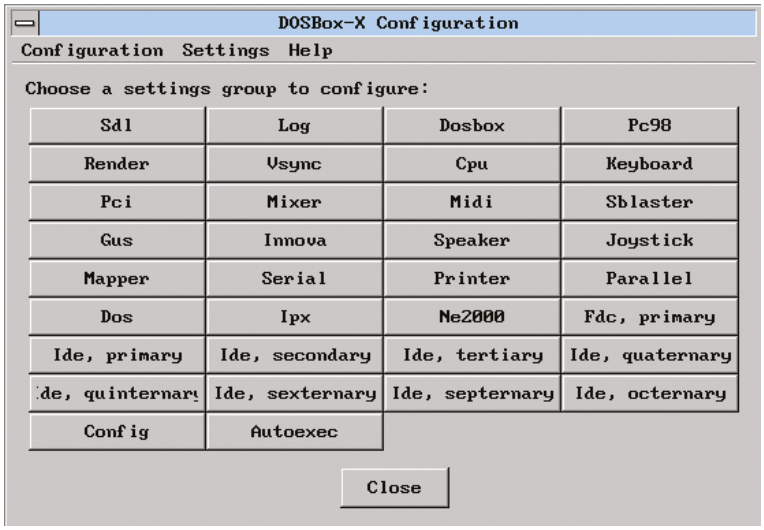
To get proper aspect ratio correction and reasonable graphical fidelity at 1280×720, you should set the output to opengl, aspect to true, and select a scaler up to interpolate low-res graphics. Scaler choice is largely a matter of personal taste, so use the Video menu options to try a few. If your sound becomes choppy, you're pushing Raspberry Pi's capabilities too far.

07 autoexec.bat

At the end of the config file is `autoexec`, where we'll put all our `MOUNT` and `IMGMOUNT` lines to assign drive letters to directories and floppy or CD images, and any commands to run at boot.

In our sample config, we've used MOUNT to set **/home/pi/dos** as DOS's drive C. It's here that we'll copy and install all our software to.

If you use the `IMGMOUNT` command with multiple file names of CD or floppy images, you'll be able to swap between those images in order to swap between media. To swap floppies, use **F12+LEFT-CTRL+D**. To swap CD or DVDs, use **F12+LEFT-CTRL+C**.



One of DOSBox-X's key advantages is a graphical interface that covers each element of your emulated PC's configuration, from CPU emulation to scaler

08 Using DOSBox-X

Like DOSBox, DOSBox-X uses the open-source FreeDOS operating system, rather than Microsoft's proprietary MS-DOS. Navigation through directories isn't too different to using a Bash terminal, particularly as a number of Bash commands have been included, such as `LS` as an alternative to DOS's `DIR`, and `CD ..` alongside the usual DOS `CD..` command. To run a .exe, .com, or .bat file, just type its name without the extension.

To capture and release your mouse, use the **LEFT-CTRL+F10** shortcut. The autolock entry under SDL config enables capture-on-clock.

> "DOSBox-X uses the open-source FreeDOS operating system"

09 Buy Windows 3.11

Now we'll install Windows for Workgroups 3.11, released in December 1993. The biggest challenge is finding a copy to install – usually ageing floppy disks, or disk images if you made backups. We're working from a set of disk images.

If you don't already have one and don't fancy the second-hand market, you can, surprisingly, find it included in Microsoft Visual Studio Subscriptions (formerly MSDN Subscriptions), currently priced at £33.54 per month, for the benefit of developers working on backwards compatibility.

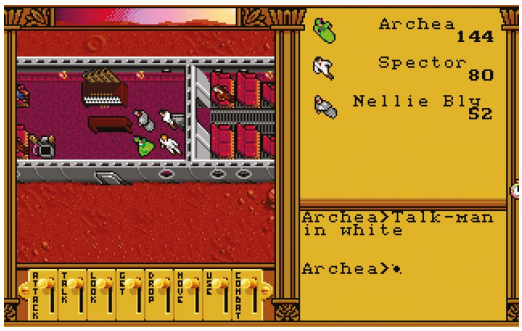
10 Install Windows

Copy the contents of each installation disk or image to a **/win311** subdirectory of the **dos** directory tree we made earlier; you can do this as you normally would on the desktop or at the command line, or by using DOSBox-X's `IMGMOUNT` to mount them and using the DOS `COPY` command while switching disks. At the command line, start DOSBox-X with a Windows-suitable config file – download ours from **magpi.cc/github**.

```
dosbox-x -conf win311.conf
CD WIN311
SETUP
```

Windows 3.11 will install itself. Reboot.

```
CD WINDOWS
WIN
```



▲ RPG classic Worlds of Ultima: Martian Dreams is legally available for free from **GOG.com**, but you'll have to use innoextract 1.8 (**constexpr.org/innoextract**) to pull the files out of it

11 Using Windows 3.x

If you've only ever used Windows 95 or later, Windows 3.x's interface may feel somewhat alien. There's no Start button and if you want to quit back to the DOS prompt, you have to open Program Manager's File menu and select Exit Windows…

The default Program Manager folders, each full of shortcuts to helpful software and settings, are clearly labelled. To explore your mounted DOS drives, open Main and then File Manager. Accessories include MS Paint precursor Paintbrush, a Sound Recorder, and even a Media Player.

A line at the top left of each opened window allows you to move and close it, while minimise and maximise buttons are at the right.

12 Boot Raspberry Pi to DOS

Once you've configured DOSBox-X – and any relevant window managers – to your satisfaction, you can complete your pitch-perfect 1990s PC simulation by booting straight to DOS. Open a Terminal window and type:

```
mkdir /home/pi/.config/autostart
mousepad /home/pi/.config/autostart/dosbox.desktop
```

Add the following to the new text file:

```
[Desktop Entry]
Type=Application
Name=DOSBox
Exec=/usr/bin/dosbox-x
```

This will use DOSBox-X's default config file. You'll need to enable fullscreen in your DOSBox-X config for this to launch correctly, and opengl-dependent aspect ratio correction is also strongly advised.

pi-dos.conf

DOWNLOAD THE FULL CODE: magpi.cc/github

› Language: **DOSBox-X config file**

```
# Basic DOSBox-X config for 90s DOS software on Raspberry Pi.
# See default config file and https://github.com/joncampbell123/dosbox-x/wiki for further documentation

[sdl]
# set fullscreen true if you want to boot to an authentic-feeling DOS environment
fullscreen = false

# Don't forget to set Raspberry Pi's desktop resolution to 1280x720
fullresolution = desktop

# opengl allows aspect ratio correction
output = opengl

[render]
# set frameskip to 1 or 2 for resource-hungry titles
frameskip = 0

# aspect ratio correction
aspect = true

# choose your favourite. Don't use scalers on games that already have high resolutions. Set scaler to none to improve performance.
scaler = advmame3x

[cpu]
# use normal core for multitasking OSes such as Win95
core = dynamic

# some software benefits from emulating a specific CPU, which can be specified here
cputype = auto

# if you experience lag or juddering audio, set CPU cycles to max.
cycles = auto


[autoexec]
# Your DOS autoexec.bat file. These commands will be run at startup, making it easy to mount lots of floppies or CDs at once, as well as your working directories.

mount c /home/pi/dos/

# uncomment and customise these lines to mount floppy and CD images. Remember that DOS isn't case sensitive, but Linux is.

# imgmount a "/home/pi/dos/floppy/disk1.img" "/home/pi/dos/floppy/disk2.img" "/home/pi/dos/floppy/disk3.img"  -t floppy
# imgmount e "/home/pi/dos/cd/a directory with spaces in/sherlock.iso" /home/pi/dos/cd/quake/QUAKE101.cue  -t iso -fs iso

c:
```

Upcycle a vintage radio

Fed up with boring black-box Bluetooth speakers? Upcycle a vintage radio and give it a new lease of life

MAKER

PJ Evans

PJ Evans is a writer, software engineer, and keeps getting told to turn it down.

@mrpjevans

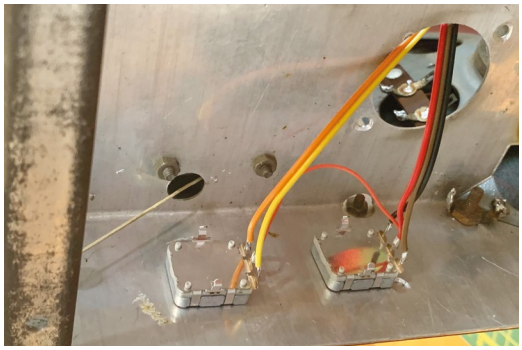
Since Marconi's remarkable wireless inventions of the 1890s, radio has been a part of our everyday lives. This has led to thousands of different types of receiver being designed, made, and sold through the years, each model reflecting the trends of the day. A radio today is more likely to have Bluetooth and streaming capability and resemble a minimalist black cylinder or cube. If you yearn for something with a little more style, why not give a classic radio a new lease of life with a Raspberry Pi computer? From a standalone player to a smart speaker, it can be whatever you want.

01 Source your radio

To upcycle a vintage radio, you need to have one. Auction sites and car boot sales are full of opportunities, and you can choose something that reflects your personality. We looked at Walkmans, 1980s giant boomboxes, and 1920s wooden standing units until settling on a 1950s-era art deco kit radio. Its Bakelite body was in remarkable condition and the controls intact. It doesn't matter if your choice works or not, although it would be a shame to 'break' a vintage piece of equipment. The insides of our unit were heavily corroded, so we decided to strip everything out and give it a good clean.

02 Set up Raspberry Pi

As ever with our tutorials, there comes a time to select and prepare your Raspberry Pi computer. Your choice should be based on your intended use of the radio. If it's just a standalone player, pretty much any model will do. If size is an issue, such as a Walkman, then a Zero W is a good



▲ Here are the rotary encoders in place. These turn freely, but you can get others that 'click' as they turn and include a push-switch

choice. For more powerful applications such as internet streaming, Spotify or AirPlay/Chromecast, consider a 3A+ or 4. The steps are the same regardless. Set up Raspberry Pi OS Lite (we don't need a desktop), connect to your WiFi network, and make sure everything is up to date with `sudo apt update && sudo apt full-upgrade`. Finally, install the GPIO Zero library for Python 3:

```
sudo apt install python3-gpiozero
```

> "Once built and configured, it's time to test everything out before going further"

03 Choose your amplification

Although all Raspberry Pi computers feature direct audio out in one form or another (only via a GPIO workaround on a Zero), there's not enough current to drive a speaker; we need to add an amplifier. You can buy a small amplifier circuit such as Adafruit's inexpensive MAX98306 (**magpi.cc/max98306**) and amplify the audio out signal. We decided to use a dedicated amplifier HAT, JustBoom's AMP HAT, which features an audiophile-grade digital-analogue converter (DAC) and amplifier all in the one HAT. This allows us to use a single power supply, although we did need to go up to 12V. Now the speaker can be attached directly to your Raspberry Pi and HAT.

04 Install your amp

If you've gone for an amplification HAT, such as the JustBoom AMP HAT or HiFiBerry AMP2, it's now time to install it. Most HATs will redirect the sound output to the GPIO and convert the digital signal to analogue using much higher-performing circuitry than you get with our favourite little computer. Finally, the on-board amplifier boosts the signal so it's powerful enough to drive the speaker cone. Setting this up can often involve changing some important configuration files in Raspberry Pi OS, such as **/boot/config.txt**, so be sure to follow all instructions carefully and make backups.

05 Testing 1-2, 1-2

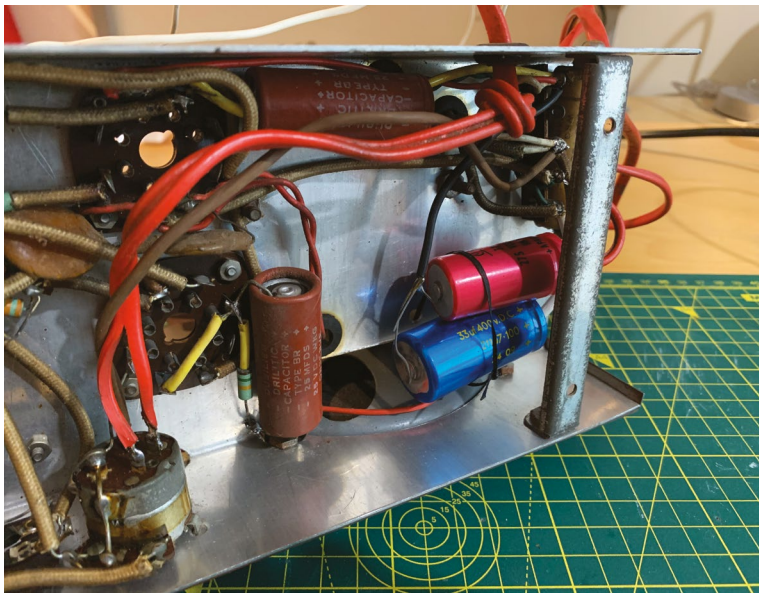
Once built and configured, it's time to test everything out before going further. With everything powered off, use some speaker wire to connect the amp to the speaker. If the speaker's connections indicate polarity, make sure everything matches up. Now boot your Raspberry Pi computer, SSH in if running headless, and set the volume of your amplifier to a low level by running `alsamixer` from the command line. How you do this will vary by HAT. Press **ESC** to quit out of alsamixer and run the following command:

```
speaker-test -c1
```

You should hear 'pink noise' coming form the speaker. If not, go back to alsamixer, increase the volume, and try again. Repeat until everything is working.

You'll Need

- Vintage radio
- JustBoom AMP HAT **magpi.cc/justboom**
- Rotary encoder **magpi.cc/rotaryencoder**
- Suitable PSU for the amp **magpi.cc/justboompsu**
- Speaker wire



▼ Upon opening up this 1950s set, we were presented with home-built electronics that we were not prepared to connect to the mains

Top Tip

Stereo or mono?

If your radio has a single speaker and your audio output is stereo, always use the left channel.

06 Twist and turn

Now we can control the volume using alsamixer, but how to do this when the radio is running? Rotary encoders are control devices that resemble volume controls (although those are normally potentiometers), but can send data to the GPIO from which we can determine whether the control is being turned clockwise or anti-clockwise and work out a value from that. The JustBoom AMP HAT has a breakout to access some unused GPIO pins, so we connected an encoder as shown in the diagram (**Figure 1**). Rotary encoders are available in all shapes and sizes, so chances are you'll find one to match your radio.

07 Reading the rotary

Rotary encoders send pulses on two lines, sometimes referred to a 'clk' and 'dt'. By reading their state we can calculate what's happening. For more on how rotary encoders work, see the projects in *The MagPi* issues 92 and 93. On Raspberry Pi, enter the code in the listing here (or download from **magpi.cc/radiovolume**) and save in your home directory as **volume.py**. You may need to change the GPIO pins used or the command to adjust volume depending on your setup.

08 Turn it up

To try out your volume control, run the following command in the same directory as the file you created in the previous step:

Figure 1

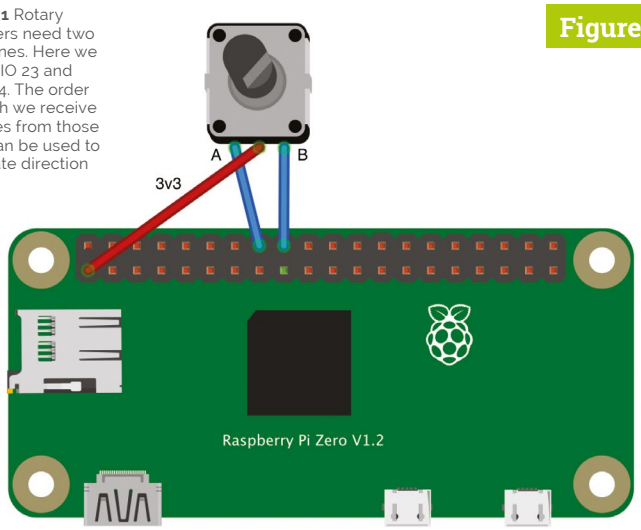


Figure 1 Rotary encoders need two GPIO lines. Here we use GPIO 23 and GPIO 24. The order in which we receive changes from those lines can be used to calculate direction

```
python3 volume.py &
```

(The '&' makes it run in the background.) If you don't see any errors, press **ENTER** and you can now run the previous test again:

```
speaker-test -c1
```

Try slowly turning your encoder clockwise. The volume should get progressively louder. The code is using the `RotaryEncoder` class to track the movement, reading a value between 0 and 100. We then send this to the ALSA audio system as a desired volume percentage. When you're done, use **CTRL+C** to stop the test and type `fg` to switch to the **volume.py** script and use **CTRL+C** to stop that too.

09 Install Mopidy

Now you have a Raspberry Pi-driven audio system with volume control. All that's left is to find something to play on it! There are too many options to cover them all here, but for standalone playback we like Mopidy.

To install Mopidy, you need to add their dedicated repository to your system:

```
wget -q -O - https://apt.mopidy.com/mopidy.gpg | sudo apt-key add -
sudo wget -q -O /etc/apt/sources.list.d/mopidy.list https://apt.mopidy.com/buster.list
```

Now you can update and install:

```
sudo apt update
sudo apt install mopidy
```

Finally, start Mopidy and ensure it restarts on boot:

```
sudo systemctl enable mopidy
mopidy start
```

10 Add the user interface

Mopidy is a headless system, which means it has no user interface, although many are available for free – including the beautiful web-based Iris, which is one of the many plug-ins available for Mopidy. Installation is as straightforward as entering the following:

```
sudo apt install python3-pip
pip3 install mopidy-Iris
```

You should now be able to view the Iris interface at **http://<hostname>:6680/iris/** (where <hostname> is the name or IP address of your Raspberry Pi computer). Here you can add music and services such as Spotify, and control playback.

> Mopidy is a headless system, which means it has no user interface

11 Automatic volume

To make sure our volume script runs when Raspberry Pi OS boots, a quick method is to add it to the **rc.local** file, which is a script that is run when the system is started. To get access, enter the following:

```
sudo nano /etc/rc.local
```

Find the end of the file and above the last line (which reads `exit 0`), enter the following:

```
/usr/bin/python3 /home/pi/volume.py &
```

Be very careful: a mistake here could prevent your system from booting. Press **CTRL+X**, then **Y** to save and exit. Now both Mopidy and your volume control script will run at boot time.

12 Add more

If you have any Apple devices, it's really easy to add AirPlay support. Run the following:

```
sudo apt-get install shairport-sync
sudo service shairport-sync start
```

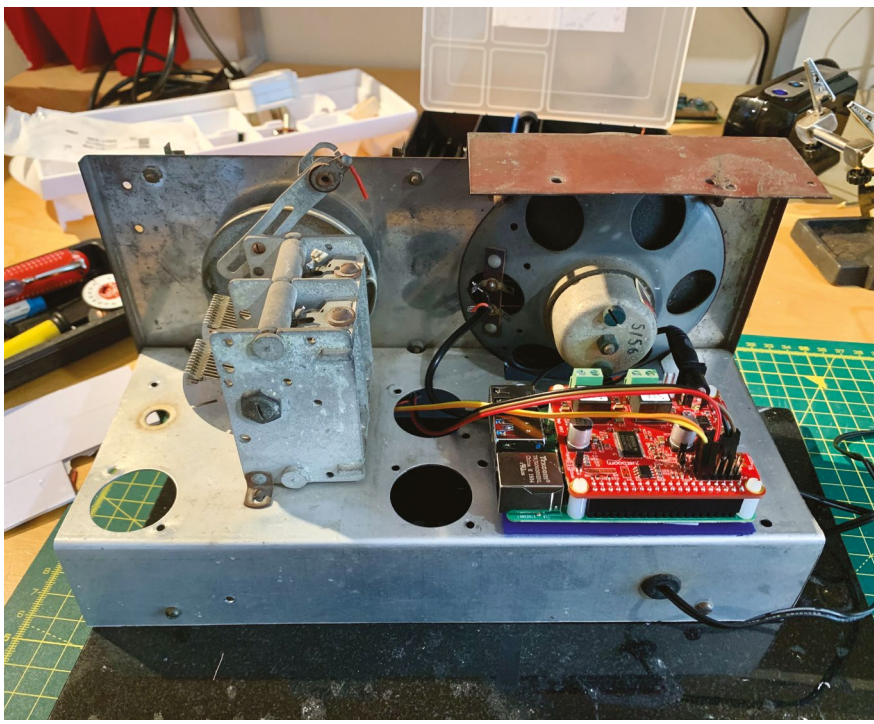
Now your radio will show up as an audio target on your iPhone or Mac. Adding support for Spotify Connect is also possible using librespot (**magpi.cc/librespot**). If you want to see what else can be done, take a look at the advanced **retroradio.py** script at **magpi.cc/retroradiogit**, which uses an additional rotary encoder to add tuning, complete with radio static effects.

volume.py

DOWNLOAD THE FULL CODE:
magpi.cc/retroradiogit

Language: **Python 3**

```
001. import os
002. from time import sleep
003. from rotary import RotaryEncoder
004.
005. vol_control = RotaryEncoder(14, 15, min_val=0, max_val=20)
006. vol_control.watch()
007. vol_last_position = 10
008. vol_control.position = 10
009.
010. # Set initial low volume
011. os.system('amixer set Digital 50%')
012.
013. while True:
014.   if vol_control.position != vol_last_position:
015.     vol_last_position = vol_control.position
016.     os.system('amixer set Digital ' +
     str(vol_last_position * 5) + '%')
017.     print(vol_last_position * 5)
018.
019.   sleep(0.001)
```



▲ Nearly everything has been removed and cleaned, which leaves plenty of space for a Raspberry Pi and amplifier. Note the insulation plate to protect the circuitry

Share your keyboard and mouse with Barrier

MAKER
Lucy Hattersley

Lucy is Editor of *The MagPi* and since getting Barrier installed, she's thrown two keyboards and mice (not 'mouses' – thanks Twitter friends) into a box in the cupboard.

@lucyhattersley

Use Barrier to move your mouse seamlessly from Raspberry Pi to the screen of another computer, and control both machines at once. By **Lucy Hattersley**

We use Raspberry Pi pretty much all the time now, but we still keep other Windows and Mac computers around for those few tasks we can't do on Linux.

It's annoying having to switch between different operating systems, and having multiple keyboards and mice on the desktop. So when Raspberry Pi Director of Software Engineering Gordon Hollingworth wrote a blog post about Barrier (**magpi.cc/sharekeyboardmouse**), we really had to take a look.

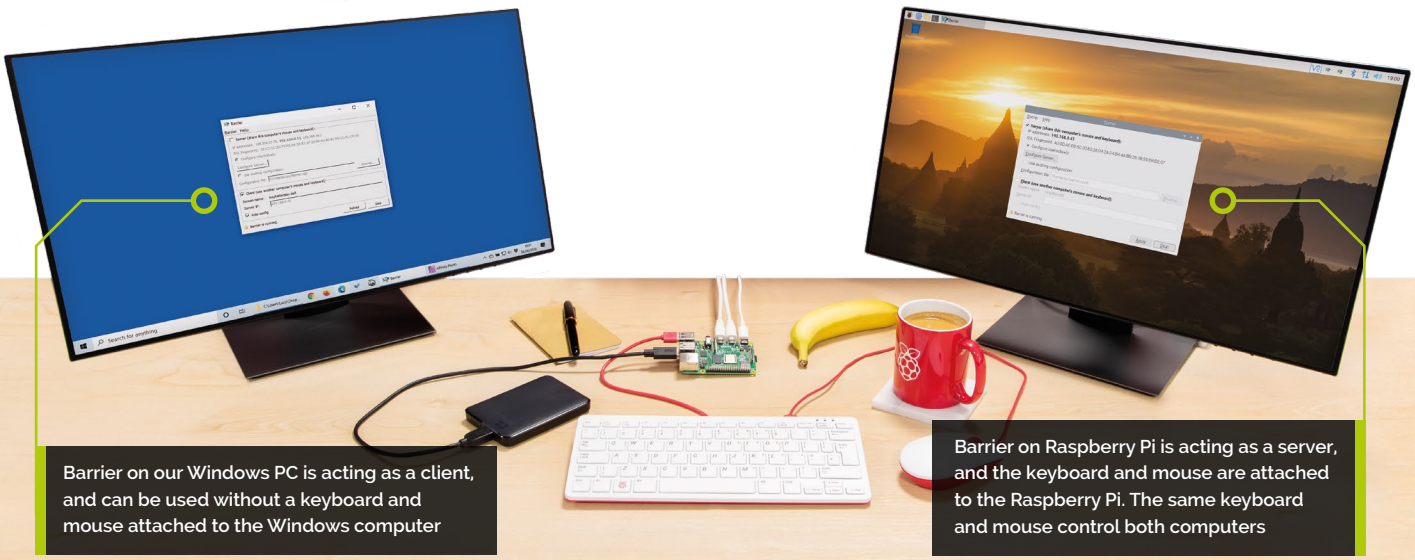
With Barrier installed on your computers, it's possible to use a single keyboard and mouse to control all of your different devices. Simply move the mouse off to one side of your PC and it will appear on the screen of the Raspberry Pi; the keyboard is used to enter text in whatever window has focus.

01 Download Barrier to Raspberry Pi

Barrier is used to share a keyboard between Raspberry Pi and other computers: Windows PC, Mac, or Linux (even a second Raspberry Pi).

We're going to use Barrier to share a keyboard and mouse connected to our Raspberry Pi to a Windows PC on the same network.

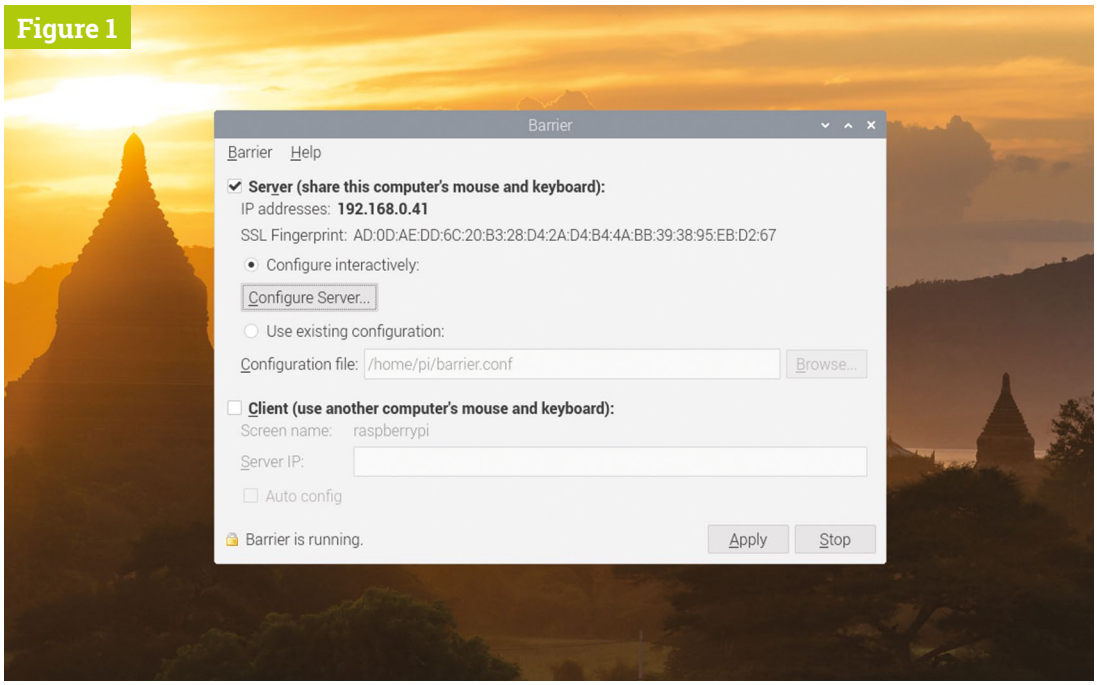
First, install Barrier on Raspberry Pi using APT. Open a Terminal and enter:



Barrier on our Windows PC is acting as a client, and can be used without a keyboard and mouse attached to the Windows computer

Barrier on Raspberry Pi is acting as a server, and the keyboard and mouse are attached to the Raspberry Pi. The same keyboard and mouse control both computers

Figure 1



You'll Need

- Raspberry Pi with Raspberry Pi OS
- A second computer on the same network
- Barrier **magpi.cc/barriergit**

Figure 1 The Barrier server configuration running on Raspberry Pi

```
sudo apt update
sudo apt install barrier -y
```

02 Install Barrier on the client

Now download Barrier to your client computer (in our case the Windows PC) from the Barrier GitHub page (**magpi.cc/barrierrelease**).

Open the **BarrierSetup-2.3.2.exe** program (you may have a later version number). Use the corresponding DMG file for macOS – or APT for a Linux, as shown in the previous step.

Make sure all of your computers are connected to the same network before going any further.

> Now Barrier is installed on both computers, you need to decide which one is going to be in control

03 Client and server

Now that Barrier is installed on both computers, you need to decide which one is going to be in control. This is the one you will have your keyboard and mouse connected to. This will be the 'server' and the other computer will be the 'client'.

Figure 2

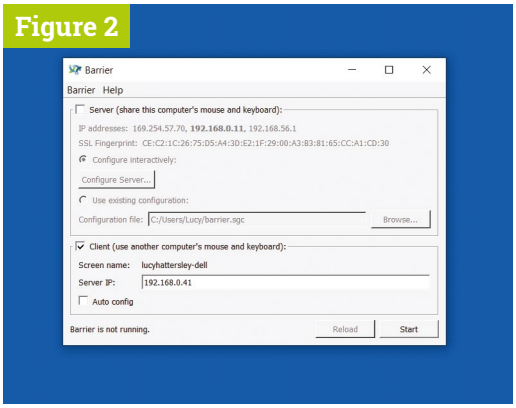


Figure 2 Barrier on the Windows PC and is set up in client mode. Here is our server configuration

Because we use our Raspberry Pi so much, we're going to connect our keyboard and mouse to it and have it control the secondary Windows PC. But it could easily be the other way around.

If you're using a laptop and Raspberry Pi, then it's probably better to set the laptop as the server (because it will always have a keyboard and mouse attached), and the Raspberry Pi as the client.

04 Set up the sever

Barrier is installed on Raspberry OS, and is opened by choosing Menu > Accessories > Barrier. The Barrier window will appear (**Figure 1**).

Warning! Security

Barrier uses SSH to create a link between two computers. Make sure you change the default password on both machines.

magpi.cc/security

Figure 3

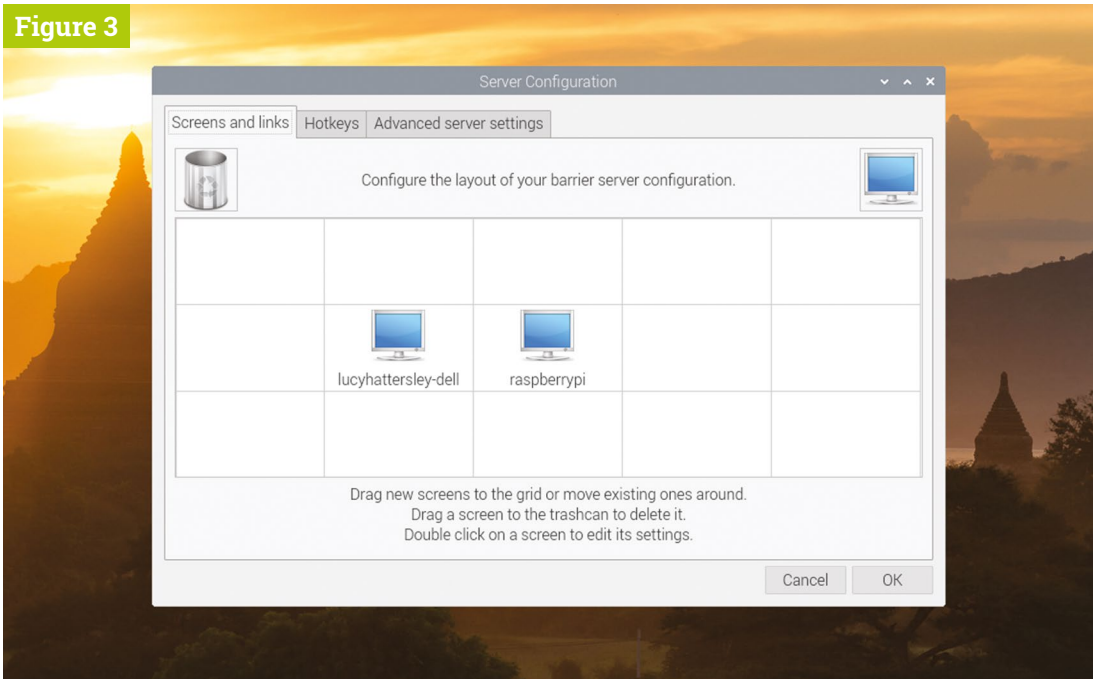


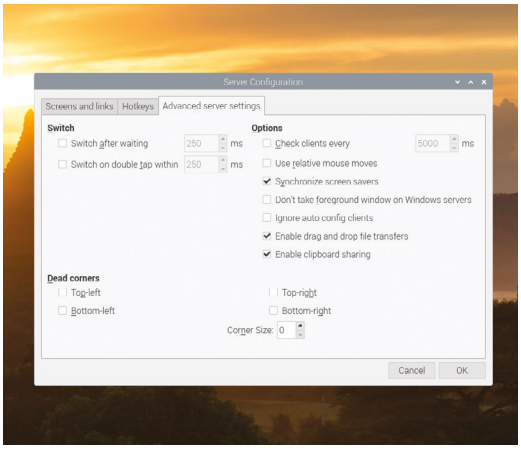
Figure 3 The Server Configuration window is used to set up the position of both monitors correctly (so the mouse flows from one screen to another)

Ensure that Server is ticked and make a note of the IP address (on our Raspberry Pi this is 192.168.0.41; on your network the IP address may be different).

05 Set up the client

Now open the Barrier app on your client machine (in our case a Windows 10 computer).

Deselect the Server checkbox and select Client instead. Enter the IP address for Raspberry Pi into the Server IP text box. On our setup, Raspberry Pi is located at 192.168.0.41 (see **Figure 2**). Your IP address may vary – it is displayed in Barrier on Raspberry Pi. Make a note of the screen name for your client computer. Ours is 'lucyhattersley-dell'.



The Advanced server settings window enables you to fine-tune interaction and share clipboards between computers

> "Now you can use Raspberry Pi's keyboard to enter text into your Windows computer"

06 Configure server

Head back to Barrier on Raspberry Pi and click Configure Server. This will open the Server Configuration window (**Figure 3**).

Top Tip

Port opening

Barrier server uses TCP port 24800. The clients use a variety of ports that can be identified using `netstat -an` and looking for the IP address of the server. If Windows Firewall has blocked the port, Barrier may not run. Take a look at this Barrier document for help: **magpi.cc/barrierfirewall**

remote.it

Can’t port forward?

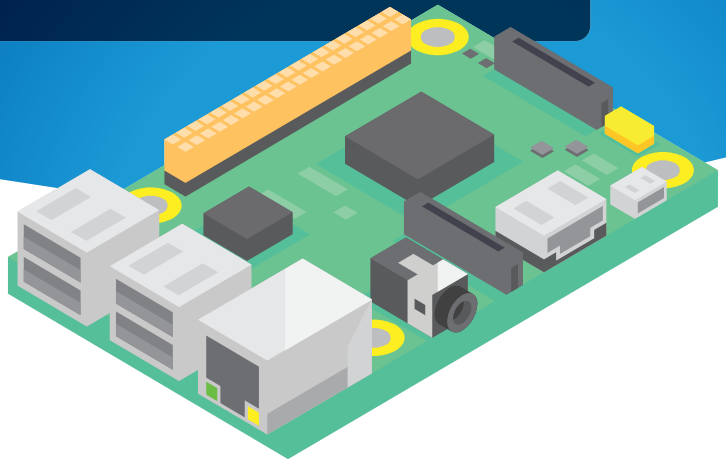
No global IP address? No admin access to router settings? On a mobile network? Device behind multiple subnets? Worried about leaving open ports? Worried about exposing your global IP?

No Problem.

Use your localhost address to create secure tunnels to your remote devices.

Install in an instant on Raspian Buster

```
> sudo apt update
> sudo apt install remoteit
```

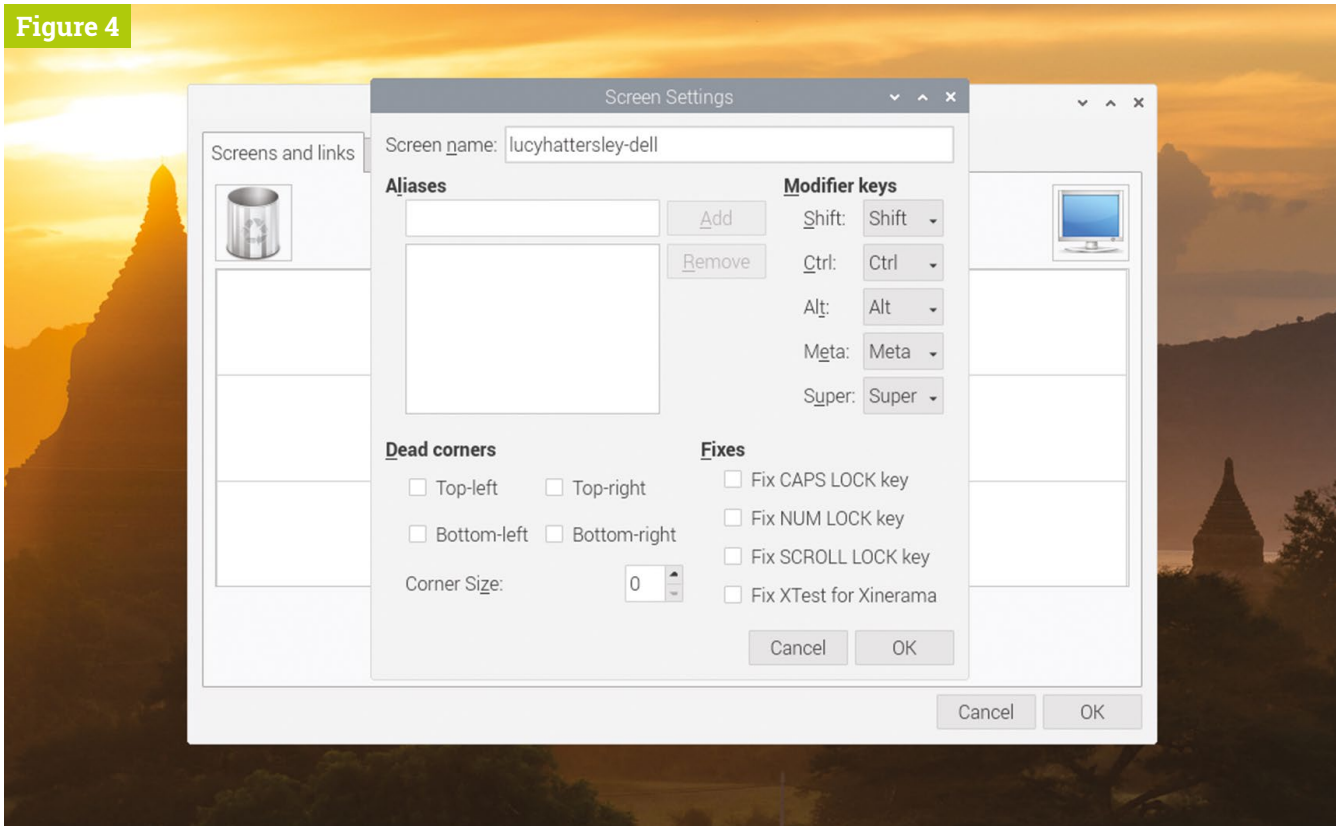


Port forwardless access to remote devices

Easily access your Raspberry Pi from anywhere without a public IP address or port forwarding. Supports SSH, VNC, web servers, Minecraft, and more. Free for personal use. Business licenses available. remote.it is an approved package for Raspian Buster. Install in moments!

Download remote.it at **remote.it/rpi**

Figure 4



▲ **Figure 4** It is important to use the correct screen name of your computer in the Screen Settings window

Now you need to add, and position, the client computer using the Screen Name. Drag the monitor icon from the top-right of the Server Configuration window and place it next to the monitor icon marked 'raspberrypi'. Ours is positioned to the left of our Raspberry Pi (to match the monitor layout).

The monitor icon will be called 'Unnamed'. Double-click it to open the Screen Settings window and change the Screen Name to match the client computer – for example in our case, 'lucyhattersley-dell' (**Figure 4**). Click OK to close the window.

07 Start it up

Click Start on Barrier on your server computer (in our case, Raspberry Pi). Wait until the lock icon in the bottom left of the Barrier window displays 'Barrier is running'.

Now click Start on Barrier on the client computer (our Windows PC). Again, wait for the 'Barrier is running' message.

Top Tip

Set up VNC

When you turn off the client computer, it will not reconnect to Barrier automatically. It's a good idea to set up VNC screen sharing on both computers, so you can remotely control the computer. This isn't as elegant a method as using Barrier, but it can be used to click Start in the Barrier app on the client computer. **magpi.cc/vnc**

08 Mouse and keyboard

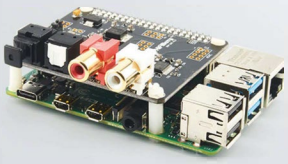
Move the mouse pointer on your Raspberry Pi over to the left of display and it will flow to the client machine. Now you can use Raspberry Pi's mouse and keyboard to control the Windows interface. Open an app and click on a text box (such as a web browser and URL box). Now you can use Raspberry Pi's keyboard to enter text into your Windows computer. It's now safe to disconnect any keyboard and mouse from the client computer. You'll be able to use the server keyboard and mouse moving forward.

Universal & Unique

UniPiCase

for Raspberry Pi 4

Perfect option for your DIY or high-volume commercial projects



Fits all kinds of HATs
Ideal PoE HAT case

Use with HAT or Pi alone
Simple and professional

Rapid, tool-free assembly
GPIO cable pass-through

Two wall mount options
Well ventilated

www.UniPiCase.com

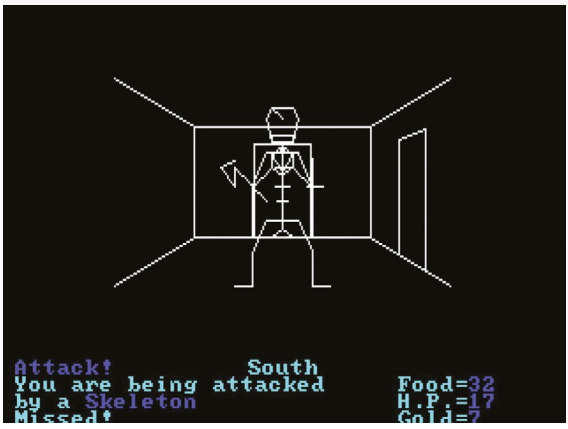
Designed and manufactured in North America

Source Code

Create a turn-based combat system

Learn how to create the turn-based combat system found in games like Pokémon, Final Fantasy, and Undertale

AUTHOR
RIK CROSS



It may look crude, but Richard Garriott's *Akalabeth* laid the groundwork for *Ultima* and was one of the earliest CRPGs.

In the late 1970s, high school student Richard Garriott made a little game called *Akalabeth*. Programmed in Applesoft BASIC, it helped set the template for the role-playing genre on computers. Even today, turn-based combat is still a common sight in games, with the recent *Pokémon Sword and Shield* revolving around a battle system which sees opponents take turns to plan and execute attacks or defensive moves.

The turn-based combat system in this article is text-only, and works by allowing players to choose to defend against or attack their opponent in turn. The battle ends when only one player has some health remaining.

Each `Player` taking part in the battle is added to the static `players` list as it's created. Players have a `name`, a `health` value (initially set to `100`), and a Boolean `defending` value (initially set to `False`) to indicate whether a player is using their shield. Players also have an `inputmethod` attribute, which is the function used for getting player input for making various choices in the game. This function is passed to the object when created, and means that we can have human players that give their input through the keyboard, as well as computer players that make choices (in our case simply by making a random choice between the available options).

A base `Action` class specifies an action `owner` and an `opponent`, as well as an `execute()` method which has no effect on the game. Subclasses of the base class override this `execute()` method to specify the effect the action has on the `owner` and/or the `opponent` of the action. As a basic example, two actions have been created: `Defend`, which sets the owner's `defending` attribute to `True`, and `Attack`, which sets the owner's `defending` attribute to `False`, and lowers the opponent's `health` by a random amount depending on whether or not they are `defending`.

"Even today, turn-based combat is still a common sight in games"

Players take turns to choose a single action to perform in the battle, starting with the human 'Hero' player. The `choose_action()` method is used to decide what to do next (in this case either attack or defend), as well as an opponent if the player has chosen to attack. A player can only be selected as an opponent if they have a `health` value greater than 0, and are therefore still in the game. This `choose_action()` method returns an `Action`, which is then executed using its `execute()` method. A few `time.sleep()` commands have also been thrown in here to ramp up the suspense!

After each player has had their turn, a check is done to make sure that at least two players still have a `health` value greater than 0, and therefore that the battle can continue. If so, the static `get_next_player()` method finds the next player still in the game to take their turn in the battle, otherwise the game ends and the winner is announced.

Our example battle can be easily extended in lots of interesting ways. The AI for choosing an action could also be made more sophisticated, by looking at opponents' `health` or `defending` attributes before choosing an action. You could also give each action a 'cost', and give players a number of action 'points' per turn. Chosen actions would be added to a list, until all of the points have been used. These actions would then be executed one after the other, before moving on to the next player's turn.

Wireframe

This tutorial first appeared in Wireframe, our sister magazine that lifts the lid on the world of video games. Every issue includes tutorials and in-depth interviews, along with news and reviews of the latest indie and triple-A games.

To find out more, visit their website at **wfmag.cc**.

Check out their subscription offers at **wfmag.cc/subscribe**.

Turn-based combat in Python

Here's Rik's code snippet, which creates a simple turn-based combat sequence in Python. You'll need to install Pygame Zero – see **wfmag.cc/pgzero**. To run the code, **cd** to its directory, then **pgzrun turn.py**


```
import random, time

class Action():
    def __init__(self, owner, opponent):
        self.owner = owner
        self.opponent = opponent

    def execute(self):
        pass

class Attack(Action):
    def __init__(self, owner, opponent):
        super().__init__(owner, opponent)

    def execute(self):
        self.owner.defending = False
        if self.opponent.defending:
            hit = random.randrange(10,20)
        else:
            hit = random.randrange(20,40)
        self.opponent.health -= hit
        print('{} is hit! (-{})'.format(self.opponent.name, hit))

class Defend(Action):
    def __init__(self, owner, opponent):
        super().__init__(owner, opponent)

    def execute(self):
        self.owner.defending = True
        print(self.owner.name, 'is defending!')

class Player():
    players = []

    def __init__(self, name, inputmethod):
        self.name = name
        self.inputmethod = inputmethod
        self.health = 100
        self.defending = False
        self.players.append(self)

    def __str__(self):
        description = "Player: {}\n{}\nHealth = {}\
nDefending = {}\n".format(
            self.name,
            '-' * (8 + len(self.name)),
            self.health,
            self.defending
        )
        return(description)

    @classmethod
    def get_next_player(cls, p):
        # get the next player still in the game
        current_index = cls.players.index(p)
        current_index = (current_index + 1) % len(cls.players)
        while cls.players[current_index].health < 1:
            current_index = (current_index + 1) % len(cls.players)
        return cls.players[current_index]

    def choose_action(self):
        print(self.name, ': [a]ttack or [d]efend?')
        action_choice = self.inputmethod(['a', 'd'])
        if action_choice == 'a':
            print('Choose an opponent')
            # build up a list of possible opponents
            opponent_list = []
            for p in self.players:
                if p != self and p.health > 0:
                    print('[{}] {}'.format(self.players.
index(p), p.name))
                    opponent_list.append(str(self.players.
index(p)))
            # use input to get the opponent of player's action
            opponent = self.players[int(self.inputmethod(opponent_
list))]
            return Attack(self, opponent)
        else:
            return Defend(self, None)

def human_input(choices):
    choice = input()
    while choice not in choices:
        print('Try again!')
        choice = input()
    return choice

def computer_input(choices):
    time.sleep(2)
    choice = random.choice(choices)
    print(choice)
    return choice

# add 2 players to the battle, with their own input method
hero = Player('The Hero', human_input)
enemy = Player('The Enemy', computer_input)

# the hero has the first turn
current_player = Player.players[0]
playing = True

# game loop
while playing:

    # print all players with health remaining
    for p in Player.players:
        if p.health > 0:
            print(p, end='\n\n')

    # current player's action executed
    action = current_player.choose_action()
    time.sleep(2)
    action.execute()

    # continue only if more than 1 player with health remaining
    if len([p for p in Player.players if p.health > 0]) > 1:
        current_player = Player.get_next_player(current_player)
        time.sleep(2)
    else:
        playing = False

for p in Player.players:
    if p.health > 0:
        print('**', p.name, 'wins!')
```

Use an Inky wHAT display with Raspberry Pi

Make a great impression with a fancy e-ink name badge or custom display that subtly shows off your tech skills. By **Rosie Hattersley**

MAKER
Rosie Hattersley

Rosie writes about how to use tech and first learnt to code on a Commodore 64.

@RosieHattersley

One of the fun things about Raspberry Pi is how easy it is to use to show off your personality, whether that's with a custom case you've designed, a complete DIY project, or a display that uses code to send a personalised message.

Here we explore the Inky Python library and look at how to use it with Pimoroni's Inky wHAT (**magpi.cc/inkywhat**). This wide version of the Inky pHAT can be used to show off a distinctive image, text, a meaningful slogan, and to tell people at a Raspberry Jam who you are.

Inky wHAT's e-ink display operates in the same way as a Kindle e-book reader. It's easy to read, even in bright sunshine, and has no eye-fatiguing bright white backlight. Better yet, it draws very little power and therefore is ideal for battery-powered use. Once the desired display data has been sent to Inky wHAT, it can even be disconnected from your Raspberry Pi. Your message will stay on its screen.

> "It draws very little power and therefore is ideal for battery-powered use"

You'll Need

- Inky wHAT
- Inky Python library

01 Attach Inky wHAT to Raspberry Pi

Carefully line up your Inky wHAT over the GPIO header pins on Raspberry Pi (use the header booster that sits between the wHAT and the board if yours is a full-sized Raspberry Pi model) and press it into place. Connect Raspberry Pi's mouse, keyboard, screen, and power on. As with using any new hardware, start by updating your Raspberry Pi. We advise using a fresh installation of Raspberry OS, which will suggest updating itself when it first boots up. Follow the prompts to check for and download any updates.

02 Turn on SPI and I²C

Inky wHAT needs both SPI (Serial Peripheral Interface) and I²C (Inter-Integrated Circuit) enabled to work. Both can be found in Raspberry Pi Configuration. Click on the menu icon and choose Preferences > Raspberry Pi Configuration. Click on the Interfaces tab and set both SPI and I²C to Enabled. Click OK.

03 Install Inky wHAT software

With the Inky wHAT hardware attached, we'll install the software. Open a Terminal (click the black rectangle icon in the top bar) and enter:

```
curl https://get.pimoroni.com/inky | bash
```

Press **Y** in response to 'Do you wish to do a full install?', then press **ENTER**.

04 Clean and reset

During the wHAT installation, some example programs will have been copied to Raspberry OS. Open a new Terminal window and type:

```
cd /home/pi/Pimoroni/inky/examples
ls
```

...to locate them. Let's start by 'cleaning' your Inky wHAT's screen (it ships with the logo displayed.) While in the **examples** directory, enter:

```
python clean.py --type what --colour yellow
```

(If you have the red wHAT, then enter `--colour red` instead, and for all following examples.)

▲ Here we see Inky wHAT displaying a name badge with The MagPi as our name. You can change it to anything you want

The Inky wHAT screen flashes as it refreshes and then cycles through yellow, black, and white. In the end you will have a clear screen. To bring back the fantastic Inky wHAT logo screen, type

```
python logo.py --type what --colour yellow
```

05 Quotes

Inside the **examples** directory is another folder marked **what**. There are more examples in here, including a selection of quotes. Enter:

```
cd what
```

…to change directory. Then run the example:

```
quotes-what.py -colour yellow
```

For these programs, you don't need to specify the type (because you're in the **what** folder).

The quotes are pulled from an editable selection at Wikiquote (**magpi.cc/wikiquotepeople**).

06 Change quote

You can edit the **quotes-what.py** code to pick quotes from any of your favourite people. Enter:

```
mousepad quotes-what.py
```

…to open the file in Text Editor. And replace the list of names with your favoured person:

```
people = [
      "Douglas Adams"
]
```

hello_inky.py

DOWNLOAD THE FULL CODE: magpi.cc/github

> Language: **Python 3**

```
001. from inky import InkyWHAT
002. from PIL import Image, ImageFont, ImageDraw
003.
004. inky_display = InkyWHAT("yellow") #change to red for red Inky wHAT
005. inky_display.set_border(inky_display.WHITE)
006.
007. img = Image.new("P", (inky_display.WIDTH, inky_display.HEIGHT))
008. draw = ImageDraw.Draw(img)
009.
010. from font_fredoka_one import FredokaOne
011.
012. font = ImageFont.truetype(FredokaOne, 36)
013.
014. message = "Hello World!"
015. w, h = font.getsize(message)
016. x = (inky_display.WIDTH / 2) - (w / 2)
017. y = (inky_display.HEIGHT / 2) - (h / 2)
018.
019. draw.text((x, y), message, inky_display.RED, font)
020. inky_display.set_image(img)
021. inky_display.show()
```

magpi_inky.py

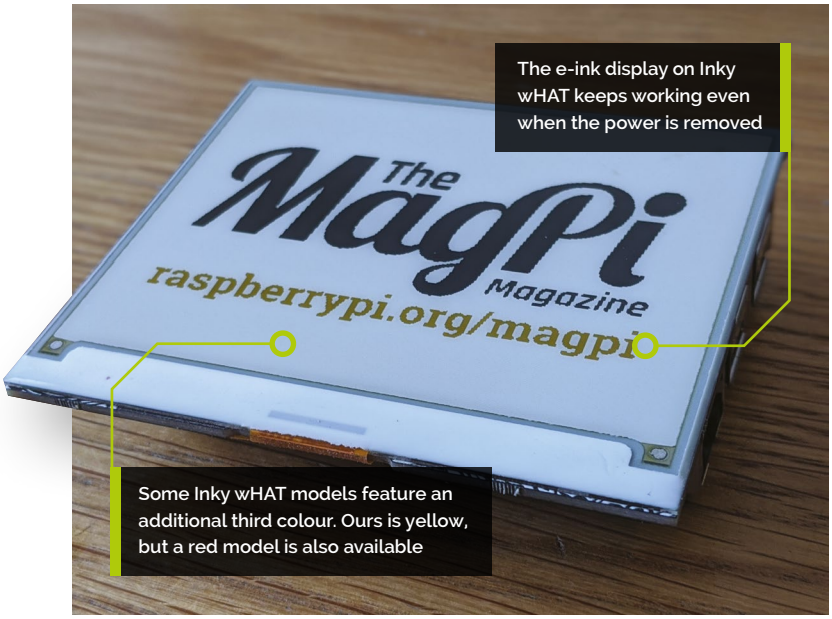
> Language: **Python 3**

```
001. from PIL import Image
002. from inky import InkyWHAT
003.
004. inky_display=InkyWHAT("yellow")
005. inky_display.set_border(inky_display.WHITE)
006.
007. img = Image.open("MagPi-Logo.png")
008. inky_display.set_image(img)
009. inky_display.show()
```

inky-palette.gpl

> Language: **ASCII**

```
001. GIMP Palette
002. Name: Inky pHAT
003. Columns: 1
004. #
005. 255 255 255      White
006.   0   0   0      Black
007. 255   255   0    Yellow
```



Top Tip

Shut down

Because the Inky wHAT can display without power, it's tempting to remove the HAT from Raspberry Pi. But make sure you shut down first: `sudo shutdown -h now`

Use File > Save As and save the file as **quotes-adams.py**. Now, in the Terminal, enter:

```
python quotes-adams.py --type yellow
```

…and you'll see a quote from the celebrated author.

07 Make a name badge

Now we've explored the options Pimoroni provides, let's tell Inky wHAT to display something we specify. Head back to the examples directory, using `cd ..` or `cd /home/pi/Pimoroni/examples`, and type:

```
python name-badge.py --type what --colour yellow --name "The MagPi"
```

Replace 'The MagPi' with your own name (or any name you like). After a few seconds, Inky wHAT's display will mimic a conference delegate's name badge, complete with the name you provided.

08 Hello Python

What about coding your own text and images from scratch? Fortunately, it's possible to code your own messages using Python. Let's start with the traditional Hello World program. Open the Thonny IDE (Menu > Programming > Thonny Python IDE) and save the file as **hello_inky.py** (we saved ours in the **Pimoroni/inky** directory).

Now enter the code from **hello_inky.py**. Click Run to display Hello World on the screen.

> "It can only handle 400×300-pixel PNG files that use three-channel indexed colour"

09 Show off your own images

You can tell Inky wHAT to display any image you wish, but it can only handle 400×300-pixel PNG files that use three-channel indexed colour. Although specific, it doesn't mean you can't display most images, just that you will need to convert them first. The GIMP image editor is ideal for this. To install GIMP, type:

```
sudo apt update
sudo apt install gimp
```

You will also need the three-channel indexed palette. Either use Text Editor to create the **inky-palette.gpl** file and save it with your code files (we've put ours in **/home/pi/Pimoroni/inky**), or download it from Pimoroni's Inky GitHub repo – in a Terminal, enter:

```
git clone https://github.com/pimoroni/inky
```

You'll find the colour palette at **inky/tools/inky-palette.gpl**. It's configured for the Inky wHAT red version, so change the third line from 255, 0, 0 to 255, 255, 0 if you have a yellow Inky wHAT.

10 Import the palette

Once installed, type `gimp` in the Terminal to run GIMP, or open it from Menu > Graphics > GNU Image Manipulation program. Choose Windows > Dockable Dialogs > Palettes. The Palettes dialog appears in the top-right of the screen. Right-click it and choose Import Palette. Choose Palette File and click the Select Palette File icon next to it. Use the window to select the **inky-palette.gpl** file. Click Import.

11 Convert an image

You can use **inky-palette.gpl** to convert an RGB image to the three channels. For best results you can create your own image using GIMP (Choose File > New and set the width to 400 and the height to 300). And then draw an image using the tools and stick to the three colours that match your Inky wHAT model (white, black, and yellow/red). If your artistic skills aren't feeling up to much, we've included a **MagPi-Logo.png** file along with the code files on our GitHub (**magpi.cc/github**). It's already sized at 400×300 and features the three colours.

Open the image (File > Open) and click Convert to the colour profile (this isn't related to our three channels). Now choose Image > Mode > Indexed and select Use Custom Palette. Click the palette graphic (typically it will be marked Web) and choose inky-palette from the list. Click Convert.

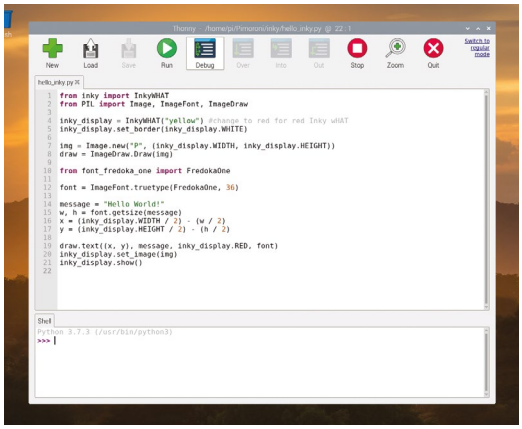
Now choose File > Export As and save the file to the same directory as your code (we saved it to **/home/pi/Pimoroni/inky**).

12 Image code

The code to display an image on the Inky wHAT is much shorter, and easier to understand, than the code for text.

Enter the code from **magpi_inky.py** and make sure that the file name and path on line 7 (**magpi_logo.png**) match the file you saved in the previous step. Run the program to view the image on your Inky wHAT:

```
python magpi_inky.py
```



▲ Use Python code to add your own text and images to Inky wHAT and incorporate it into your own projects

Part 04

Rotary encoders: Raise a Glitch Storm

Control a storm of sound with rotary encoders

MAKER

Mike Cook

Veteran magazine author from the old days, writer of the Body Build series, plus co-author of *Raspberry Pi for Dummies*, *Raspberry Pi Projects*, and *Raspberry Pi Projects for Dummies*.

magpi.cc/mikecook

A Glitch Storm is an explosive torrent of musical rhythms and sound, all generated from a single line of code. In theory, you can't do this with a Raspberry Pi running Python – this month, we throw theory out the window and show you how.

01 What is a Glitch Storm

A Glitch Storm is a user-influenceable version of bytebeat music. We love definitions like that here at the Bakery: something you have never heard of is simple a development of something else you have never heard of. Bytebeat music was at the heart of the old Commodore 64 demo scene, a competition to see who could produce the most impressive graphs and music in a very limited number of bytes. This was revived/rediscovered and christened by Viznut, aka Ville-Matias Heikkilä, in 2011. And then JC Ureña of the 'spherical sound society' converted the concept into the interactive Glitch Storm.

02 So what is it?

Most random music generators work on the level of notes; that is, notes are chosen one at a time and then played, like our Fractal Music project in *The MagPi* #66 (**magpi.cc/66**). However, with bytebeat music, an algorithm generates the actual samples levels that make up the sound. This algorithm performs bitwise operations on a tick variable that increments with each sample. Depending on the algorithm used, this may or may not produce something musically interesting. Often, the samples produced exhibit a fractal structure, which is itself similar on many levels, thus providing both the notes and structure.

> "The algorithm contains variables that a user can change in real-time while the sample is playing"

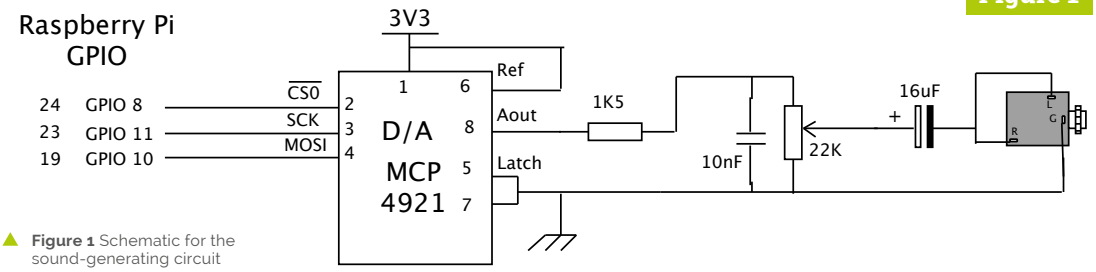
03 Enter the 'Glitch Storm'

With a Glitch Storm, three user-controlled variables – a, b, and c – can be added to this algorithm, allowing the results to be fine-tuned. In the 'Algorithms' box, you can see that the bytebeat algorithms simply run; they all repeat after a certain time, but this time can be long, in the order of hours for some. A Glitch Storm algorithm, on the other hand, contains variables that a user can change in real-time while the sample is playing. This exactly what we can

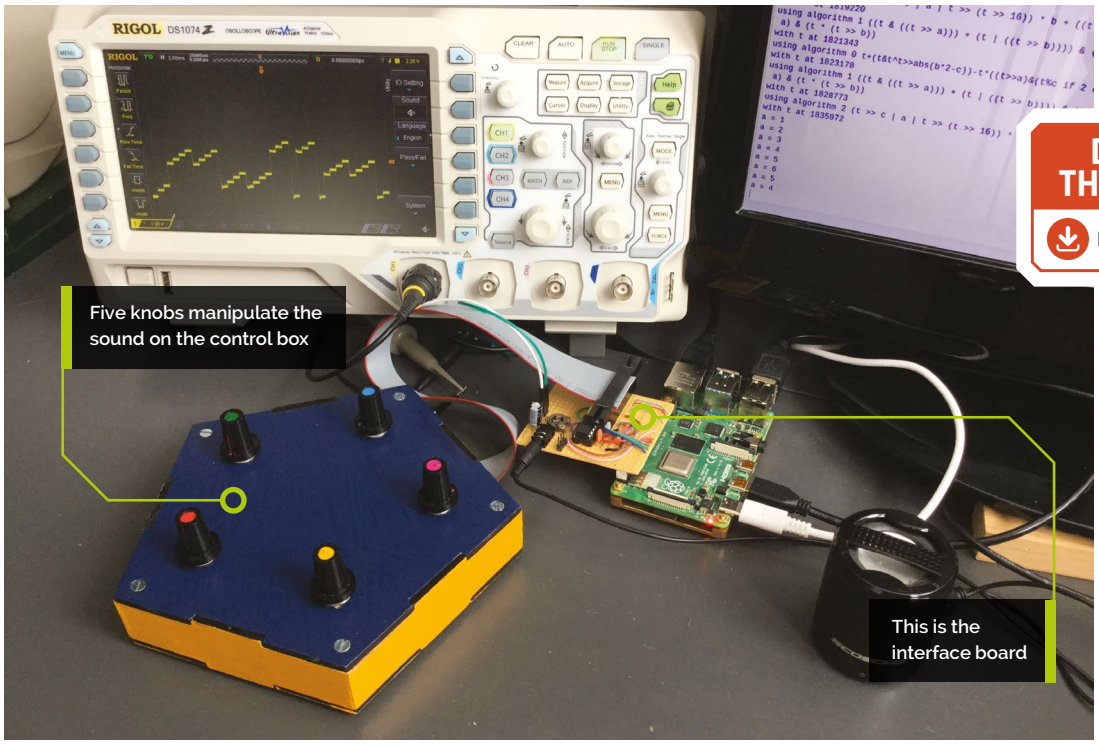
You'll Need

- MCP4921 D/A converter **magpi.cc/mcp4921**
- 5 × KY-040 rotary switches, with nuts **magpi.cc/rotary**
- 3.5 mm stereo jack socket **magpi.cc/stereopcb**

Figure 1



▲ **Figure 1** Schematic for the sound-generating circuit



DOWNLOAD THE FULL CODE:
magpi.cc/pibakery

Algorithms

After each sample is calculated, t is incremented.

Bytebeat

```
Sample = t  # this produces a simple sawtooth wave
Sample = t & t >> 8 # a minimal Sierpinski harmony
Sample = t * (42 & t >> 10) # "the 42 melody"
```

Glitch Storm

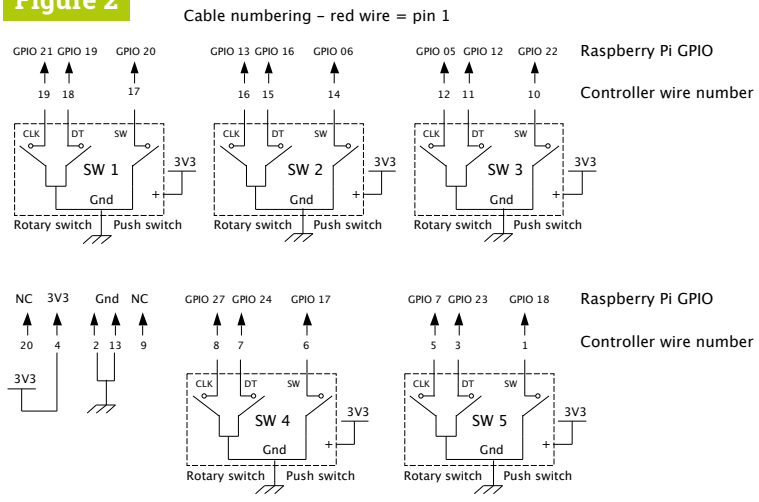
```
Sample = t * (t >> a) & (b * t >> 7) & (8 * t >> c)
```

do with rotary encoders, without having the algorithm interrupted by checking the state of them all the time.

04 What hardware?

In order to produce music like this on the Raspberry Pi, we need some extra hardware to generate the sound samples, and also a bunch of rotary encoders to control things. The samples are produced by using a 12-bit A/D converter connected to one of the SPI ports. The schematic of this is shown in **Figure 1**. The clock rate for the transfer of data to this can be controlled and provides a simple way of controlling, to some extent, the sample rate of the sound. **Figure 2** shows the wiring diagram of the five rotary encoders we used.

Figure 2

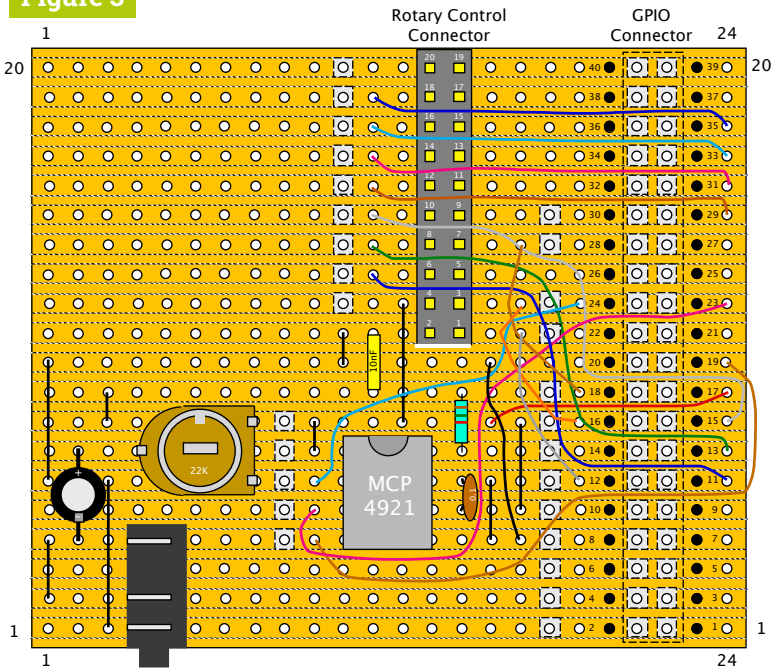


▲ **Figure 2** Schematic for the control box

05 Making the hardware

The hardware comes as two parts: the D/A converter and associated audio components. These are built on a board that hangs off Raspberry Pi's GPIO pins. Also on this board is a socket that carries the wires to the control box. We used an IDC (insulation displacement connector) to connect between the board and the box, as we wanted the D/A connection wires to be as short as possible because they carry a high frequency signal. We used a pentagonal box just for fun, with a control in each corner, but the box shape is not important here.

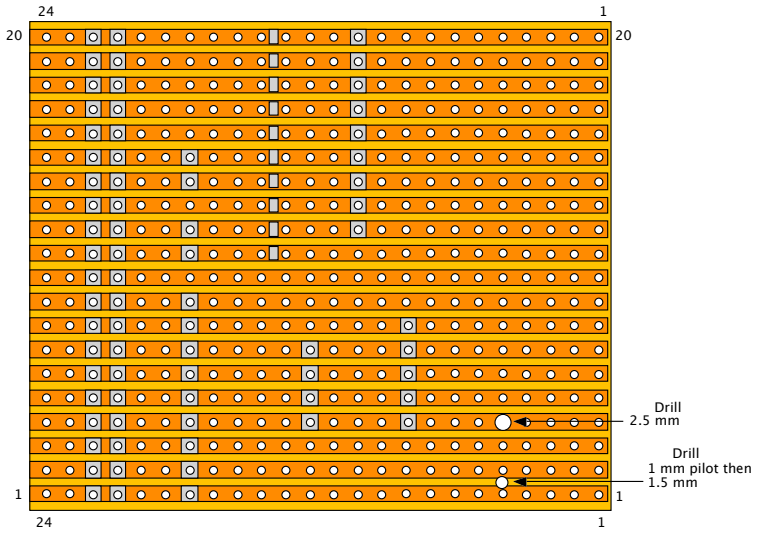
Figure 3



▲ **Figure 3** Front physical layout of the interface board

> The knobs control the user variables as well as the sample rate and what algorithm to use

Figure 4



▲ **Figure 4** Rear physical layout of the interface board

06 Construction

The board is built on a 20-row by 24-hole piece of stripboard. **Figure 3** and **Figure 4** show the physical layout for the front and back of the board. The hole number 5 on row 4 is enlarged to 2.5 mm and a new hole is drilled between rows 1 and 2 to accommodate the audio jack socket. A 40-way surface-mount socket connector is soldered to the back of the board, and a 20-way socket is soldered to the front. You could miss this out and wire the 20-way ribbon cable direct to the holes in these positions if you want to economise.

07 Further construction notes

Note: as always, the physical layout diagram shows where the wires go, not necessarily the route they will take. Here, we don't want wires crossing the 20-way connector, so the upper four wires use 30 AWG Kynar wire to pop under the connector and out through a track hole, without soldering, on the other side. When putting the 20-way IDC pin connector on the ribbon cable, make sure the red end connector wire is connected to the pin next to the downward-pointing triangle on the pin connector. **Figure 5** shows a photograph of the control box wiring.

08 Testing the D/A

The **live_byte_beat.py** listing on GitHub is a minimal program for trying out a bytebeat algorithm. It will play until stopped by pressing **CTRL+C**. The variable `v` holds the value of the sample, which is then transferred to the D/A over SPI in two bytes. The format of these two bytes is shown in **Figure 6**, along with how we have to manipulate `v` to achieve an 8-bit or 12-bit sample output. Note that all algorithms were designed for an 8-bit sample size, and using 12 bits is a free bonus here: it does sound radically different, and not always in a good way.

09 The main software

The main software for this project is on our GitHub page (**magpi.cc/pibakery**), and contains 24 Pythonised algorithms. The knobs control the user variables as well as the sample rate and what algorithm to use. You can add extra algorithms, but if you are searching online for them, you will

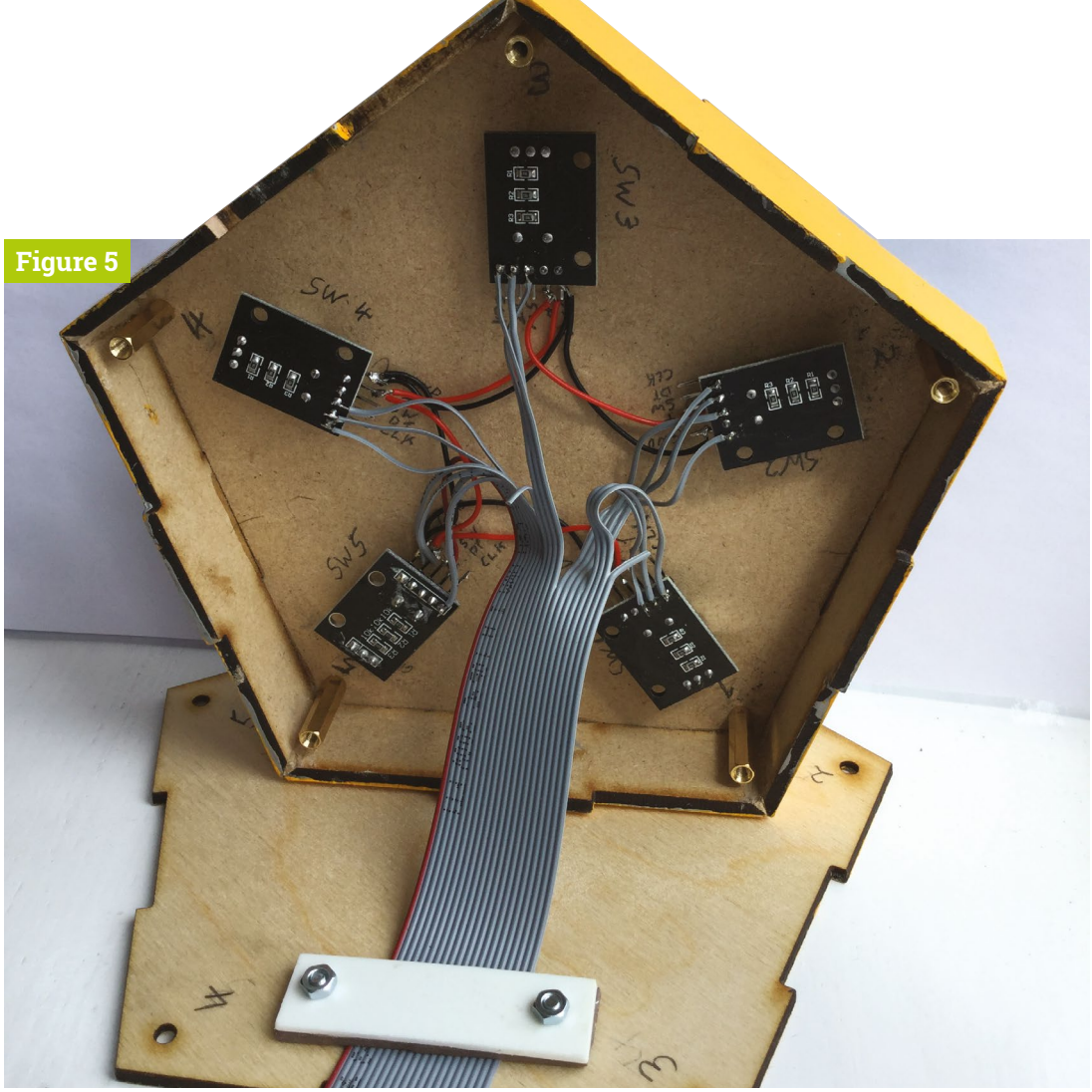


Figure 5

Top Tip

Ribbon cable connector problems

If you get red wire in the wrong pin, you will have to compensate by wiring the encoders differently. A revised schematic for this is on our GitHub page.

Figure 5 Wiring of the control board

Figure 6 How to program the registers in the D/A converter

find they are written in C. There are two major differences you need to note when converting from C to Python. The first is the ternary operation which in C is a question mark, and the second is the modulus operator with a percent sign. See the notes that accompany the main code about these.

10 Why does this work?

There are a few reasons why you would not expect this to work on a Raspberry Pi in Python. The most obvious being that of the interruptions made by the operating system, regularly interrupting the flow of output samples. Well, it turns out that this is not as bad as you might fear, and the extra 'noise' this causes is at a low level and is masked by the glitchy nature of the sound. As Python is an interpreted language, it is just about fast enough to give an adequate sample rate on a Raspberry Pi 4.

Make some noise

You can now explore the wide range of algorithms for generating a Glitch Storm and interact with the sound. On our GitHub page there's a list of useful

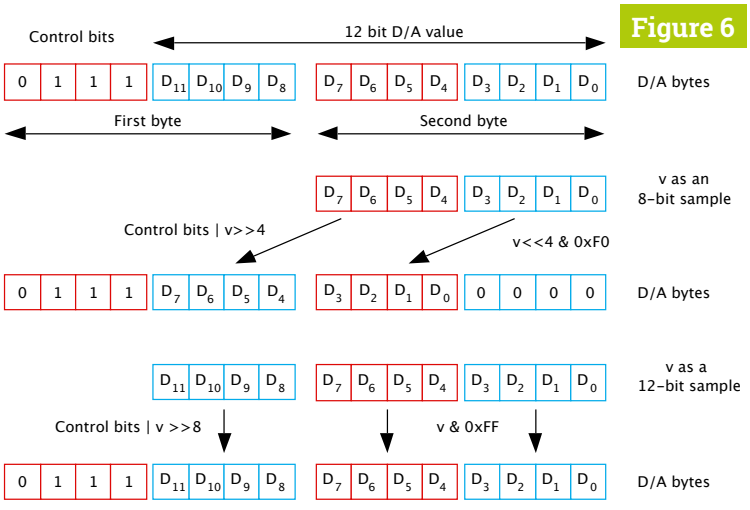


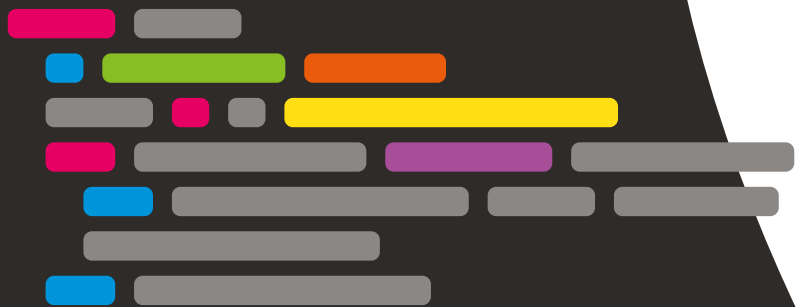
Figure 6

links allowing you to explore what others have done so far. For a sneak preview of the bytebeat type of sound, visit **magpi.cc/bytebeatdemo**; you can even add your own algorithms here. For interaction, however, there's no substitute for having your own hardware. The best settings are often found by making small adjustments and listening to the long-term effects – some algorithms surprise you about a minute or two into a sequence by changing dramatically.

LEARN Computing and ICT AT HOME

Gain a new skill or even a new career with these great resources

Are you stuck at home this summer? Maybe you're looking for a challenge, something to wipe away the boredom. Sure, you can catch up on another box set, but you could also try something new. Whether it's getting your computer to ask your name and repeat it back to you or maybe making a T-shirt quilt with embedded LEDs, the Raspberry Pi community is bursting at the seams with ideas to make your summer a bit more fun. If you find something you like, maybe you can carry on and turn it into a new skill or career? Here's some inspiration.



Learn to speak computer

It's a lot of fun telling computers what to do, but getting started can be intimidating. Here's a gentle introduction

Someone once joked that a software developer is a machine for turning coffee into computer programs. Yes, it can sometimes feel that way, but in reality coding can be a highly enjoyable, rewarding experience not unlike solving a crossword puzzle or a tricky Sudoku. The only real barrier is understanding the language the computer speaks. There are many different languages, some easier to understand than others. Here we'll look at the skills that a developer requires and how to get started without the whole thing becoming overwhelming.

Coding is the art of telling a computer what to do. We say 'art' and we mean it. Yes, it's known as 'computer science' but more often than not, the imagination and creativity that is associated with the arts can mean the difference between a working piece of code and a great piece of engineering. If you're new to coding, it would be fair of you to assume that a solid grasp of mathematics is required. Yes, it does help to know your numbers, but what is far more

Scratch comes built into Raspberry Pi OS with Desktop. It's a perfect and fun introduction to coding

Your Raspberry Pi computer can handle many different programming languages, including Scratch, Python, and JavaScript

▲ There's lots of support available for new coders, such as this club in Milton Keynes

important is logic. A programmer needs to take the task required and break it down into its component parts which can then be converted into instructions a computer can understand. We call this process computational thinking.

Where to start?

You can write code in a variety of languages. It can often be bewildering to the newcomer: Which language to use? What's the difference? In reality, most common languages are general purpose, meaning they are suited to most tasks. In the Raspberry Pi community we champion Python as a great 'first' language. This has (somewhat unfairly) given it a reputation as a less capable language than others, but it has become the first choice for new projects in both the artificial intelligence and financial business communities. There are dedicated languages for certain uses, such as R for data analysis, but most – such as Python, JavaScript, Java, and newcomers like Go – are similar in terms of capability.

Three core principles

These general-purpose languages break down any task into three possible categories: Sequential, repetitive, or conditional. Sequential means a list of instructions such as 'fill kettle', 'switch kettle on'. Repetitive means tasks that are repeated until a condition is met, so you would repeatedly watch the kettle filling up until it was full, then go on to the next instruction: 'Turn the tap off'. Finally, a conditional means 'if in one condition, do this; if in another, do that'. So if you're out of tea, go and buy teabags.

Fun, frustration, and new friends

Learning a language can be challenging and fun, but sometimes frustrating. The curve comes in two parts: learning computational thinking (breaking down a problem into our three types of task) and learning the 'syntax', which can differ significantly from language to language.

Luckily, there is a wealth of information and help available. You have a lot of choice when it comes to how you like to learn, where you want to learn, and the pace at which you will do your best. We also have a rich community of code clubs open to children and people of any age. Some cater to specific communities, such as Girlscode MK, a free club founded in Milton Keynes to help women of all ages and backgrounds get started in coding.

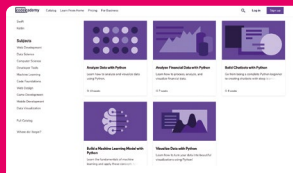
If you are looking to get support in your coding journey, check local libraries and schools to see what's going on. There are plenty of online groups as well that offer help and support.

Start learning to code

Take a Scratch course

Scratch is a remarkable drag-and-drop visual programming language from MIT. Google offers a structured course perfect for newcomers.

magpi.cc/csfirst



Codecademy

A great next step with a catalogue of courses, starting with language-agnostic computation principles and progressing to actual software development.

codecademy.com

Udemy

A massive collection of courses for all capabilities. Learn Python with zero experience, or discover the latest developments in AI.

udemy.com

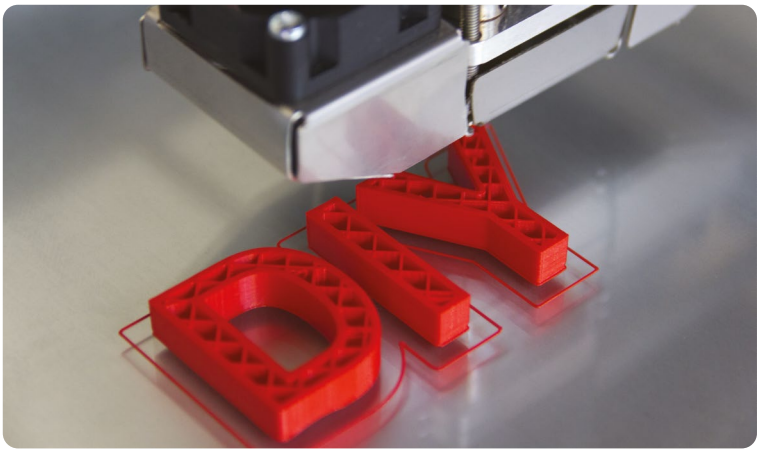
Become a maker and enhance the physical world

Making things can be one of the most rewarding pastimes, but what to do? Here are some ideas

A **'maker' has several definitions. Typically a maker is someone who creates something new by creating or following a design.** This is technology and the arts married together in the physical world. The choices are endless, as are the complexity and skills required. This is great because it means there's something for everyone in the hacking world. You can get involved with physical computing, robotics, or master your cross-stitch. Whatever appeals, you can find communities and new friends that can help inspire and encourage you to create and improve. A central tenet of the community is 'learning through failure', so there's no need to feel intimidated – we were all beginners once!

> Naturally, these old and new disciplines cross over, creating wonderful objects and experiences

Just as learning to code can present seemingly endless choices, the maker community appears to be limited only by your imagination. As an abstract term, making is just the concept of creating something new. To do that, you're probably going to need a skillset to operate the machinery required or know the techniques to fabricate something yourself. The maker community offers the support and guidance needed to acquire those skills through online forums, publications, clubs, and makerspaces – well-equipped 'super sheds' where you can gain access to expensive equipment as part of group.



▼ Learn how to do it yourself with 3D printing

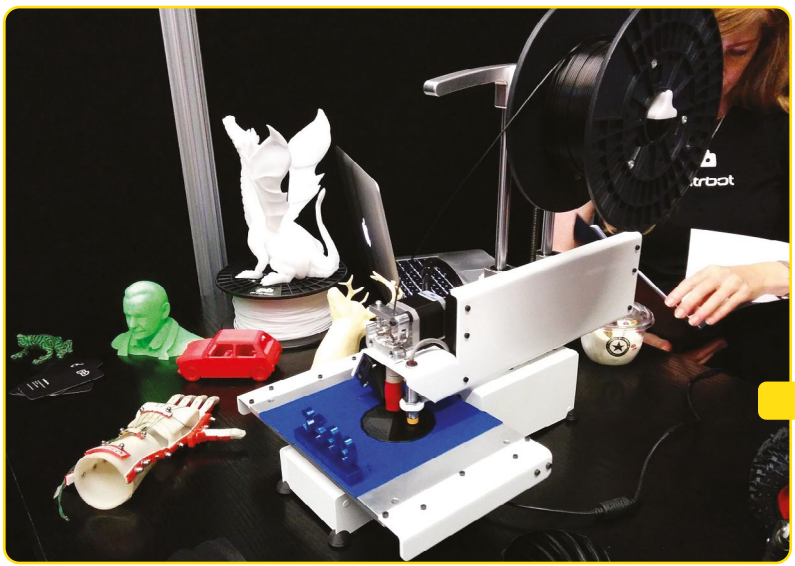
Exploding makers

The maker movement has exploded in recent years. The reasons are many. Some people have tired of electronic consumer equipment that cannot be repaired or are otherwise locked down and made obsolete too soon. Others enjoy upcycling items that would otherwise find themselves in landfill, giving all manner of electronics, furniture, and equipment a new lease of life. Although carpentry, fabrics, and blacksmithing have been with us for centuries, devices such as Raspberry Pi computers and microcontrollers have created whole new physical computing movements. Naturally, these old and new disciplines cross over, creating wonderful objects and experiences.

Bug bites

If you want to get involved in making, you can choose just how much you want to dedicate yourself to this fascinating hobby. Be warned,

▲ In makerspaces like this one, you can work with expensive equipment in a safe and supportive environment



▲ A 3D printer is an affordable and fun introduction to making

though: once bitten by the making 'bug', you may find yourself obsessed. The maker world is littered with stories of evening hobbies that resulted in new careers. Take Spencer, who, out of his own curiosity, attempted to build his own computer based on a 1980s Z80 8-bit processor. He described the resulting mess of wires on a breadboard as 'a bit of fun'. The reaction to his videos was such that just a couple of years later, the mess had become an in-production computer and he now had his own company, RC2014 Computers. This is the joy and reward making can bring.

When you're coding, it's quite hard to do any real damage (well, unless you delete everything, but you do have backups, don't you?). This is not true of making. From snagging your finger on a needle to the maker's greatest enemy, fire, making comes with some strict rules and safety concerns. Most makerspaces have a famous 'Rule Zero' which is 'Do not be on fire'. So, if you fancy building your own Tesla coils, or getting creative with a band-saw, always get help and guidance to stay safe. The community will be only too happy to help. Best of all, join your local makerspace where you can get hands-on training, support, and advice. Many makerspaces run courses on the more dangerous or technical equipment.

Further reading

If you want to read about the latest and greatest in making, try our sister publication, HackSpace magazine (**hsmag.cc**). Every month, it showcases great projects, reviews new equipment, and provides in-depth tutorials. It's a must-read for beginners and experienced professionals alike. There are also numerous meet-ups and festivals, such as Electromagnetic Field, where you can marvel at projects big and small, serious and silly.

Learn maker skills

3D printing

A perfect hobby for the Raspberry Pi enthusiast. Learn to design objects and then see them come to life in front of your eyes.

3dprintingforbeginners.com

CNC milling

The classic discipline of wood- and metalworking with a modern twist. Build and control a robotic milling machine.

magpi.cc/cncguide

Crochet

Or knitting, or cross-stitch. Whether it's to relax or create an awesome piece of cosplay, working with fabrics can be extremely rewarding.

magpi.cc/crochetguide

Laser cutters

3D printing's flatter and faster friend. Design and cut objects in minutes, ranging from puzzles to Strandbeests.

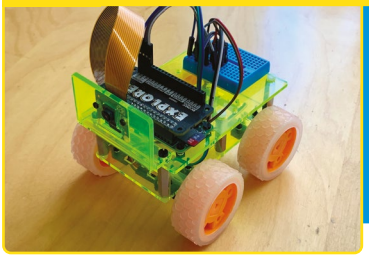
magpi.cc/lasertraining

Electronics

Raspberry Pi is perfect for physical computing projects. Learn about motors, LEDs, and sensors with the many fun kits available.

magpi.cc/learnelecbook

Robotics



Is your Raspberry Pi going places? Add a chassis, some sensors and a motor, throw in some code, and you have your own robot.

magpi.cc/learnrobotics

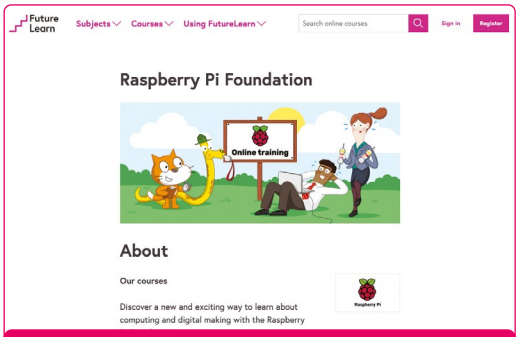
Taking it to the next level

Has your new hobby become all encompassing? Ready to take the leap? Maybe consider a professional qualification

There comes a point in any vocation, including programming and making, where you hit a natural barrier. You've learnt the basics, practised, failed, failed again, probably hit your thumb more times than you would like to admit, then succeeded. You're good enough that you think you may be able to do this for yourself, full-time. It's at this point you can make a choice: either follow your nose and learn through experience, or make time to study for a formal qualification. Your choice will probably come down to whether you're pitching your skills to a potential employer or going it alone. Either way, here are some examples of professional qualifications relevant to this feature.

> You've learnt the basics, practised, failed, then succeeded

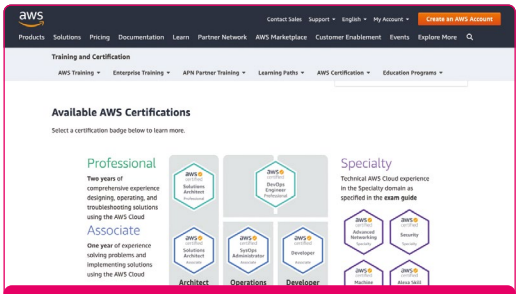
▲ Expand your skill set on a professional course



National Centre for Computing Education

Would you like to teach computing? With the new wave of computer science being taught in UK schools, teachers are finding themselves with a range of new challenges compared to the older ICT curriculum. The newly formed National Centre for Computing Education, created in partnership with the Raspberry Pi Foundation, offers a range of certified courses for teachers involved in GCSE Computer Science recognised by the British Computing Society. There's also Picademy, offering training in Raspberry Pi for teachers all over the world.

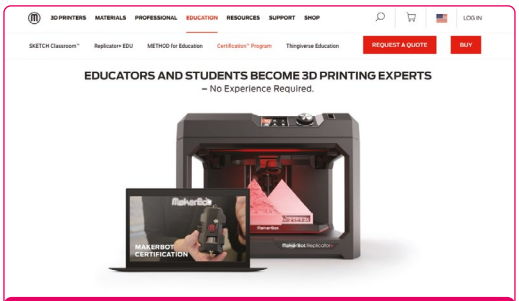
magpi.cc/futurelearn



Amazon AWS

This may sound a little specific, but Amazon Web Services (AWS) has truly been revolutionary across the IT world. Comprising a range of services such as data storage, virtual computers, search engines, and databases, all AWS offerings are 'in the cloud' and you only pay for what you use. Companies across the world have flocked to AWS and similar services. So if you're thinking about moving into IT, especially as a system administrator, it would be a great advantage to have recognised exposure to the AWS ecosystem.

magpi.cc/awscert



MakerBot Certification

Aimed at schools, MakerBot offers courses for both students and teachers in the world of 3D printing. These wonderful devices are becoming more and more commonplace in schools across the world as they can encompass many facets of design and technology. Safe and productive use of such printers can be tricky, so these courses not only provide a series of lesson plans for learning about fabrication, but also give teachers the skills to conduct those lessons safely. All course lessons are conducted online, with certificates of completion for both student and teacher.

magpi.cc/makerbotcert

British Computer Society

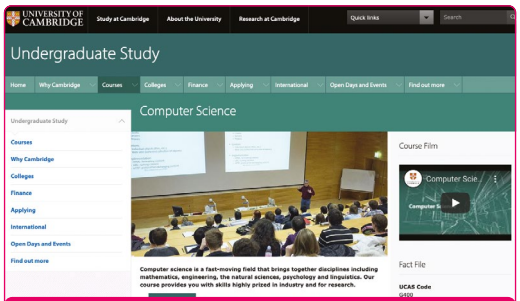
The British Computer Society (BCS) is a Chartered Institute with a 60-year history. If you're serious about learning computer science at an academic level, this may be the place for you. A wide range of certifications are available for existing IT professionals as well as newcomers under the apprenticeship programme. There's also a dedicated scholarship track for teachers. Once you've got some experience, you can also apply to become chartered and join the BCS. Finally, if you are not working directly in IT but want to show you have digital literacy, BCS offers a series of beginner's courses.

magpi.cc/bcsqualified

City & Guilds

The City & Guilds group is over 140 years old and is dedicated to training for vocational occupations. It provides certification that is recognised the world over. If you're looking to really boost your maker skills and go professional, it may well be worth your while investigating the wide range of courses on offer. Being of a 'maker' nature, these are rarely online, so you'll need to find a nearby college that's running the course you want. For example, the carpentry course gives a solid grounding in woodwork as well as good business practices.

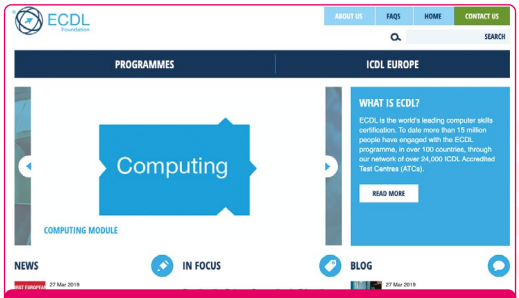
magpi.cc/cgcarpentry



Computer science degree

You may be surprised to learn that many IT professionals return to university later in their careers to study computer science, whether it be at bachelor or master's level. This is often because their personal development has stalled. Working in software development can become a routine of keeping up with technological developments and learning the latest language or framework. Computer science transcends any one language or platform and concentrates on the logic and computational thinking behind the code. If you're interested in the deeper areas of computing such as artificial intelligence or financial algorithms, a solid grounding in computer science is a must.

magpi.cc/unicamcs



ECDL qualification

The European Computer Driving Licence (also known as the International Computer Driving Licence) can give a boost to your CV if you're looking to demonstrate that you have digital skills. If you have little IT experience, this can be the boost you need. From using email to becoming a master of the word processor, the qualification is recognised as level 2 by Ofqual, making it equivalent to a GCSE. It is taught throughout the world, so check with your local college.

ecdl.org

The Open University

If you're already working or can't easily travel to study, The Open University can provide a lifeline with its remote study courses. There's a wide range of courses available and you can choose a study level to suit you. Fancy a BSc in IT and statistics? You can do it full-time or part-time, or if you prefer a Certificate of Higher Education, you can undertake a shorter course. The modular system means you can study 'as you go' and at your own pace. The Open University provides valuable education for new and mature students around the world.

magpi.cc/ouitcourses

Argon Neo & Argon Fan HAT

SPECS

CASE:
Aluminium alloy, passive cooling, sliding magnetic top

FAN:
40 mm cooling fan, status LEDs, power button

▶ Argon40 ▶ **argon40.com** ▶ £15 / $15, Argon Neo case; £10 / $10, Argon Fan

Argon40 makes some of the best cases for Raspberry Pi. We take a look at its new entry-level offering. By **Lucy Hattersley**

Argon Neo is the latest case from Argon40, an impressive case maker which came onto everybody's radar last year following a successful Kickstarter campaign to build a high-quality case. The result was Argon One, which won our group test of thermal cases for Raspberry Pi (**magpi.cc/thermaltest**).

The Argon Neo and Argon Fan HAT split the case and cooling system into two separate purchases. These fit together neatly to provide a single integrated solution.

Whereas the Argon One has a fan integrated into the lid and a breakout board to move connections to the rear, Argon Neo surrounds the Raspberry Pi board and provides access holes for the original connections.

Alongside the fan, The Argon Fan HAT provides a power button, which can perform safe shutdown, forced shutdown, and reboot Raspberry Pi.

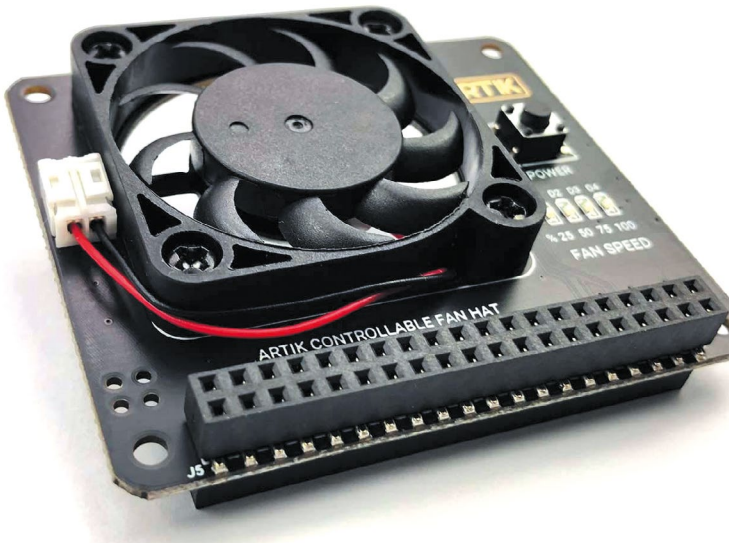
Putting it together

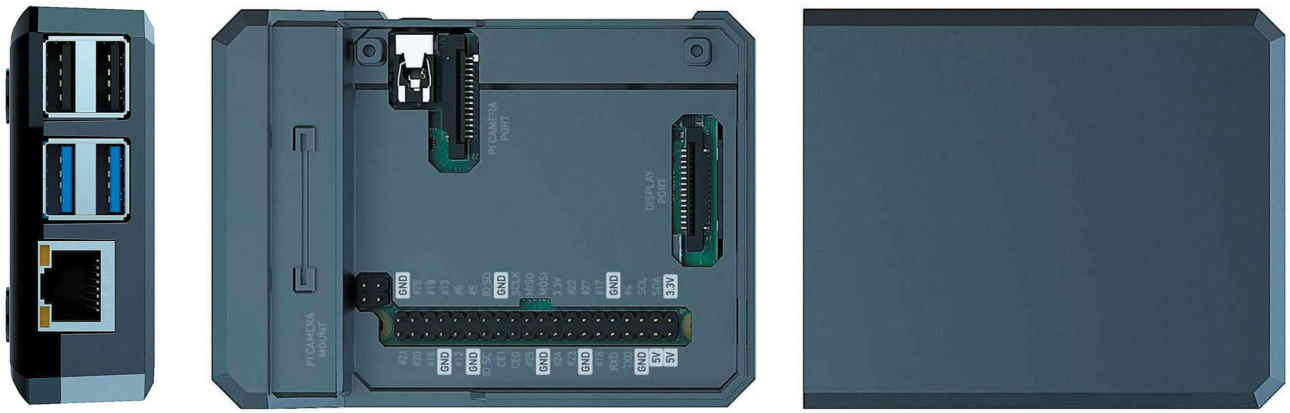
As with other cases (such as Flirc, **magpi.cc/flirc**), the metal part of the case connects to the CPU and RAM (with included thermal paste). The metal case acts as a passive heatsink.

A neat touch it shares with Argon One is the GPIO pin layout guide printed on the case. (Although with the Argon Fan HAT attached, half of the GPIO pins numbers are hidden behind the board.)

Unlike Argon One, you can also access Raspberry Pi's Display and Camera ports (the latter only

▼ The layers fit together to sandwich Raspberry Pi, and the metal case acts as a passive cooling system





Argon Neo is a stylish case with a lid that can be removed to break out the GPIO pins

The Argon Fan HAT sits on top of Raspberry Pi and provides additional cooling at temperatures above 55°C

without the fan connected). When you're done with accessing ports and GPIO pins, the metal lid snaps the whole kit shut via a magnet to form a stylish metal case.

Keeping it cool

The fan speed is related to the CPU temp, kicking in at 55°C and increasing to 100% fan speed at 65°C.

We stress-tested a Raspberry Pi 4 (4GB) CPU at full capacity for 15 minutes with both Argon Neo and Argon Fan HAT attached.

Things started at 37°C and the temperature slowly rose and hovered around 55°C, before the fan duly kicked in and held the temperature in check for the full 15 minutes, maxing out at 58°C.

We ran the same test with the Argon Neo case without the fan attached and found the temperature maxed out at 58°C – exactly the same temperature as with the fan attached.

None of this is anywhere close to threatening Raspberry Pi performance. As the CPU temperature approaches 85°C, Raspberry Pi OS throttles the CPU speed to bring down the temperature. Either with or without the fan, we didn't get anywhere near that temperature.

> The whole thing snaps together into a stylish metal case

With that in mind, we overclocked our Raspberry Pi 4 to 2.0GHz (**magpi.cc/overclock**) and reran the test. Without the fan, the temperature ran up to 82°C (and was kept in check by the passive cooling of the case alone). Still not enough for Raspberry Pi OS to begin throttling the CPU.

So, we are left wondering whether you actually need the fan?

Which to buy?

It's $15 for the Argon Neo case and $10 for the Fan HAT, which makes the package the same price as the larger Argon One.

To its credit, the Argon Neo is a more minimalist design that's slimmer and more in keeping with the design aesthetic of Raspberry Pi. It's certainly a neat solution, and is cheaper if you opt for the case without the fan.

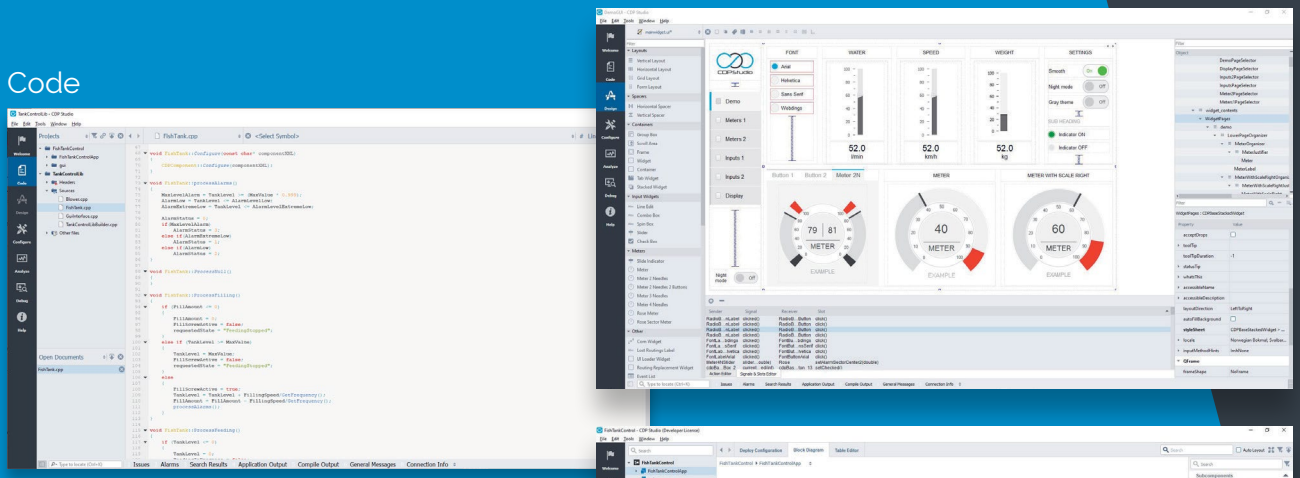
Verdict

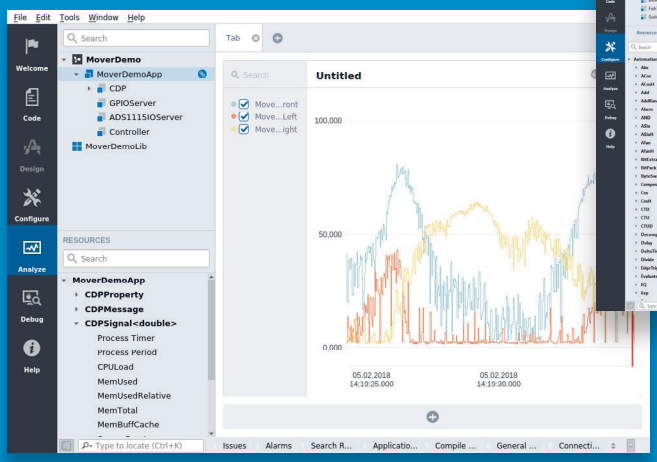
We like this an awful lot, but we like the original Argon One case a little more. However, the Neo is stylish and cheaper if you opt for the case without the fan.

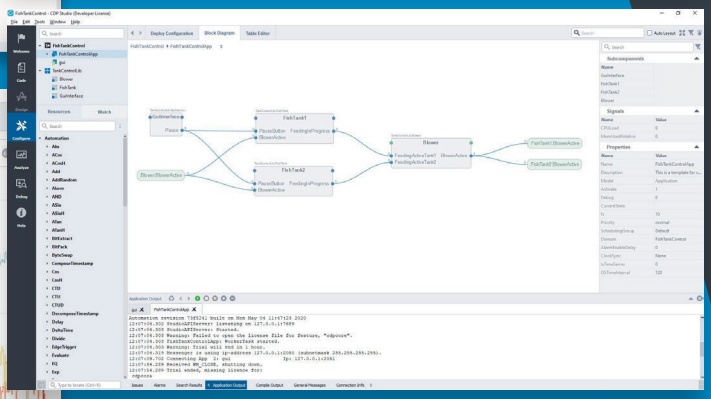
8/10

Design

Code







Configure

Analyze

CDP STUDIO: HOME PROJECTS MADE EASY

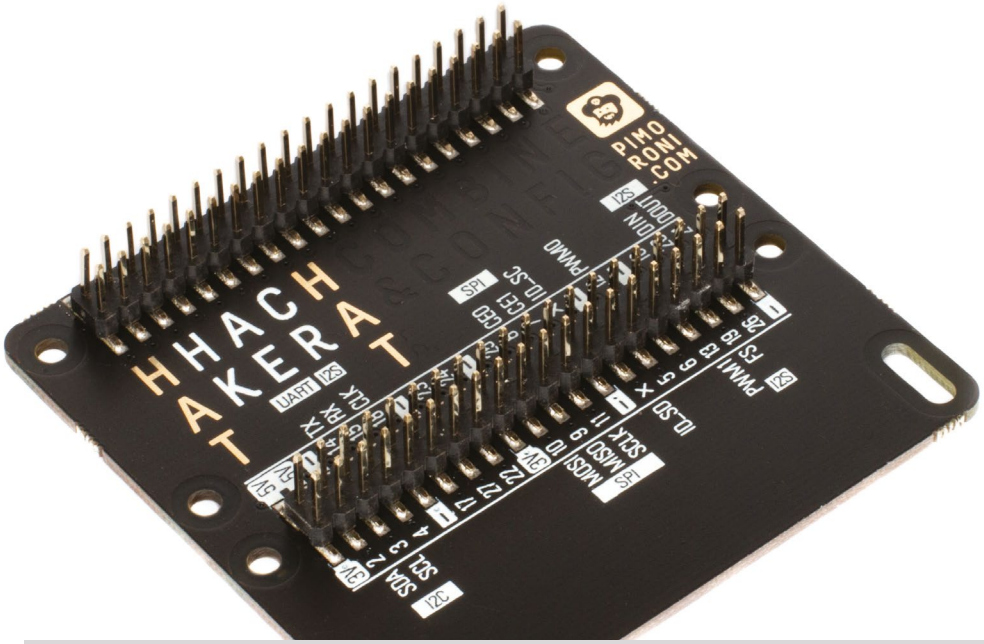
Professional control system development tool

CDP Studio is a great software development tool for your home projects. Build systems for Raspberry Pi, use C++, open source libraries, out of the box support for GPIO, I2C, MQTT, OPC UA and more. Create user interfaces in the integrated designer. Code less and do more!

NOW FREE FOR NON-COMMERCIAL USE.

CDP Technologies AS
Hundsværgata 8, 6008 Ålesund, Norway
Tel: +47 990 80 900 • info@cdptech.com
www.cdpstudio.com

CDPStudio



With two 40-pin male headers, you can mount two mini HATs, or one mini HAT and one full-size one

HAT Hacker HAT

▶ Pimoroni ▶ **magpi.cc/hathacker** ▶ £9 / $10

SPECS

CONNECTIONS:
2 × 40-pin male headers for mounting two mini HATs

FIXINGS:
6 × black metal PCB standoffs, 12 × screws

DIMENSIONS:
65×62×16 mm

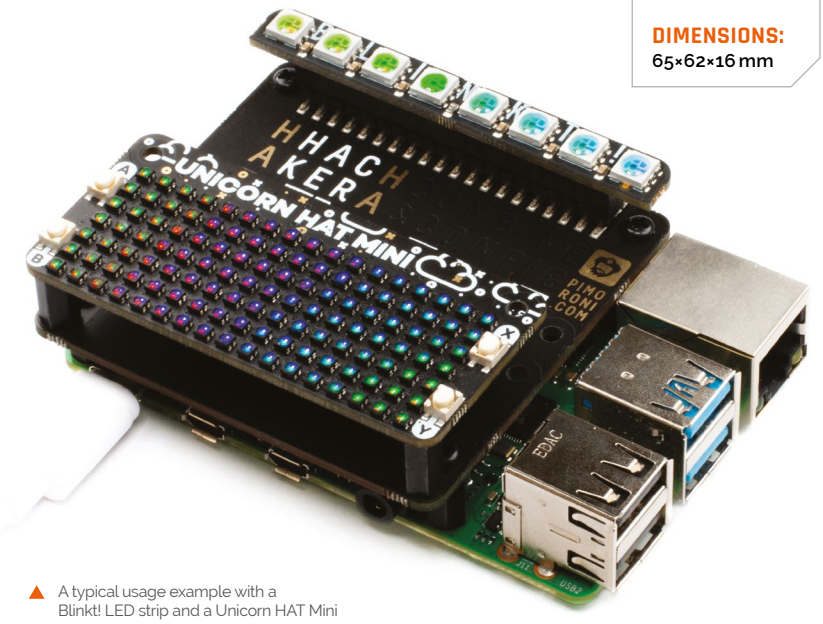
Enabling Raspberry Pi to wear two mini HATs. By **Phil King**

With two sets of 40 GPIO pins, the HAT Hacker HAT enables you to connect two mini HATs (aka pHATs) – or one mini HAT and one full-size HAT – to Raspberry Pi. It serves a similar function to Pimoroni's earlier (and still available) Black Hat Hack3r, but rather than being connected via a ribbon cable, it is a HAT itself and so mounts directly on Raspberry Pi's GPIO header.

Supplied fully assembled, it'll work with any 40-pin Raspberry Pi model. Pimoroni has thoughtfully included six metal PCB standoffs (with screws) for spacing and securing it, and/or the attached mini or full-size HATs, via pairs of the eight mounting holes provided.

Pin checking

No extra software is required. So, in theory, you just mount your two mini HATs on the HAT Hacker HAT and you're all set to go. Well, you do need to check that the same pins aren't used by both HATs (unless they're I²C pins with different addresses), which is easily done by looking them up on **pinout.xyz** – or using the pHAT Stack Configurator (**pinout.xyz/phatstack**), although the latter offers a limited selection of HAT options and could do with updating.



A typical usage example with a Blinkt! LED strip and a Unicorn HAT Mini

If you need to attach more than two HATs to Raspberry Pi, a pHAT Stack (**magpi.cc/phatstack**) would be a better option, as you can attach up to five mini HATs or three full-size HATs.

For most project setups, however, the HAT Hacker HAT should suffice. You might well want to connect just one mini HAT, e.g. a display, and use the other set of pins for connecting electronic components – for which the full labelling of the second header's pins is very useful. Unlike the similar FullpHAT (**magpi.cc/fullphat**), however, there are no bonus GPIO breakout holes in addition to the two main headers.

Verdict

A compact and solidly built solution to mounting more than one mini / full-size HAT on Raspberry Pi at a reasonable price.

8/10

SFERALABS

The best of the maker world for the industrial automation market!

strato

Servers

iono

I/O Modules

exo

Sensors

POWERED BY Raspberry Pi

SFERALABS

www.sferalabs.cc

VueScan

▶ Hamrick Software ▶ **hamrick.com** ▶ £32 / $40

▲ From a preview, you can look at changing settings and see how the final scan might look

The powerful scanner software is now available for Raspberry Pi OS. **Rob Zwetsloot** puts it through its paces.

The promise of VueScan is very alluring. Its goal is to help you use a vast number of scanners on various operating systems, thanks to some fancy reverse engineering and a massive list of internet-accessible drivers for over 6000 different scanners.

Recently, VueScan released a version of its software that works on Raspberry Pi, so we decided to check it out on Raspberry Pi OS.

> Any function the scanner has, you can change

Problem solver

Getting VueScan is pretty easy – head to **hamrick.com** and it will recommend you a version to download for the OS you're using. There are instructions on how to install the version from this download; however, we recommend hitting the link for other versions and finding the ARM32 DEB file for installation. It's a bit easier, only involves a single file, and will then appear in the program menu under Accessories. VueScan can be used for free, although it will embed a watermark onto your scans. It's a good way to test if your scanner will work with it, though, especially as we had some issues with a scanner that was listed as being supported.

Troubleshooting was quite easy, however. Common problems are listed and, once you're through that shortlist, you're taken to the website to find out more. The developers behind the software are a contact form away from giving you some support on the software.

Once you have the scanner working, while you can just hit Scan and get a good picture, you have a lot of options to play with as well. Any function the scanner has, you can change, and you can even add some post-processing to the scanned images: fixing colours, making the imager sharper, and even allowing for character recognition (OCR) so the text of the image can be highlighted. Your mileage might vary on how well that OCR works, though – it's highly dependent on how good your scanner is and the item you're scanning.

SPECS

SCANNERS SUPPORTED: 6000+

ARCHITECTURE: 32- and 64-bit

OPTICAL CHARACTER RECOGNITION: Yes (Pro version)

Verdict

It's a great piece of software that makes scanning on Raspberry Pi much easier than it's ever been.

8/10

10 Best:

Raspberry Pi cases

Keep your Raspberry Pi safe with these amazing cases

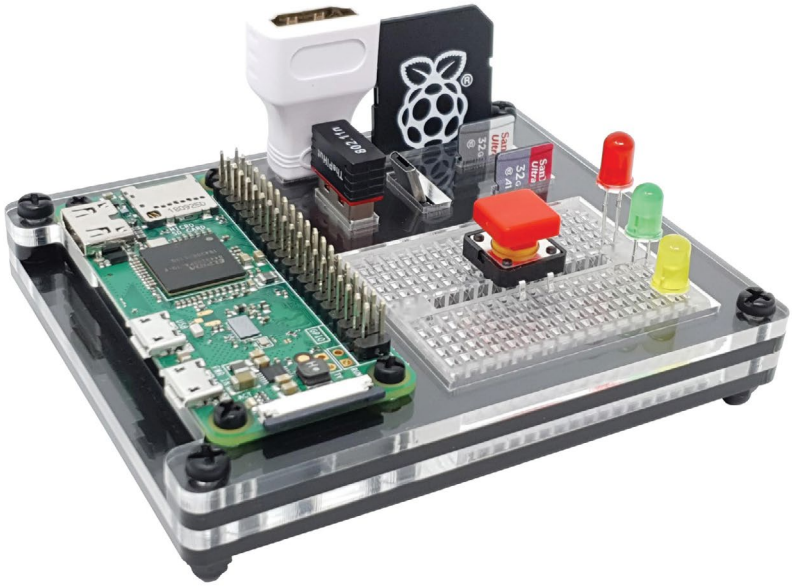
When you love something, you want to protect it. We love our Raspberry Pi, so it seems natural to get a case for it to make sure it lives a safer life. We've seen a ton of cases over the years, with a massive variety of functionality. Here are ten of our favourites.

▲ Official Raspberry Pi 4 case

Official and classy

Simple and sleek, the official case is your go-to protector for a Raspberry Pi. The version for Raspberry Pi 4 is no different, and also comes in a lovely black and grey combination.

£5 / $8 | **magpi.cc/pi4case**



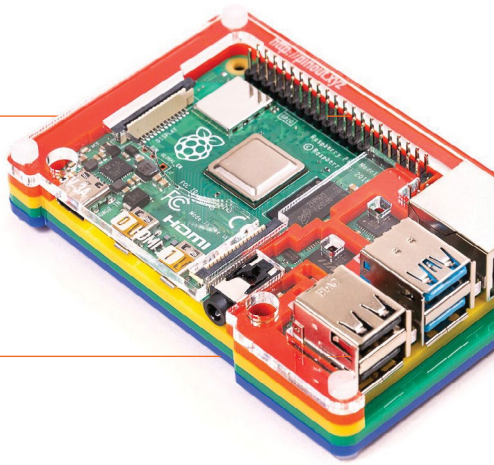
▲ ZeroDock

Stay organised

Keeps your Raspberry Pi Zero safe? Check. Access to all its ports? Check. Ability to keep all your important components neatly stowed next to it? Double check.

£10 / $13 | **magpi.cc/zerodock**

▸ Pibow Coupé



Stackable protection

The original Raspberry Pi case reimagined for Raspberry Pi 4. You'll now only find the Coupé version, as it uses far less plastic and makes it easy to add HATs and other accessories.

£9 / $10 | **magpi.cc/pibow**

▾ Argon ONE

Keep it cool

The winner of our thermal management case group test for Raspberry Pi 4, the Argon One case makes your Raspberry Pi look like a futuristic supercomputer.

£25 / $32 | **magpi.cc/argonone**

▲ SUPERPi Case

Console conversion case

As well as being a nice-looking case, it allows you to redirect ports to appropriate locations, enabling you to plug USB controllers into the front, like on the original console.

£30 / $38 | **magpi.cc/superpi**

▲ SmartiPi Touch 2

Screen base, Lego beginnings

This case allows you add a touchscreen to a Raspberry Pi, while also providing a stand for it to stay upright. There are Lego add-ons you can use to make it extremely fun as well.

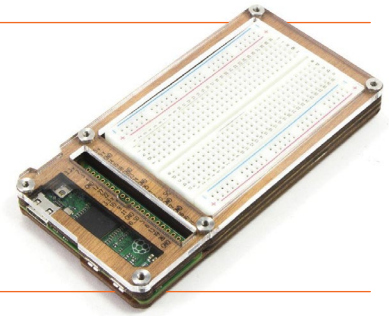
£24 / $31 | **magpi.cc/smartipi2**

▸ Zebra Zero Plus

Easy prototyping

While it does create a much larger footprint than a Raspberry Pi Zero on its own, it makes circuit creation nice and easy, with clearly labelled GPIO pins and an integral breadboard.

£10 / $13 | **magpi.cc/zebrazero**



▲ Aluminium Heatsink Case

Cool speed lines

This cool-looking case will help keep your Raspberry Pi cool. It also comes in several excellent colours (we like the purple), so it should fit any mood.

£12 / $13 | **magpi.cc/alumheatsink**

▼ FLIRC case

Stylish media enclosure

These FLIRC cases are a great, sturdy enclosure with a media PC use in mind. The version for Raspberry Pi 4 has great heat-dissipating abilities as well.

£13 / $17 | **magpi.cc/flirczero**

▲ SecurePi

Maximum security

This is probably more for enterprise use, but if you're concerned about the physical security on your Raspberry Pi, being able to lock off the ports on this case is a great addition.

£10 / $11 | **magpi.cc/securepi**

PRINT YOUR OWN!

None of the cases above catching your eye? 3D-printing a Raspberry Pi case is always an option. Taking one from Thingiverse (**thingiverse.com**) and printing it, or doing a quick mod in Blender, is a great way to personalise your enclosure.

Learn command line with Raspberry Pi

Get under the hood of your computer and control everything from a text interface. By **Lucy Hattersley**

Conquer The Command Line

CREATOR

Richard Smedley

Price:
£3 / $3

magpi.cc/clibook

Underneath the swish exterior of Raspberry Pi OS sits Bash (Bourne-again shell), a command-line interface used to control Raspberry Pi.

This text-based interface forms the heart of Debian Linux, and everything you can do in the visual GUI (graphical user interface) and much more can be done from the command line. Learning to use the command line (typically accessed via the Terminal program) is a rite of passage for serious computer users. The command line enables you to access, and control, every aspect of your computer.

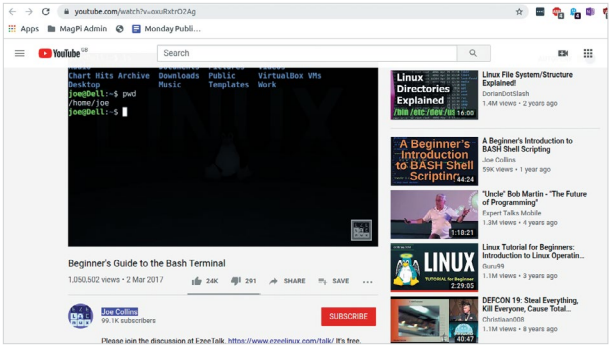
If you're a complete newcomer to the command line, then *Conquer The Command Line (2nd Edition)* is the book to get. This Essentials Guide covers every aspect of the command line, from working with files, installing and removing software, managing drives, remote access, and controlling startup and shutdown procedures.

A recent update covers even more advanced topics such as building software from source code and using the internet from the command line.

Keep this book on your desk and it'll help you get started and be the perfect reference. As with all Raspberry Pi Press books, you can download it as a free PDF.

Three videos to watch

Like learning on YouTube? Check out these vids



BEGINNER'S GUIDE TO THE BASH TERMINAL

Joe Collins has an easy manner and a great tutorial for learning the command line from a single video.
magpi.cc/beginnersbash

COMMAND LINE CRASH COURSE

This talk by Robert Axelsen at the FreeCodeCamp Vienna June 2018 meetup is a great lesson in using the command line.
magpi.cc/clcrash

RASPBERRY PI – GETTING STARTED WITH TERMINAL

This video by Core Electronics guides you through the command line from a Raspberry Pi perspective.
magpi.cc/corecl

Learning the Shell

CREATOR

William Shots

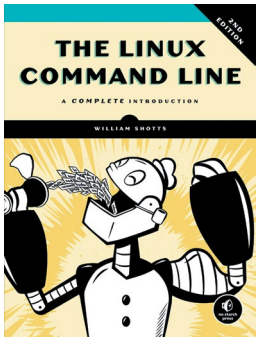
Price:
Free

magpi.cc/learningtheshell

Learning the Shell is a great online page packed with vital information for the newcomer.

It covers navigation, working with files, commands, permissions, and job control. It's highly regarded, and a great guide.

The website has all the information to help you get started, and there is an accompanying book, *The Linux Command Line* (**linuxcommand.org/tlcl.php**) that you can purchase from No Starch Press or download as a free PDF under a Creative Commons licence.



And once you've learned the basics, a follow-on tutorial called 'Writing shell scripts' walks you through more advanced functionality, where you get Raspberry Pi to do the work for you.

Learn the Command Line

CREATOR

Codecademy

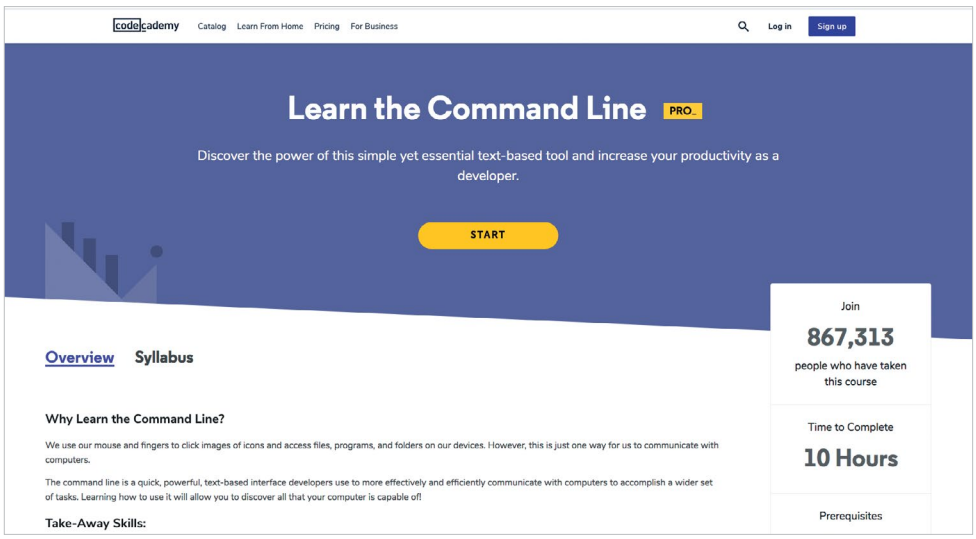
Price:
Free

magpi.cc/codecadamycl

Codecademy's Learn the Command Line is a free interactive website that is hugely popular.

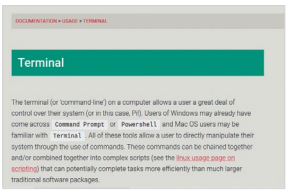
Basic commands are covered, and it tries to keep everything game-like, with usage 'streaks' and points.

Whether you are better off learning from an interactive website or inside the shell itself is debatable (we prefer the shell). But if you find books dry and the shell daunting, then the hand-holding found here will be a breath of fresh air.



Bookmark these webpages

These reference guides are useful to have around



RASPBERRY PI DOCUMENTATION

The Raspberry Pi Terminal documentation page gives an overview of the Terminal and how to use it on Raspberry Pi.
magpi.cc/terminaldocs

DEBIAN WIKI

Raspberry Pi OS is built on top of Debian Linux. So, it makes sense to check out and bookmark the Debian wiki.
magpi.cc/debiancl

BASH REFERENCE MANUAL

This is a fantastic resource that covers a wide range of features and functions beyond the entry-level commands.
magpi.cc/bashref

Helene Virolan

A social entrepreneur aiming to encourage more girls to get involved in tech

- Name Helene Virolan | Occupation Director of Girls Into Coding
- Community role STEM ambassador and educator | Website magpi.cc/gic

Helene Virolan and her daughter Avye Couloute are two incredibly important people in the maker community, making huge steps to help young girls explore an interest in STEM. Not to mention, they're one of the rare mother and daughter tech teams in the community.

"Two years ago, I helped my now twelve-year-old daughter Avye to set up Girls Into Coding (**magpi.cc/gic**)," Helene tells us. "[It's to help] get girls involved in coding and tech activities and engage them to consider STEM education and careers. My tech journey with her has been fantastic. Working with her made me realise that I wanted to continue this journey and be part of those who are contributing to making a difference, to close the gender gap in the tech industry."

Avye and Helene showing the award-winning Voice O'Tronik Bot created by Avye

The Girls Into Coding (GIC) events were incredibly successful, so Helene has decided to commit fully to the organisation: "Earlier this year, we set higher ambitions and established GIC as a Community Interest Company aiming to offer more girls more opportunities to develop."

When did you first learn about Raspberry Pi and CoderDojo?
I first learnt about the Raspberry Pi about five years ago at a Raspberry Jam in Covent Garden, Central London. Someone at a Code Club meetup recommended going, so Avye and I did.

Avye really enjoyed going to these kinds of events so we continued to go to other Jams. They encouraged her to craft, code and design, and allowed me to discover new things.

At one in particular, I was approached by the organiser of a CoderDojo and was invited to become a mentor. I accepted and soon volunteered as a Scratch

> We've created workshops and education events to immerse young people in a range of STEM activities

▲ As well as being a mentor at a CoderDojo, Helene is a Raspberry Pi Certified Educator

Favourite builds

Voice O'Tronik Bot

Avye's favourite project is the Voice O'Tronik Bot. It's a voice command robot made out of upcycled materials. In 2018, she showcased it at the Coolest Projects UK and it won first place in the hardware category.

Web-controlled robot activity

Avye and I worked alongside Llewelyn [Fernandes] from Think Learn Create to develop and deliver a web-controlled robot activity. We used a Raspberry Pi Zero and Raspberry Pi camera mounted on chassis made up of parts from previous iterations of Avye robotics kits. Collaborating is great anyway, but this was probably the most fun I've had on a project.

We tested the web-controlled robots with a group of participants in May; they had a balloon-popping fest on our floor.

mentor. Since my introduction to Raspberry Pi, I've enjoyed tinkering with the various iterations, and last year I attended the Picademy in Cambridge to become a Raspberry Pi Certified Educator.

What is Girls Into Coding?

Girls Into Coding was an idea that Avye had when she was ten. By this age she had moved from attending workshops at CoderDojo to preparing and delivering them herself at the Kingston University Dojo and at other community events. After repeatedly observing that most attendees at her workshops were boys, Avye identified a need to help more girls to access these events and was determined to encourage more girls to give STEM a go. As a result, she decided to set up 'Girls Into Coding'(GIC), a series of events targeting girls.

Girls and women are under-represented in STEM and the new generation has a chance to change that. Girls Into Coding is our way of contributing to that change. Moreover, GIC was set up to encourage girls to explore and enjoy STEM subjects and to encourage them to pursue further activities, education, and careers.

We provide learning opportunities for girls aged 10 and 14. We've created workshops and education events to immerse young people in a range of STEM activities. Our workshops are led by Avye and specialist STEM educators, and assisted by volunteers. The workshops consist of robotics, physical computing, and 3D printing. As well as participating in hands-on workshops, the events give the girls an opportunity to listen to lightning talks throughout the day, delivered by inspiring female role models who are doing cool stuff in the tech world.

Do you have any advice for anyone wanting to mentor or help teach computing subjects?

I would encourage anyone to be a mentor, even if they do not have the computing knowledge at first. You can learn as you go along. Preparation is key. Spend some time preparing the session in advance, give yourself enough time to get used to the content, and practise.

Mentoring is a very rewarding experience. It can be daunting at first but after a few mentoring sessions, your confidence will grow and before you know it you will soon find out that you can do it and have fun in the process! Be brave and do not overcomplicate it! There are lots of online tutorials that are available for those starting out. They are a very good place to start.

This Month in Raspberry Pi

MagPi Monday

Amazing projects direct from our Twitter!

Every Monday we ask the question: have you made something with a Raspberry Pi over the weekend? Every Monday, our followers send us amazing photos and videos of the things they've made. Here is a small fraction of them. Follow along at the hashtag #MagPiMonday.

01. This is actually quite a neat little replica!

02. The oak frame on this is very nice – we're looking forward to seeing how this goes!

03. Lego is always dazzling, to be fair

04. This is actually very useful for those of us that sometimes podcast or make videos

05. So far, this is one of our favourite Raspberry Pi High Quality Camera projects

06. We often yell at ancient relics too

07. It balances! Good work

08. We always love to see model railways that use a Raspberry Pi

09. Whatever could this be...?

01

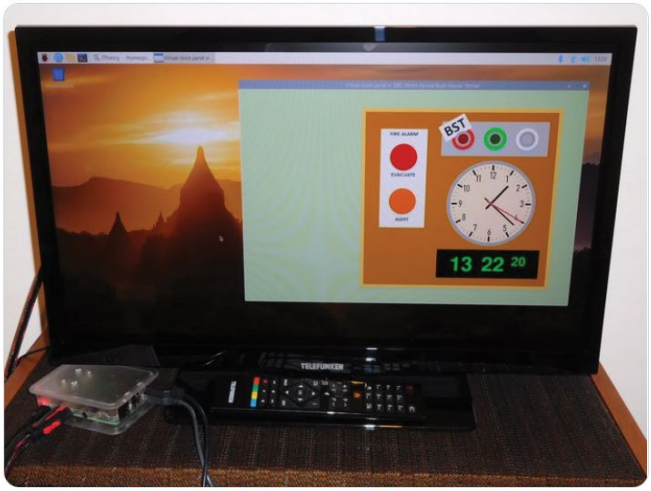
Paul Webster
@G7KVE

Replying to @TheMagPi

This week I developed a fully-functional virtual replica of @BBCRosAtkins' studio clock panel (a souvenir of the @bbcworldservice Bush House era).

Viewers of @BBCOS on TV will know what this is about...

Since yesterday the code is running on a Pi 3B+ driving a 22" LED screen.



02

Martin Parker
@Mr_MartinParker

Replying to @TheMagPi

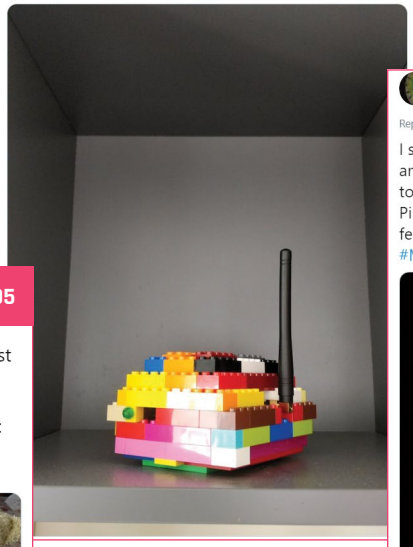
I've made a start on my heating / security project using a @Raspberry_Pi 3B+ and a 5" lcd screen in a frame I made from off cuts from my OakPi Laptop. Here I'm testing the normal screen being a kind of screen saver of time then to access menu which I've still to work on

03

damian koblintz
@dkoblintz

Replying to @TheMagPi

Built a case for text transponder (I know I'm dazzled too)

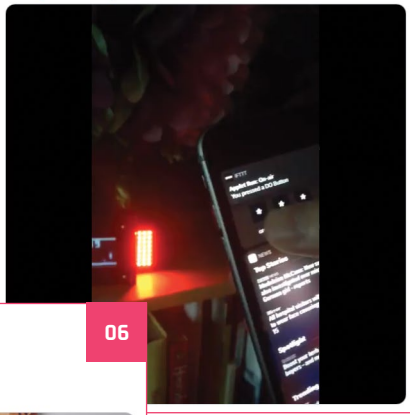


04

Scott Young
@thebot75

Replying to @TheMagPi

I set up an 'on air' indicator as I'm working from home and on video calls a lot, but want my wife and daughter to be able to move about their home freely 😂. Uses a Pi Zero W, @pimoroni Unicorn PHAT, @adafruit IO feed, and @IFTTT button widgets on my phone to set. #MagPiMonday



05

Pierre-yves Baloche
@FunkyPiwy

Replying to @TheMagPi

Finalised my last projet : The #PiHQCam with the latest #RaspberryPi HQ Camera ! Including some great @pimoroni components to integrate in my own #3Dprint enclosure. You can read more about it here : medium.com/@baloche.pierr... . Ideal for some upcoming #ShotOnRaspberryPi

06

8 Bits and a Byte
@8BitsandaByte

Replying to @TheMagPi

A delightfully useless ancient relic

07

Raspibotics
@raspibotics

Replying to @TheMagPi

I've finally managed to get the robot balancing to some extent! 😀

08

PenguinTutor
@penguintutor

Replying to @TheMagPi

I've wiring my outdoor model railway to a Raspberry Pi. Now with analog control, IoT web browser speed control or fully automated mode (stop and start from station).

penguintutor.com/projects/model...

Now starting a YouTube video series on how it works. youtu.be/x3bClfb_ZxU

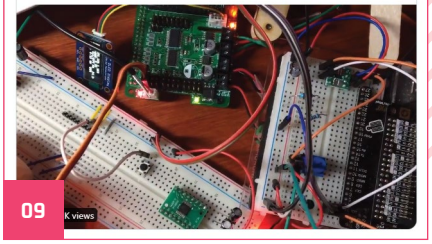
09

Dr Footleg - Roboteer
@drfootleg

Replying to @TheMagPi

I got all the components for my new HAT design working together and finally got the #PCB order submitted for fabrication!
twitter.com/drfootleg/stat...
#MagPiMonday

> Dr Footleg - Roboteer @drfootleg · May 26
> I admit my test rig has scope for improvement, but this confirms all the circuits which will be included on my next revision of this board work together. Board is off the Pi so I can test the 3.3V regulator which will be used instead of the Pi 3.3V line.



Coolest Projects 2020 goes online

Show off your amazing projects online in the revamped Coolest Projects

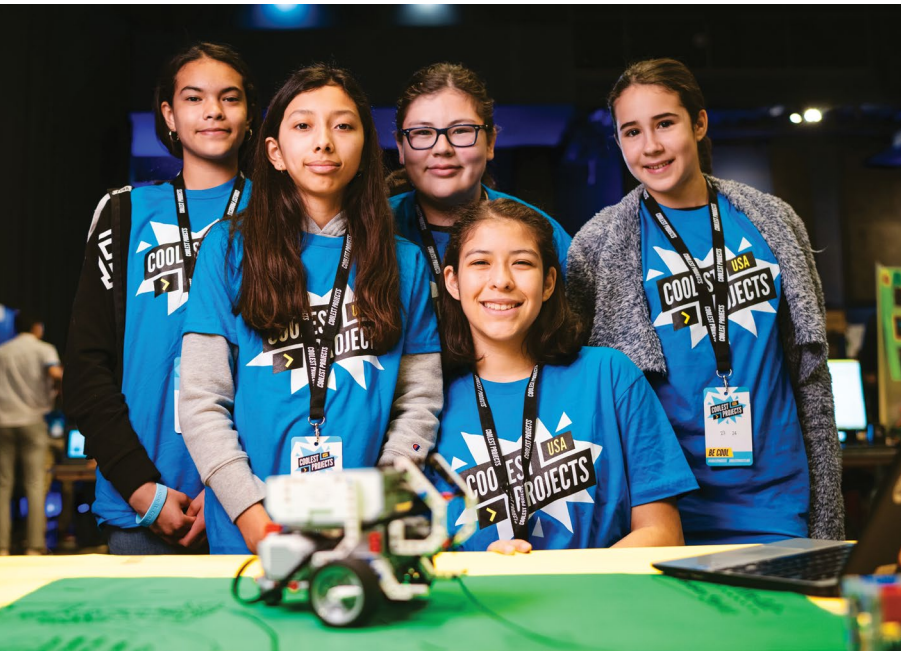
If you're feeling a bit gutted at the lack of Coolest Projects UK and International this year, you'll be excited to know that Coolest Projects is going online!

The online version will be truly international, with everyone up to the age of 18 able to register their projects for free.

It may be a bit late to enter by the time you read this as entries close on 28 June. However, they don't need to be completely finished so if you want to enter, here's what you'll need to do below. Head to **coolestprojects.org** to find out more.

How it works

Projects will become available for anyone to view once the digital showcase opens in June, and in July, a series of high-profile guests will announce their favourite projects!



▲ Coolest Projects USA 2020 went ahead in early March, but the UK and International events will take place online

Projects should be entered into one of six categories: Hardware, Scratch, Mobile Apps, Web, Games (excluding Scratch), and Advanced Programming. Once registration opens, you will need to upload your project and share some details with us about what you have created.

> A series of high-profile guests will announce their favourite projects!

If you are registering a project made in Scratch: when registration opens, save your project to the Coolest Projects Scratch Studio (**magpi.cc/cpscratch**), and answer a few questions about your project in the Scratch description, then complete the registration form.

If you are registering a project in another category: make a short video (maximum of two minutes) in which you screen share your code and answer a few questions about your project. When registration opens, upload the video to YouTube or Vimeo, then complete the registration form.

Crowdfund **this!** Raspberry Pi projects you can crowdfund this month

NanoSound ONE

The NanoSound ONE is described as a '192kHz 24-bit hi-res network audio streamer/media PC with hi-fi DAC and Raspberry Pi 4 hosted within sleek aluminium enclosure'. The enclosure in question is a modified Argon ONE – one of our favourite Raspberry Pi cases, especially if you want to keep a Raspberry Pi nice and cool. The NanoSound ONE uses technology from the great NanoSound DACs which is then built into the case, making construction nice and simple. It even allows full access to the GPIO pins, which is very snazzy.

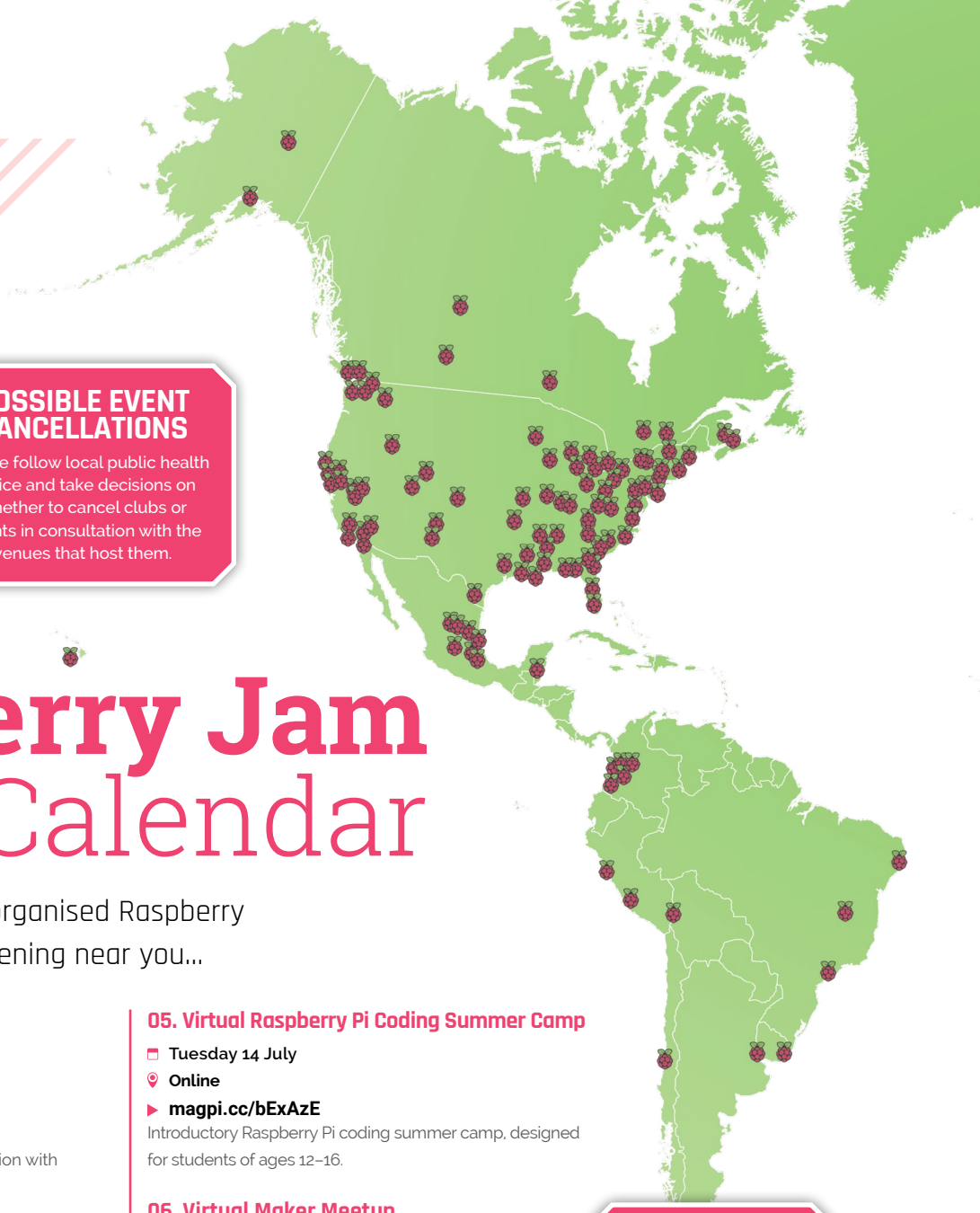
▶ **kck.st/2ZVPXma**

Hi-Fi DAC Case for Raspberry Pi

CROWDFUNDING A PROJECT?

If you've launched a Raspberry Pi-related project, let us know!

magpi@raspberrypi.org



POSSIBLE EVENT CANCELLATIONS

Please follow local public health advice and take decisions on whether to cancel clubs or events in consultation with the venues that host them.

Raspberry Jam Event Calendar

Find out what community-organised Raspberry Pi-themed events are happening near you...

01. Computer Vision on Depth Cameras: Spatial AI

- Wednesday 1 July
- Online
- **magpi.cc/iN8Sc3**

A webinar for learning about computer vision with OpenCV on a Raspberry Pi.

02. Houston Raspberry Pi Meetup

- Wednesday 1 July
- Online
- **magpi.cc/qfSx5y**

A community of coders, artists, educators, and engineers interested in project-based learning with Raspberry Pi.

03. Virtual Camp: Retro Arcade Raspberry Pi

- Monday 6 July
- Online
- **magpi.cc/bPpepE**

A virtual engineering summer camp for high school students, hosted by the University of Kentucky College of Engineering.

04. 100th Preston Raspberry Jam

- Monday 6 July
- Online
- **magpi.cc/bpQNkc**

An evening of online talks and networking for people with an interest in Raspberry Pi.

05. Virtual Raspberry Pi Coding Summer Camp

- Tuesday 14 July
- Online
- **magpi.cc/bExAzE**

Introductory Raspberry Pi coding summer camp, designed for students of ages 12–16.

06. Virtual Maker Meetup

- Tuesday 14 July
- Online
- **magpi.cc/u63uef**

Join other makers for a virtual meetup where folks share ideas, inspire and encourage each other, and highlight the things they've made.

07. Internet of Things (IoT) Using Raspberry Pi

- Monday 20 July
- Online
- **magpi.cc/QK1iPi**

Learn how to use Raspberry Pi without a monitor, keyboard, and mouse. Deploy programs remotely.

08. Medway Makers Tech Meetup

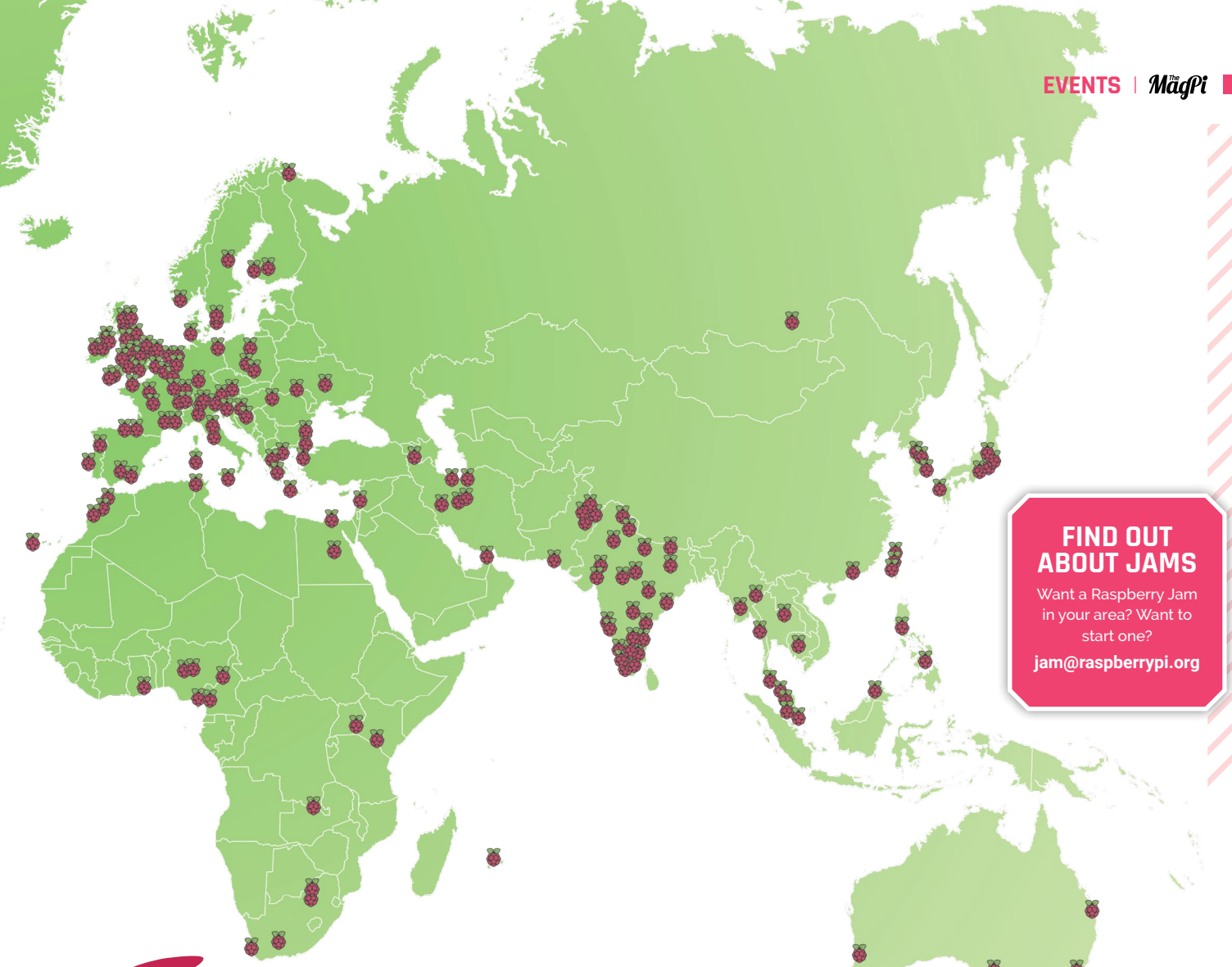
- Sunday 2 August
- Online
- **magpi.cc/cWiYbd**

Workshops in building cool things with Arduino, how to code for beginners, how to use a Raspberry Pi, etc.

FULL CALENDAR

Get a full list of upcoming events for July and beyond here:

rpf.io/jam



FIND OUT ABOUT JAMS

Want a Raspberry Jam in your area? Want to start one?

jam@raspberrypi.org

We've highlighted some of the areas in need of a Jam! Can you help out?

Raspberry Jam advice:

Planning and support

"If you want your Jam to be child-friendly, involve a local school or teacher in the planning stage."

Cat Lamin – Wimbledon Raspberry Jam

Every Raspberry Jam is entitled to apply for a Jam starter kit, which includes magazine issues, printed worksheets, stickers, flyers, and more. Get the book here: **rpf.io/guidebook**

Your Letters

Tips for labelling

Hi. Like most people, I have tens if not hundreds of microSD cards, and the number of Raspberry Pi computers in the house isn't far behind!

I have a Raspberry Pi 4 that is running my amateur radio station, a Raspberry Pi Zero running a flight radar program, and two more Raspberry Pi Zeros running MMDVM (one for Fusion, the other for DMR / DSTAR).

I also completed a project for work that uses a Raspberry Pi Zero to control the pH of a solution used in a chemical process.

All these Raspberry Pi boards and SD cards can be a pain: what Raspberry Pi does what? And, more importantly, what SD card does what? My solution: QR, and Micro QR codes. QR codes allow you to store up to 4296 characters, more than enough to describe essential information, and the Micro QR codes allow around 21. If you needed more information, you could set up a spreadsheet that gave more information – i.e. use the Micro QR code to store a number, then reference that number on a spreadsheet.

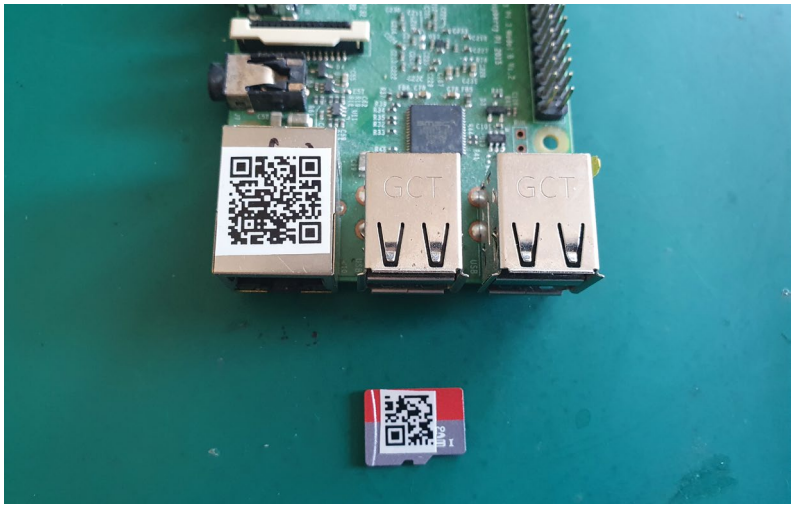
Using these two codes allows me to write a short description of what the card / Raspberry Pi is used for, as well as information such as IP address and any other information I may need.

I use a cheap Brother P-Touch P700 printer to make the labels, and need never worry if the card I am about to format contained something I had been working on for months!

Some of the advantages of the QR system are that the labels are removable and can store enough information to allow you to keep those precious projects safe.

Attached is a picture of the SD / Raspberry Pi labels I use – there is no personal information on the QR codes, they are just an example.

Lee via email



▼ A few people have been sending in their SD labelling ideas – we'd love to hear yours!

Thanks for the tip, Lee. While we don't think hundreds of SD cards is the norm, we definitely have several that could benefit from a mini QR code database. We appreciate the explanation.

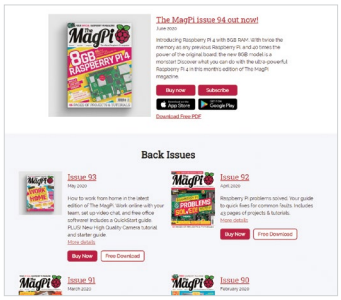
Inspirational

I see every Monday you have 'what did you build at the weekend?' tweets. I have two Raspberry Pi boards – one is running pi-hole and the other Grafana, Telegraf, etc. People's projects are amazing; where have I gone wrong?

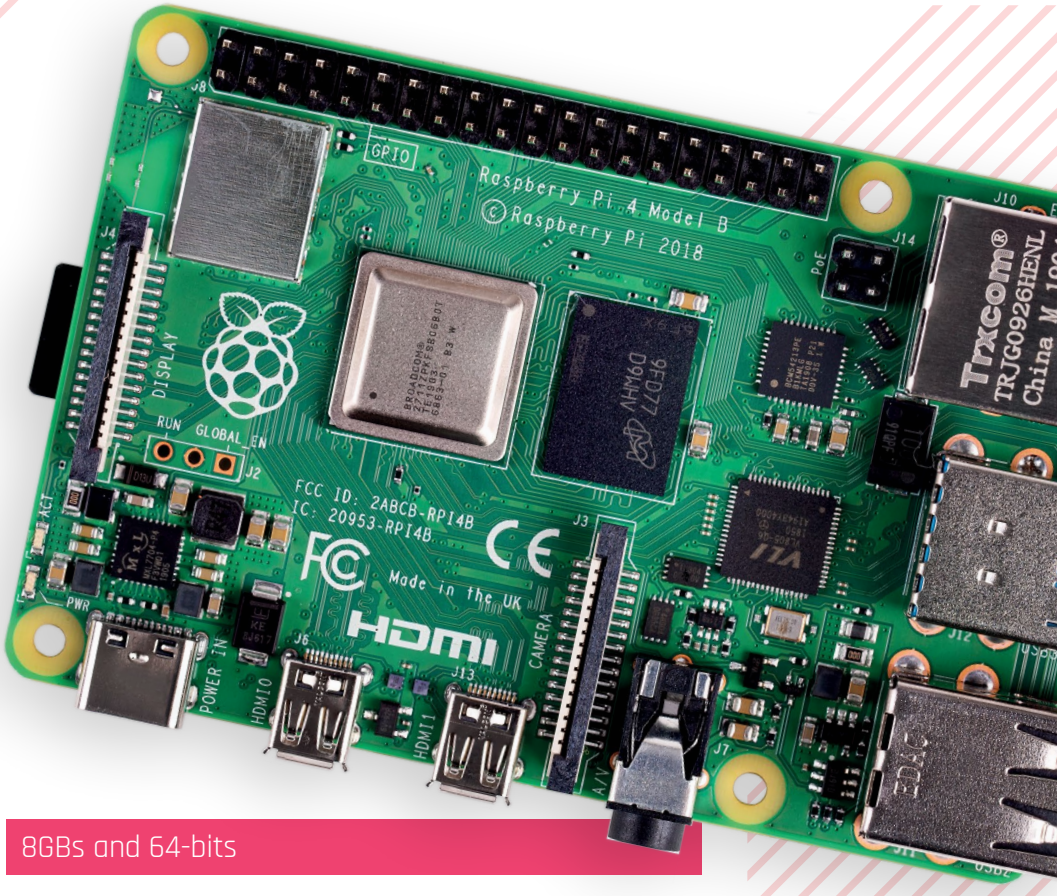
Where can I get inspiration for cat feeders and the like?

Mark via Twitter

Well, Mark, we generally hope that sharing all these incredible community projects will give folks a few sparks of inspiration. We understand that not all of them may be achievable by every follower, though, so do make sure to browse through the magazine (remember, all our issues are available as free PDFs on our website) for anything simpler for you to get started with.



▲ Your regular reminder that all issues of *The MagPi*, along with our many books, are available as free PDFs from our website: **magpi.cc/issues**



8GBs and 64-bits

It's good to see that the 8GB Raspberry Pi 4 is no longer a typo in a manual! I have been wondering if it would ever exist and was hoping it might come with a full-fledged 64-bit version of the operating system. Is there a version coming?

Farah via Facebook

Currently Raspberry Pi OS 64-bit is in beta, so you can try it out on a Raspberry Pi but it's not quite considered ready for release yet. You can find out more on it, and what the current differences are, in this blog post: **magpi.cc/osupdate**.

One of the benefits of 64-bit Raspberry Pi OS is that it will make better use of the 8GB of memory available. Once it's working to a satisfactory standard, we'll be doing more tests.

Contact us!

- Twitter **@TheMagPi**
- Facebook **magpi.cc/facebook**
- Email **magpi@raspberrypi.com**
- Online **raspberrypi.org/forums**

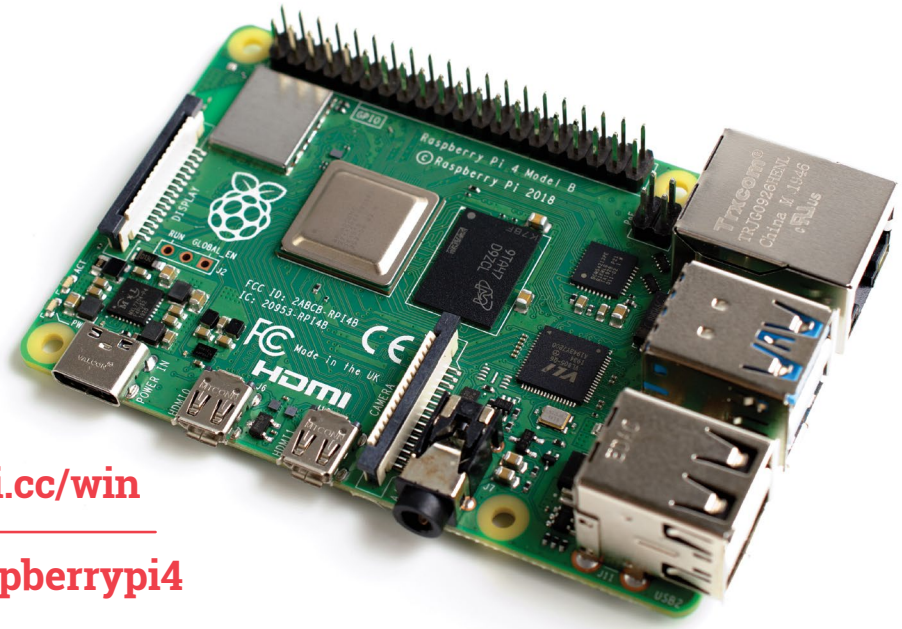
EXPLORE YOUR PASSION FOR PHOTOGRAPHY

ISSUE 164 ON SALE NOW

VISIT **WWW.DIGITALSLRPHOTO.COM** TO LEARN MORE

WIN ONE OF FIVE 8GB RASPBERRY PI 4

The latest version of Raspberry Pi has a whopping 8GB of memory, giving it even more power than ever before.



Head here to enter: magpi.cc/win

Learn more: magpi.cc/raspberrypi4

Terms & Conditions

Competition opens on **24 June 2020** and closes on **30 July 2020**. Prize is offered to participants worldwide aged 13 or over, except employees of the Raspberry Pi Foundation, the prize supplier, their families, or friends. Winners will be notified by email no more than 30 days after the competition closes. By entering the competition, the winner consents to any publicity generated from the competition, in print and online. Participants agree to receive occasional newsletters from The MagPi magazine. We don't like spam: participants' details will remain strictly confidential and won't be shared with third parties. Prizes are non-negotiable and no cash alternative will be offered. Winners will be contacted by email to arrange delivery. Any winners who have not responded 60 days after the initial email is sent will have their prize revoked. This promotion is in no way sponsored, endorsed or administered by, or associated with, Instagram or Facebook.

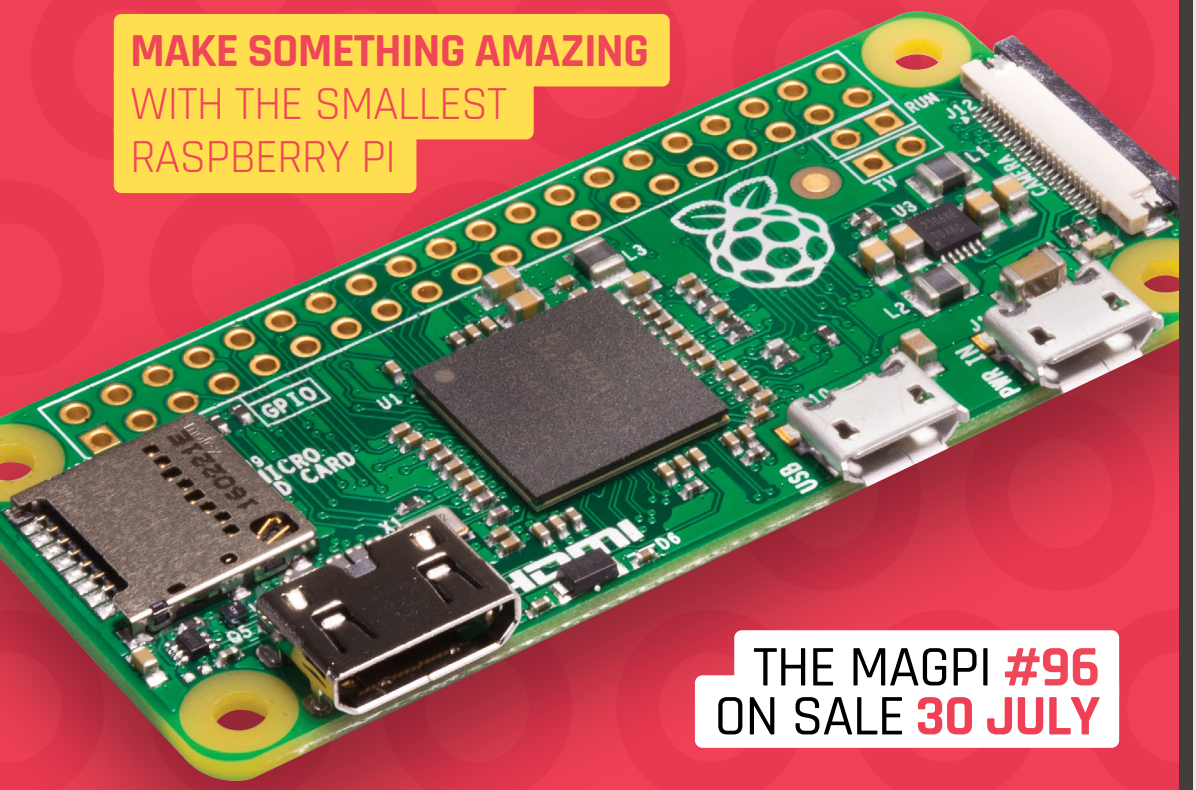
Wireframe

Join us as we lift the lid on video games

Visit **wfmag.cc** to learn more

RASPBERRY PI ZERO PROJECTS

MAKE SOMETHING AMAZING
WITH THE SMALLEST
RASPBERRY PI



THE MAGPI #96
ON SALE 30 JULY

Plus!

The 3D-printed microscope

Build a calendar-controlled busy light

Use DOSBox-X with Windows 98 software

DON'T MISS OUT!
magpi.cc/subscribe

TWITTER	@TheMagPi
FACEBOOK	fb.com/MagPiMagazine
EMAIL	magpi@raspberrypi.com

EDITORIAL
Editor
Lucy Hattersley
lucy@raspberrypi.com

Features Editor
Rob Zwetsloot
rob.zwetsloot@raspberrypi.com

Sub Editors
Phil King and Nicola King

ADVERTISING
Charlotte Milligan
charlotte.milligan@raspberrypi.com
+44 (0)7725 368887

DESIGN
criticalmedia.co.uk

Head of Design
Lee Allen

Designer
Sam Ribbits, Harriet Knight

Illustrator
Sam Alder

CONTRIBUTORS
Mike Cook, David Crookes, PJ Evans, Rosemary Hattersley, Nicola King, Rik Cross, KG Orphanides

PUBLISHING
Publishing Director
Russell Barnes
russell@raspberrypi.com

Director of Communications
Liz Upton

CEO
Eben Upton

DISTRIBUTION
Seymour Distribution Ltd
2 East Poultry Ave,
London EC1A 9PT
+44 (0)207 429 4000

SUBSCRIPTIONS
Unit 6 The Enterprise Centre
Kelvin Lane, Manor Royal,
Crawley, West Sussex, RH10 9PE
+44 (0)1293 312193
magpi.cc/subscribe
magpi@subscriptionhelpline.co.uk

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

ISSN: 2051-9982.

Opening up

Lucy Hattersley takes a breath of fresh air

As I write, I'm sitting on a bench, at the top of a park, gazing out to the city. It's a warm, muggy day with the threat of a thunderstorm later. Right here, right now, I take a deep breath and enjoy life.

Other people seem happy too: the park is full of children, families catching up, and couples sitting around – relaxing. Cups of take-away coffee are being drunk. The local ice-cream van does a roaring trade. One guy is doing his Joe Wicks-inspired training regime. He appears to be head-butting the ground. Nobody minds.

> I'm going to don my face mask, wear a smile, and pop down to WHSmith

I'm aware that all this may not last. Behind the smiles, many folks are struggling. Life isn't completely normal; it may never be. But right here, right now. It's all OK.

The MagPi magazine has had a rough few months. I can see the shopping centre from here and only the supermarket and a lonely health food store remained open. All other shops closed. Some will reopen today; others may never reopen. We haven't sold many magazines in stores over the last few months.

Still, the Raspberry Pi community is strong and we've always had our own online store. Lots of you took up our offer of a subscription and for that we are grateful, thank you. Our subscribers saved us.

The team has struggled on and off. Fortunately, we didn't all fall at the same time, but like everybody we've felt illnesses, anxiety, and distraction. As the saying goes: 'you're not working from home; you're at home, during a crisis, trying to work.'

At one point we ran out of Raspberry Pi Zero computers for new subscribers. We've fixed the problem, and will have shipped all the free computers to new subscribers by the end of the month.

We're not 100% sure what the next few weeks will bring. But hopefully more events will open up, and we may even be able to report on community gatherings. Rest assured, we'll be there letting everybody know when it's safe to start showing off all their projects in public.

Be a reader

When I first started out as a magazine editor, I met Dylan Jones, editor of GQ, and asked him for some advice. He said to pop down to the newsagents once a month and take a photograph of the newsstand. Be a reader of your own magazine, Dylan told me.

I've done that every month since, bar the last few months. Every month I wander to a newsagent, buy a copy of *The MagPi*, and sneakily take a photo. We used to use the snaps for vociferous cover meetings, arguing over headlines, colours, and fonts; mostly, these days, I like the archive.

I've missed it. Wholeheartedly. So if you don't mind, I'm going to don my face mask, put on my smile, and pop down to WHSmith.

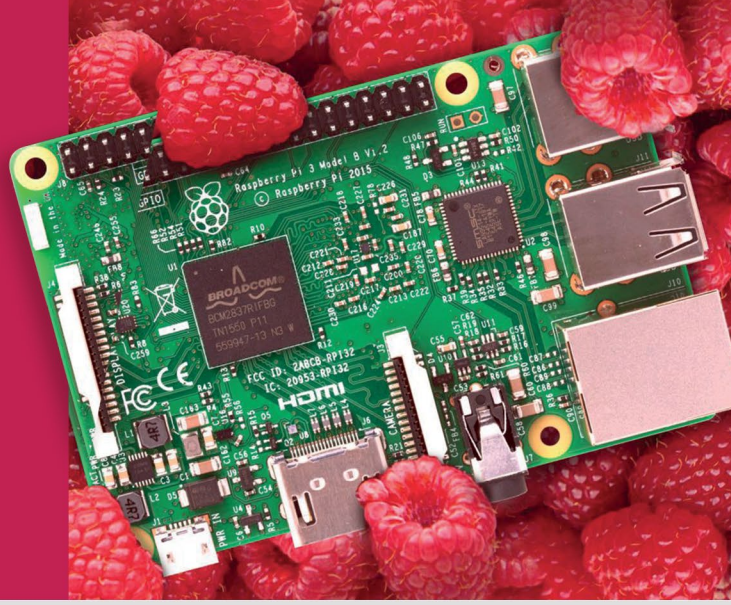
AUTHOR

Lucy Hattersley

Lucy is editor of *The MagPi* and reads it every month. The batteries never run out on print magazines.

magpi.cc

American **Raspberry Pi** Shop

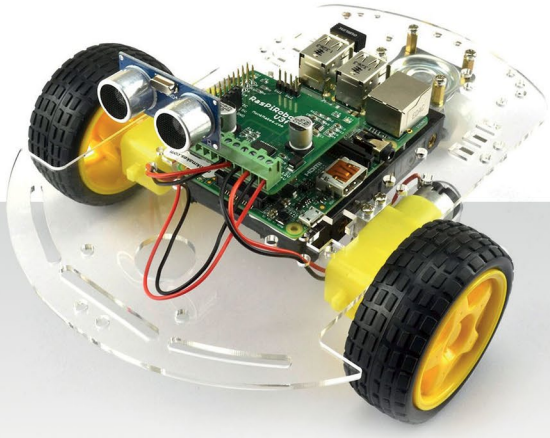


- Displays
- HATs
- Sensors
- Cases
- Arcade
- Swag
- Project Kits
- Cameras
- Power Options
- Add-on Boards
- Cables and Connectors
- GPIO and Prototyping

Partner and official reseller for top Pi brands:

adafruit · sparkfun ELECTRONICS · PIMORONI · HiFiBerry · and many others!

Price, service, design, and logistics support for **VOLUME PROJECTS**



USA PiShop.us

Canada BuyaPi.ca

Raspberry Pi APPROVED RESELLER

RASPBERRY PI 4 STARTER KIT

This 8-piece kit includes everything you need to get going in the world of Raspberry Pi– from all the hardware in a single box, to access to an online step-by-step guide as well as projects to get up and running faster than ever

okdo™
DESIGN THE WORLD

Find out more about this and many other products at

www.okdo.com

@letsokdo