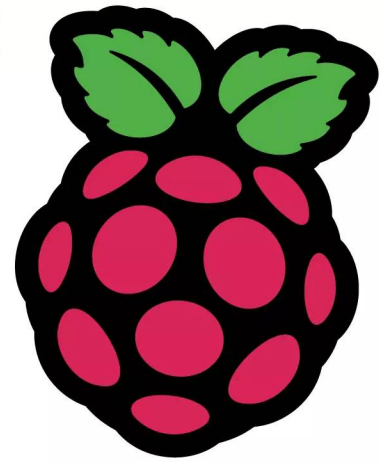




YOUR OFFICIAL RASPBERRY PI MAGAZINE

The MagPi



Issue 136 | December 2023 | magpi.cc

The official Raspberry Pi magazine



Make a
Pico Bubble
Machine



COMPLETE RASPBERRY PI BUYER'S GUIDE

The essential
Raspberry Pi shopping list



£5.99

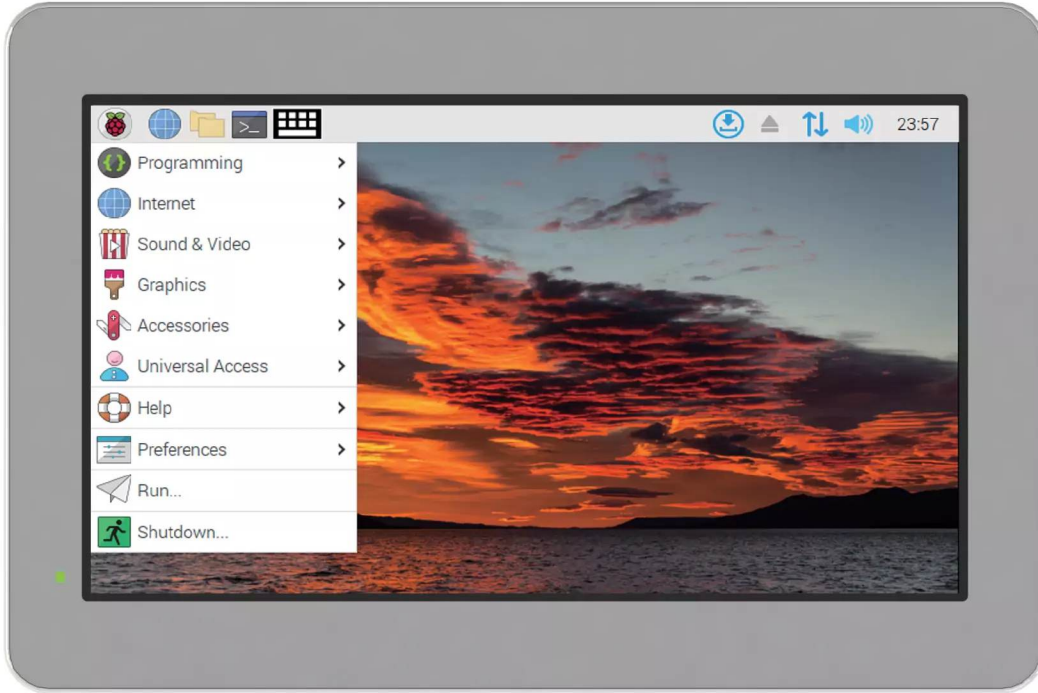


Raspberry Pi 5
Cloud Gaming

OVERCLOCK & BENCHMARK RASPBERRY PI 5



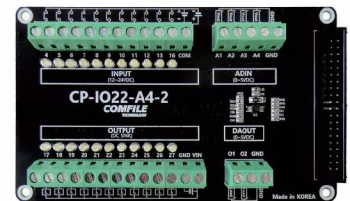
Industrial Raspberry Pi



ComfilePi

The ComfilePi is a touch panel PC designed with high-tolerant components and no moving parts for industrial applications. It features a water-resistant front panel, touchscreen, color LCD (available in various sizes), RS-232, RS-485, Ethernet, USB, I2C, SPI, digital IO, battery-backed RTC (real-time clock), and piezo buzzer.

Use the rear-panel 40-pin GPIO header to expand its features and capabilities with additional I/O boards. The ComfilePi is UL Listed and employs Raspberry Pi Compute Module.



WELCOME

to The MagPi 136

There's a lot of excitement around Raspberry Pi thanks to the ongoing waves of Raspberry Pi 5's epic launch.

We've got a lot of Raspberry Pi 5 coverage this month. From in-depth heating, cooling and thermal testing (page 66) to benchmarking (page 72). We've also got an in-depth guide to Overclocking (page 42).

Once you're done speed testing and overclocking your new Raspberry Pi 5 you'll want to make something with it. Check out our Buyer's Guide (page 32). This is packed with present ideas for the holiday season. From dynamic robots to weather trackers this feature is packed with gifts that will put a smile on any maker's face.

Happy holidays!

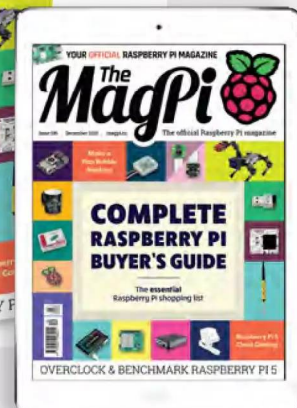
Lucy Hattersley Editor



EDITOR **Lucy Hattersley**

Lucy is editor of The MagPi and has thoroughly enjoyed overclocking her Raspberry Pi. Next up is a seriously fast retro gaming project.

magpi.cc



Raspberry Pi **5**

PRIORITY BOARDING

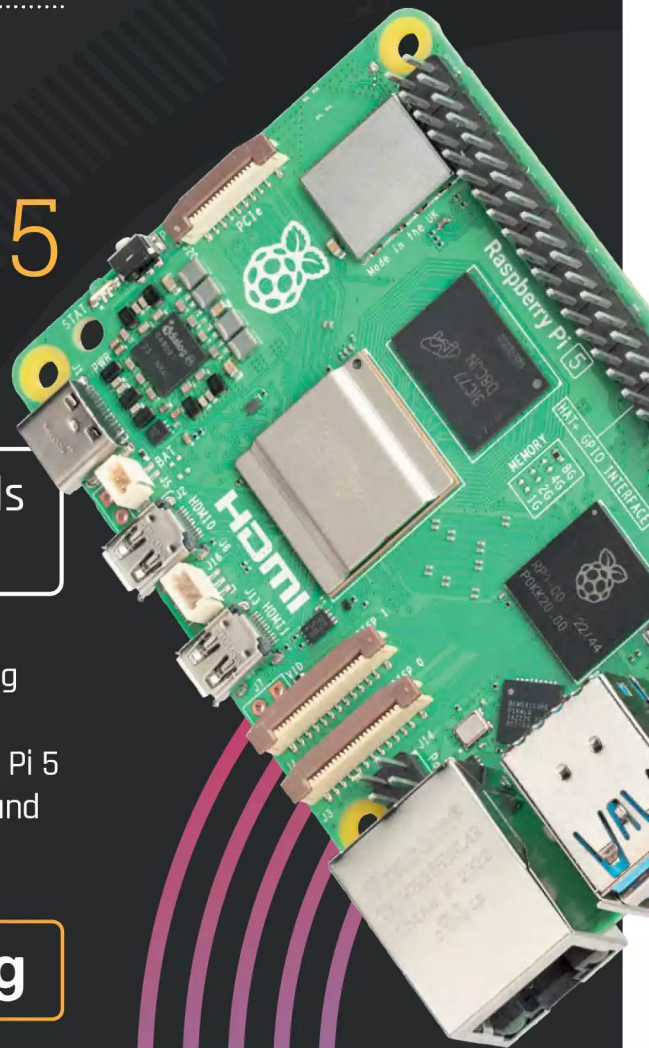
GET YOUR
RASPBERRY PI 5
FIRST!

We've reserved Raspberry Pi 5 boards
for The MagPi subscribers

New subscribers to The MagPi will get a Priority Boarding code when they sign up. So don't delay. Take out a subscription today! You'll be able to buy your Raspberry Pi 5 first and get incredible Raspberry Pi tutorials, projects, and community events every month.

magpi.cc/priorityboarding

TERMS & CONDITIONS A Priority Boarding code will be emailed to everybody who takes out a print subscription to The MagPi or HackSpace magazine. People who subscribe to both magazines will get two codes. Print subs only! Priority Boarding does not apply to people with App Store, Google Play, Zinio, PDF contributions, or other paid-for subscriptions. Each code will entitle you to purchase 1x (one) Raspberry Pi 5 model (either 4 GB or 8 GB) for the standard retail price and delivery. Multiple codes need to be used individually. This is a limited offer and is subject to change or withdrawal at any time.

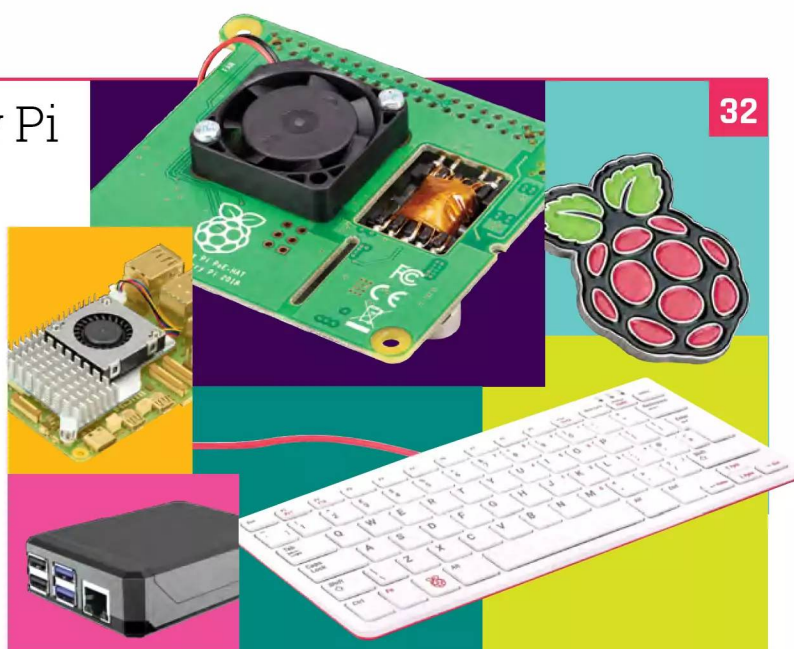


Contents

► Issue 136 ► December 2023

Cover Feature

32 Complete Raspberry Pi Buyer's Guide



32

Regulars

- 90 Your Letters
- 92 Community events calendar
- 97 Next Month
- 98 The Final Word

Project Showcases

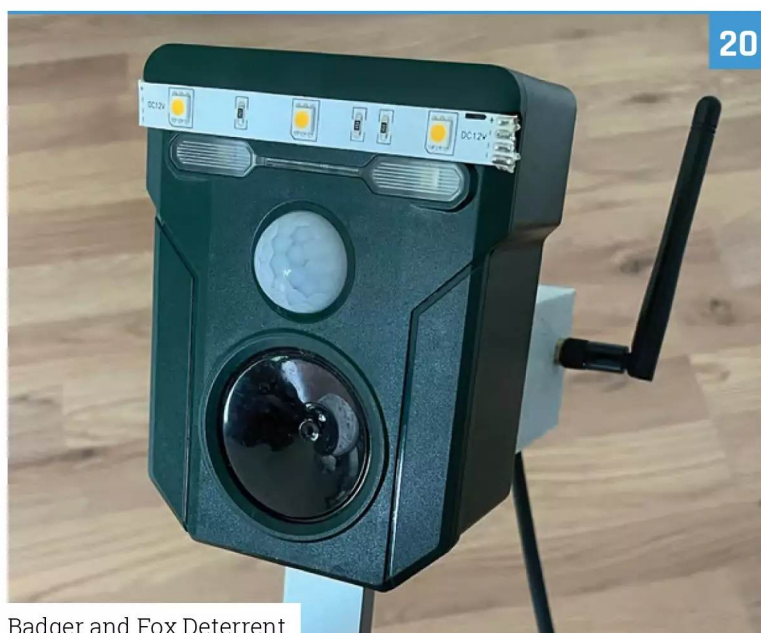
- 08 Retro Pi World Radio
- 12 Confetti Bot
- 14 Pico Bubble Machine
- 16 RePalm
- 20 Badger and Fox Deterrent
- 24 Tide Clock
- 26 Pihome
- 28 Bullet Time Wedding Booth

08



Retro Pi World Radio

20

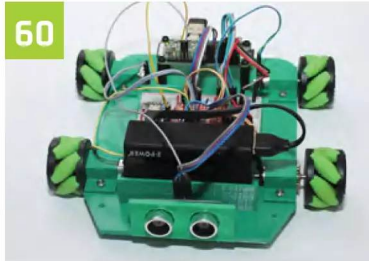


Badger and Fox Deterrent

The MagPi is published monthly by Raspberry Pi Ltd, Maurice Wilkes Building, St. John's Innovation Park, Cowley Road, Cambridge, CB4 0DS, United Kingdom. Publishers Service Associates, 2406 Reach Road, Williamsport, PA, 17701, is the mailing agent for copies distributed in the US and Canada. Application to mail at Periodicals prices is pending at Williamsport, PA. POSTMASTER: Send address changes to The MagPi, c/o Publishers Service Associates, 2406 Reach Road, Williamsport, PA, 17701.

Tutorials

- 42 Overclock Raspberry Pi 5
- 46 Cloud gaming on Raspberry Pi 5
- 50 Python coding on Raspberry Pi 5
- 54 Remote control with TigerVNC
- 56 Using Ubuntu – part 2
- 60 Make a mecanum robot – part 2

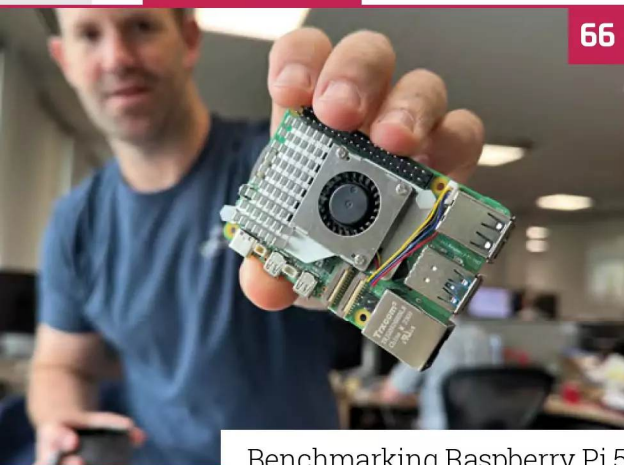


Make a mecanum robot – part 2



Cloud gaming on Raspberry Pi 5

The Big Feature



Benchmarking Raspberry Pi 5



PicoVision review

Reviews

- 76 PicoVision
- 79 10 amazing 3D prints
- 82 Learn Scratch

Community

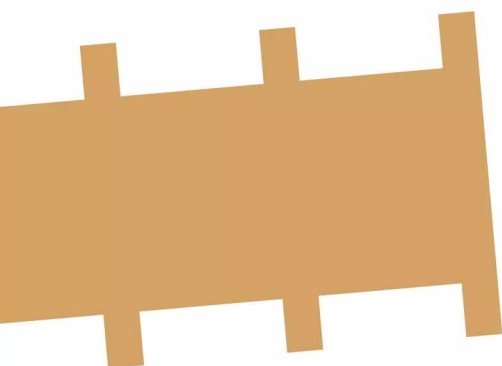
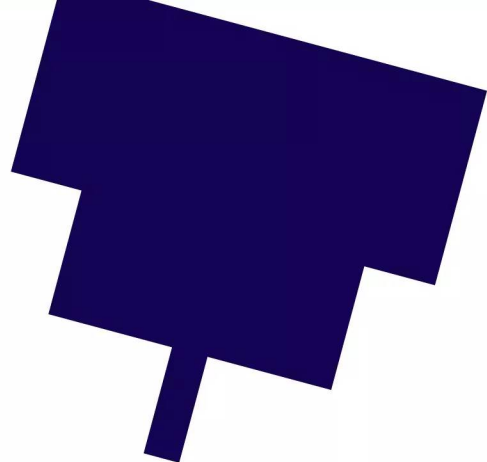
- 84 Odd Jay interview
- 86 This Month in Raspberry Pi



Odd Jay interview

WIN 1 OF 10 **RASPBERRY PI 5 ACCESSORY PACKS** 95

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



Innovation demands selection

Let us help you engineer the future with millions of products from over 2,300 name-brand, industry-leading suppliers.

Find what you need at [digikey.co.uk](https://www.digikey.co.uk) or call 0800 587 0991.

DigiKey

we get technical

DigiKey is a franchised distributor for all supplier partners. New products added daily. DigiKey and DigiKey Electronics are registered trademarks of DigiKey Electronics in the U.S. and other countries. © 2023 DigiKey Electronics, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA

ECIA MEMBER
Supporting The Authorized Channel

Retro Pi World Radio

An old valve radio provides the ideal housing for this nostalgic Raspberry Pi build, finds **Rosie Hattersley**



Raju Subramanian

MAKER

Software engineer Raju loves tinkering with tech, assembling classic characters from LEGO and helping kids get to grips with code

magpi.cc/worldradio

Most of the maker-interviewees who feature in The MagPi's project showcase pages have a deep love of technology and, often, a career in computing. Retro Pi World Radio creator Raju Salubramanian is just such a fellow, but he is also on a mission to teach the next generation to code: something that exactly chimes with why Raspberry Pi exists in the first place.

Raju describes himself as an AFOL (adult fan of LEGO) and, as well as building his own Optimus Prime, Hulkbuster and other complex kits, he has used the brightly-coloured bricks in his mission to teach his kids and their friends how to code. He does this as a 'coach' for the FIRST LEGO League (firstlegoleague.org) in his home town of San Jose, California. He is almost as big a fan of Raspberry Pi as he is of LEGO, with multiple projects under



▶ He posted a video of his project on YouTube showing station selection

**Quick FACTS**

- > Raju and his partner both love 3D printing projects
- > And sell their designs online
- > This build would cost roughly \$275 if you made it from scratch
- > But Raju was able to repurpose many components he already owned
- > Next, he's going to tackle a Raspberry Pi LEGO Robotics project

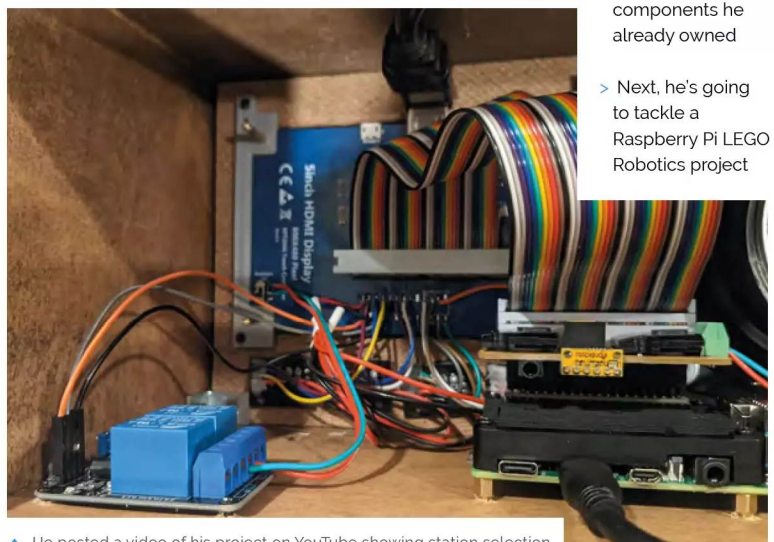
“ Raju wanted something that would force us to slow down in our current fast-paced digital world ”

his belt and six Raspberry Pi boards in use around his home (not all disclosed to his partner!).

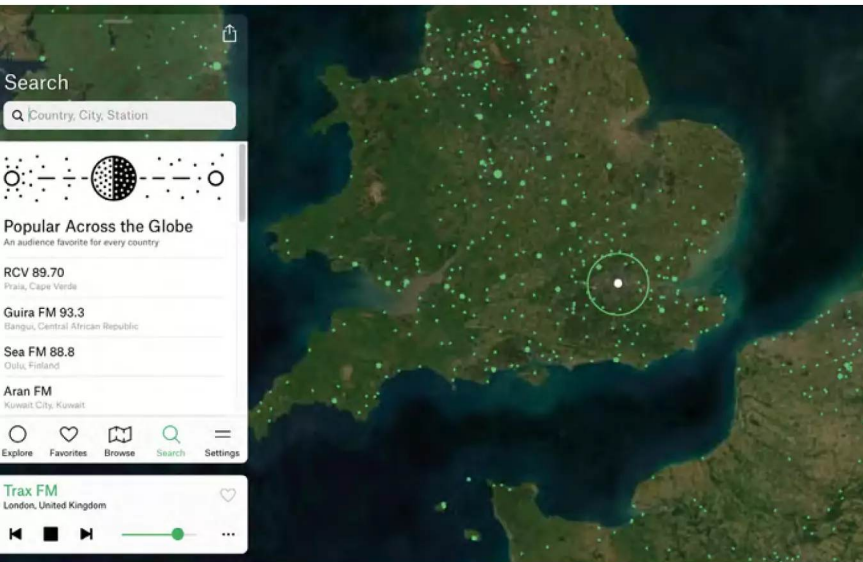
Despite his thoroughly modern career as a software engineer in Silicon Valley and a CV that lists VMware and Google among his credits, Raju has a fondness for the old days. “I have always loved to tinker and repurpose old hardware to build something new”, he says, favouring Raspberry Pi boards because they are “perfect” for the projects he likes to work on since “they are so easy to setup and get started with”.

A nod to nostalgia

For the retro radio build Raju wanted something that would force us to slow down in our current fast-paced digital world. He says the analogue controls of the radio are in some way the opposite of how we interact with our devices today with touch and voice. The project was also meant to



▲ He posted a video of his project on YouTube showing station selection



▲ Radio.Garden provides a user-friendly global station selector

remind Raju of his childhood of using analogue knobs to change stations. His lengthy search for the ideal old radio eventually led to him choosing one with a speaker to one side and enough space to cut out a window and fit a new 5in screen. His Retro World Radio can tune in to a vast collection of internet streaming radio stations from around the globe, and features a vintage map to browse and find radio streams.

Keep on improving

Raju had seen other retro radio upcycling projects online, and was particularly keen on creating one similar to this internet radio project:

magpi.cc/internetradio. He used a 2GB Raspberry Pi 4 because he “needed the extra horsepower of Raspberry Pi 4 to get experience with the map display”. He bought the main components from online stores such as Adafruit.

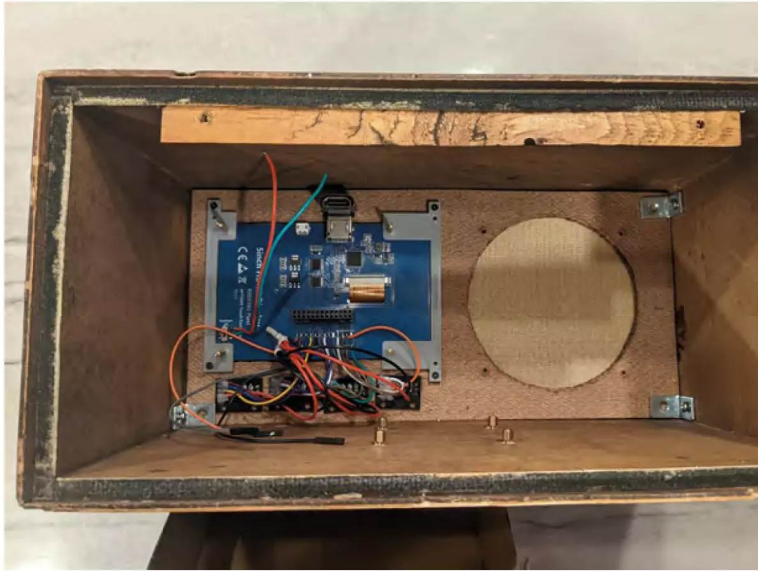
Sourcing a suitable radio and writing the software interface were his main challenges,

“ Retro World Radio can tune in to a vast collection of internet streaming radio stations ”

since backend engineering is his professional forte. “I enjoyed the experience of writing all the Javascript code for the display and I am still thinking of ways to improve the overall aesthetic. Raju designed the new parts himself including a new front face for the radio, using CAD software to design and create some



▲ The searchable Radio Garden site serves up stations from around the globe



3D-printed parts. He also wrote software for the Retro World Radio, making liberal use of existing SDKs and open source or free online libraries.

He began with a development platform using a breadboard, which allowed him to test the software. The first version had three rotary knobs mapped to Raspberry Pi's GPIO pins, but Raju later decided to use four knobs. These pull in internet streams from Radio.Garden and can be chosen by location as well as genre.

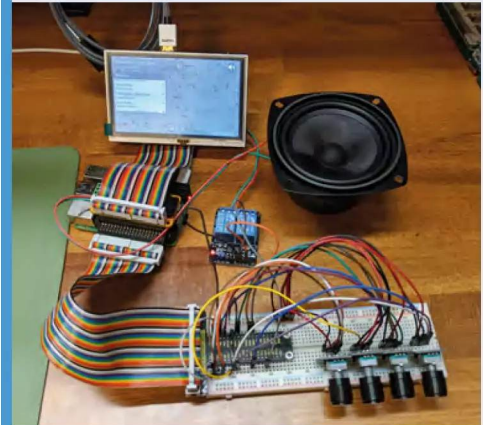
Having completed the radio project, Raju is now gearing up to combine his love of tinkering with tech with LEGO, and has recently acquired a Raspberry Pi Zero W. [M](#)

▲ Raju bought an old radio from eBay for its off-centre speaker and warm acoustics

▼ He used the remaining cabinet space for the new radio menu



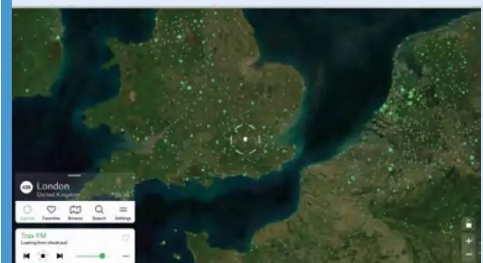
Rig up a radio



01 Having sourced a lovely vintage radio through eBay, Raju replaced the old backing board and fitted new rotary encoders and a Raspberry Pi and AudioDAC.



02 He used the preconfigured version of Raspberry Pi OS Lite he found on Comitup to launch an automatic wireless connection: magpi.cc/comitup



03 NodeJS controls backend functions and serves up static images for the display. Raju's build details are at magpi.cc/worldradio.

Confetti Bot

Want a party to go off with a bang? Then you want a device that fires confetti, as **David Crookes** explains



Hazal Mestci

Hazal Mestci is a Turkish technologist and artist who lives in NYC. Her day-to-day life involves building fun robots, coding, drawing, reading, doing sports, and cooking.

magpi.cc/confettibot



ALERT!

Make sure you handle and use confetti cannons carefully.

▶ The motor driver is an inexpensive but important part of the build

If you're going to celebrate, then you should do so in style. So what better way than sending a stream of confetti flying through the air at a press of a button? That's the premise behind the Confetti Bot, a small cannon that shoots small pieces of coloured paper (biodegradable, of course) skywards with a decent amount of power. With a Raspberry Pi 3 B+ at its heart, it's a relatively inexpensive way to get a party started.

The device was created to mark the public launch of Viam, a software platform for smart machines. The company's project manager, Fahmina Ahmed, wired a button and a motor together to pop confetti in celebration of the milestone.

Developer advocate Hazal Mestci then suggested taking the device further. "I thought we should make this project presentable and fully functional and provide a step-by-step tutorial for Viam users to follow," Hazal says. What emerged was a device

“Fitting all the components inside was the hardest part”

that uses Raspberry Pi 3 B+ and an L298N motor driver to turn a GPIO motor when the button is pressed to trigger the release of confetti. "I chose to use Raspberry Pi due to its versatility and ability to function as a compact and affordable computing platform," Hazal says. "It provided the necessary computing power to control the button, motor and other hardware components involved, and its GPIO allowed me to program and control the actions of the Confetti Bot."

Tight squeeze

The device is connected to a 12V battery to make it portable, and the components are placed within an enclosure designed with the assistance of senior robotics engineer Jeremy Hyde using the 3D CAD package SolidWorks. "Fitting all the components inside was the hardest part," Hazal says. "I had to design multiple versions of the enclosure and test the prototype many times. I was also unaware that

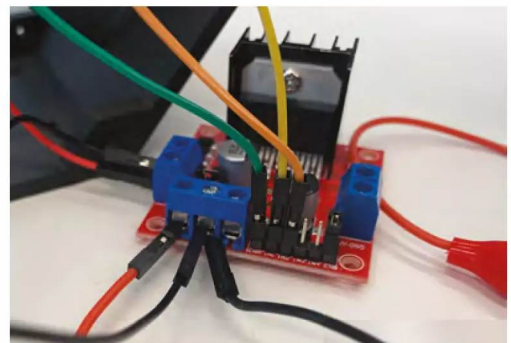
all confetti cannons are not the same size so, even between brands, sometimes the clasp wouldn't hold the cannon strongly and the cannon would turn with the motor without popping anything. The cables to the battery are still hanging out, so hopefully a second version will enclose everything."

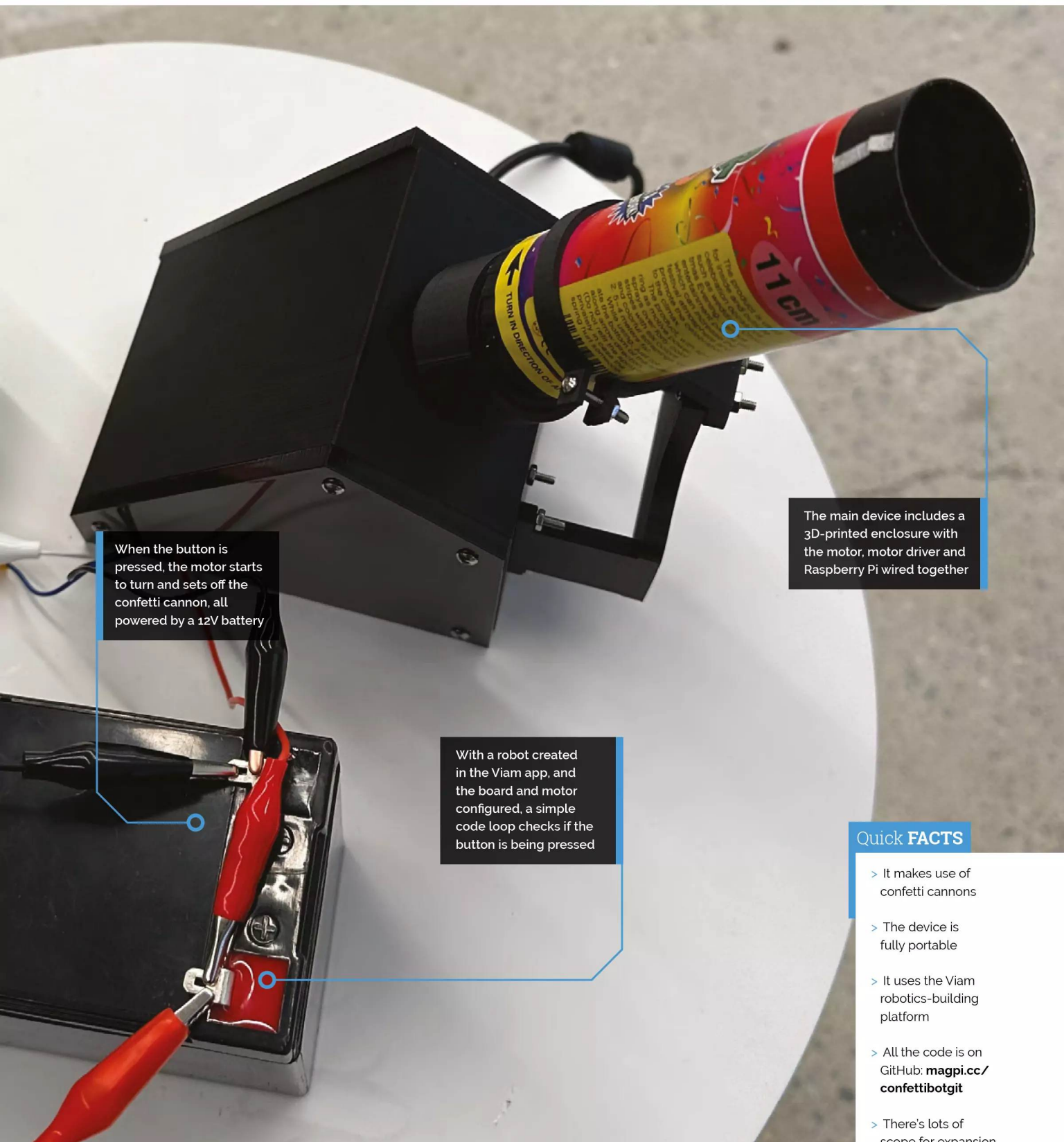
Party time

The software was written in Python, making use of the Viam Python SDK. "Viam's compatibility with a variety of hardware components allowed seamless integration," Hazal explains. "I didn't have to write backend code to activate the specific motor I was using, and I didn't have to understand the nits and grits of computing boards to be able to get a reading from a pin.

"Thanks to the Python SDK, most of the boilerplate code was already dynamically created for me. In the Viam app, as you add your components to your config, the Code Sample tab automatically gives you working code to get your resources. So it initiated the board and the motor as I added them, and imported all relevant libraries for me."

Although the Confetti Bot works well as it is, Hazal says it can certainly be improved, and she has one particular idea in mind: "I want to incorporate a camera and machine learning capabilities so that it can automatically release confetti when a person approaches my desk," she says. "It will surprise them on a good day and scare them on a bad one." **M**





When the button is pressed, the motor starts to turn and sets off the confetti cannon, all powered by a 12V battery

The main device includes a 3D-printed enclosure with the motor, motor driver and Raspberry Pi wired together

With a robot created in the Viam app, and the board and motor configured, a simple code loop checks if the button is being pressed

Quick FACTS

- > It makes use of confetti cannons
- > The device is fully portable
- > It uses the Viam robotics-building platform
- > All the code is on GitHub: magpi.cc/confettibotgit
- > There's lots of scope for expansion

Pico Bubble Machine

A Syrian academic uses MicroPython to teach kids the joy of coding. By **Rosie Hattersley**



Aula Jazmati

Aula has a doctorate in electronic engineering and uses her Raspberry Pi projects to inspire other makers

magpi.cc/aulaj

Syrrian electronic engineer Aula Jazmati was amazed when she encountered her first Raspberry Pi in 2014. “It was like a miracle in a country suffering from difficult circumstances”, so much so that Aula set about sharing the joy with students and children as a means to inspire their love of making. Many Raspberry Pi and Hexabit projects later, Aula decided to take on a more whimsical creation – a Pico Bubble Machine – figuring “soap bubbles have always fascinated children and adults alike. They bring joy and excitement, creating beautiful floating spheres that reflect light in mesmerising colours.” This DIY project also offers an exciting opportunity for children to learn about electronics.

At its simplest, washing up liquid and a loop in which to form bubbles is all that’s needed to create this low-cost and playful distraction. But having set a precedent delighting pupils with her builds and science experiments, Aula was determined to design something that would catch their imagination while also encouraging them to build their own bubble machine. “With this project, children can explore the science behind soap bubbles while having fun building their own machine”, she reasons.

An amazing mentor

Aula has been using Raspberry Pi as a teaching resource in her lab since 2014. She first



▲ The water bath and motor prior to attaching Raspberry Pi Pico

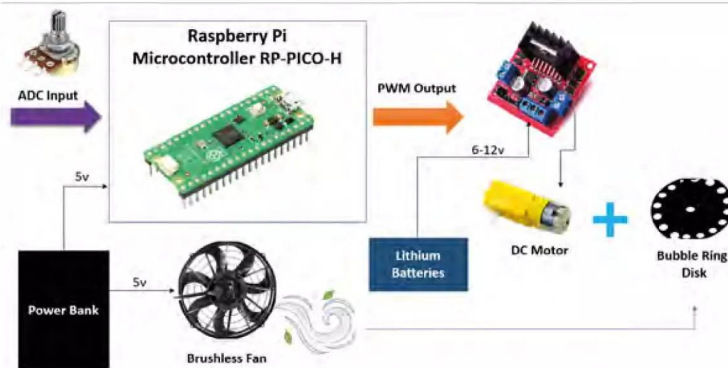
encountered the low-cost computer when on the hunt for a good value processor with an easy-to-read temperature sensor inside for use in thermal studies. “Thus began my friendship with the wonderful Raspberry Pi board, and I transferred the energy of love and optimism that the Raspberry Pi gave me to my students and everyone around me.” She particularly appreciates the open-source approach, and the attitude that science is for everyone from a community that ensures others can research and overcome any technical difficulties they encounter.

Spinning things out

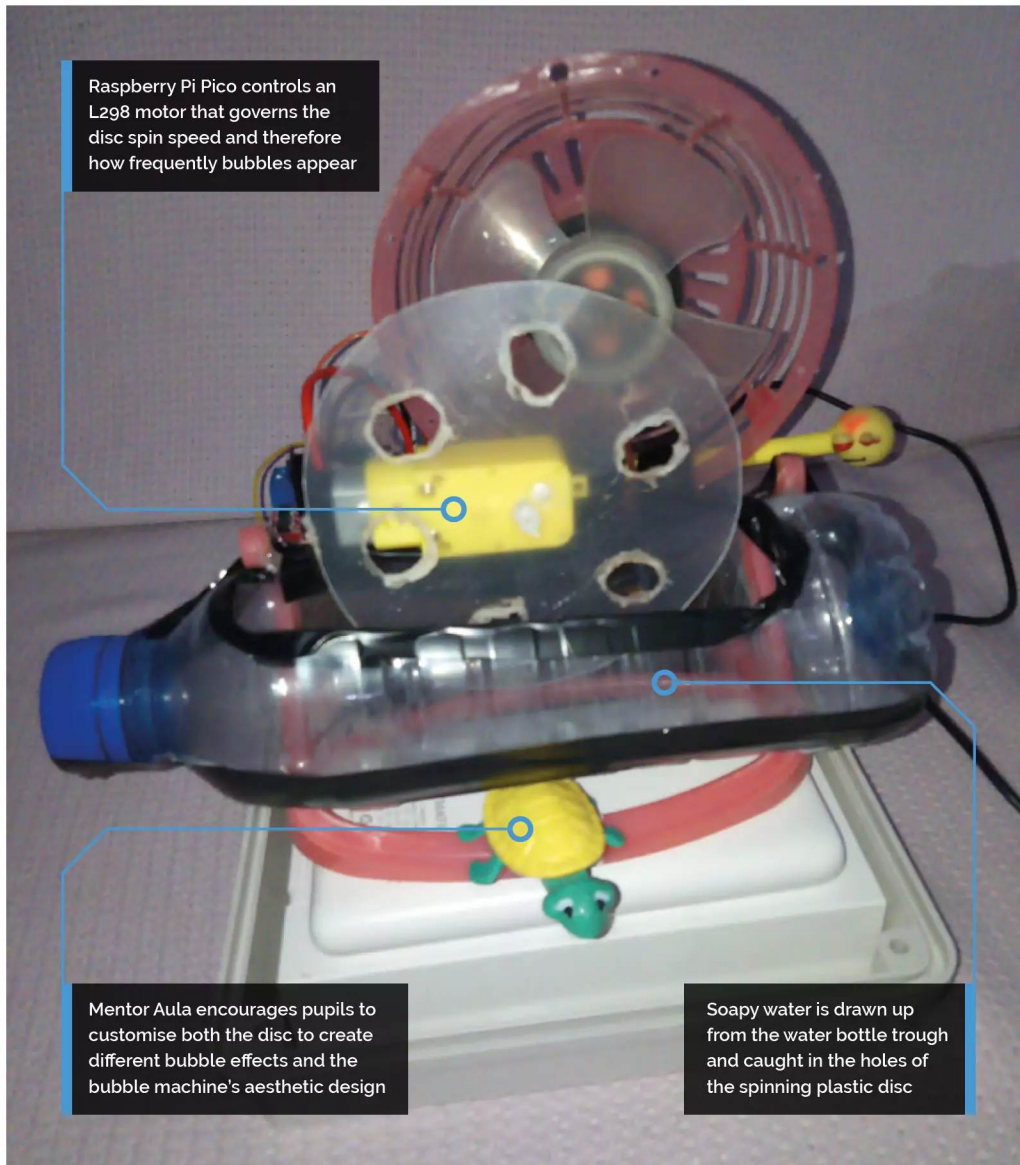
This project introduces children to basic electronics and encourages their creativity with options to customise their bubble machine, designing and decorating their own plastic discs or experiment with different fan sizes for varying bubble sizes. By programming the Raspberry Pi Pico, children can learn about coding concepts such as loops, conditionals, and functions.

The heart of Aula’s bubble machine is its perforated circular plastic disc, which performs the same role as a bubble wand. When the disc rotates, it dips into a soap solution, creating a film of soap on its surface, she explains, while air from the fan blows through the holes, forming bubbles that detach from the disc and float away. A DC motor connected to an L298 driver circuit controls both

▼ This diagram shows how Pico is used to control the motor and rotating disc



DIY Bubble Machine Block Diagram




Raspberry Pi Pico controls an L298 motor that governs the disc spin speed and therefore how frequently bubbles appear

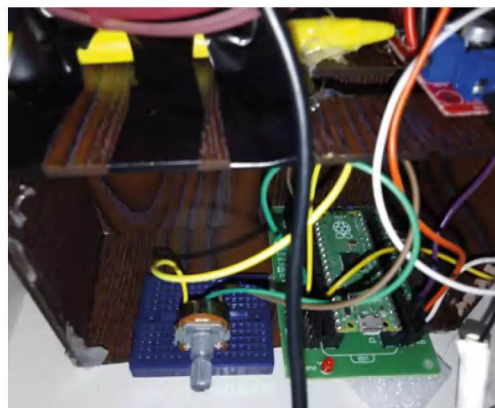
Mentor Aula encourages pupils to customise both the disc to create different bubble effects and the bubble machine's aesthetic design

Soapy water is drawn up from the water bottle trough and caught in the holes of the spinning plastic disc

“ Soap bubbles have always fascinated children and adults alike ”

the fan speed and disc rotation. These signals are provided by Raspberry Pi Pico and a potentiometer connected to it by an ADC pin.

“Raspberry Pi Pico controls when to start and stop the motor, as well as adjusting its speed, and was the ideal choice”, says Aula. It allows her to calibrate the speed at which the disc releases the bubbles and helps to optimise energy consumption so both the battery and kids’ enjoyment lasts as long as possible. 



▲ Raspberry Pi Pico and motor hooked up and ready for Aula to test them

Quick FACTS

- > Aula made a point of upcycling materials for the project
- > Reusing plastic tools and repurposing electronic components
- > Though you could design and 3D print your own
- > Aula loves the Raspberry Pi community
- > And is a long-term translation volunteer for it

rePalm

You've got to hand it to Dmitry Grinberg – squeezing PalmOS 5 onto Raspberry Pi Pico is no mean feat, as **David Crookes** discovers



Dmitry Grinberg

Dmitry is a tinkerer from Austin, TX who specialises in squeezing every cycle out of microcontrollers

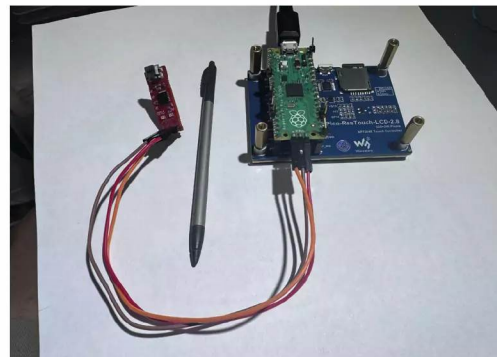
magpi.cc/repalm

You can't run Raspberry Pi OS on a RP2040 microcontroller, but that doesn't mean Raspberry Pi Pico is too underpowered for an operating system. As Dmitry Grinberg has shown, it's capable of running a port of the classic PalmOS 5 – the mobile operating system unveiled in 2002 – although it's not been without considerable effort.

PalmOS 5 was the first version of Palm's operating system to support ARM devices. Before that, the OS, which dates back to 1996, ran on the Motorola 68K processor line. "The move to ARM was a bit more interesting," Dmitry says. "There were rumours about the large internal re-architecting of the OS at the time but no details were known. In the end, the ACCESS company of Japan bought the OS and that was that. The source code was never released."

With an ongoing fascination with PalmOS – "it was the first mobile OS with any sort of success and it gave the world the term 'smartphone'" – Dmitry says he wanted to see if he could put the operating system on another device with minimal changes. He looked into emulating the hardware with success (check it out at magpi.cc/uarm). "But that was cheating," he continues. "The goal was to port it to new hardware, which is usually hard without source code, but I figured I'd try."

"I knew that, in theory, most of the drivers were contained in the Device Abstraction Layer (DAL) module, with a few others being separate (SD card, serial port, DMA). I figured that if I managed to replace those modules with some I wrote myself, that the OS should, again in theory, run on any ARM device with a compatible processor – any processor supporting ARMv4T or ARMv5T architecture." This tech adventure spanned multiple years, but the effort has certainly been worthwhile.



▲ Raspberry Pi Pico connected to the CortexProg (cortexprog.com), a Cortex-M debugger Dmitry created to read and write data from and to a microcontroller

"It culminated in the release of the source earlier this year, and now one can bring up PalmOS even on processors that it never actually supported, like tiny Cortex-M0 cores," Dmitry continues. Having tried numerous other boards from STMicroelectronics' STM32F429 to the STM32H7, RP2040 showed the most promise. "I had never encountered a Cortex-M0 microcontroller that had enough RAM for PalmOS," he says. "RP2040 does – barely – so I wanted to try."

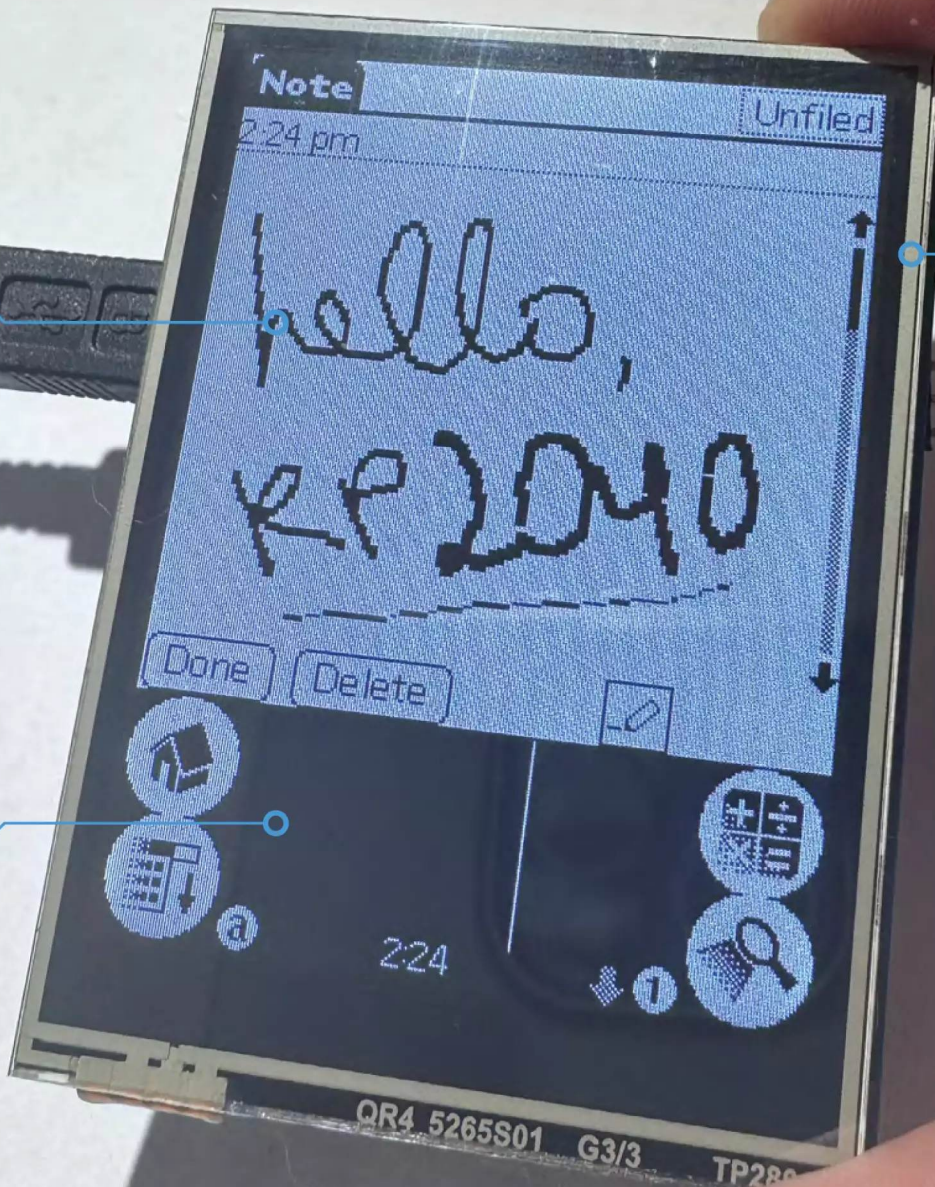
Screen time

The idea was to get PalmOS 5 running on a Raspberry Pi Pico or other RP2040-based development board and connect the device to a suitable touchscreen. Dmitry says the only one that fit his requirements was a low-cost \$15.99, 2.8-inch touchscreen display from Waveshare offering a resolution of 320 × 240px.

"It is a resolution that PalmOS supports well and really this is about the applications – the OS itself does not care about screen resolution," he says.

RP2040 is able to run an unmodified PalmOS 5.2.8 but only after a lot of hard work

Dmitry hasn't gone to the bother of creating a case. "Really, the PS2040 PalmOS is more me showing off," he says



Capable of displaying 262,000 colours, this Waveshare 2.8-inch touchscreen display is nevertheless being used to output in greyscale

Quick FACTS

- ▶ The project ports PalmOS 5 to Raspberry Pi Pico
- ▶ It uses a custom kernel to run on Cortex-M processors
- ▶ rePalm can be replicated for less than \$20
- ▶ The biggest problem is a lack of RAM
- ▶ The software can be downloaded from Dmitry's website

▶ Raspberry Pi Pico is mounted on the back of the touchscreen display. As Dmitry says, a \$1 microcontroller is being used to replicate what was once a \$400 device



“Applications work best at resolutions that are integer multiples of 160×160 , and extra space can be used for the dynamic input area. A 320×240 resolution is not super common but is supported relatively well.”

Dmitry says it helped that the screen was easy to purchase. “It is a pet peeve of mine when people publish a project and their parts list includes ‘that one thing I bought at a garage sale once’ or ‘an old tube my grandpa left me in his will’,” he explains. But it was also important to have a resistive touchscreen.

“Devices today are designed for capacitive touch, hence the large touch targets,” he says. “PalmOS predates cheap capacitive touch so it is designed for resistive touch screens which needed quite a bit of pressure to work, hence the fact that styluses had a fine tip. The fine tip

also allowed the thing being tapped to be seen, while fingers are relatively large and obscure what you tap.”

Ramming in RAM

The more Dmitry used RP2040, the more he realised it was a good fit for the project. “First, the PIO system is very versatile, allowing me to pretend to be a RAM and a ROM to the host 68k processor (in the case of PalmCard) while also driving my display with no CPU cycles needed. No other micro out there has anything quite like it,” he says.

“Second, the performance is good. Cortex-M0 may not be a very performant core per-cycles, but the fact that I can easily run it at 200MHz+ surely papers over a lot of that! Third, the documentation does not suck, and this cannot be overstated. Whoever wrote the RP2040 docs needs a medal, a bonus or a hug because it is clear and accurate.” [That would be Alasdair Allan – Ed].

But that’s not to say Pico is perfect. “There is barely enough RAM to turn around in, so it’s not a practical PalmOS device,” he laments. “PalmOS 5 needs at least 64KB of RAM for the storage heap where the databases are stored and at least 128KB of dynamic heap, where temporary allocations come from, to boot.

“My kernel and DAL need about 30KB for itself and we also need to fit at least one full frame buffer [screen image] into the RAM. At $320 \times 240 \times 2$ bits per pixel, the framebuffer is almost 19KB. Adding that up gives us 241KB. Now, RP2040 has

▶ Dmitry has also found a way to wire up external RAM to the RP2040 in a way that it’s writable, readable, and cached (magpi.cc/romram)





256KB of RAM – the other pieces are not easily usable for various reasons. This is tight, but it fits. That 30KB was after a lot of extra work I put in to shrink that footprint. Initially, it was closer to 100KB.”

Looking ahead

This all means PalmOS 5 just about barely fits on Raspberry Pi Pico and, once it boots, only basic programs run with a few KB of RAM left over for a memo pad note or two. There’s not enough RAM to load a game, and Dmitry would love an RP2040 with more RAM. “It would work better,” he says.

“ There is barely enough RAM to turn around in, so it's not a practical PalmOS device ”

Still, rePalm is a huge feat that deserves a lot of applause. Hours and hours of disassembly went into its creation and using it shows how well created PalmOS was. “The rules were simple: any common activity should be no more than two device interactions away,” Dmitry says. “People who want a modern PalmOS device don’t seek to replace their phone. They seek to escape the constant advertisements and distractions.”

With rePalm, he is showing once more what is possible. “Current plans include making a PalmOS-based smartwatch as well as a new PalmOS-based device,” he adds.

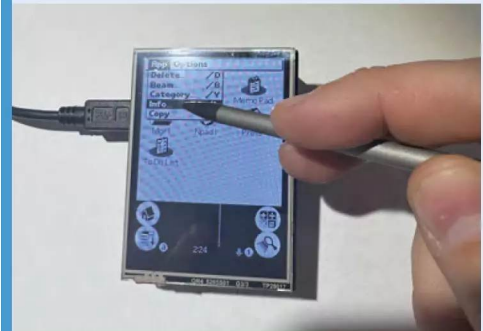
Why? “Just because.”

▲ The rePalm device looks very much at home here. It can force-pause threats, disable task switching, and supports local thread storage

Getting to grips with PalmOS



01 The project began as an experiment. Dmitry wanted to figure out how PalmOS 5 was built and how it worked. Other than replacing the DAL module, the rest of the ROM image that runs is the unmodified PalmOS.



02 Every feature that existed in PalmOS is available – including every bug – and that makes the build great for software compatibility, Dmitry says. The OS can be navigated using a stylus, just as Palm devices were back in the day.



03 Dmitry has considered using Raspberry Pi Pico W: “But to use Bluetooth, you’d need to send or receive things and to use Wi-Fi, you’d want a web browser, email client or chat app. There is not enough RAM for that.”

Badger and Fox Deterrent

Scaring off unwelcome garden visitors with a nocturnal scarecrow simply screamed Raspberry Pi project for one maker, learns **Rosie Hattersley**



Warning!
Working with animals

Please be mindful when creating devices that interact with animals. Read the RSPCA's guide on deterrents and animal population control.

magpi.cc/rspcadeterrents



James Milward

James is a software engineer for a company that manages data and uses AI and enjoys 3D printing and creating electronic projects

medium.com/@james.milward

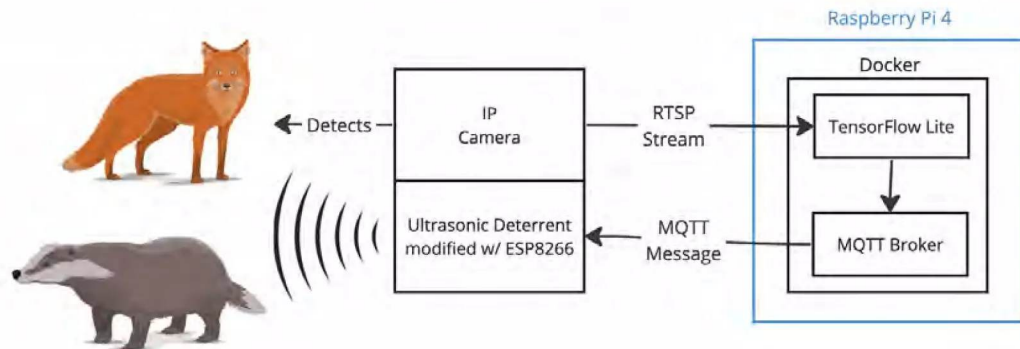
Moving house is stressful enough; unpacking only to discover your new garden is favoured by unexpected local wildlife that doesn't respond to unsubtle hints is unsettling too. Software engineer James Milward decided on a technology-based approach to tackling his night-time visitors, creating a Badger and Fox Deterrent based around Raspberry Pi 4 and some machine learning. With a day job involving data and AI, James already had a fair clue of how to go about it.

He had a number of Raspberry Pis at home and built a solar-powered watering system using Raspberry Pi A+ back in 2015. He says the low-powered footprint paired with the

significant compute capabilities of Raspberry Pi 4 prompted him to explore its machine learning potential at home.

Inaudible alarm bells

Having moved house in 2022, James and his family quickly discovered an issue with their new home: local foxes and badgers "loved ripping the garden up, digging deep holes and leaving 'gifts' everywhere for us". Most disturbingly, they brought in insects carrying who knew what in terms of germs and potentially harmful diseases. Once local wildlife experts confirmed these fleas and ticks were definitely from their garden visitors, James was "set on a



▶ Different frequencies warn off foxes and badgers, respectively



James repurposed an ultrasonic wildlife deterrent to automatically detect different types of nocturnal visitor

He used TensorFlow Lite machine learning on Raspberry Pi to detect foxes and badgers, but trained it to ignore other moving objects

Footage from the wireless deterrent can be viewed remotely, while unwelcome visitors get a warning audible only to them

Quick FACTS

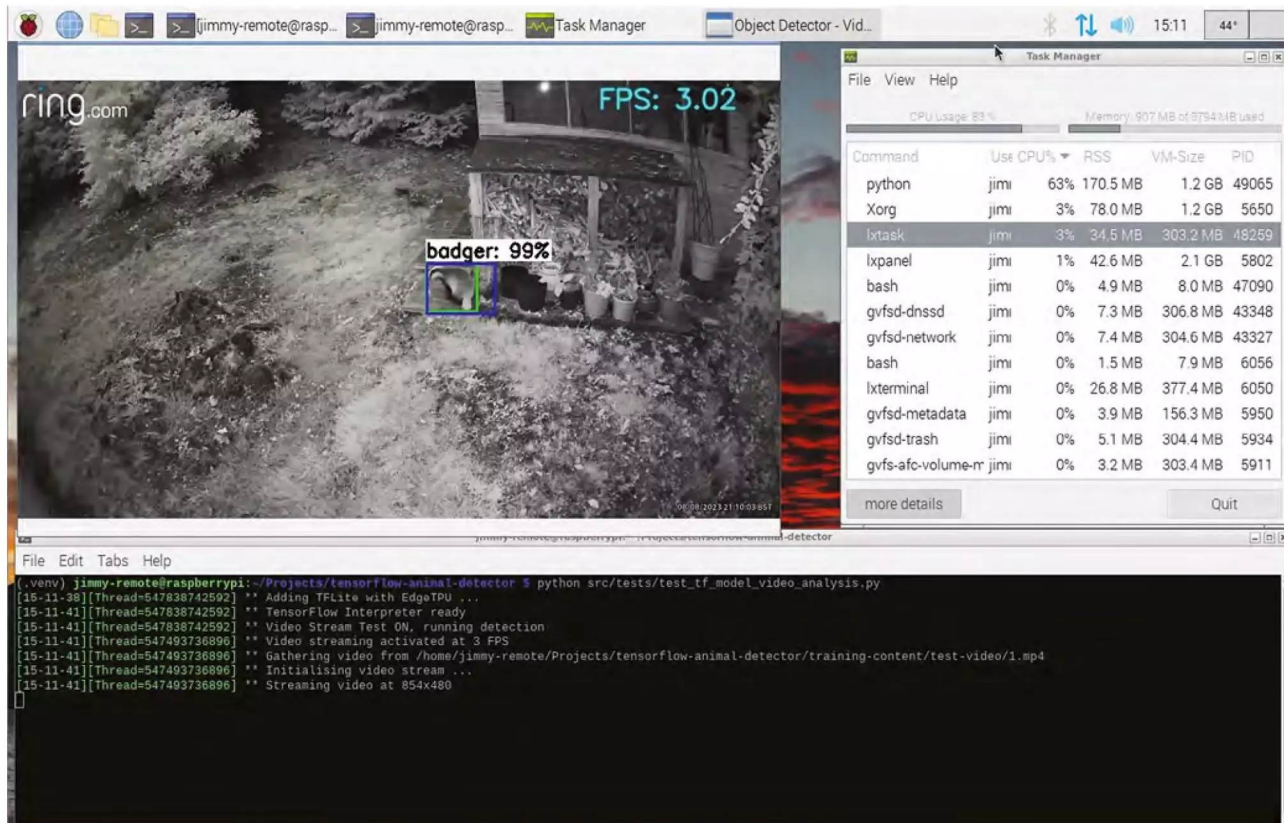
- ▶ TensorFlow Lite draws less than 6W on Raspberry Pi
- ▶ Even while actively processing images and cross-checking them
- ▶ The challenge for James was implementing this on night images
- ▶ Extracting usable, accurate video footage across a large area
- ▶ And at more than two frames per second

path to create a deterrent that could help with our problem”.

He initially tried scent-marking repellent and natural mixes of citronella and chilli, blocking entry points along the fences in the garden, but “This just resulted in our fluffy friends ignoring my initial attempts and digging more holes”. He tried one solution that offered a glimmer of hope: solar-powered ultrasonic repellents. The Milwards duly placed a few around their garden. However, the repellents required manual setting depending on whether it was badgers or foxes they wanted to warn off, as well as deactivating so as not to trigger when the family was in the garden. This led to occasions when no ultrasonic repellent was in use, and a further incursion of wildlife. Combined with inconsistent charging of the solar-powered units, James realised he needed something smarter:



▲ The TensorFlow Lite system successfully recognises a visiting badger exploring James' garden



▲ A Ring doorbell camera added to the setup also triggers an alarm

“ Raspberry Pi was the perfect choice to add some intelligence to these very basic devices ”

Raspberry Pi would be the perfect choice to help add some intelligence to these very basic devices.

Tense, nervous, headache

James had already experimented with TensorFlow and figured the combination of this open-source machine learning platform’s object recognition capabilities and Raspberry Pi could work well for his Fox and Badger Deterrent, modifying the ultrasonic repellent for remote use. By adding wireless connectivity and an ESP8266 microprocessor, he would even be able to have the repellent operate at a targeted frequency range to “accurately deter targets”.

The project’s real headache came when trying to get video frame rates good enough for object detection at a



▶ The fox and badger detector machine learning to detect the two animals



◀ The machine learning model identifying a fox

resolution of 640×640 pixels. James used existing YouTube tutorials designed for 320×320 models, but soon found that more detailed images would be needed, and set about reducing the complexity of the model without losing too much accuracy. Unfortunately, this method brought the frame rate up to just under 1FPS, which was fine for a single video stream, but James had built the code to work on multiple cameras at once. He then discovered that the Coral USB Accelerator Edge TPU works superbly with Raspberry Pi 4, which has the necessary USB 3.0 ports. “I was able to use this device to get the frame rate for object detection up to 5FPS for one video feed and about 2FPS for two feeds running concurrently.” [M](#)

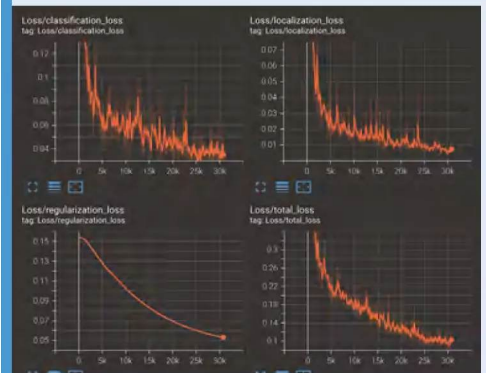
▼ A badger detected around the flower pots in the garden



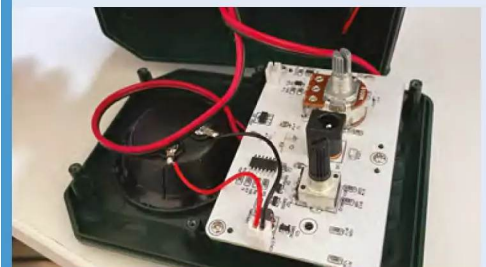
Scare the cat!



01 You will need at least one solar ultrasonic repellent device, Raspberry Pi 4, a Ring camera or compatible RTSP webcam, Arduino ESP2866, a wireless connection and power supply. James details the build at magpi.cc/deterfox.



02 Install and configure James' Python application onto Raspberry Pi 4 or 5 alongside a Coral USB3 Edge TPU (Tensor Processing Unit).



03 Modify the ultrasonic deterrent following James' GitHub, which has pretrained the object detection module to discern badgers and foxes (magpi.cc/tfanimal). You will also need a housing for the device and a means of mounting it.

Tide Clock

How do you know the tide is right for a walk? If you're Levi Maai, you create a special - and very pretty - tide clock. **Rob Zwetsloot** braves the seas for a look



Levi Maai

MAKER

A researcher, film-maker, and educator, Levi likes to tinker with tech and learn about the people who tinker with it too

pathwaysto invention.org

If you've ever lived near the sea, you'll know that it never gets old to walk along the beach, especially when the weather is good.

For Levi Maai, it turned out the weather wasn't the main problem.

"I live 200 feet above the Pacific on an ocean cliff," Levi explains. "I try to make it a weekly ritual to walk down the cliff to the shore. However, it's disappointing to reach the seaside stairs, expecting a nice sandy walk, only to find waves lapping against the base of the bluff."

For us landlubbers, tides aren't always something we have to think about, although Levi was familiar with the tide clocks where he grew up. Not being able to find one for his new Californian home, Levi decided to make his own: "It displays current tide height in feet, predicted hours until next tide and predicted height of the next tide using analogue meters and LED lights."

Tidal changes

Making a clock wasn't quite as easy as he first thought, though.

"Growing up in Rhode Island, round, four-segmented tide clocks were a common sight in homes near the Atlantic," Levi tells us.

"But after moving to California, I noticed that these simple analogue clocks were nowhere to

be found here. As it turns out, there's a good reason for that. Similar to many coastal areas around the Pacific and some of the northern Mediterranean coast, the US west coast has a more complex tidal pattern, which means that without constant adjustment, a basic tide clock would get out of sync with the 'mixed semi-diurnal' tides here in just a few days."

This meant Levi needed a way to keep the clock up to date with the NOAA (National Oceanic and Atmospheric Administration) tide data, which led him to turn to Raspberry Pi. "The device stays

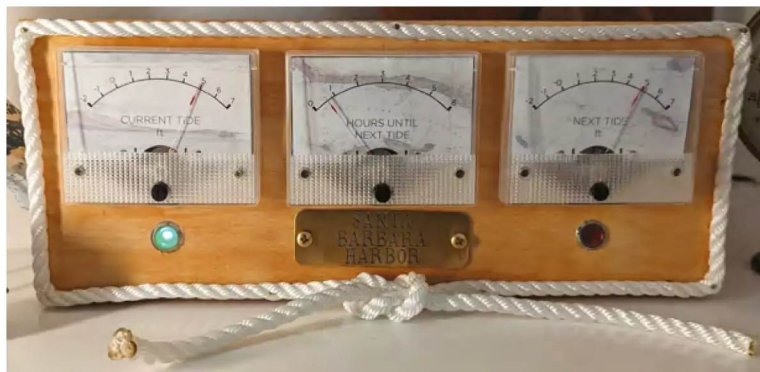
“ All of the meters and lights are driven by the GPIO pins on a Raspberry Pi 3B+ running a Python script that gathers tide predictions ”

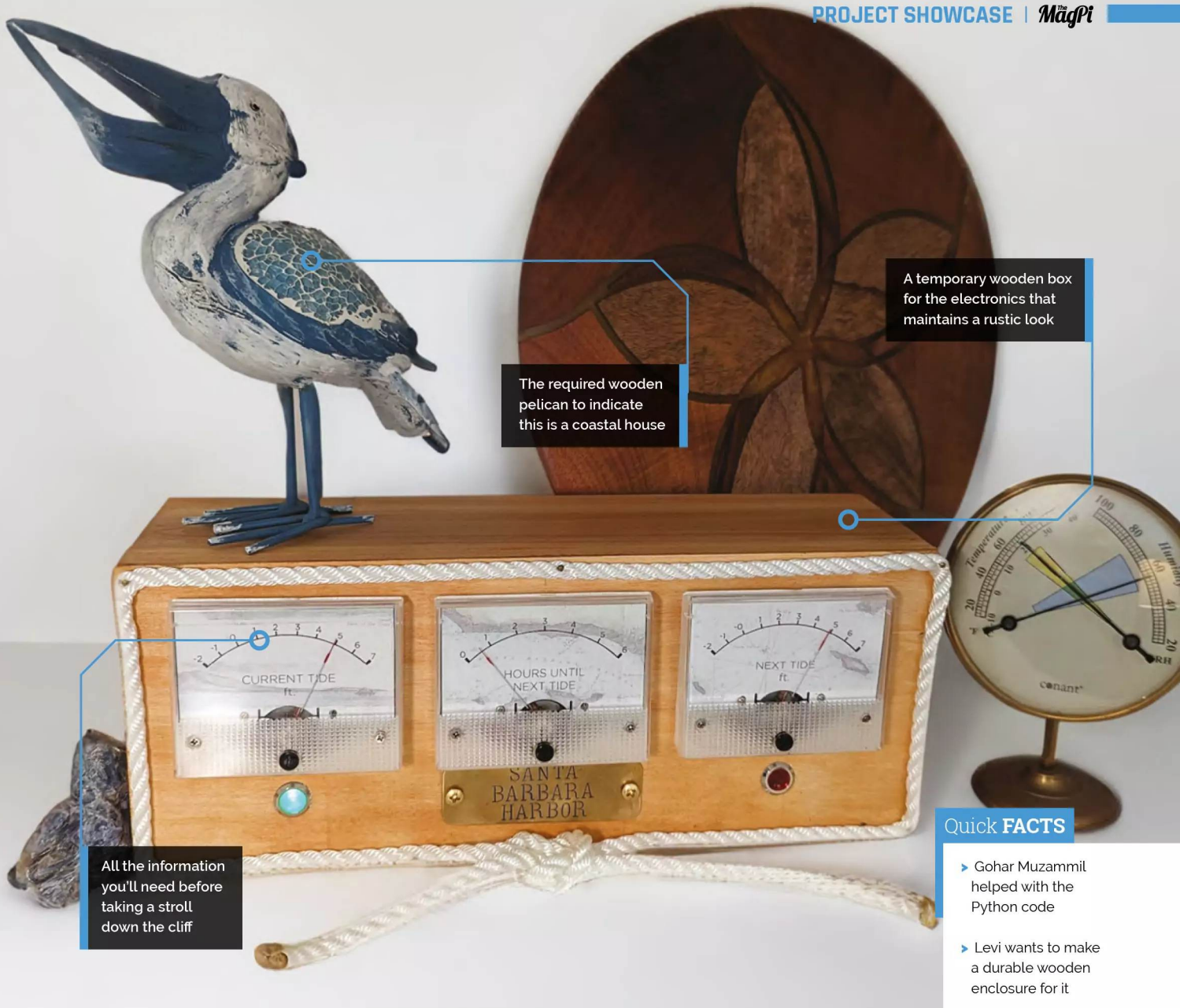
accurate as long as it has a Wi-Fi connection," Levi says. "The red light on the right side of the front panel indicates a rising tide and the green light on the left illuminates when the tide is falling. All of the meters and lights are driven by the GPIO pins on a Raspberry Pi 3B+ running a Python script that gathers tide predictions from the US National Oceanic and Atmospheric Administration's API. The script converts the raw tidal data into PWM values to drive the meters and the binary state voltage for the LEDs."

Sea worthy

As well as the electronic aspect of the project, Levi was keen to make it look more rustic to camouflage the internal tech, using analogue gauges and a wooden fascia. "Obviously there

▼ The LEDs let you know if it's a good time to head out, with green as the usual 'go' signal

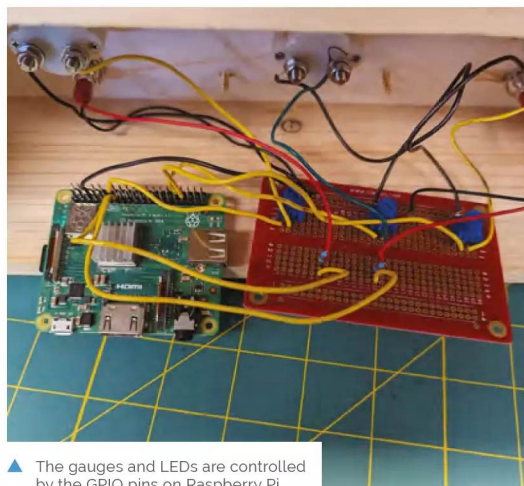




are more efficient ways to communicate tide predictions than with analogue meters,” Levi admits. “A digital display driven by a Raspberry Pi could easily show graphs and numbers based on the tide API. But I wanted something more rustic but just as accurate. They say measure twice and cut once. I must have tinkered mentally with the physical build design a thousand times over the past two years.

“So when it finally came time to assemble it, I had a very clear vision of how it should look, down to the brass plaque declaring that the information displayed was for ‘Santa Barbara Harbor’.”

According to Levi the finished project works ‘really well’, which hopefully means he’s been able to go for his weekly walks without worrying about the tide since completing it. [M](#)



▲ The gauges and LEDs are controlled by the GPIO pins on Raspberry Pi

A temporary wooden box for the electronics that maintains a rustic look

The required wooden pelican to indicate this is a coastal house

All the information you'll need before taking a stroll down the cliff

Quick FACTS

- ▶ Gohar Muzammil helped with the Python code
- ▶ Levi wants to make a durable wooden enclosure for it
- ▶ A Raspberry Pi Zero was tried originally...
- ▶ ...however the needles were a bit jittery, possibly due to lack of processing power
- ▶ The API uses live data on tide height for the harbour

Pihome

Save money on energy bills using Raspberry Pi Pico? It's more likely than you think. **Rob Zwetsloot** takes a reading.



Thomas Ackermann

A software engineer since the '90s, Thomas started his career writing C on Unix systems.

magpi.cc/pihome

In the current climate, saving money on heating is something a lot of people are interested in doing, so it's no surprise that someone decided to tackle this head on using Raspberry Pi Pico W.

"I'm using temperature and humidity trend data for optimising my heating systems and monitoring individual room temperatures and humidity," Thomas Ackermann tells us. He lives in Bavaria, which has a similar climate to the UK with distinct seasons and a temperature ranging from freezing in the winter to too hot (25–30 °C) in the summer. "Up until now I have had three systems running: two systems for room monitoring which I am using for rooms which are not occupied currently and need monitoring for deep temperatures and high humidity in the winter. The other system is for optimising an old oil heating system. The sensors are attached to the piping and so indicate when the individual pumps for heating, hot water and hot water circulation, and the oil burner, are active."

Thomas began this project with the goal of refreshing his Unix/Linux skills and learning Python on Raspberry Pi – and has ended up with a saving of about €700 (£609/\$741) a year on oil.

Efficient code

While initial work was done on Raspberry Pi, Thomas switched to Pico for the project.

"Raspberry Pi Pico W is used to collect temperature and humidity data sent by ThermoBeacons and transmits it to a cloud application for further analysis," Thomas says. "Actual data is additionally shown on a local display and provided by a local web server running on Pico W."

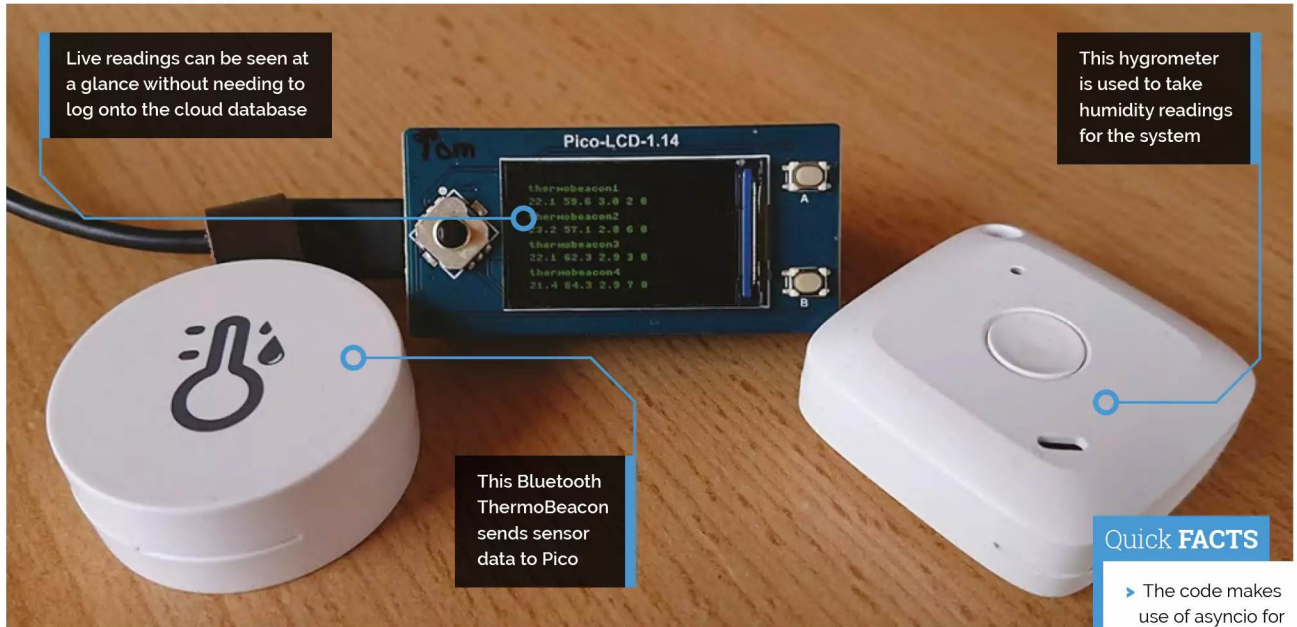
Thomas had to do some serious coding wizardry to get everything to run properly on Pico – including tracking down a memory leak in MicroPython.

"I had to extend the original Pico LCD driver to support a configurable frame buffer," explains

"This helped to identify a useless circulation pump which could be disconnected and so power could be saved"



▶ The hygrometer is hidden away by some pipes to take readings



Thomas. “This is used to have enough free memory for the https communication to the cloud service”

You can get the full code at magpi.cc/pihome

What a savings

With the data collected from his sensors, Thomas was able to make some interesting discoveries.

“This helped to identify a useless circulation pump which could be disconnected and so power could be saved,” Thomas says. “It was also used to track down a problem with an incorrect hot water temperature measurement in the

heating system; the real temperature was 5°C higher than the temperature measured from the heating controller and so the hot water pump was activated too often. Finally it was used to measure the effect of added insulation. The burner temperatures are significantly lower with the new insulation.”

Now that Thomas has taken his project this far, he’s keen to add more Bluetooth beacons and more types of sensor to the overall system. Here’s hoping it helps him save even more money... [M](#)



◀ You can get readings from Picos dotted around the house

Bullet Time wedding photo booth

One maker used a dozen cameras and Raspberry Pi Pico to build the ultimate action-packed photo booth. By **Ashley Whittaker**



Sebastian Staacks

Sebastian is a physicist and the developer of phyphox (phyphox.org). He is a father of two, a drummer, coder, and maker.

staacks.de

▼ The green button is used to start the recording countdown and to confirm that a recording should be kept when the guests review their result. The red button discards the recording



If you're already tiring of the wedding season, this project isn't going to make you feel any better. Sebastian Staacks has out-gifted us all. No serving spoon sets or gift vouchers here. For his cousin's wedding he made a bullet time photo booth using an array of DSLR cameras triggered by Raspberry Pi Picos.

What does bullet time mean?

Bullet time photo booths comprise multiple cameras rigged together. Each camera takes a photograph at the same moment simultaneously, or with a small delay between them. Software then joins each of the photographs to create a video with the striking "bullet time" effect. The effect is best known for – and indeed named after – its use in *The Matrix* (magpi.cc/matrixbullettime) in

which the chosen one, played by Keanu Reeves, dodges bullets with an impressive backbend.

You might also have heard of it as a frozen moment, dead time, flow motion, or time slice video.

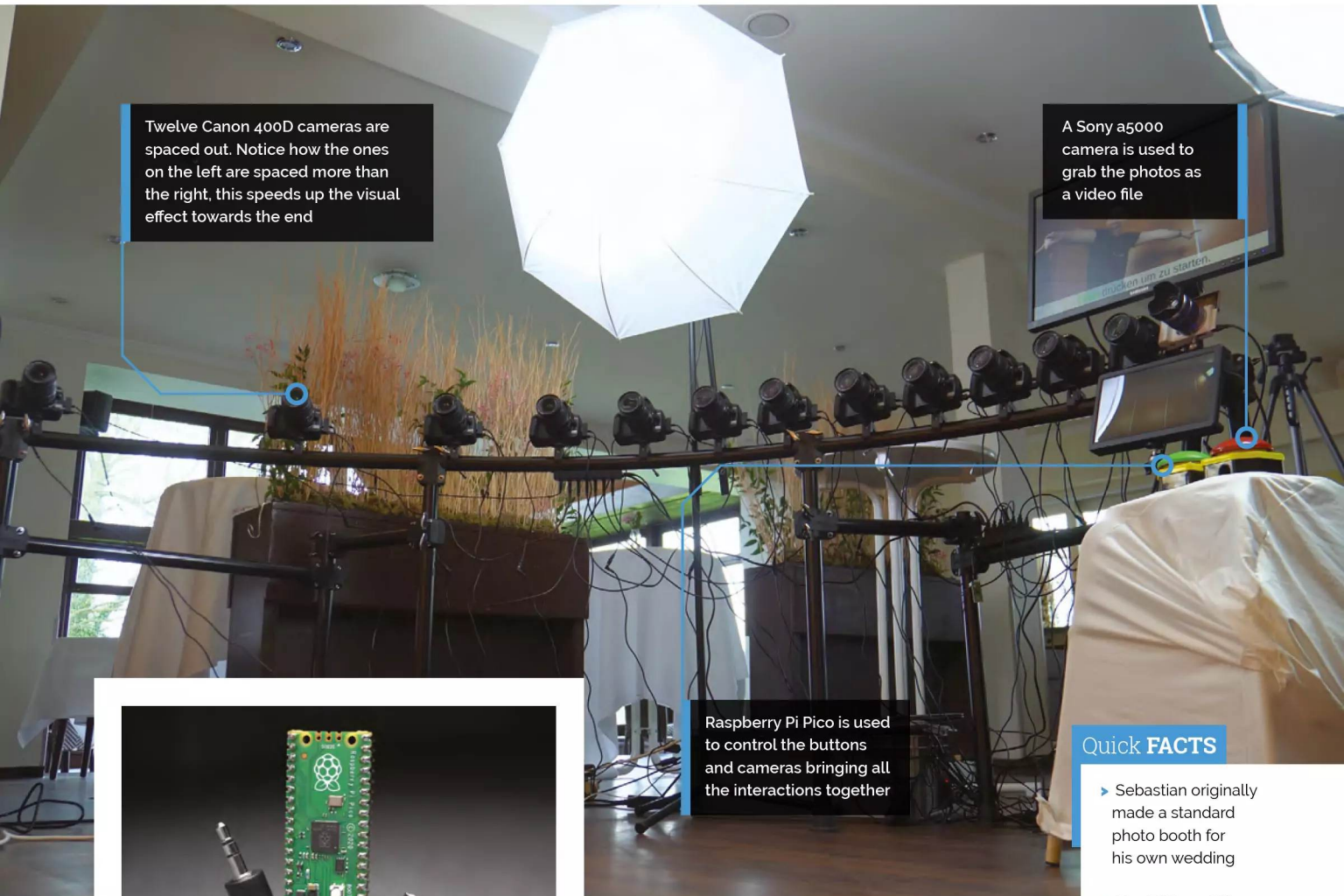
"My cousin asked me if I could build a photo booth for his wedding," explains Sebastian, "where guests could create fun memories with props like hats and giant glasses. I wanted to try something fancier. A video booth with a bullet time effect".

A three-Pico wedding

Twelve cameras are attached to a curved rig that looks a bit like a ballet barre. Look at the image and you'll notice the cameras are placed at greater distances apart towards the wall on the left, and much closer together on the other end. This achieves the impression of the action accelerating when the shots are played back in sequence.

"I set myself a limit of €50 [£43.50] per camera," says Sebastian. "which means that the 12 cameras alone would cost €600 [£521]."

Sebastian did consider Raspberry Pi Camera Module but decided on picking up extremely old DSLR cameras instead, picking "the oldest mainstream beginner DSLR that has a decent resolution". He settled on 12 10MP Canon EOS 400D cameras, which he picked up on Craigslist. A Raspberry Pi Pico (magpi.cc/pico) acts as the camera trigger. It has to make sure all the camera shutters go simultaneously in order for the bullet time effect to work. Two further Raspberry Pi Pico Ws sit inside Bluetooth wireless buttons so the wedding guests can control the action. The green button starts the countdown to a recording and saves it, and the red button can cancel a countdown, or trash it if everyone looks idiotic.



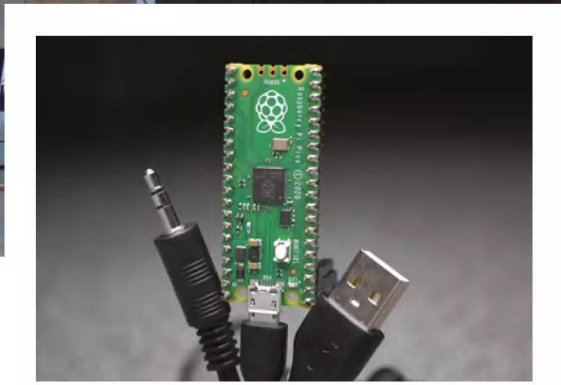
Twelve Canon 400D cameras are spaced out. Notice how the ones on the left are spaced more than the right, this speeds up the visual effect towards the end

A Sony a5000 camera is used to grab the photos as a video file

Raspberry Pi Pico is used to control the buttons and cameras bringing all the interactions together

Quick FACTS

- ▶ Sebastian originally made a standard photo booth for his own wedding
- ▶ The EOS 400D has a photo resolution of 3888x2592 pixels
- ▶ More than the 3840x2160 used in the final video
- ▶ But the EOS 400D cannot record video...
- ▶ ...so a Sony a5000 is used as a HDMI grabbing device



▲ Raspberry Pi Pico offers a serial interface via USB, which pulls either of the two signal lines of the trigger cable to ground

DaVinci Resolve software created the final video of all the shots stitched together. Its Optical Flow Estimation generated artificial additional frames, and allowed sections to be blurred or slowed down to making the stitching look more fluid.

Details, details, details

Sebastian created an incredibly detailed project page (magpi.cc/tobbullettime) delving into each element of the build, from the camera triggers to building the rig and post-processing the images. His GitHub ([magpi.cc/bullettimegit](https://github.com/magpi/bullettimegit)) repo is also a treasure trove of all the code you'll need to build your own and become the most popular wedding guest ever. Well, maybe next wedding season. [M](#)

“ Raspberry Pi Pico acts as the camera trigger ”



◀ Raspberry Pi Pico W fits into the push buttons with two AA batteries

SUBSCRIBE TODAY FOR JUST £10

Get 3 issues + FREE Pico W



Subscriber Benefits

- ▶ **FREE Delivery**
Get it fast and for FREE
- ▶ **Exclusive Offers**
Great gifts, offers, and discounts
- ▶ **Great Savings**
Save up to 23% compared to stores

Subscribe for £10

- ▶ Free Pico W
- ▶ 3 issues of The MagPi
- ▶ Priority Boarding code
- ▶ £10 (UK only)

Subscribe for 6 Months

- ▶ Free Pico W
 - ▶ 6 issues of The MagPi
 - ▶ Priority Boarding code
- | | |
|----------|---------------------|
| £30 (UK) | \$43 (USA) |
| €43 (EU) | £45 (Rest of World) |

📞 Subscribe by phone: **01293 312193**

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

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

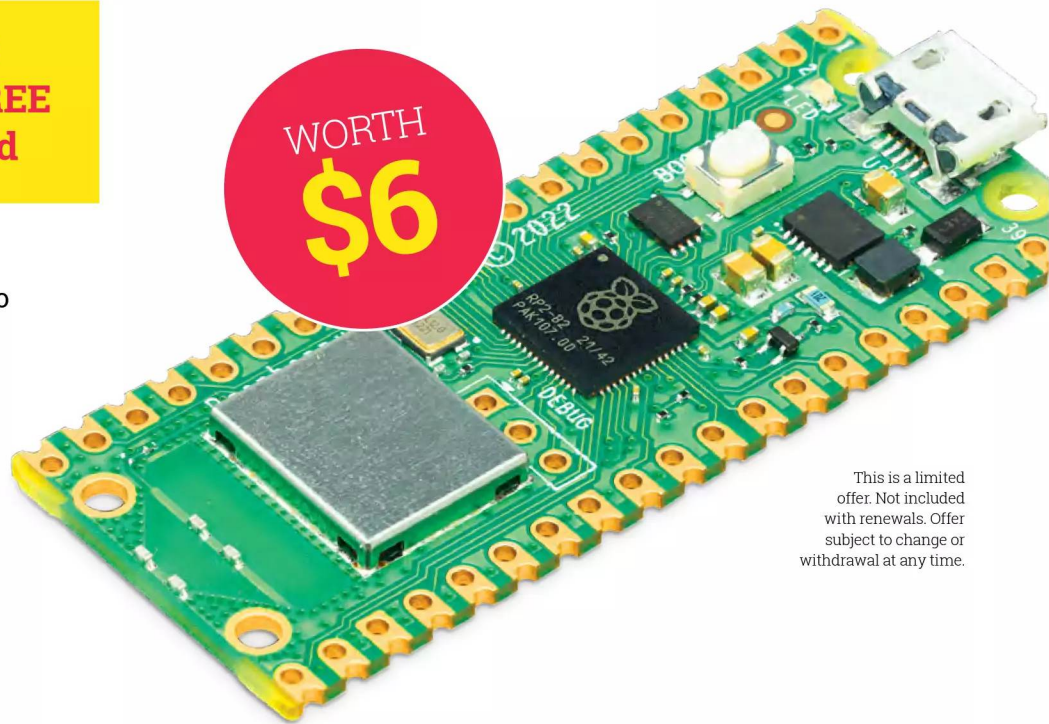
Subscribe for £10 is a UK only offer. The subscription will renew at £15 every three months unless cancelled. A free Pico W is included with a 6-month subscription in USA, Europe and Rest of World.

SUBSCRIBE TODAY AND GET A

FREE Raspberry Pi Pico W

Subscribe in print today and get a **FREE** development board

- ▶ A brand new RP2040-based Raspberry Pi Pico W development board
- ▶ Learn to code with electronics and build your own projects
- ▶ Make your own home automation projects, handheld consoles, tiny robots, and much, much more



This is a limited offer. Not included with renewals. Offer subject to change or withdrawal at any time.

Raspberry Pi **5** PRIORITY BOARDING

We've reserved Raspberry Pi 5 boards
for *The MagPi* subscribers

GET YOUR RASPBERRY PI 5 **FIRST!**

New subscribers to *The MagPi* will get a Priority Boarding code when they sign up. So don't delay. Take out a subscription today! You'll be able to buy your Raspberry Pi 5 first and get incredible Raspberry Pi tutorials, projects, and community events every month.



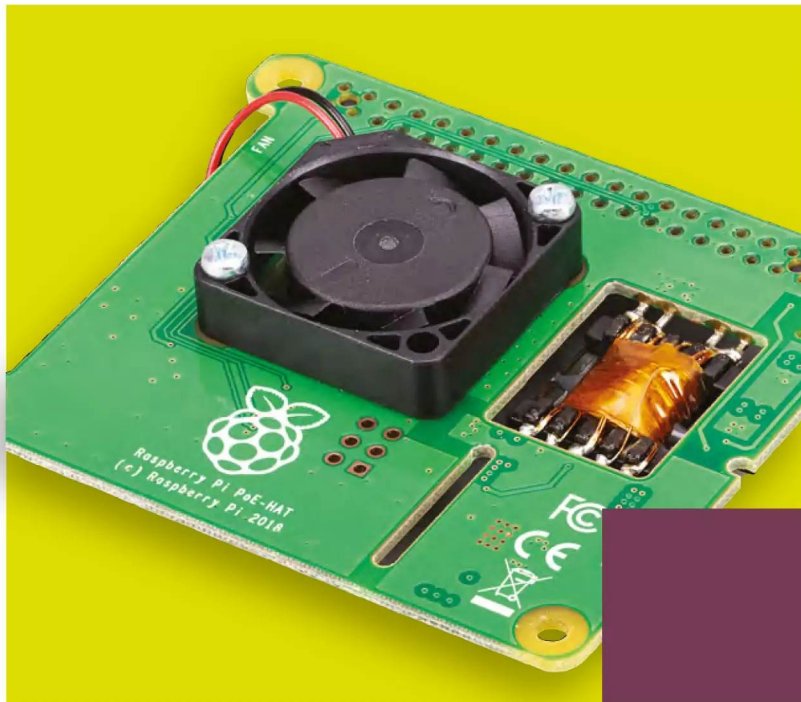
magpi.cc/priorityboarding

Terms & Conditions: A Priority Boarding code will be emailed to everybody who takes out a print subscription to *The MagPi* or *HackSpace* magazine. People who subscribe to both magazines will get two codes. Print subs only! Priority Boarding does not apply to people with App Store, Google Play, Zinio, PDF contributions, or other paid-for subscriptions. Each code will entitle you to purchase 1x Raspberry Pi 5 model (either 4 GB or 8 GB) for the standard retail price and delivery. Multiple codes need to be used individually. This is a limited offer and is subject to change or withdrawal at any time.



COMPLETE RASPBERRY PI BUYER'S GUIDE

Everything you
need for starting
your Raspberry
Pi adventure

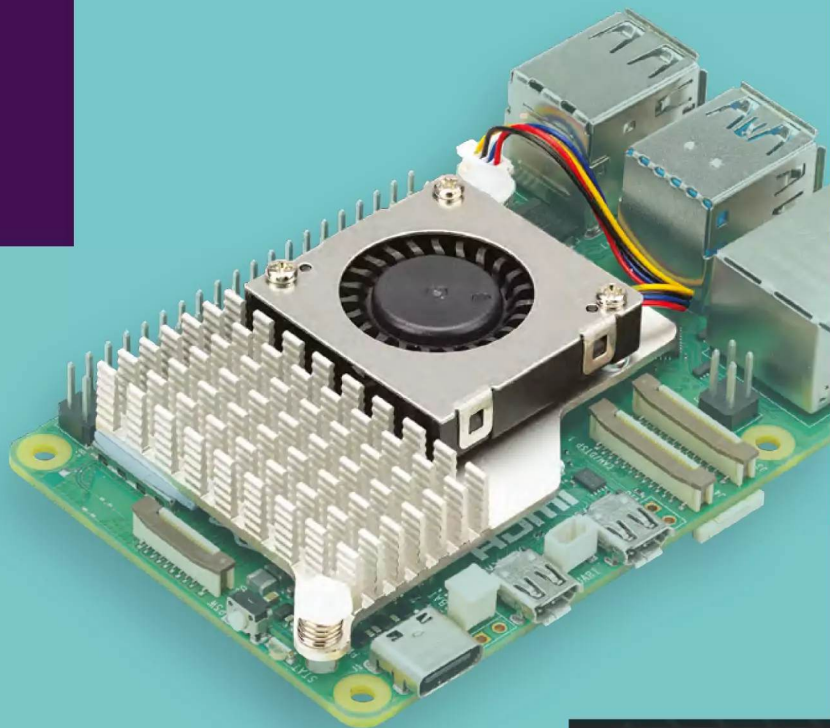


Raspberry Pi is an incredible device that can empower people to do, well, just about anything. It can't do it all on its own though without the help of the incredible accessories, add-ons, and kits available.

Whether you got a new Raspberry Pi 5 over the holiday season or looking for something new to do with your existing boards, we've put together a definitive list of essential and not-essential-but-very-cool items for you to peruse that could help kickstart your next project.

Let's get making.





HOW TO BUY Raspberry Pi 5

The best place to buy a Raspberry Pi 5 – or any other kind of Raspberry Pi! – is from an **Approved Reseller**. This is a network of shops and online stores that work with Raspberry Pi to distribute hardware, and you can find lists of which ones have stock in your country from the product pages on the Raspberry Pi website: raspberrypi.com/products



Raspberry Pi 5 PRIORITY BOARDING

There's still time to get **Priority Boarding codes!** Existing and new subscribers to the print edition of *The MagPi* and *Hackspace* magazine will get a code that lets them order and receive their Raspberry Pi 5 earlier than others. Check out page 4 for more details, or head to magpi.cc/priorityboarding



THE ESSENTIALS

Accessories and add-ons that every Raspberry Pi owner should have

THE BASICS

OFFICIAL CASE

PRICE £10 / URL magpi.cc/case

Protect and cool your Raspberry Pi

OFFICIAL MOUSE

PRICE £9 / URL magpi.cc/mouse

A USB mouse in Raspberry Pi colours

HDMI TO MICRO HDMI CABLE

PRICE £9 / URL magpi.cc/hdmi

Connect Raspberry Pi with no converters

OFFICIAL KEYBOARD

PRICE £20 / URL magpi.cc/keyboard

Type in style with this nifty keyboard

OFFICIAL POWER SUPPLY

PRICE £12 / URL magpi.cc/psu

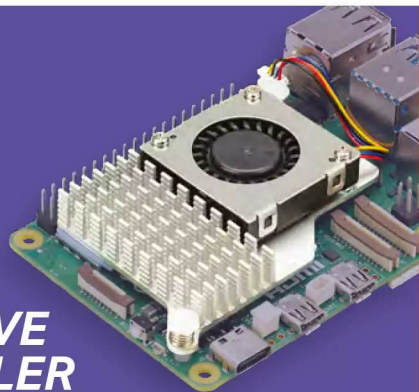
Make sure your Raspberry Pi is properly powered

ARGON NEO CASE

PRICE £15 / URL magpi.cc/argonneo

Add passive cooling with a great looking aluminium case

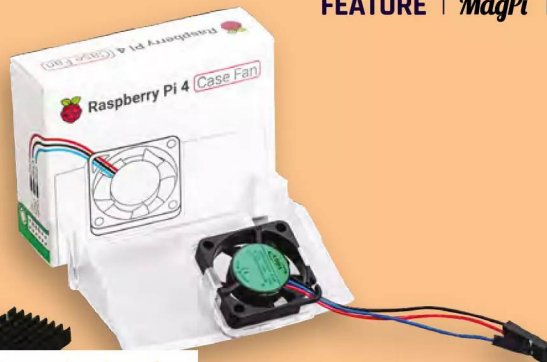




ACTIVE COOLER

PRICE £5 / URL magpi.cc/activecooler

Keep your Raspberry Pi 5 cool



CASE FAN

PRICE £6 / URL magpi.cc/casefan

Raspberry Pi 4 cooling solution

OTHER ACCESSORIES



MEDIA CENTRE CASE

PRICE £12 / URL magpi.cc/flirczero

Camouflage Raspberry Pi under your TV



FLIRC IR DONGLE

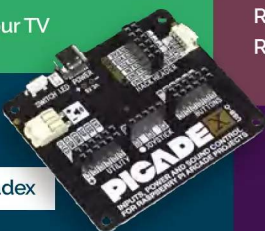
PRICE £12 / URL magpi.cc/flirc

Remote control support for Raspberry Pi media centres

PICADE X HAT

PRICE £16 / URL magpi.cc/picadex

Turn your Raspberry Pi into an arcade cabinet



POE HAT

PRICE £23 / URL magpi.cc/poehat

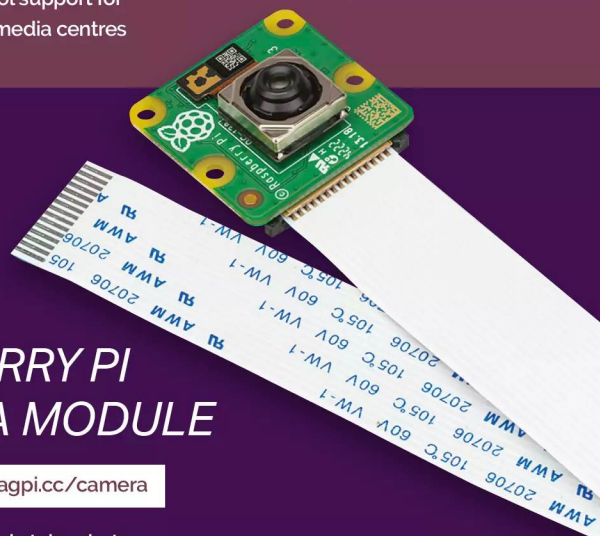
Power-over-Ethernet for industry



RASPBERRY PI CAMERA MODULE

PRICE £34 / URL magpi.cc/camera

Use AI vision or simply take photos



TV HAT

PRICE £16 / URL magpi.cc/tvhat

Watch Freeview via Raspberry Pi



ELECTRONICS & CIRCUITS

Take components and wires, and program them with Raspberry Pi

MULTIMETER

A good way to test your circuit for any faults



ESSENTIAL ITEMS

SOLDERING IRON

Use liquid metal to join components



SOLDERING ACCESSORIES

A soldering stand, some solder, flux, and a solder sucker



BREADBOARD

For quickly setting up test circuits, a breadboard is essential



PROTOTYPING WIRE

Using different wire colours can help you follow your circuit and identify issues



HELPING HANDS

Hold fiddly parts together to make soldering easy



RESISTORS

We like to have a variety of resistors to hand for a range of uses



WIRE CUTTERS

A good pair of wire cutters can also strip them



RASPBERRY PI PICO ADVANCED KIT

PRICE £31 / URL magpi.cc/advpicokit

Another Pico-themed electronics kit, this one comes with a huge number of different electronic components to choose from. There's 32 tutorials that come with it, all leading up to building your own robot, and once again a lot of the parts can also be used with a regular Raspberry Pi for bigger projects.



WHERE TO BUY COMPONENTS

Components like LEDs, buttons, sensors, etc, are not too hard to find but we recommend these sites in the UK for buying components:

shop.pimoroni.com
thepihut.com
coolcomponents.co.uk

ELECTRONICS KITS

STARTER KIT FOR RASPBERRY PI PICO

PRICE £10 / URL magpi.cc/starterpico

Pico is a fantastic board for learning electronics, and this kit comes with some basic components to get you started with lighting LEDs. It can also be used with Raspberry Pi with different jumper cables. It also comes with a Pico H – a Pico with a GPIO header attached.



GROVEPI+ STARTER KIT

PRICE £110 / URL magpi.cc/groveplus

If you want to get stuck in with some home automation and other IoT projects, this kit is a great way to start. The special GrovePi+ connects on top of Raspberry Pi and has connectors to several included modules, which can be programmed in Python and C.



STORAGE SOLUTIONS

Electronic components can be small but they also tend to come in resealable bags – for some, putting them in a drawer in their bags is good enough, but storage boxes can make organising many components much easier. The one in the photo can be found at magpi.cc/compbox.



ROBOTS, ROBOTS, ROBOTS

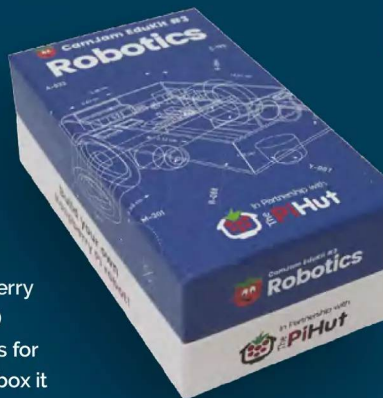
Make an automaton with a Raspberry Pi or Pico

CAMJAM EDUKIT 3

PRICE £20 / URL magpi.cc/edukit3

This is our go-to recommendation for a robot kit for beginners – it's low cost and easy to program, and the coding and components can be used in future, bigger robot projects.

It works with just about every kind of Raspberry Pi (and probably with Raspberry Pi Pico if you're a bit more experienced) and there are a few 3D printable chassis for it if you'd rather not use the cardboard box it comes in as its frame.

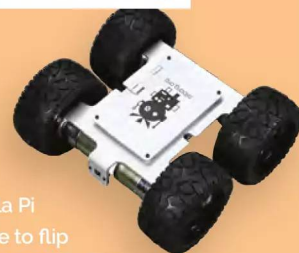


MONSTERBORG

PRICE £210 / URL magpi.cc/monsterborg

Probably the greatest wheeled robot for makers, this stylish masterpiece from the folks at PiBorg was used in the Formula Pi racing series and is able to flip over and drive upside down like the classic Tyco Rebound RC car from the nineties.

Powerful and easy to program, it includes the fantastic ThunderBorg robot controller which can be used for this and other robot projects, as well as the ability to make use of a Raspberry Pi camera module for computer vision/machine learning tasks when in automated mode. It's also very sturdy.

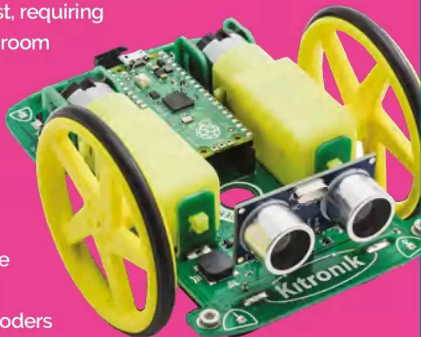


AUTONOMOUS ROBOTICS PLATFORM FOR PICO

PRICE £41 / URL magpi.cc/autorobot

If you have a Raspberry Pi Pico and you're interested in robotics, you can't go far wrong with this kit from Kitronik. It's a little easier to assemble than most, requiring no soldering, and has room for growth with extra components.

Like a lot of Raspberry Pi Pico projects, tutorials and examples are in MicroPython so they're easy to pick up and customise by rookie coders



XGO-LITE ROBOT DOG

PRICE £484 / URL magpi.cc/cm4xgo

This five-limbed robot dog is a very powerful piece of kit, able to walk along on four of those limbs and having great articulation with the fifth to pick up objects aided by its camera and computer vision.

It also has machine-learning powered face and expression recognition so it will react to you with cute animations on its screen, and can be programmed in Python and Blockly remotely for excellent automation.



3D PRINTING

Level your projects up by making your own parts

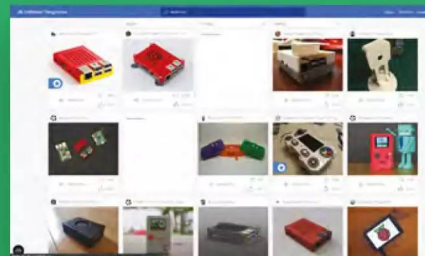
CHOOSE A 3D PRINTER

With so many 3D printers to choose from, it's best to ask the experts, which in our mind is our colleagues on our sibling publication *HackSpace* magazine. Issue 59 (hsmag.cc/59) has a guide on building an open source 3D printer, as well as reviews of 3D printers throughout its other issues, such as the new Prusa MK4 in issue 67 (hsmag.cc/67).



FIND 3D PRINTS

There's an absolutely massive community of 3D printers sharing their files on the internet, and mostly for free! If you have an idea for a part or print in general, you should look on Thingiverse (thingiverse.com) or Printables (printables.com).



We've also seen files on Etsy for sale of great items, so don't forget to look around for prints.

MAKE YOUR OWN 3D PRINTS

If you can't find what you need, FreeCAD (freecad.org) is online software that helps you create 3D objects that can be used in printing, and it's very easy to use, especially for simpler projects like cases.



RASPBERRY PI MERCH!

Show off your love for Raspberry Pi with official merchandise



Enamel pin

PRICE £4

URL magpi.cc/pin

Mug

PRICE £9

URL magpi.cc/mug



Badges

PRICE £6

URL magpi.cc/badges

Water bottle

PRICE £18

URL magpi.cc/bottle



THE *Official*

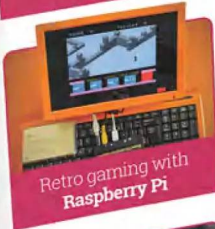


RASPBERRY PI HANDBOOK

2024



Serious fun with
electronics



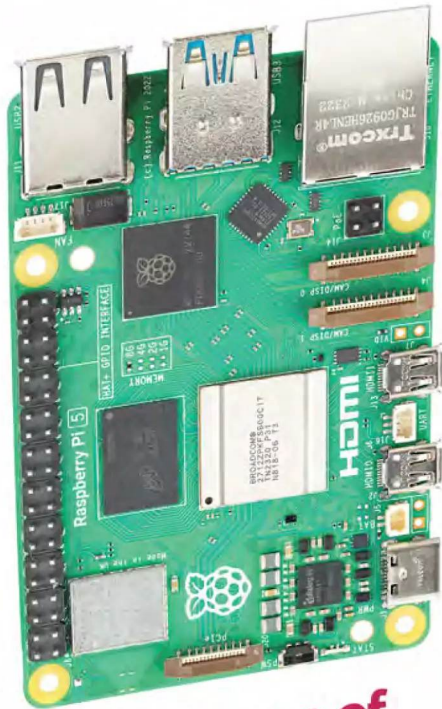
Retro gaming with
Raspberry Pi



Upcycling
Big Mouth Billy Bass



Recreate
Doctor Who's K-9



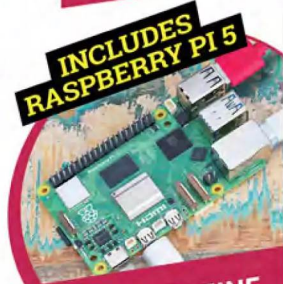
200 Pages of
Amazing Makes!



Super Simple
Robotics



Build Iron Man's
Arc Reactor



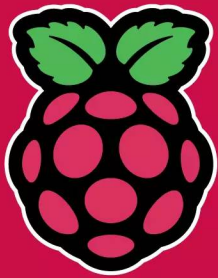
INCLUDES
RASPBERRY PI 5

FROM THE MAKERS OF *MagPi* THE OFFICIAL RASPBERRY PI MAGAZINE

THE OFFICIAL RASPBERRY PI HANDBOOK 2024

magpi.com/raspberry_pi_handbook_2024

THE *Official*



RASPBERRY PI HANDBOOK

2024



200 PAGES OF RASPBERRY PI

- Get started guide covering every Raspberry Pi
- Everything you need to know about the brand new Raspberry Pi 5
- Inspiring projects for your next build idea
- Have some serious fun with electronics
- Learn robotics with our super-simple robotics tutorial
- Play retro games with the diminutive Raspberry Pi Pico

Buy online: magpi.cc/handbook

* Raspberry Pi hardware not included

Overclock Raspberry Pi 5



Take the already speedy Raspberry Pi 5 all the way up to 3.0GHz. By **Lucy Hattersley**

MAKER

Lucy Hattersley

Lucy is editor of The MagPi magazine and enjoys computers and her cat. Overclocking the cat is probably a bad idea though. Probably?

magpi.cc

By default, Raspberry Pi 5 offers a huge speed increase over previous models.

Having said that, more is always a bonus and it's possible – if not always desirable, or necessary – to push the speed pedal a little harder.

With some Raspberry Pi devices, you can make the clock speeds run even faster, a process known as “overclocking”.

Overclocking Raspberry Pi 5 via Raspberry Pi OS isn't dangerous, but it isn't supported either (and not all Raspberry Pi 5 boards respond to overclocking, see “YMMV”).

Overclocking causes Raspberry Pi 5 to use more power, and the board will consequently run a lot hotter. This isn't dangerous because Raspberry Pi OS will throttle performance to bring the heat levels down.

But more speed is what we're after here, so we heavily suggest using a cooling device, such as the Active Cooler or official Raspberry Pi Case for Raspberry Pi 5. These devices contain fans which will keep the temperature down and the speed up.

Thanks go out to Jeff Geerling whose work predated ours (magpi.cc/jeffoverlock) and forms the basis for the **config.txt** code in this tutorial.

Let's put the pedal to the metal!

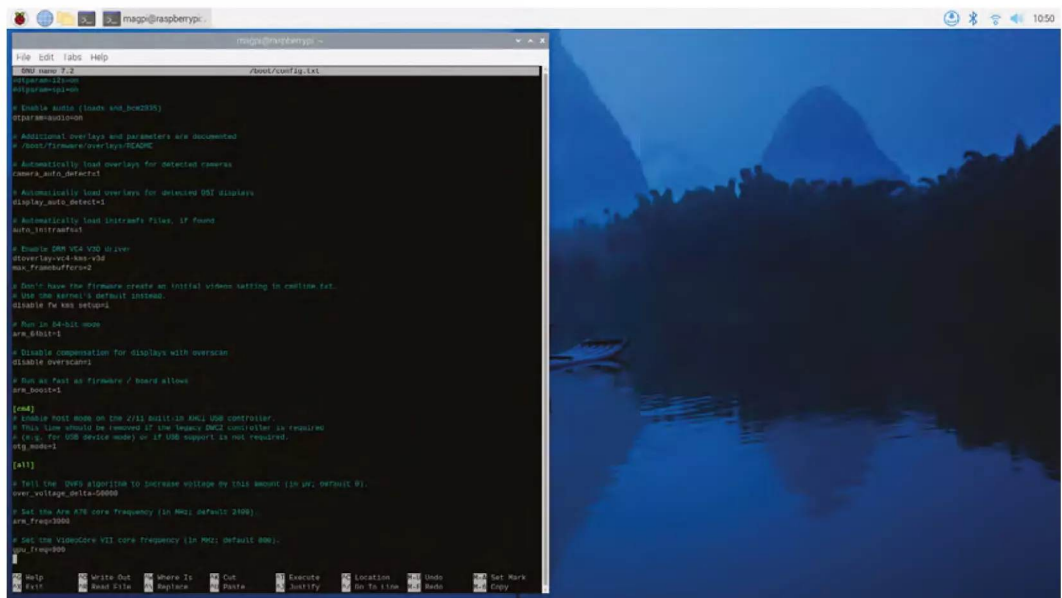
01 Edit config

Set up a fresh installation of Raspberry Pi 5 OS. Be prepared for some experimentation here so don't use an SD card with anything on it you don't mind losing. If you have an Active Cooler or case

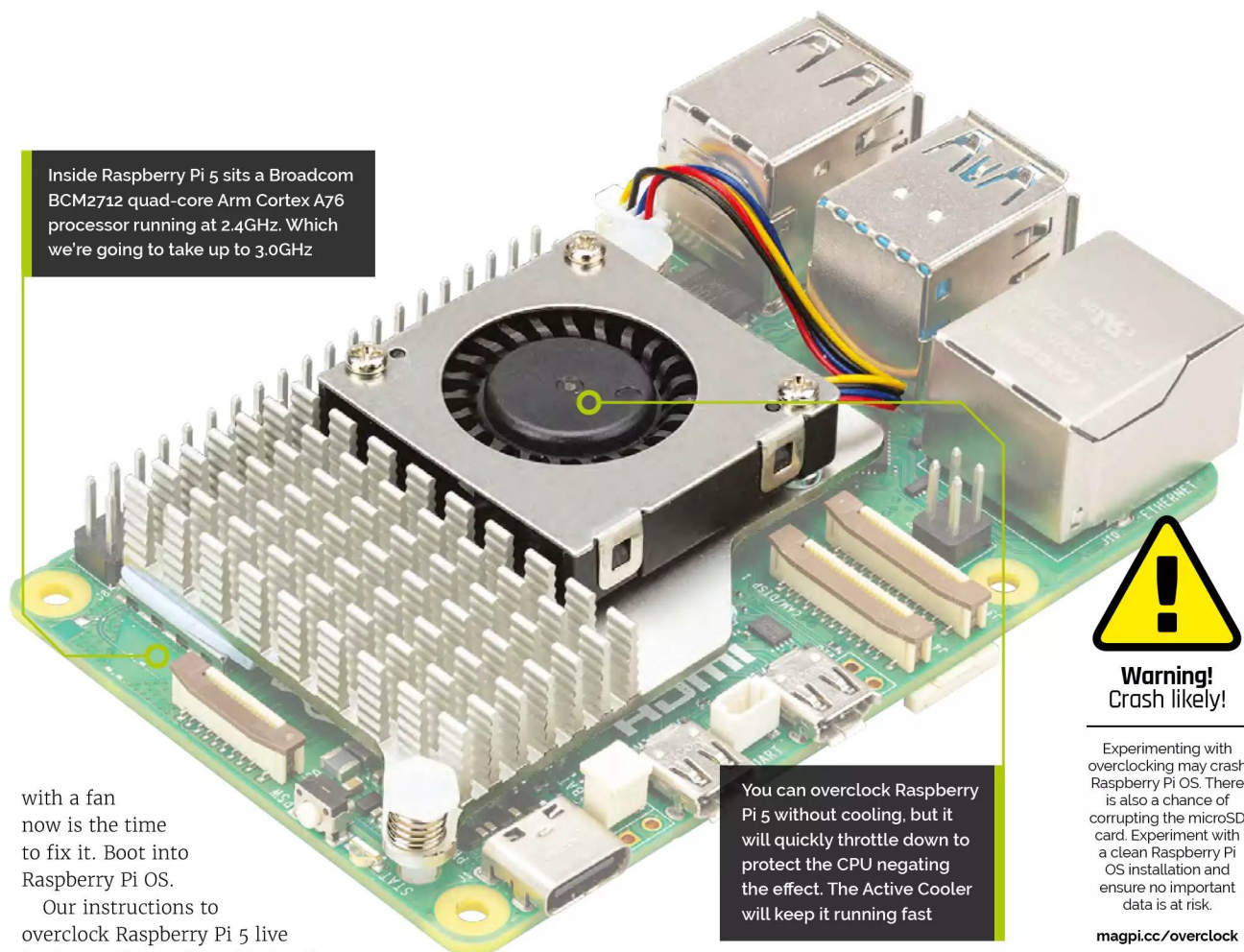
▶ The config.txt file is edited to adjust CPU clock and GPU core speeds

You'll Need

- ▶ Raspberry Pi 5
magpi.cc/raspberrypi5
- ▶ Raspberry Pi OS (Bookworm)
magpi.cc/software
- ▶ Active Cooler or Fan Case recommended
magpi.cc/activecooler



Inside Raspberry Pi 5 sits a Broadcom BCM2712 quad-core Arm Cortex A76 processor running at 2.4GHz. Which we're going to take up to 3.0GHz



Warning!
Crash likely!

Experimenting with overclocking may crash Raspberry Pi OS. There is also a chance of corrupting the microSD card. Experiment with a clean Raspberry Pi OS installation and ensure no important data is at risk.

magpi.cc/overlock

with a fan now is the time to fix it. Boot into Raspberry Pi OS.

Our instructions to overclock Raspberry Pi 5 live in the **config.txt** file. This itself lives inside the **/boot/** directory. Open a terminal window and edit this file with the following command

```
$ sudo nano /boot/config.txt
```

Scroll down to the end of the file and locate [all]. We're going to add code beneath here.

02 Increase voltage

The first step is to increase the voltage amount. In previous Raspberry Pi models. For previous iterations of Raspberry Pi we recommended using `over_voltage`, but now the Raspberry Pi engineers recommend `over_voltage_delta` (magpi.cc/ovdelta). This is because Raspberry Pi 4, 400 and now 5 use a feature called DVFS (Dynamic Voltage and Frequency Scaling, magpi.cc/dvfs). This technique allows Raspberry Pi 4 and 5 devices to run at lower temperatures while still providing the same performance.

According to Raspberry Pi's documentation:

You can overclock Raspberry Pi 5 without cooling, but it will quickly throttle down to protect the CPU negating the effect. The Active Cooler will keep it running fast

Underclock

An interesting aspect of `over_voltage_delta` is that it's a signed value. This means you can use negative numbers to use less power (and have less speed). This could have interesting applications in projects that use solar, or battery power.

Use this code in **config.txt** to take the power level down:

```
[all]
# Decrease the voltage
over_voltage_delta=-10000

# Set Arm core frequency in MHz
(default 2400)
arm_freq=2200
```

You can use this command in a Terminal window to view the current power consumption:

```
$ vcgencmd measure_volts core
```

Top Tip

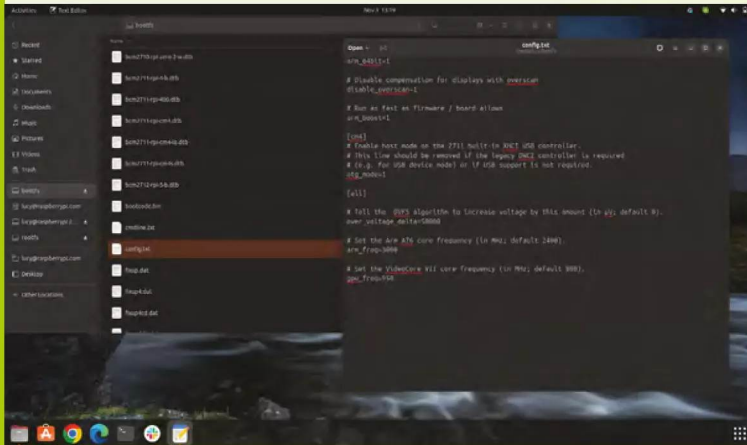
YMMV

Your mileage will vary and there is an overclock lottery. The stock speeds of 2400Mhz and 800Mhz are what all Raspberry Pi 5 models can operate at. Some units will overclock better than others. If Raspberry Pi OS behaves erratically or freezes, reduce the amounts or stop overclocking altogether.

Recover from crash

In this tutorial we are slowly increasing the CPU and GPU speed. On many Raspberry Pi 5 boards (including ours) this will eventually result in a crash. Typically, Raspberry Pi OS will freeze.

The challenge here is that you cannot edit config.txt in Raspberry Pi OS to decrease the CPU speed. The easiest way to fix this is to power off Raspberry Pi 5 and remove the microSD card. Next, insert the microSD card into another computer. Open the microSD card drive in the Files/Finder/Explorer app and locate the config.txt file on the root of the drive. Open the file in a text editor and adjust, or remove, the overclock values back to the point where Raspberry Pi 5 booted. Place the microSD card back in Raspberry Pi 5 and boot up again.



▲ The config.txt file is being edited on a PC laptop running Ubuntu to restore the values to a point where Raspberry Pi 5 will boot and run correctly

“Various clocks (e.g. ARM, Core, V3D, ISP, H264, HEVC) inside the SoC are monitored by the firmware, and whenever they are not running at full speed, the voltage supplied to the particular part of the chip driven by the clock is reduced relative to the reduction from full speed. In effect, only enough voltage is supplied to keep the block running correctly at the specific speed at which it is running. This can result in significant reductions in power used by the SoC, and therefore in the overall heat being produced.”

Add the following code to the **config.txt** file underneath [all]:

```
# Increase the voltage
over_voltage_delta=50000
```

03 Increase frequency

Now we are going to increase the frequency of the CPU from its core speed of 2400MHz. Eventually we plan to take this up to 3000MHz,

but let's try smaller values first. Add these lines below the over_voltage_delta line:

```
# Set Arm core frequency in MHz (default 2400)
arm_freq=2600
```

Restart Raspberry Pi OS. Click the Raspberry Pi logo and choose Shutdown, then Reboot. Or enter sudo reboot in a terminal window.

Now test out Raspberry Pi OS for a few minutes to ensure that it remains operating properly without freezing up. Open a web browser and browse a few sites to check everything is running. If Raspberry Pi OS freezes you will need to remove the overclock commands from **config.txt** (see the “Recover from crash” boxout).

04 Speed test

If Raspberry Pi OS runs without crashing take a look at the increased speed in action. To check the new maximum clock speed open a terminal window and enter:

```
$ sudo cat /sys/devices/system/cpu/cpu0/cpufreq/cpuinfo_max_freq
```

In use, the core frequency will adjust between the maximum speed and a slower frequency (to reduce energy consumption). To view the actual speed enter this command:

```
$ vcgencmd measure_clock arm
```

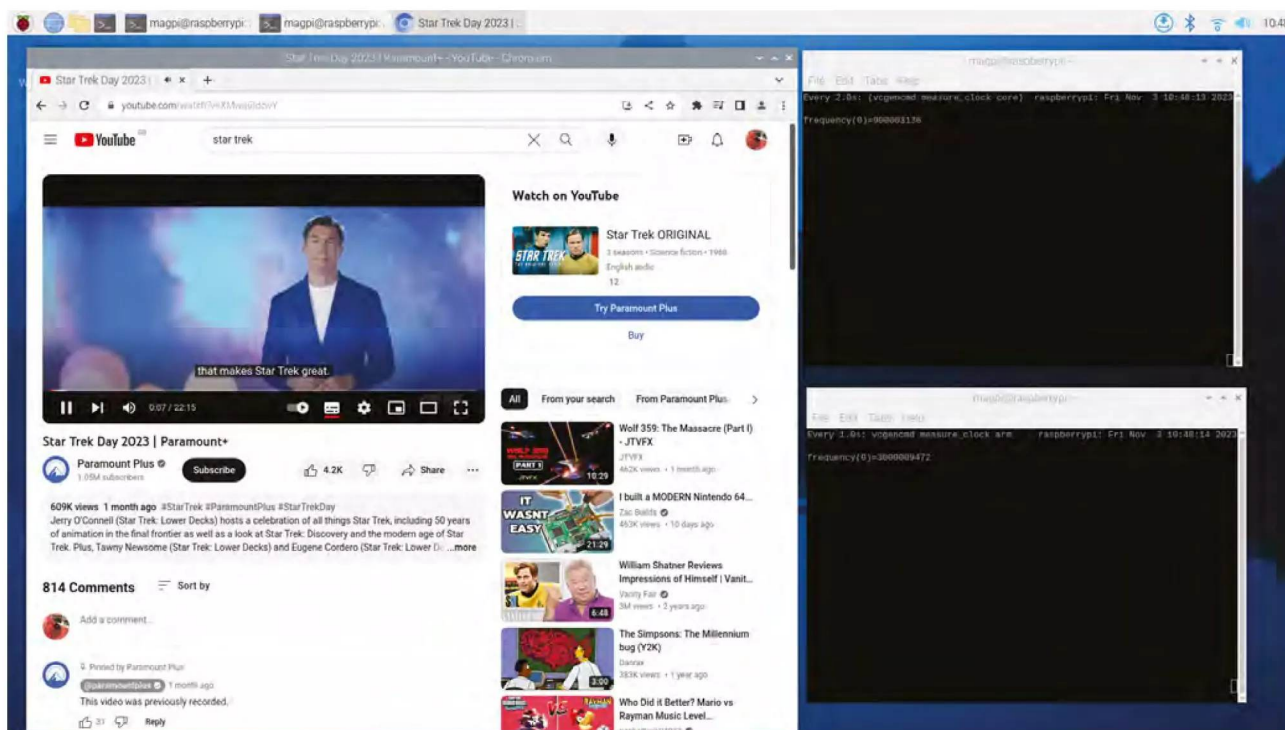
This will return the value that the Arm CPU is running at × 1,000. To monitor the clock in real time enter this command:

```
$ watch -n 1 vcgencmd measure_clock arm
```

Fire up the web browser and visit **youtube.com**. Open up a video and you will see the clock speed spin up. After a while it may reduce back down as the Arm chip manages its speed efficiently. Open another window and run another video to spin up the speed.

05 Take it further

Now we're going to take the Arm clock up to the maximum we have found reliable on



our test devices, 3MHz. Open a Terminal window and enter `nano /boot/config.txt` to edit the configuration files:

```
# Set Arm core frequency in MHz (default 2400)
arm_freq=3000
```

Again, reboot Raspberry Pi OS and check that everything is running correctly. Repeat Step 4 to test the speed. If your Raspberry Pi crashes follow the “Recover from crash” boxout and reduce the speed to 2800. Some Raspberry Pi boards are more amenable to overclocking than others (see the “YMMV” boxout).

06 GPU frequency

Our next step is to increase the GPU frequency. By default, this is 800Mhz, and we have successfully increased this to 900MHz on our test Raspberry Pi boards. Edit `config.txt` and add:

```
# Set GPU frequency in MHz (default 800)
gpu_freq=900
```

Reboot and test Raspberry Pi OS by launching a web browser and testing YouTube. Some users have reported that `gpu_freq=1000` also works on their devices, so feel free to test that out. But we

have maxed out at 3000 and 900 for our CPU and GPU, respectively.

Overclocking does use more power than running at the standard speed. And, we can't stress enough that the regular 2.4GHz and 800MHz clock speeds for the CPU and GPU are more than sufficient for all the tasks we've used Raspberry Pi for. So this is more of a novelty to play around with for us. Perhaps of more use would be underclocking Raspberry Pi 5 for projects that demand less power, and need to make the most use of battery power (see our “Underclocking” boxout). [\[1\]](#)

▲ Using a Terminal command to monitor clock speeds while running YouTube videos

config.txt

► Language: **Bash**

DOWNLOAD THE FULL CODE:



magpi.cc/github

```
001. # Increase the voltage
002. over_voltage_delta=50000
003.
004. # Set Arm core frequency in MHz (default 2400)
005. arm_freq=3000
006.
007. # Set GPI frequency in MHz (default 800)
008. gpu_freq=900
```

Cloud Gaming with Raspberry Pi 5



PJ Evans

PJ is a writer, software engineer and tinkerer. He really enjoyed the testing phase of this article.

mastodon.
social/@mrpevans

MAKER

Play the latest AAA console games with only a Raspberry Pi 5 and a good internet connection. No downloads, no updates. Here's how

Did you know your Raspberry Pi can also double as an Xbox or a high-end gaming PC? You can now play the latest games in full HD using just your humble credit-card-sized computer. Your inputs (controller, keyboard) are sent to the remote 'rig' and a video stream of the game is sent back. Recent improvements mean lag is barely perceivable and the video quality is excellent. It's not quite as easy as powering up and jumping into *Baldur's Gate III*, however, and you'll need to make a few decisions along the way. This tutorial will walk you through it all.

01 Choose your platform

There are a handful of cloud gaming services available. Out of all of them we've chosen Xbox Game Pass Ultimate as a great service for our Raspberry Pi 5. For a monthly subscription, you can access hundreds of games that are instantly playable, and many top games are



▲ The 8BitDo Pro 2 controller is a perfect choice for this project, and is endorsed by Microsoft for Xbox Games Pass

You'll Need

- ▶ Fast internet connection (ideally fibre)
- ▶ 8BitDo Pro 2 controller magpi.cc/pro2
- ▶ Wireless keyboard/mouse magpi.cc/k400

included. We'll also cover other options such as Nvidia GeForce Now. Sadly, Sony's PlayStation equivalent is PC-only, and we encountered issues using Amazon's Luna. Xbox and Nvidia can work with Raspberry Pi as they both support streaming to the browser, including Chromium.

02 Choose your hardware

A good controller is essential for any gaming, cloud or otherwise. We've chosen the 8BitDo Pro 2. Not only is it a great Bluetooth controller, but it's also proven with Raspberry Pi hardware and approved by Xbox. You can also use an official Xbox controller if you prefer. You'll also need to interact with a keyboard and mouse. Of course, you can use a regular wired keyboard and mouse connected to the USB ports, but if you're setting up a gaming environment you may want to consider a wireless keyboard such as the Logitech K400 series which combines a keyboard and trackpad in one.

03 Initial setup

We're going to need the 'full' version of Raspberry Pi OS, complete with desktop, as we need the Chromium web browser. Using the Raspberry Pi Imager (magpi.cc/imager) select Raspberry Pi OS (64-bit) and write the image to your SD card to get the latest evolution of the operating system. We strongly recommend using a wired internet connection to your Raspberry Pi if possible, for best results, but if you're using Wi-Fi, set it up now (and SSH too if you need remote access). Once booted and ready to go, check and install any software updates (or run `sudo apt update && sudo apt upgrade` in the Terminal).



▲ You'll need to use the keyboard and mouse as well as a game controller, so choose a wireless option like these

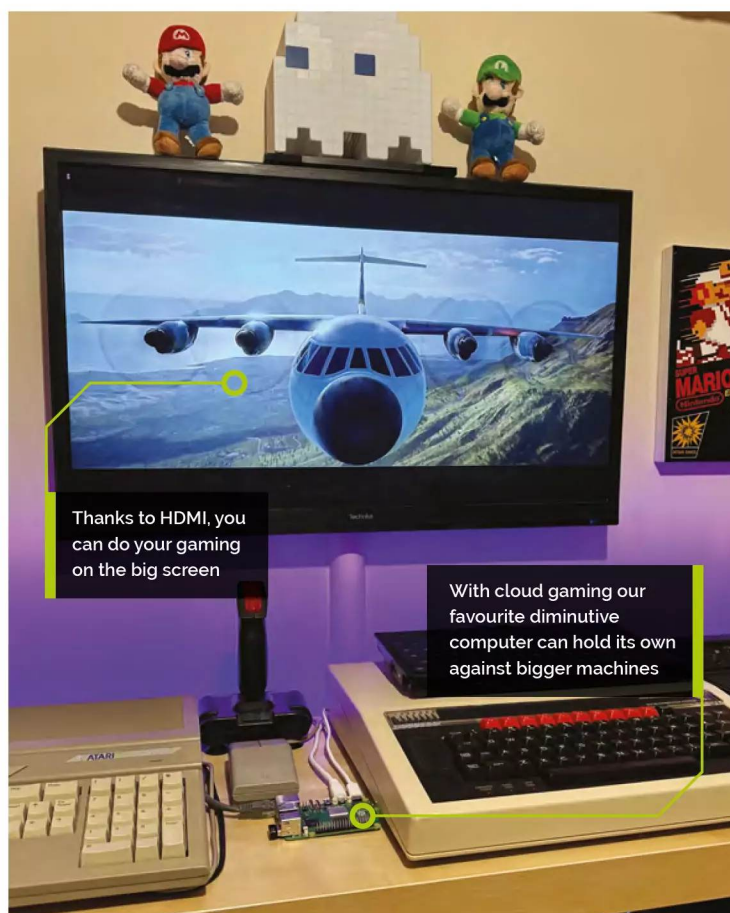
04 Set up your controllers

The keyboard (with built-in trackpad) we have chosen is very simple to set up because it uses a dedicated USB dongle. Just plug in the receiver, and make sure the keyboard has some batteries and is switched on. Both it and the trackpad should work immediately. Bluetooth devices, such as our controller, need to be paired first. Click on the Bluetooth icon at the top-right of the screen and select 'Add Device'. When the scanning window appears, put your controller into pairing mode (on the Pro 2, press Start + X then press the sync button for a few seconds). The device should appear in the window. Now click 'Pair' to connect. You only have to do this once.

“ Recent improvements mean lag is barely perceivable and the video quality is excellent ”

05 Get an account

It is a necessity of cloud gaming that you have an account with the provider. In Xbox's case, this is Microsoft. So if you haven't got one, now is a good time to get yourself set up with a Microsoft ID. Go to xbox.com, click 'Sign in' then 'Create account'. Go through the steps to get set up, and don't forget to choose a strong password and multi-factor authentication. Now you can log into xbox.com and get ready to start gaming.



Thanks to HDMI, you can do your gaming on the big screen

With cloud gaming our favourite diminutive computer can hold its own against bigger machines

06 Sign up for Game Pass Ultimate

Xbox's Game Pass service comes with different tiers of access. Cloud gaming is only provided with the Ultimate package, which at time of writing is £12.99 per month with a £1 offer for the first month. For your money you get full access to over 300 games, from classics like *Fortnite* and *GTA*, indie greats like *A Short Hike* and newer top titles such as *Starfield* and *Football Manager 2024*. Sign up will require credit card details which will be debited monthly. You can cancel your subscription at any time. Get yourself signed up and then you're ready to start gaming.

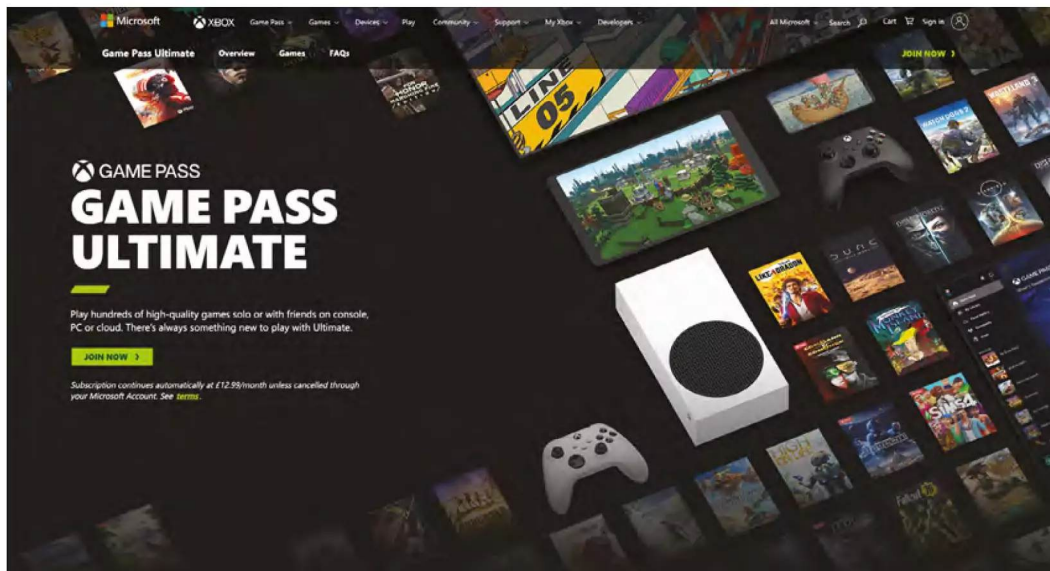
07 Try it out

Open up Chromium on your Desktop and go to xbox.com. If you haven't already, log in now. When you are returned to the front page, click 'Games' then 'Cloud games'. You'll have a selection of games chosen for you, or you can scroll to the bottom and click 'See all games'. Choose a game and then click 'Play'. You may get a warning that a controller has not been detected.

Top Tip

Choose the right mode

Using the 8BitDo Pro 2? Always begin by pressing the X + Start buttons to ensure maximum Xbox compatibility.



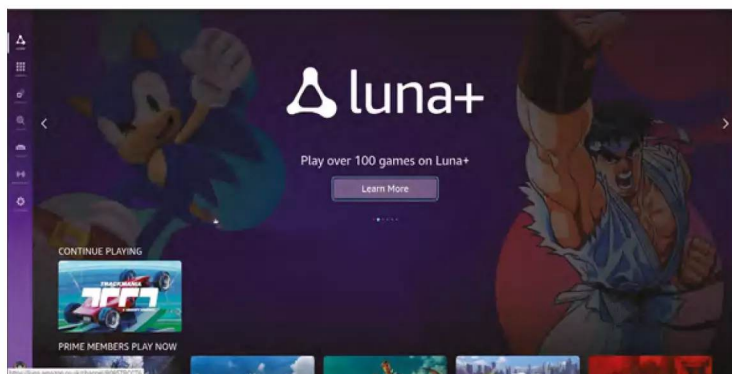
► You'll need to sign up for Xbox Games Pass Ultimate. It comes with access to over 300 games

If so, click A on your controller and it should clear. If that doesn't work, check your pairing. You'll now be put in a queue for the first available rig.

08 Playtime!

Your wait time will depend on the game's requirements and popularity. In our testing, most games kept you waiting no more than a couple of minutes, but popular and resource-hungry games can see waits of 30 minutes or more at peak times. If this happens, try choosing another game. The wait screen will keep you informed of progress. When the game starts, it will be as if you were in front of a real console and you play accordingly. Pressing the 8BitDo controller's 'heart' button will bring up an overlay so you can end the game or adjust settings.

▼ Our experience streaming games with Luna wasn't good, but it might work for you



09 Troubleshooting

Now you will find out whether your setup has a good enough connection. The service will try to adjust video quality, prioritising speed over resolution. If you find it inadequate, try moving from Wi-Fi to a wired connection. Check whether your sibling is streaming HD video while uploading 5,000 selfies. If nothing works, contact your internet provider to see if there are any faster options available to you. We've found that fibre-based services are the only way to get close to a real console experience. If you're finding response times laggy, nearby radio interference may be the cause, and using an Ethernet cable to connect to your router can fix this too.

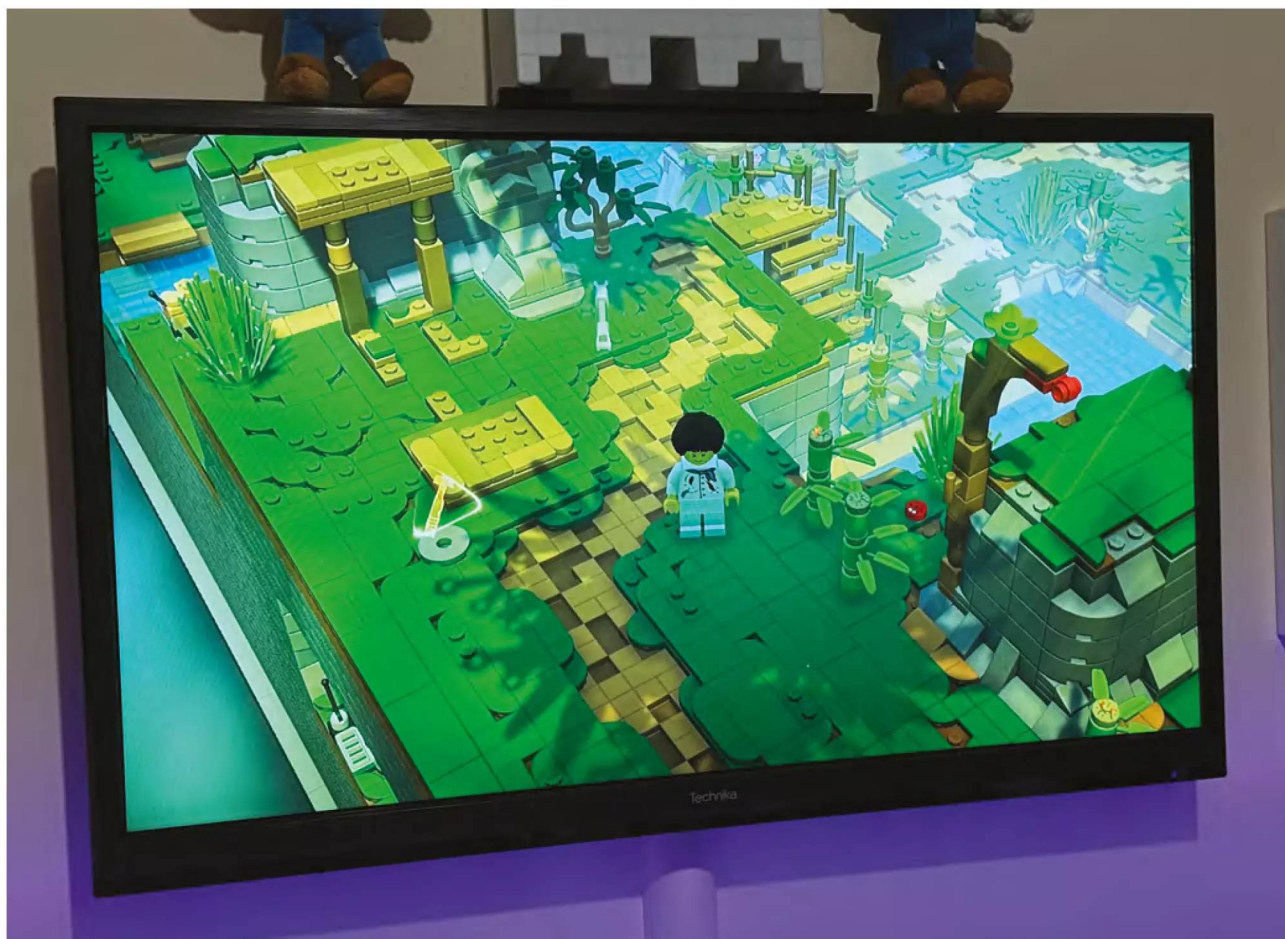
10 Boot straight in

To finesse your Xbox experience, we can configure Raspberry Pi OS so you can boot straight into a full-screen browser. Open up a Terminal window or SSH into your tiny gaming rig. Now issue this command:

```
nano ~/.config/wayfire.ini
```

Go to the bottom of the file and add this as a single line:

```
[autostart]
chromium = chromium-browser https://www.xbox.com/en-GB/play?xr=shellnav/ --kiosk
```

```
--noerrdialogs --disable-infobars --no-first-run --ozone-platform=wayland
```

Save the file and quit Nano (**CTRL+X**). Reboot, and Chromium will start in fullscreen mode automatically, and go straight to the Xbox home page.

11 Nvidia GeForce Now

If you want to try cloud gaming completely for free, check out Nvidia GeForce Now, a similar service to Xbox Game Pass. It has a free tier that allows you to access a small subset of games, including an excellent demo of *Lego Bricktales*. It runs in the browser just like Xbox's service, but can link in with popular game libraries such as Epic and Steam to allow access to your purchased games (if supported). You can also choose a paid tier to get access to the games libraries and better hardware. Visit magpi.cc/geforcenow to check it out.

“ The service will try to adjust video quality, prioritising speed over resolution ”

12 Other platforms

We also tried Amazon Luna, which is a similar offering to Nvidia's. However, although streaming worked well, we found it struggled with controller input, with seemingly random keypresses ruining the gaming experience. We were unable to establish whether this was a temporary glitch, however, so do try Luna if you have Prime membership. If you're looking at a permanent gaming setup, don't forget that cloud-based streaming is not your only option. Raspberry Pi is a brilliant platform for retro gaming (check out RetroPie) or even having a go at making your own with Python PyGame. Whichever way you go, have fun! 🎮

▲ Nvidia's GeForce Now platform comes with some great demos for free, such as *Lego Bricktales*

Top Tip 🟡

Steam Link

If you want to play your PC games in another room, check out Steam Link for Raspberry Pi.

Python on Raspberry Pi



Alasdair Allan

Alasdair Allan is a scientist, author, hacker and journalist. In the past he has mesh networked the Moscone Center, caused a US Senate hearing, and contributed to the detection of what was – at the time – the most distant object yet discovered in the Universe.

magpi.cc/docs

There's been a major change to Python. Now is a great time to learn about virtual environments. By **Alasdair Allan**

Raspberry Pi OS Bookworm brings a big change for Python users, as we implement a change introduced by the Python community in upstream Debian: from Bookworm onwards, packages installed via pip must be installed into a Python Virtual Environment. This will affect a lot of people, and we understand it will be disruptive for many as we all get used to the new way of doing things.

In this tutorial, we take a look at what has changed and how to effectively use Python on Raspberry Pi OS Bookworm.

Python 3 is installed by default on Raspberry Pi OS and is used for many important functions. Interfering with the system Python installation can therefore cause problems for your operating system, so it's important that if you install third-party Python libraries you use the correct package management tools.

There are two routes to installing libraries into the default python distribution. Firstly you can use apt and install pre-configured system packages, or alternatively, you can use pip to install packages that are not distributed as part of Raspberry Pi OS.

From Bookworm onwards, packages installed via pip must be installed into a Python Virtual Environment using venv (magpi.cc/venv). This has been introduced by the Python community, not Raspberry Pi, see PEP 668 for more details (magpi.cc/pep668).

Installing packages using apt is the preferred method for installing Python libraries on Raspberry Pi OS Bookworm. Packages installed via apt are tested, usually pre-compiled so they install faster, and are designed for Raspberry Pi OS. They won't break your system. Additionally, installing via this route means that all required dependencies are also installed, and a log of installation is maintained by the OS so the installation can be easily rolled back (uninstalled) if needed.

For instance, to install the Python 3 library to support our Build HAT:

```
$ sudo apt install python3-build-hat
```

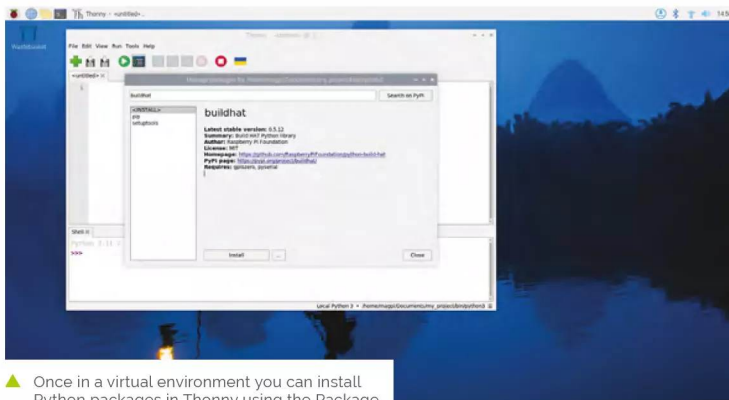
...to install the pre-built library.

Using apt makes installing larger packages, like numpy which has many native dependencies including a Fortran compiler, much simpler than installing individual packages using Python's own package management system.

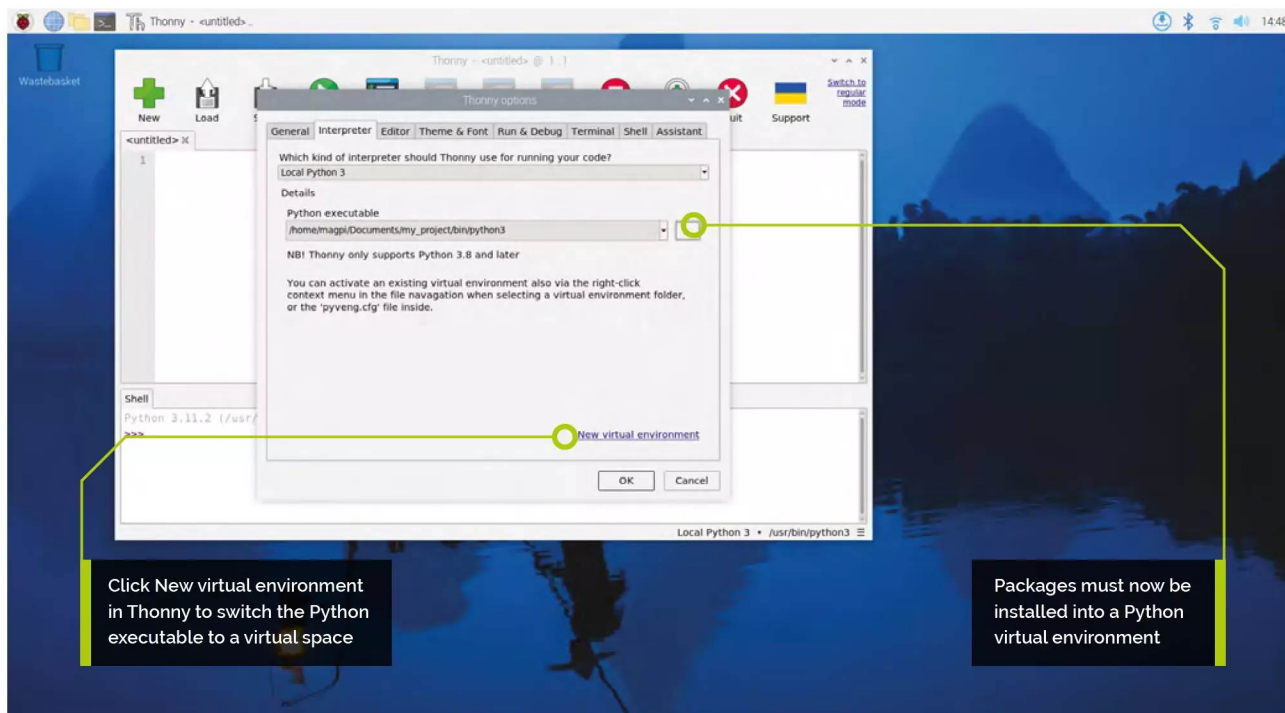
If you want to install a Python library called "numpy" you can use `apt search foobar` to search for the exact package name. In most cases, you'll find that the required package is going to be called `python-foobar` or `python3-foobar`.

Virtual environments

In previous versions of the operating system you used to be able to directly install libraries, system-wide, using the package installer for Python, commonly known as pip. You'll find the following sort of command in many tutorials online:



▲ Once in a virtual environment you can install Python packages in Thonny using the Package



```
$ pip install buildhat
```

However, in newer versions of Raspberry Pi OS, and other operating systems, this is disallowed. If you try and install a Python package system-wide you'll receive an error similar to this:

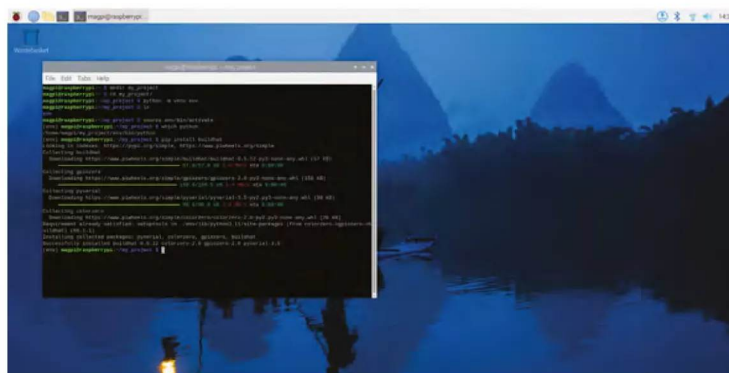
```
$ pip install buildhat
error: externally-managed-environment
```

× This environment is externally managed
 → To install Python packages system-wide, try apt install python3-xyz, where xyz is the package you are trying to install.

If you wish to install a non-Debian-packaged Python package, create a virtual environment using `python3 -m venv path/to/venv`. Then use `path/to/venv/bin/python` and `path/to/venv/bin/pip`. Make sure you have `python3-full` installed.

For more information visit <http://rptl.io/venv>

Note that if you believe this is a mistake, you should contact your Python installation or OS distribution provider. You can override this, at the



risk of breaking your Python installation or OS, by passing `--break-system-packages`. See PEP 668 for the detailed specification.

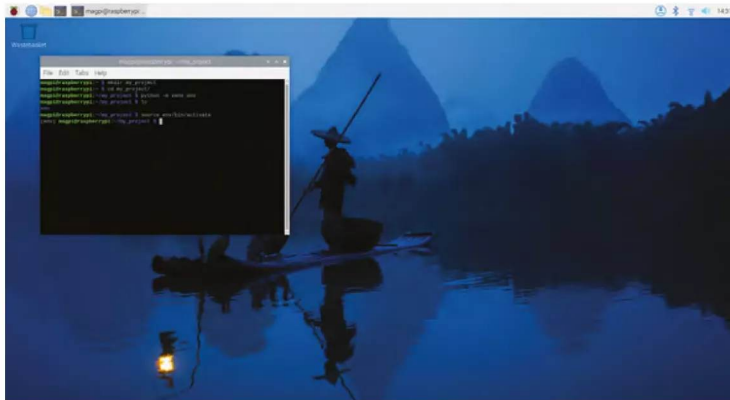
This error is generated because you're trying to install a third-party package into the system Python. A long-standing practical problem for Python users has been conflicts between OS package managers like apt, and Python-specific package management tools like pip. These conflicts include both Python-level API incompatibilities and conflicts over file ownership.

Therefore from Bookworm onwards, packages installed via pip must be installed into a Python Virtual Environment using `venv`. A virtual environment is a container where you can safely install third-party modules so they won't interfere with, or break, your system Python.

▲ Installing the buildhat package inside a Python virtual environment allows the package and its dependencies to be added

You'll Need

- ▶ Raspberry Pi
- ▶ Raspberry Pi OS (Bookworm)



▲ Creating a virtual environment in the command line

Top Tip

System Python

If you want to inherit the currently installed packages from the system Python you should create your virtual environment using `python`

```
-m venv
--system-site-packages env.
```

▼ Trying to use pip to install buildhat in the default Python environment now causes an error message to show

Using pip with venv

To use a virtual environment, the first thing you need to do is to create a container to store the environment. There are several ways you can do this, depending on how you want to work with Python.

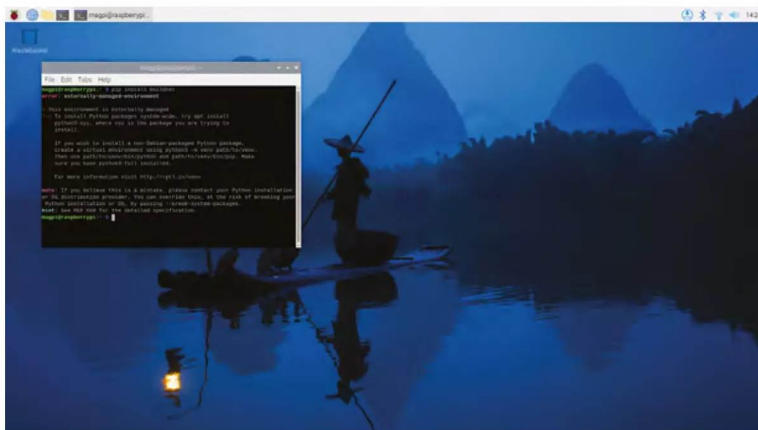
One way you can proceed is to create a new virtual environment (also known as “venv”) for each Python project you make. Here you’ll create a directory to hold your code along with a virtual environment directory:

```
$ mkdir my_project
$ cd my_project
$ python -m venv env
```

if you now look inside the `my_project` directory you’ll see a directory called `env`.

```
$ ls -la
total 12
drwxr-xr-x  3 pi pi 4096 Oct  3 14:34 .
drwx----- 20 pi pi 4096 Oct  3 14:34 ..
drwxr-xr-x  5 pi pi 4096 Oct  3 14:34 env
$
```

Inside this directory is a full Python distribution. To activate your virtual environment and make that



version of Python the one you’re currently using you should type:

```
$ source env/bin/activate
(env) $
```

... and you’ll see that your prompt is now prepended with `(env)` to indicate that you’re no longer using the system Python. Instead, you’re using the version of Python contained inside your virtual environment. Any changes you make here, or modules you install into your environment, won’t cause problems for your system Python.

```
(env) $ which python
/home/pi/my_project/env/bin/python
```

If we go ahead and install a third-party package, it’ll now install into the Python distribution in our virtual environment,

```
(env) $ pip install buildhat
Looking in indexes: https://pypi.org/simple,
https://www.piwheels.org/simple
Collecting buildhat
  Downloading https://www.piwheels.org/
simple/buildhat/buildhat-0.5.12-py3-none-any.
whl (57 kB)
-----
57.8/57.8 kB 2.8 MB/s eta 0:00:00
Collecting gpiozero
  Downloading https://www.piwheels.org/
simple/gpiozero/gpiozero-2.0-py3-none-any.whl
(150 kB)
-----
150.5/150.5 kB 6.9 MB/s eta 0:00:00
Collecting pyserial
  Downloading https://www.piwheels.org/
simple/pyserial/pyserial-3.5-py2.py3-none-any.
whl (90 kB)
-----
90.6/90.6 kB 7.5 MB/s eta 0:00:00
Collecting colorzero
  Downloading https://www.piwheels.org/
simple/colorzero/colorzero-2.0-py2.py3-none-
any.whl (26 kB)
Requirement already satisfied: setuptools
in ./env/lib/python3.11/site-packages (from
colorzero->gpiozero->buildhat) (66.1.1)
Installing collected packages: pyserial,
colorzero, gpiozero, buildhat
Successfully installed buildhat-0.5.12
colorzero-2.0 gpiozero-2.0 pyserial-3.5
(env) $
```

...and if you `pip list` you'll see that your current version of Python has your new modules:

```
(env) $ pip list
Package      Version
-----
buildhat    0.5.12
colorzero   2.0
gpiozero    2.0
pip          23.0.1
pyserial    3.5
setuptools  66.1.1
```

After writing your code, you can run it from the command line inside the virtual environment as you'd expect, by invoking Python as normal.

```
(env) $ ls -la
total 12
drwxr-xr-x  3 pi pi 4096 Oct  3 14:34 .
drwx----- 20 pi pi 4096 Oct  3 14:34 ..
drwxr-xr-x  5 pi pi 4096 Oct  3 14:34 env
-rw-r--r--  1 pi pi   0 Oct  3 14:45 my_
code.py
(env) $ python my_code.py
Hello World!
(env) $
```

You can leave your virtual environment and return to using the system Python by typing:

```
(env) $ deactivate
$
```

... and prove to yourself you've done so by checking the installed packages using `pip list`. You'll see there is a considerable difference!

User environments

An alternative method to creating a virtual environment for each of your Python projects is to create a single virtual environment for your user account and then activate that environment before running any of your Python code. This approach may be preferred if you commonly install the same set of modules for each project, and don't want to have to bother creating individual Python environments for each project, essentially just duplicating your environment.

```
$ python -m venv ~/.env
$ source ~/.env/bin/activate
(env) $
```

Using Thonny editor

Thonny is our recommended IDE (integrated development environment) when working with Python on Raspberry Pi. By default, Thonny uses the system Python.

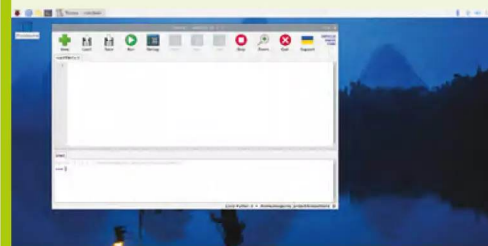
However, you can switch to using a Python virtual environment by clicking on the interpreter menu located at the bottom right of the Thonny window. Clicking on this will offer you a menu to select a configured interpreter or to Configure an interpreter.

Choose Configure interpreter and click 'New virtual environment'. An alert window will appear that says "After clicking 'OK' you need to choose an empty directory, which will be the root of your new virtual environment".

Click OK and choose 'Documents' in the sidebar or the Files app. Click 'Create Folder' and enter "my_project" as the folder name and click OK. Wait for the 'Creating virtual environment' window to appear, and close. Check that the Python executable selection contains "my_project/bin/python3" and click OK.

You can now install packages into your virtual machine using Tools > Manage packages. If you cannot see Tools then first click on 'Switch to regular mode' and restart Thonny.

Search for 'buildhat' in the Manage packages window and click 'buildhat' in the search results window. Click 'install' to add the package to your virtual environment.



We can again check we're in a separate environment by using `pip list`:

```
(.env) $ pip list
Package      Version
-----
pip          23.0.1
setuptools  66.1.1
```

... and leave it using `deactivate`.

```
(.env) $ deactivate
$
```

Remote control Raspberry Pi with VNC



PJ Evans

PJ is a writer, software engineer and tinkerer. He once pranked a colleague by using VNC to randomly change his desktop image. You should not do this.

[mastodon.social/@](#)

Use TigerVNC to remotely access the latest and greatest version of Raspberry Pi OS desktop

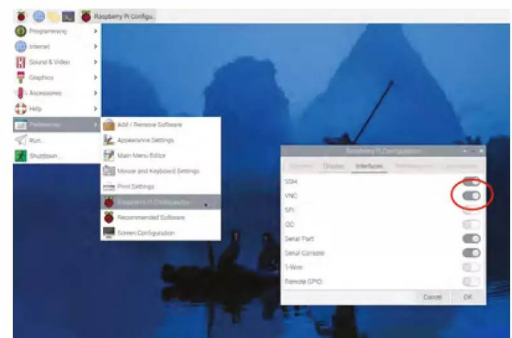
As is so often the case in computing, a new version of an operating system brings advancements but also some compatibility headaches. The old X11 window system that is the underlying engine of the Raspberry Pi desktop has grown too restrictive, so Raspberry Pi OS Bookworm has moved to the new whizzy Wayland engine. Wayland is a great and necessary step forward, but if you want to remote control your desktop using VNC, you need to move to a service that understands Wayland and a client that's compatible. And that's TigerVNC. So, here's how it's done.

01 Let's get ready to remote!

The latest Raspberry Pi OS uses the newer Wayland subsystem to manage the desktop (as opposed to the old-guard X11), compatibility with many VNC servers has (for now) been lost. Luckily, Raspberry Pi OS Bookworm ships with a working service, WayVNC, installed and ready to go. So, start by using Raspberry Pi Imager to install Raspberry Pi OS and make sure you've got all the usual networking up and running. Finally, update everything by checking the System Updates app or using `apt update && sudo apt upgrade` in the terminal.

02 Enable VNC

There are two ways to enable your WayVNC server, which will allow you to connect to the Raspberry Pi remotely. The first, and easiest, method is to use the desktop. Log in locally and click on the Raspberry Pi logo to bring up the menu. Now go down to Preferences and select Raspberry Pi Configuration. On the resulting



▲ Here we enable VNC using the Desktop app. Just slide the switch to the right and click OK

window, click the Interfaces tab and select VNC. Now click OK and within a couple of seconds, your VNC server will be running. You won't see anything different on the screen, it's all in the background.

03 Help! I've got no head!

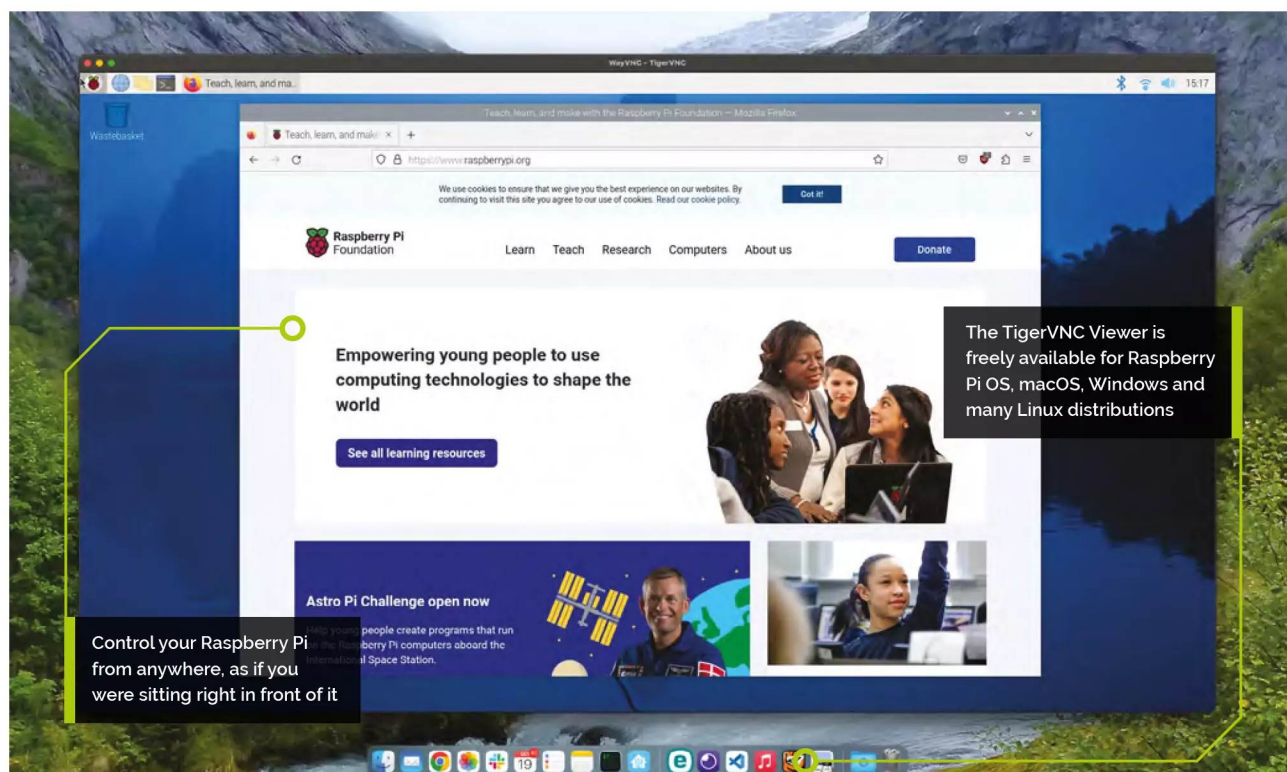
If you are setting up a headless Raspberry Pi (one that has no display attached), you can still enable VNC remotely. Using a terminal app on your local computer, SSH into the Raspberry Pi (you'll need to know its IP address or hostname) then run `sudo raspi-config` to start the Configuration Tool. Use the cursor key to select Interface Options, select VNC and press **ENTER**. On the next screen use **TAB** to select Yes, press **ENTER** again and the server will be enabled. You then can go back to the main menu and select Finish. There's no need to reboot.

04 A local app for local people

Just as much as we need something to VNC

You'll Need

- ▶ Raspberry Pi
- ▶ Raspberry Pi OS Bookworm
- ▶ Computer with TigerVNC Viewer [tigervnc.org](#)

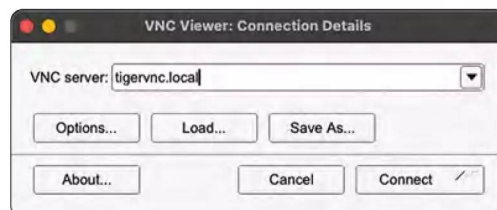


to, we need something to VNC from. We now turn our attention to the remote computer. WayVNC, Raspberry Pi's VNC server, has some particular security requirements that reduce the number of clients with which it is compatible. We've seen the best results with TigerVNC, which is based on the popular TightVNC. You can get the TigerVNC client for free from tigervnc.org although on a Raspberry Pi you can install it by searching for `tigervnc-viewer` in Add/Remove Software, in Terminal by running `sudo apt install tigervnc-viewer`.

“ You should now see your Raspberry Pi's desktop in all its glory ”

05 Get yourself connected

Time to try it out! Start up TigerVNC. A small connection box will appear. Enter the IP address or hostname of the Raspberry Pi followed by “.local” (you can find the hostname using `hostname` in the Terminal). Or enter `hostname -I` to get the IP address. Unlike many VNC clients, you do not put “:1” or similar after the address. That will cause an error. Click Connect. You may be alarmed to see a warning about a server certificate or a large dialogue asking about



◀ The TigerVNC UI is easy to use. Enter the IP address or hostname (normally followed by “.local”) and click Connect

accepting a key. These are nothing to worry about, just some of the advanced security we discussed. Go ahead and say OK to both prompts. Finally, enter your username and password (the same as you use for login). You should now see your Raspberry Pi's desktop in all its glory.

06 Remote access

You will now be able to control your Raspberry Pi as if you were sitting in front of it. You can even cut and paste across the two machines. Don't expect a perfect frame rate, as the networking will introduce lag. If you're interested in accessing the desktop outside of your network, remember security should be firmly at the front of your mind. You'll need to use your router to assign your Raspberry Pi a static IP address and some method of VPN access (such as PiVPN Wireguard) for security access to your network. 📌

Top Tip 📌

What's in a name

Here we are using TigerVNC as the client, not the popular TightVNC. Tiger is an evolution of TightVNC, which does not yet work with Bookworm.

Part 02

Using Ubuntu on Raspberry Pi



Phil King

MAKER

A long-time contributor to The MagPi, Phil is a freelance writer and editor with a focus on technology.

@philkingeditor

Exploring more features of the Ubuntu Desktop Linux OS on Raspberry Pi

In the first part of this tutorial series (magpi.cc/135), we showed how to install and set up the Ubuntu desktop operating system on a Raspberry Pi, and took you through some of its key features.

This time we'll look at how to access the command line, set up remote connections via SSH and RDP, install media codecs, use Raspberry Pi's GPIO pins, change the desktop environment, and use alternative ways of installing software.

01 Access the command line

As well as navigating Ubuntu's desktop GUI, you can open a CLI (command-line interface) Terminal to use powerful text-based commands

and get under the hood of the operating system. We'll be using it to perform various tasks.

To open a Terminal window in Ubuntu, click the Activities button at the top of the left sidebar and search for 'Terminal', then double-click the icon to launch it. Alternatively, use the keyboard shortcut: **CTRL+ALT+T**.

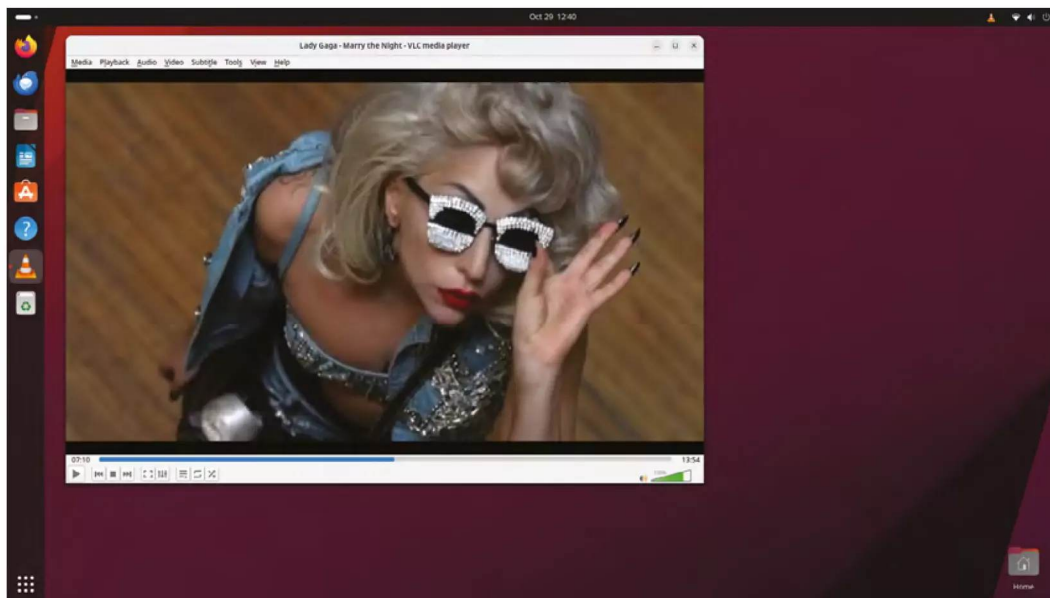
The terminal window will show the usual command-line prompt where you can enter a host of commands. To see the path for the current working directory (i.e. the folder you're in), enter `pwd`. To list its contents, enter `ls`. You can use the `cd` command to change directory; for example, `cd Downloads`.

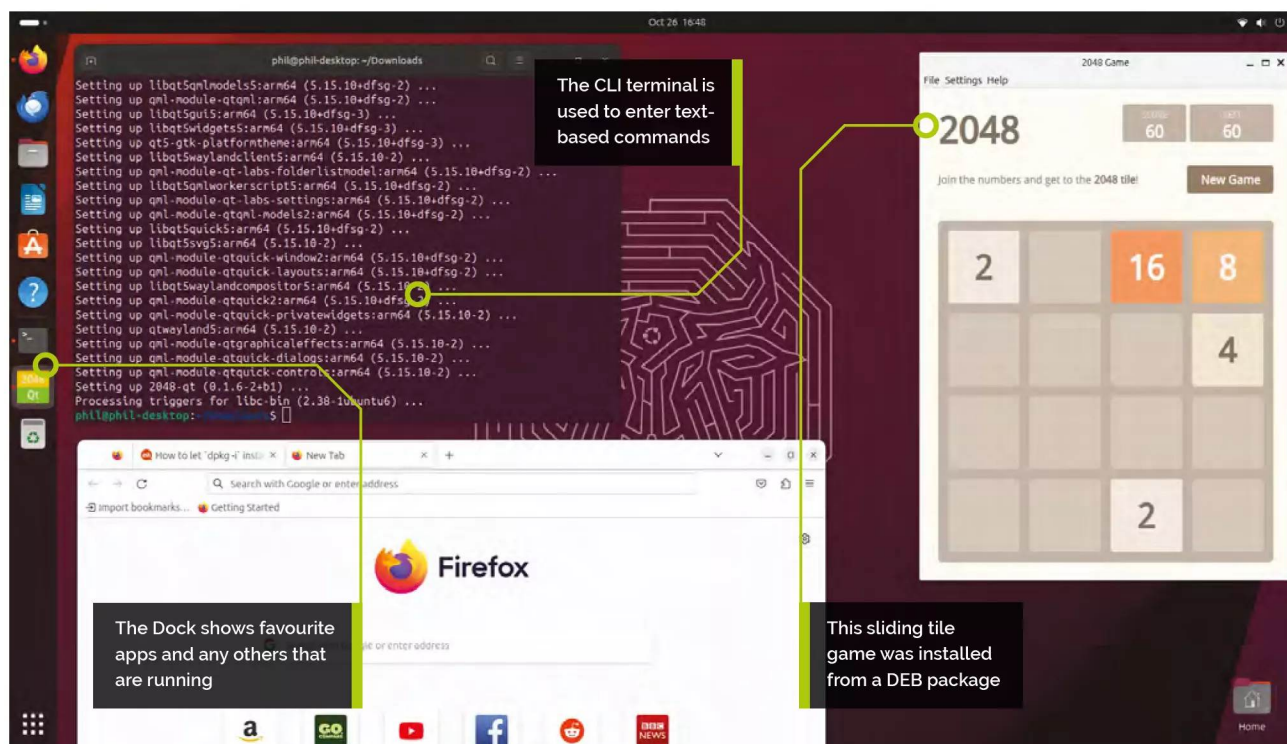
For lots more useful commands, check out the command-line tutorial in The MagPi issue 125 (magpi.cc/125).

You'll Need

- ▶ Raspberry Pi 4, 400, or 5
- ▶ 16GB (or greater) microSD card with Ubuntu Desktop 23.10 installed

- ▶ After installing Ubuntu-restricted-extras, you can play MP4 videos in a media player such as VLC





02 Install media codecs

As a pure FOSS (free open-source software) operating system, Ubuntu doesn't come with common non-free codecs pre-installed. So, if you attempt to play a media file such as an MP3 music track or MP4 video, you may see a message to that effect.

Luckily, there's an easy fix, as you can install a single package – called `ubuntu-restricted-extras` – to install all the essential media codecs. To do so, open a Terminal window and enter the following command:

```
$ sudo apt install ubuntu-restricted-extras
```

You'll be prompted to enter your Ubuntu user password. A list of packages to be installed will then be shown; press **ENTER** to confirm. You'll then need to confirm the EULA (press **TAB** to highlight 'OK', then press **ENTER**). Press **TAB** to highlight the options, then **ENTER** on 'Yes' to accept the terms. You should now be able to play files with those restricted codecs.

If you also want to play DVDs via a USB-attached drive, you'll need to install `libdvdread4`. Enter this Terminal command:

```
$ sudo /usr/share/doc/libdvdread4/install-css.sh
```

“ Since Ubuntu (like Raspberry Pi OS) is based on Debian Linux, you can also install software from DEB files ”

03 Install DEB files

Since Ubuntu (like Raspberry Pi OS) is based on Debian Linux, you can also install software from DEB files. To find packages suitable for Raspberry Pi, head to packages.debian.org/stable and look for any labelled 'arm64' or 'all'.

To install a downloaded DEB file, there are two main ways. You can use the Terminal command `sudo dpkg -i` followed by the filename (and path if not in the same directory). For instance, to install the 2048 Qt game:

```
$ sudo dpkg -i 2048-qt_0.1.6+b1_arm64.deb
```

If you get messages about missing dependencies, follow this up with:

```
$ sudo apt -f install.
```

...to install them. Alternatively, to open DEB files straight from the desktop, you can install the Gdebi tool using this command:

Top Tip

Desktop icons

When switching back to the default desktop environment, it may retain the icon settings etc. from the previous one. Revert them in the Tweaks app.

appended with '.local', or use its IP address. Then enter your username and the Remote Desktop password. Ignore any warnings that the certificate couldn't be verified and continue.

You should then be able to view Raspberry Pi's Ubuntu desktop on the remote machine. Note the sharing icon that appears in the top taskbar; click it when you want to close the remote connection.

07 Change the look

In Settings > Appearance, you can switch to dark mode and change the background image, while Settings > Ubuntu Desktop has options for desktop icon size/position, the Dock (left sidebar), and tiling. You can do more customisation by installing the `gnome-tweaks` tool:

```
$ sudo apt install gnome-tweaks
```

Opening the Tweaks app will enable you to alter the appearance of the cursor, icons, fonts and other settings. You can add more options from custom themes by installing the Extension Manager:

```
$ sudo apt gnome-shell-extension-manager
```

Open the Extension Manager app, click the Browse tab, then User Themes and Install. In the web browser, download a theme or icons from **gnome-look.org**, such as Orchis. Extract the file (double-click in the file browser).

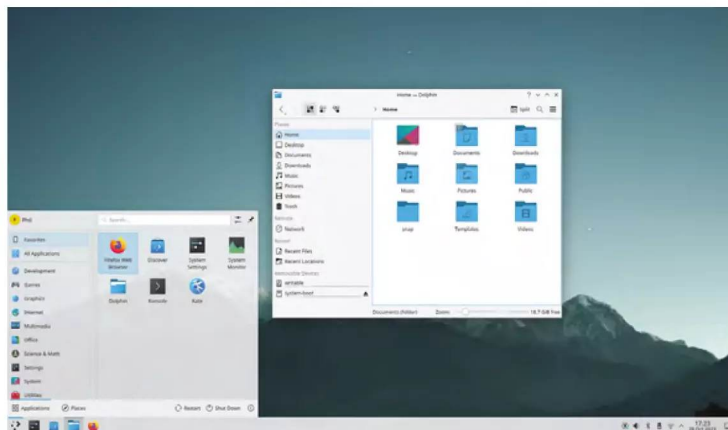
In Terminal, create two new directories (for GTK/GNOME Shell and icon/cursor-based themes respectively) in your home folder:

```
$ mkdir ~/.themes
$ mkdir ~/.icons
```

In the Files browser, press **CTRL+H** to show hidden files/folders. Now move the extracted theme folder(s) into the relevant folder, such as `.themes` for Orchis. Reopen the Tweaks app, and you should now see new options in the Shell and Legacy Apps drop-downs. You may get a system prompt to install missing snaps; click 'Yes'.

08 Switch desktop environments

You can even switch to a different desktop environment (DE) from the user login screen – to get back there, click the top-right power icon and



Log Out. Click your username, then the gear icon at the bottom right of the login screen to select a DE from the menu.

▲ You can switch to a different desktop environment, such as KDE Plasma

By default, there's only a choice of two DEs: Ubuntu (the default) and Ubuntu on Xorg (which switches the display manager from Wayland to X11). You can install extra environments from the command line, however. For example:

```
$ sudo apt install kde-plasma-desktop
```

You should then be able to select it – Plasma (X11) in this case – from the login screen DE selection menu.

09 Coding with Python

You can still use Raspberry Pi's GPIO pins in Ubuntu, but you'll need to install a new Python module:

```
$ sudo apt install python3-igpio
```

You can then import it like any other Python module at the top of your code, with `import igpio`. You can write programs in Ubuntu with a text editor such as nano, or a Python IDE such as Spyder: `sudo apt install spyder`.

For details of the `igpio` module's functionality, see **magpi.cc/igpio**. For some code examples, visit **magpi.cc/igpioexamples**. The 'basic GPIO example' will blink an LED connected to GPIO 23 and a GND pin. Paste it into a new document, named `blink.py`, using Nano (or another text editor), then run it with the command `python3 blink.py`. Alternatively, you can install a Python IDE such as Spyder – `sudo apt install spyder` – and run it from there. **14**

Top Tip

IP address

To obtain Raspberry Pi's IP address, to connect from another device via SSH or Remote Desktop, enter the command `hostname -I`.

Raspberry Pi: Mecanum robot with sensors



**Stewart
Watkiss**

Also known as Penguin Tutor. A Maker and YouTuber who loves all things Raspberry Pi. Author of Learn Electronics with Raspberry Pi.

penguintutor.com

[@stewartwatkiss](https://twitter.com/stewartwatkiss)

You'll Need

- ▶ HC-SR04 Ultrasonic distance sensor magpi.cc/hcsr04
- ▶ 330 ohm resistor magpi.cc/330r
- ▶ 470 ohm resistor magpi.cc/470r

- ▶ Figure 1. The HC-SR04 module has a transmitter and receiver. It sends out ultrasonic pulses and detects an echo response

Expanding on our mecanum robot project, add a distance sensor to avoid crashing into obstacles. Learn how collision avoidance is used to make driving safer

The mecanum robot in the earlier tutorial (MagPi issue #135, magpi.cc/135) could be controlled with a gamepad. If you didn't react fast enough, then it was quite easy to crash the robot. Adding an ultrasonic distance sensor means that the robot can automatically stop before colliding with a wall or other object.

The sensor used is similar to those found in car parking sensors. The robot senses the distance to objects in front and automatically stops forward movement when an object is too close. With additional code, this could be used as the basis of a robot that could navigate a maze.

01 Ultrasonic distance sensors

There are various different sensors that can be used to detect distance. The one used here is based on ultrasonic distance measurement, and works by emitting an ultrasound signal and then listening for an echo response. The time

taken between the signal being transmitted and the response received can be used to work out the distance to an object.

Figure 1 shows the HC-SR04 sensor used for this project. There are different models of the HC-SR04, some of which can work down to 3.3V. This circuit is based around the 5V version for maximum compatibility.

02 Sensor signals

In addition to the power supply, the HC-SR04 sensor module has two pins for an input and output signal. The trigger needs to be sent a short 10µs pulse to start the measurement. The transmitter then sends eight sonic bursts and the receiver listens for an echo.

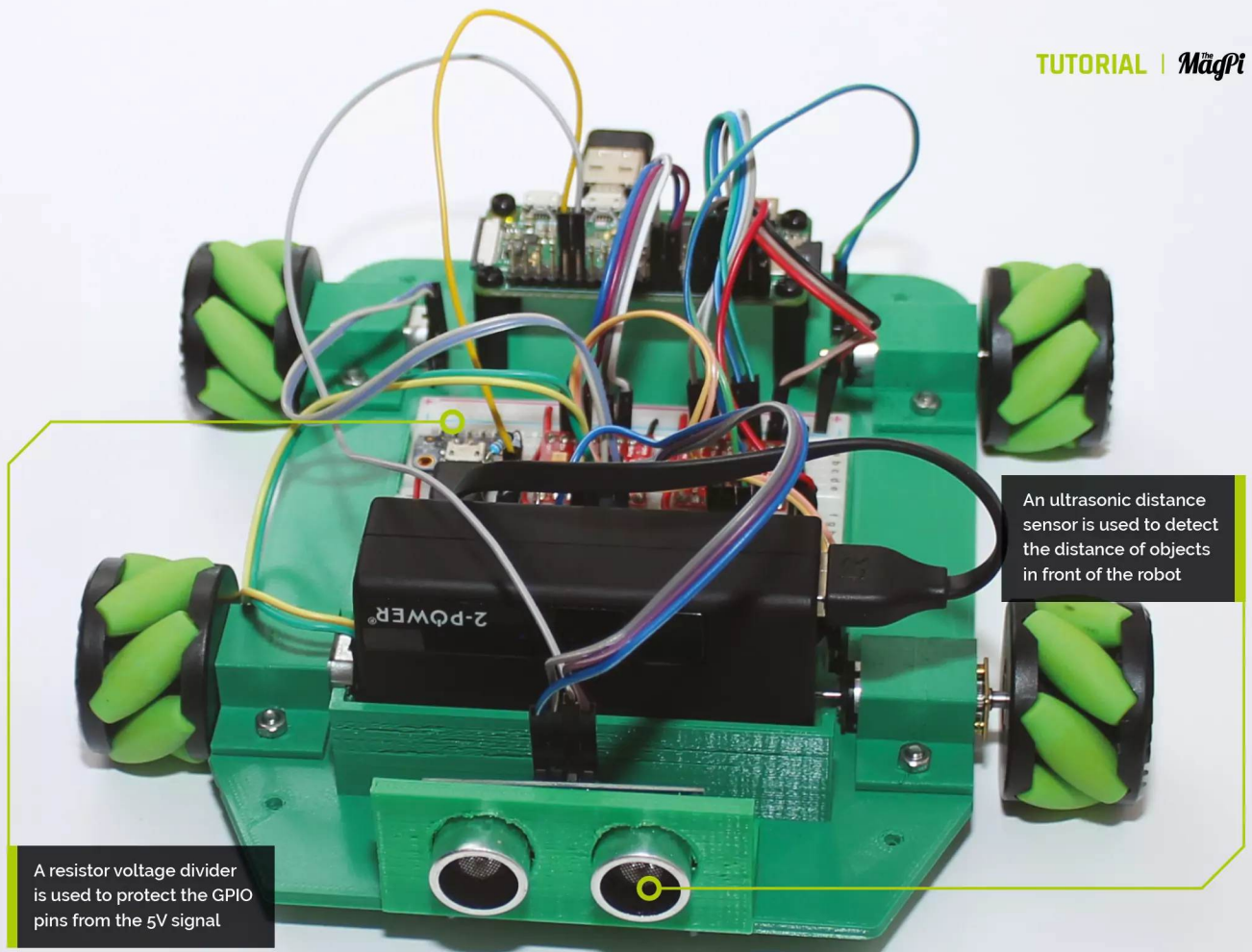
After determining how long it took to receive a response at the receiver, the echo pin is then raised high for a length of time proportional to the distance. This is shown in **Figure 2**.

The output from the echo pin is 5V, which is too high for the GPIO pins. Therefore a resistor voltage divider can be used to drop the signal to approximately 3.3V.

03 Wiring the distance sensor

The distance sensor needs a 5V power supply. As the robot can have up to 6V, this is connected after the diode used to connect to the 5V pin on your Raspberry Pi.





The distance sensor needs to be connected to two GPIO ports. The chosen pins are GPIO 5 (physical pin 29) for the trigger, and GPIO 6 (physical pin 31) for the echo.

The trigger pin works with a 3.3V input signal so is connected directly to the GPIO pin. To drop the echo signal from 5V to 3.3V you need two resistors. If using USB for power then there is not much space for these, but they can be squeezed into the gap between the USB connector and the motor drivers as shown in **Figure 3**.

04 Resistor voltage divider

The breadboard diagram shows the resistors with long leads, which are needed to get them to fit into such a tight space. This makes it difficult to see how it is wired, so the schematic diagram of the voltage divider is also shown in **Figure 4**.

The output is from pin 3 of the HC-SR04, which goes through a voltage divider created by R1 and R2. The voltage is shared across these resistors proportional to their resistance. The voltage across R2 will be:

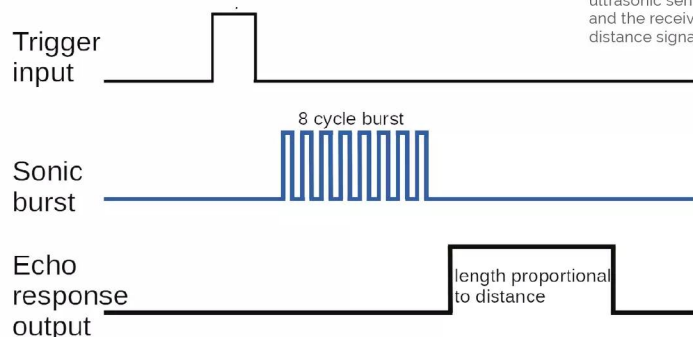
$$V \times (R2 / (R1+R2))$$

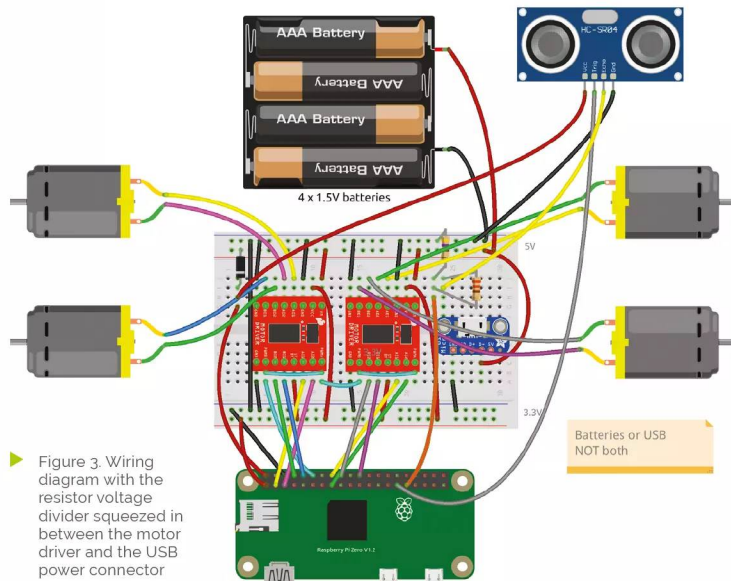
... which is approximately 3V. This is the signal that is passed to GPIO 6.

05 Using PiGPIO

GPIO Zero is used to control the motor driver and can also be used for the ultrasonic sensor. Using its default native pin configuration GPIO Zero can communicate with most sensors with acceptable performance. For the ultrasonic

▼ Figure 2. Waveforms showing signals used to trigger the ultrasonic sensor and the received distance signal





► Figure 3. Wiring diagram with the resistor voltage divider squeezed in between the motor driver and the USB power connector

sensor the timing is critical, and using the native library can give incorrect results. Instead, you can use PiGPIO. This allows direct memory access (DMA), which reduces the requirement on the processor. To use PiGPIO the PiGPIO daemon first needs to be run as root, and then the appropriate `pin_factory` needs to be set in your Python program. The daemon is started using

```
sudo pigpiod
```

06 Using Python Pin Factory

With the `pigpiod` daemon running, then the device factory can be set within the Python code. First, you need to import the PiGPIOFactory library from `gpiozero.pins.pigpio` import `PiGPIOFactory`. Then set the `pin_factory` using:

```
Device.pin_factory = PiGPIOFactory()
```

This will then use PiGPIO for all devices that use the GPIO pins from GPIO Zero.

07 Measuring the distance

To measure the distance first create a `DistanceSensor` object:

```
dist_sensor = DistanceSensor(echo=6, trigger=5)
```

The distance can then be queried by looking at the `distance` property:

```
dist_sensor.distance
```

The returned value is in meters. The maximum distance set in the code is 0.09, which works out as 9cm.

08 Using Pygame Zero

The earlier version of the robot software was written as a command-line Python program. One issue with that is that the code would stop running whilst waiting for the next key press. This means it is not possible to check for a potential collision whilst the robot is moving.

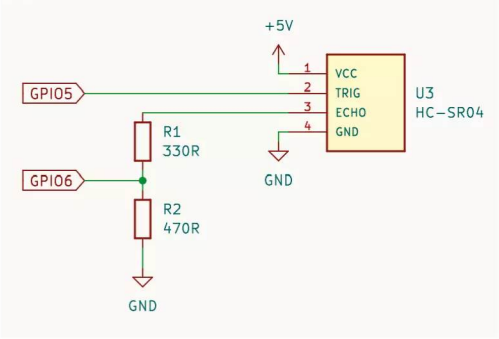
An alternative is to use Pygame Zero, which can create a GUI application which can respond to keyboard presses as well as regularly checking the value from the distance sensor. It also provides a way to create a graphical screen which can be controlled using a mouse or touchscreen. A screenshot is included in **Figure 5**.

09 How to code for Pygame Zero

The main difference with Pygame Zero is that instead of running the program in sequence the code is event-driven. The main way that code is run is through two functions `draw()` and `update()` which are each called approximately 60 times per second (depending upon computer performance). The `draw()` function is used for displaying the interface on the screen and the `update()` function normally handles any other code that needs to be run.

It is also possible to set code that runs when an event occurs which in this example is the `on_key_`

▼ Figure 4. Schematic diagram showing the wiring between the GPIO pins and the ultrasonic sensor. The two resistors form a voltage divider reducing the voltage



`down(key)` function which is called each time that a key is pressed on the keyboard.

10 The update function

The collision avoidance code is included in the update function. The code detects if the distance is less than the minimum distance. If it is then it also checks that the robot is going forward, otherwise you would end up with a robot that couldn't reverse either.

The code includes error handling by placing checks for the sensor inside a try block. If the code in the try block has an exception, then instead of crashing the program it will ignore the rest of that block and call the except block instead. In this case, if the sensor isn't detected, then it prints out an error message.

11 Running the code

To run the code then it needs to be run using `pgzrun`. This can be done using Pygame Zero mode in the Thonny editor, or by running on the command line using:

```
pgzrun mecanum-pgz.py
```

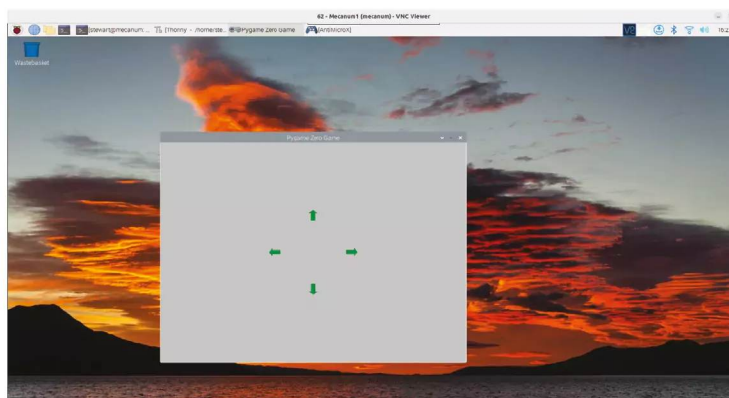
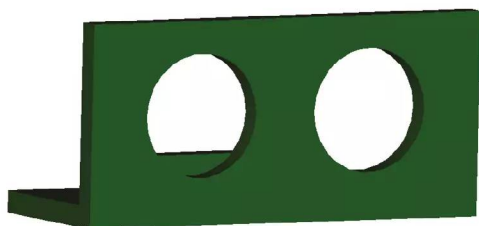
The program can still be used with AntiMicroX to allow the robot to be controlled using a gamepad.

If you get an error, then make sure you started pigpiod first, as mentioned earlier.

12 Mounting the sensor

Finally, you will need somewhere to mount the sensor. If you are using the 3D-printed chassis then there are mounting holes included for a printable bracket. There are small holes

▼ Figure 6. The bracket is designed to screw to the front of the chassis to mount the ultrasonic sensor



▲ Figure 5. A very basic GUI display has been created using Pygame Zero. This allows the sensor code to run regularly to check for a possible collision

included on the module PCB to allow mounting, but due to their small size compared to the resolution of the 3D printer these were not used, and the board can be held with a small amount of tape or adhesive putty. The bracket is shown in **Figure 6**. Download the bracket design file from magpi.cc/robotmecanum.

For a different chassis, it may be possible to drill holes for the sensor or make your own bracket using thick cardboard. [\[7\]](#)

Pironman Mini PC Case for Raspberry Pi 4



Aluminum Tower Case

Ice Tower Cooler

Support M.2 SATA SSD

Programmable OLED Display

IR Receiver

Safe Shutdown

mecanum-pgz.py

> Language: Python Pygame Zero

DOWNLOAD THE FULL CODE:

 magpi.cc/mecanumrobotgit

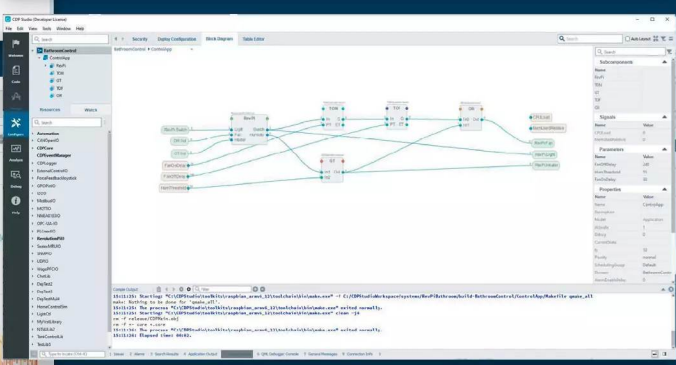
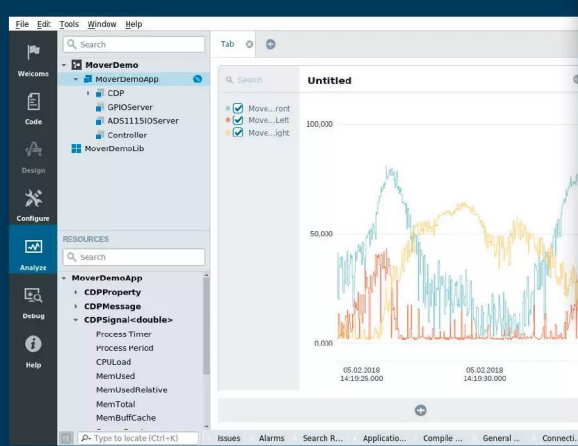
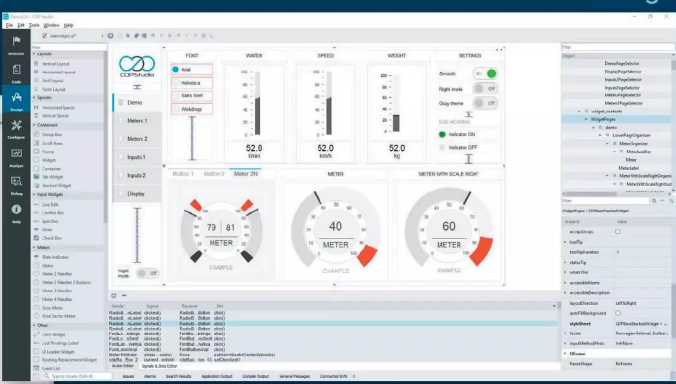
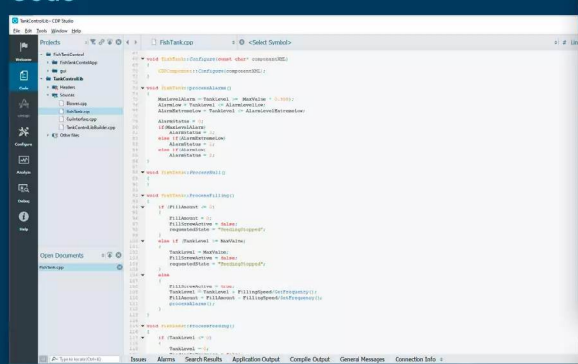
```

001. # Minimal version of mecanum-pgz
002. from gpiozero.pins.pigpio import PiGPIOFactory
003. from gpiozero import Device, PWMOutputDevice,
004. Motor, DistanceSensor
005.
006. WIDTH=800
007. HEIGHT=600
008.
009. Device.pin_factory = PiGPIOFactory()
010.
011. pwm_pin = 18
012. m_f_l = (2,3)
013. m_f_r = (22,23)
014. m_r_l = (14,15)
015. m_r_r = (24,25)
016.
017. motors = [
018.     Motor(m_f_l[0], m_f_l[1], pwm=False),
019.     Motor(m_f_r[0], m_f_r[1], pwm=False),
020.     Motor(m_r_l[0], m_r_l[1], pwm=False),
021.     Motor(m_r_r[0], m_r_r[1], pwm=False)
022. ]
023.
024. pwm_out = PWMOutputDevice (pwm_pin)
025. dist_sensor = DistanceSensor(echo=6, trigger=5)
026.
027. arrows = [
028.     Actor("arrow.png", center=(500,300)), # Right
029.     Actor("arrow.png", center=(400,200)), # Up
030.     Actor("arrow.png", center=(300,300)), # Left
031.     Actor("arrow.png", center=(400,400)) # Down
032. ]
033. arrows[1].angle = 90
034. arrows[2].angle = 180
035. arrows[3].angle = 270
036.
037. # distance in meters - a value between 0 and 1
038. min_distance = 0.09
039.
040. # list to convert key into motor on/off values to
    correspond with direction
041. # direction based on number keypad
042. direction = {
043.     keys.K_2 : (-1, -1, -1, -1), # Backwards
044.     keys.KP2 : (-1, -1, -1, -1), # Backwards
045.     keys.K_4 : (-1, 1, 1, -1), # Left
046.     keys.KP4 : (-1, 1, 1, -1), # Left
047.     keys.K_5 : (0, 0, 0, 0), # Stop
048.     keys.KP5 : (0, 0, 0, 0), # Stop
049.     keys.K_6 : (1, -1, -1, 1), # Right
050.     keys.KP6 : (1, -1, -1, 1), # Right
051.     keys.K_8 : (1, 1, 1, 1), # Forwards
052.     keys.KP8 : (1, 1, 1, 1), # Forwards
053. }
054.
055. # Track direction
056. current_direction = (0, 0, 0, 0)
057.
058. # speed is as a percentage (ie. 100 = top speed)
059. speed = 50
060. pwm_out.value = speed/100
061.
062. def draw():
063.     screen.fill((192,192,192))
064.     for arrow in arrows:
065.         arrow.draw()
066.
067. def update():
068.     global current_direction
069.     try:
070.         if (dist_sensor.distance <= min_distance):
071.             # Is robot going forward (or diagonal)
072.             if (current_direction[0] == 1 and
current_direction[1] == 1):
073.                 print ("Warning - crash imminent")
074.                 # Force stop using stop key number
075.                 set_direction (direction[keys.K_5])
076.     except:
077.         print ("No distance sensor detected")
078.
079. def on_key_down(key):
080.     global speed, current_direction
081.     # Get next key pressed
082.     if (key in direction.keys()) :
083.         set_direction(direction[key])
084.
085. def set_direction(direction):
086.     global current_direction
087.     current_direction = direction
088.     for i in range (0, 4):
089.         if direction[i] == -1:
090.             motors[i].backward()
091.         elif direction[i] == 1:
092.             motors[i].forward()
093.         else:
094.             motors[i].stop()
095.

```


Design

Code



Configure

Analyze

PROFESSIONAL CONTROL SYSTEM DEVELOPMENT TOOL

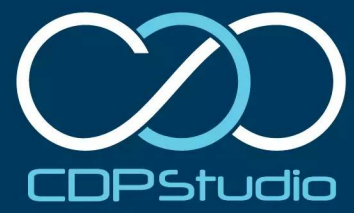
Home projects made easy.

CDP Studio, a great software development tool for your home projects. Build systems for Raspberry Pi, use C++ or NoCode programming, open source libraries, out of the box support for GPIO, I2C, MQTT, OPC UA and more. Create beautiful user interfaces. Built for industrial control system development, **FREE for home projects.**

cdpstudio.com

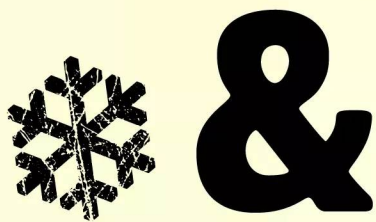
Tel: +47 990 80 900 • info@cdpotech.com

CDP Technologies AS // Hundsværgata 8, 6008 Ålesund, Norway





HEATING



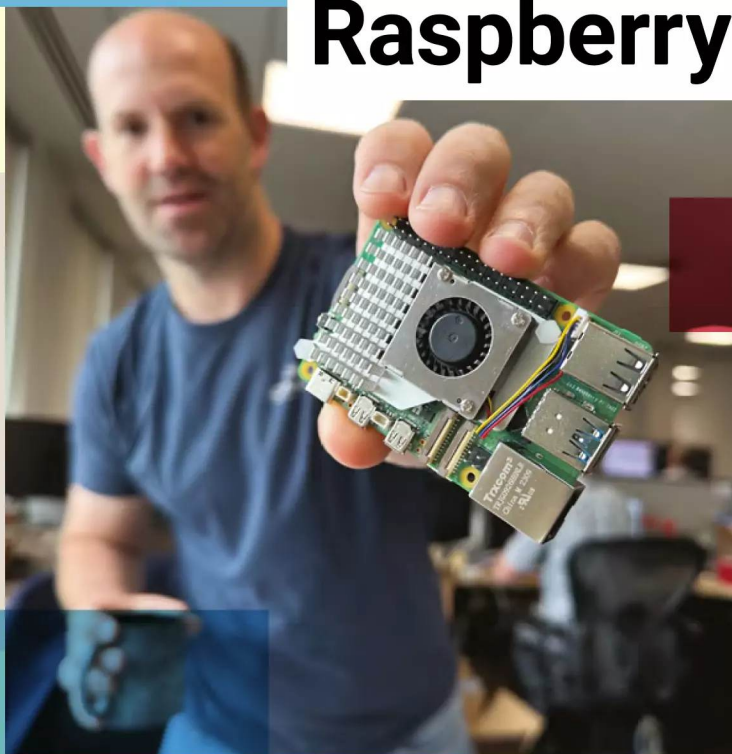
With the release of Raspberry Pi 5, we're introducing two new official hardware solutions for cooling.

COOLING



Raspberry Pi 5

BY ALASDAIR ALLAN

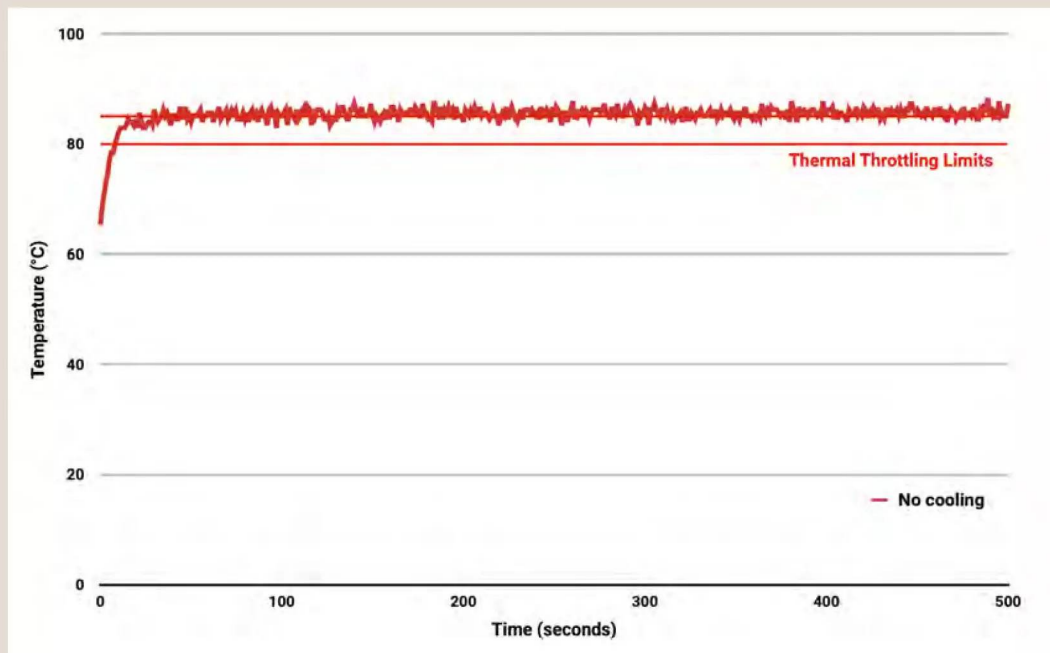


▲ Eben showing off a prototype of the Active Cooler for Raspberry Pi 5

Every time a new Raspberry Pi is released, there are questions about thermal control of the new board. People want to know whether it's necessary, and if so, what you'll need to do to make it happen.

For normal usage of your Raspberry Pi, adding cooling is entirely optional. The idle performance of a Raspberry Pi 4, and a Raspberry Pi 5 is about the same, and under typical loads, Raspberry Pi 5 will run cooler than a similarly loaded Raspberry Pi 4. However, a heavy continuous load will mean that the board could potentially go into thermal throttling. Throttling happens as there are software controls to limit CPU speeds if things get start to get too toasty. Although, even when fully throttled, a Raspberry Pi 5 is still going to run faster than a Raspberry Pi 4.

But data makes everything better, so I decided to grab some early production hardware and run some tests to help you make up your mind about whether you're going to need to cool your Raspberry Pi 5.



◀ Processor temperature in °C against time in seconds. At time $T=T_0$ stress testing was started

HOW WE MEASURE CPU TEMPERATURE

The `vcgencmd` command is an amazingly useful source of information about the things that are happening on your Raspberry Pi, and the Python bindings (magpi.cc/vcgencmd) surface all of that and let you programmatically monitor pretty much everything that needs monitoring. Here we're going to use the `vcgencmd` Python bindings to monitor and log the temperature, along with the current CPU clock speed and throttling state, to a file (see `measure_temp.py`, p69).

Once we have the script up and running in a Terminal window, we can open up another and kick off a stress test on all four cores to load the CPU. To do that, I'm going to use the `stress` (magpi.cc/stress) command line tool to impose a heavy workload on all four of the CPU cores.

```
$ sudo apt install stress
$ stress --cpu 4
```

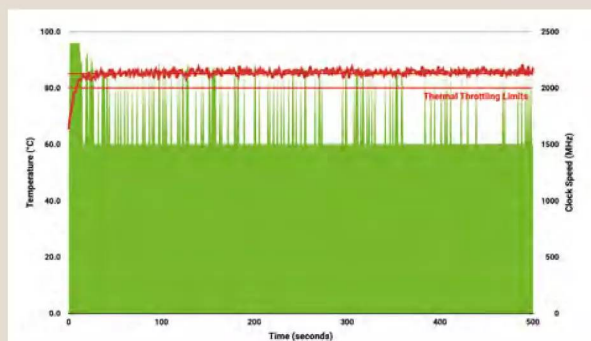
To prevent overheating, all Raspberry Pi boards begin to throttle the processor when the temperature reaches 80°C, and throttle even further when it reaches the max temp of 85°C.

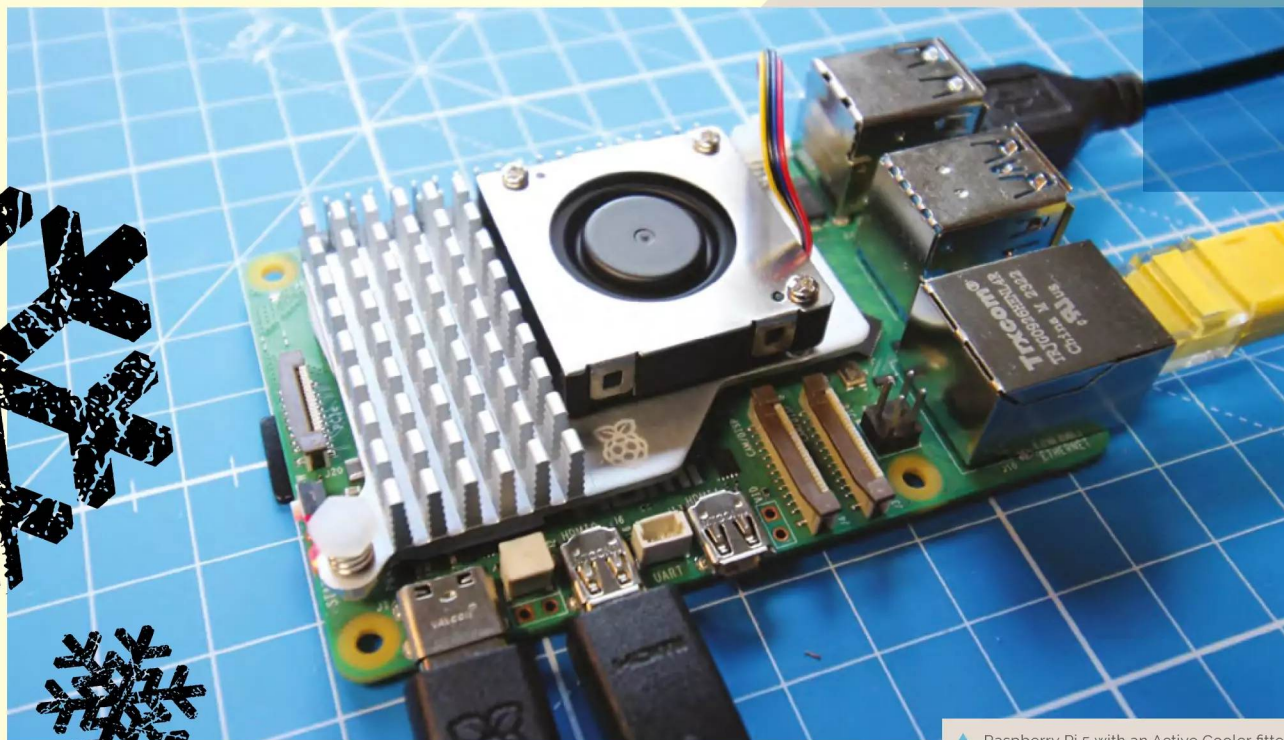
NO COOLING

The first thing to do is to measure what happens when your Raspberry Pi 5 is not cooled. Without any cooling in place, Raspberry Pi 5's CPU idle temperature is around 65°C when sitting out in the open air on the lab bench.

For normal use adding cooling is optional. If you're watching a YouTube video, or working on the desktop, you aren't going to be stressing the CPU like we did in this test. But, unsurprisingly, with the heavy sustained load we're imposing on the CPU with no cooling the maximum temperature climbs to and then remains stable just above the 85°C thermal limit during extended testing. This leads to sustained thermal throttling after the temperature reported by the processor rises above the throttling limits.

▶ The clock speed intermittently throttled on an uncooled Raspberry Pi undergoing a stress test to maintain a constant maximum thermal temperature





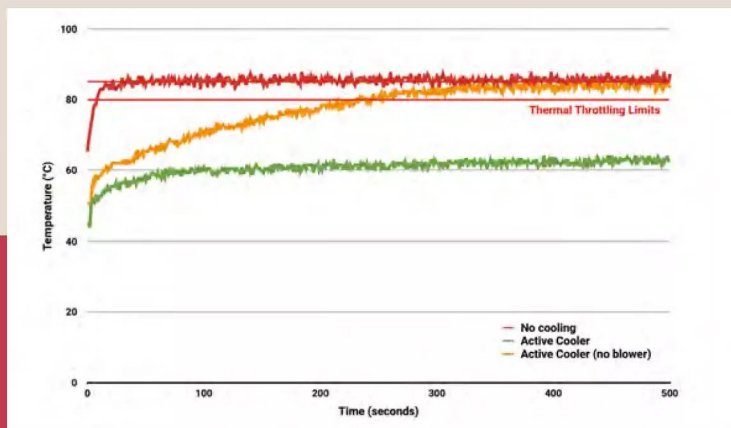
▲ Raspberry Pi 5 with an Active Cooler fitted

FITTING THE ACTIVE COOLER

I then ran the same test with managed active cooling using the new Active Cooler, and then with the Active Cooler still fitted but with the blower's fan disconnected. Both these tests were done with the Raspberry Pi sitting in the open air on the lab bench.

The Active Cooler is a single-piece anodized aluminium heatsink with an integrated blower. It has pre-applied thermal pads for heat transfer, and is mounted to the Raspberry Pi 5 board directly using spring-loaded push pins. It is actively managed by Raspberry Pi's firmware. At 60°C the blower's fan

▼ Processor temperature in °C against time in seconds. At time T_0 stress testing was started



will be turned on, at 67.5°C fan speed will be increased, and finally at 75°C the fan will increase to full speed. When the temperature drops back below these limits, the blower's fan will spin down automatically.

Thanks to the passive heatsink, with the Active Cooler fitted we see a much lower idle temperature, around 45°C. During extended testing under load, the fan of the Cooler spins up at low speed to stabilise the CPU temperature at 60°C, with a maximum temperatures of 62 to 63°C being seen during the tests.

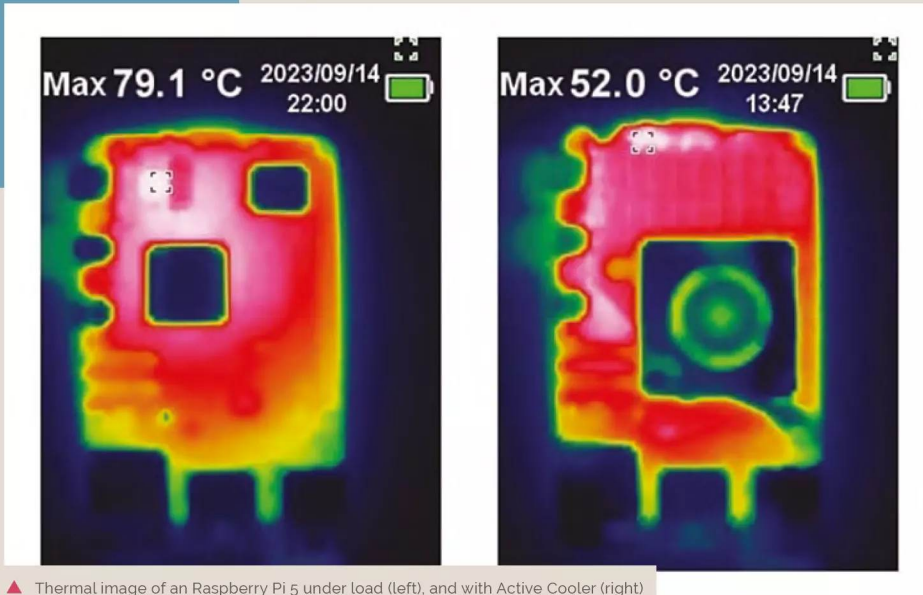
Noise levels of between 35 to 40dB were measured during the load test while the fan was in operation – that's about as much noise as you'll make turning the page of a book. During the extended stress testing the fan never actually needed to run at full speed to maintain temperature control of the Raspberry Pi.

Unplugging the fan and relying solely on the passive cooling provided by the aluminium heatsink, the idle temperatures were similar, but under extended load the CPU temperature eventually reaches the point at around $T_0 + 200$ seconds where thermal throttling occurs.

Reattaching the cable causes the fan to spin up to full speed immediately, and with the load removed, the CPU is cooled back to an idle temperature of around 45°C within a further 300 seconds, with the fan spinning back down to lower speeds as the temperature falls back to normal.



FOR NORMAL USAGE OF YOUR RASPBERRY PI, ADDING COOLING IS ENTIRELY OPTIONAL



▲ Thermal image of an Raspberry Pi 5 under load (left), and with Active Cooler (right)

BUT WHAT ABOUT A HAT?

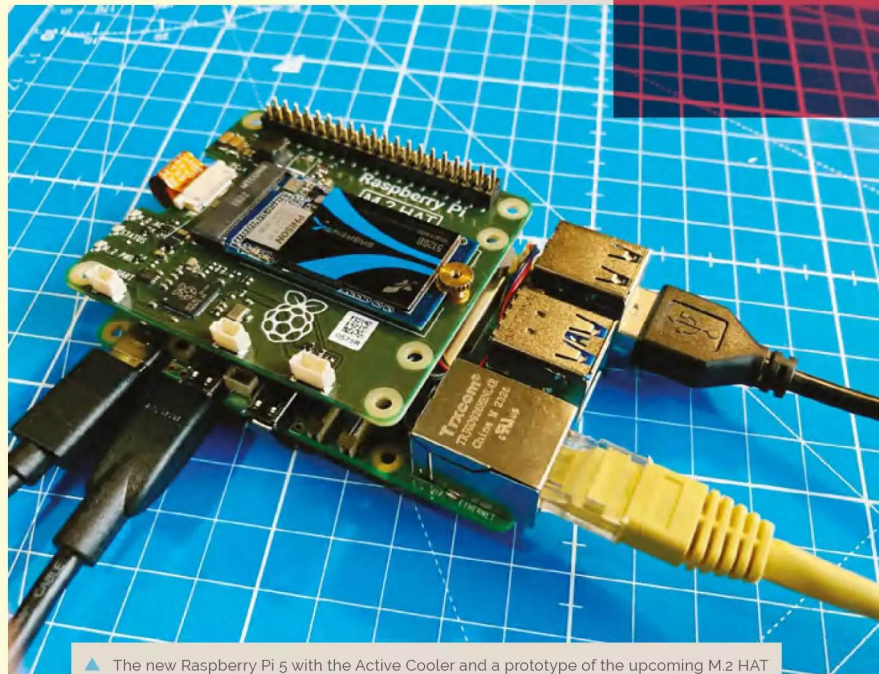
The big question a lot of folks will have at this point is, what happens when you add a HAT?

Well, you can mount a HAT above the Active Cooler using a set of 16mm GPIO extenders. Inevitably there is some disruption to the air flow which will cause the Raspberry Pi to run hotter, but the Active Cooler is still able to handle extended stress tests without significant temperature rises.

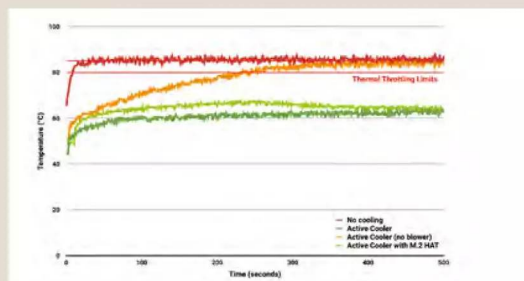
Testing was done with a prototype of the new M.2 HAT, booting Raspberry Pi from the NVMe drive – both because I happened to have one on my desk, and also because this is going to be a pretty common use case for the Raspberry Pi 5 – and just a reminder: the only thing you really need to remember about the prototype M.2 HAT is that the production version will almost inevitably look nothing like the one in this picture!

With the M.2 HAT fitted above the Active Cooler, the idle temperature of the Raspberry Pi was slightly higher than without the HAT present, at around 49 °C.

Under sustained load the CPU temperature initially rose to the second 67.5 °C trigger point, spinning the blower's fan up from low to its middle speed. However, this quickly dropped the CPU temperature below the trigger point, which in turn dropped the fan speed back to its lower setting. The CPU temperature then stabilised at around 64 °C for the remainder of the sustained testing.



▲ The new Raspberry Pi 5 with the Active Cooler and a prototype of the upcoming M.2 HAT



◀ Processor temperature in °C against time in seconds. At time $T = T_0$ stress testing was started



▼ Raspberry Pi 5's clip-together case comes with a built-in fan



USING THE NEW CASE

Next on the test bench was the new fan case. I removed the Active Cooler from my board, and went ahead and fitted the Raspberry Pi 5 inside the new case. The new case comes as four components; the base which the Raspberry Pi clips into, then a frame and fan assembly, and finally a lid that clips on top.

Like the Active Cooler, the fan assembly is actively managed by Raspberry Pi's firmware: at 60°C the blower's fan will be turned on, at 67.5°C fan speed will be increased, and finally at 75°C the fan increases to full speed. When the temperature drops back below these limits the fan will spin down automatically.

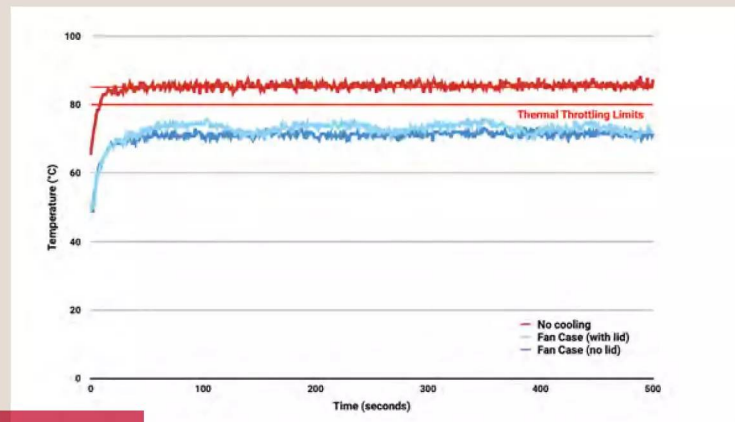
Testing was carried out in the same fashion as before, first with the fan assembly in place, but with the lid removed. Then again with both

LIKE THE ACTIVE COOLER, THE FAN IS ACTIVELY MANAGED BY RASPBERRY PI'S FIRMWARE

fan assembly in place, and this time with the lid clipped on top.

Using the fan case we see idle temperatures a couple of degrees hotter than with the Active Cooler on its own, at around 48°C. With the lid removed we see maximum temperature of approximately 72°C under sustained load, and with the lid in place we see a marginally higher maximum of around 74°C under load.

We can see that while temperature under load is higher than with the Active Cooler, the maximum temperature under load is still well below the 80 and 85°C throttling temperatures.

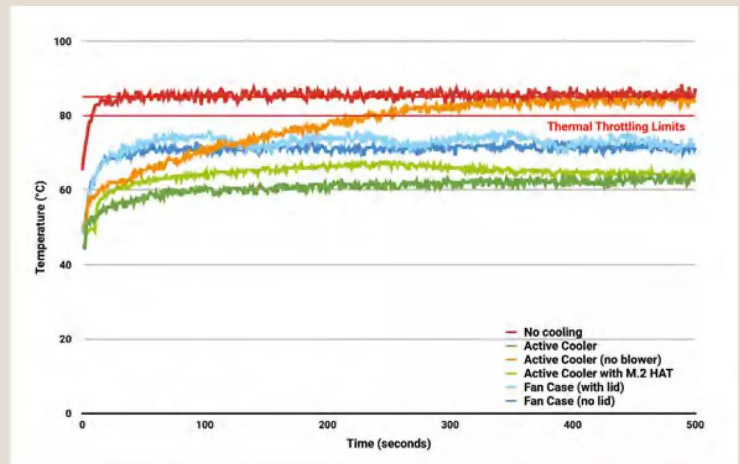


◀ Processor temperature in °C against time in seconds. At time T=T₀ stress testing was started

CONCLUSIONS

For normal use adding cooling is optional, although performance may be improved with the addition of active cooling. However a heavy continuous load, such as rebuilding the Linux kernel, will force the new Raspberry Pi 5 into thermal throttling. For heavy loads thermal throttling can extend processing times, and passive cooling is probably going to be insufficient thermal management for heavy loads that extend beyond 200 or 300 seconds of duration, with active cooling necessary to prevent thermal throttling from occurring.

When deciding on a cooling solution you should consider what sort of use you're going to put your Raspberry Pi 5 to, and make a decision on cooling based on that, rather than just arbitrarily adding cooling. Because for a lot of day-to-day use cases, it's not going to be needed.



Cooling of any type isn't mandatory, no harm will come to your Raspberry Pi if it's left uncooled — and even while throttling under heavy load, a Raspberry Pi 5 is still faster than an unthrottled Raspberry Pi 4.

▲ Processor temperature in °C against time in seconds. At time $T = T_0$ stress testing was started

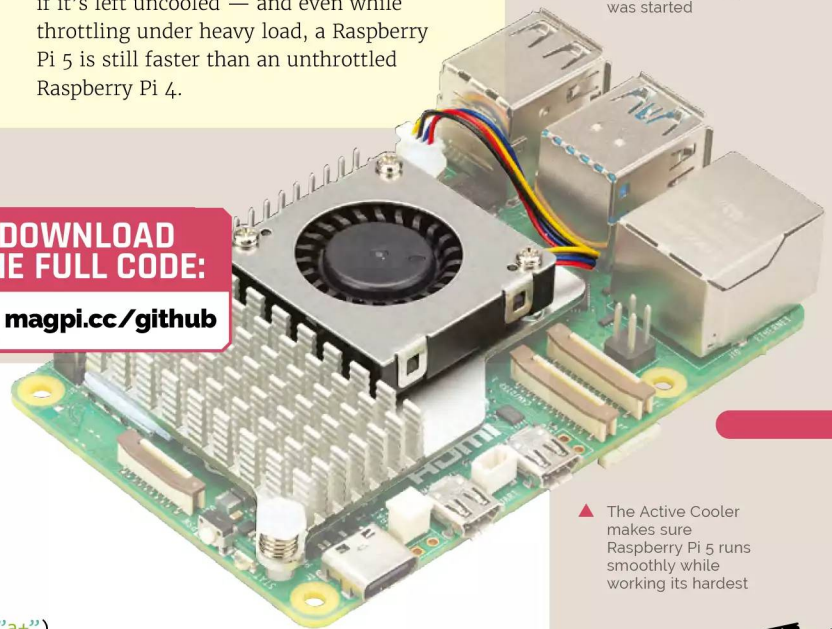
measure_temp.py

► Language: Python

DOWNLOAD THE FULL CODE:

magpi.cc/github

```
001. import sys
002. import os
003. import time
004.
005. from vcgencmd import VcgenCmd
006.
007. def main():
008.     start_time = time.time()
009.     fb = open("/home/pi/readings.txt", "a+")
010.     fb.write("Elapsed Time (s),Temperature (°C),Clock Speed (MHz),Throttled\n")
011.     vcgm = VcgenCmd()
012.     while True:
013.         temp = vcgm.measure_temp()
014.         clock = int(vcgm.measure_clock('arm')/1000000)
015.         throttled = vcgm.get_throttled()['breakdown']['2']
016.
017.         string = "%.0f,%.0f,%.0f,%.0f\n" % ((time.time() - start_time), temp, clock, throttled)
018.         print(string, end="")
019.         fb.write(string)
020.         time.sleep(1)
021.
022. if __name__ == '__main__':
023.     main()
```



▲ The Active Cooler makes sure Raspberry Pi 5 runs smoothly while working it's hardest

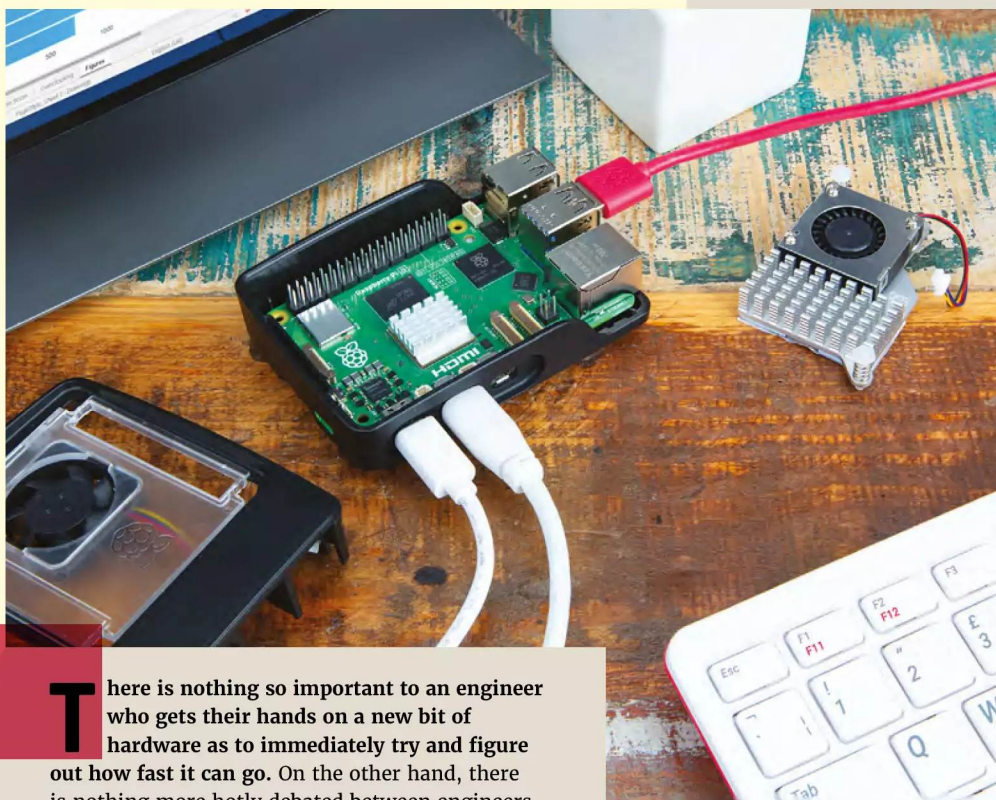


BENCHMARKING

Raspberry Pi 5

Raspberry Pi 5 benchmarks reveal a significant performance boost.

BY ALASDAIR
ALLAN



There is nothing so important to an engineer who gets their hands on a new bit of hardware as to immediately try and figure out how fast it can go. On the other hand, there is nothing more hotly debated between engineers as to how to measure how fast things go. But like many things, what you should measure really depends on how and why you will be using the hardware in the first place.

As our American friends might say, your mileage may vary (magpi.cc/ymmv).





BENCHMARKING WITH GEEKBENCH

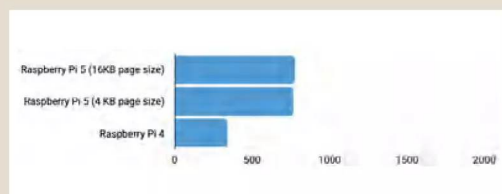
There are many benchmarking frameworks, but Geekbench is mine (geekbench.com), and why it's mine comes back to the question of how and why. Instead of just straight-up crunching some pointless numbers, Geekbench takes a stab at running tests that attempt to reflect how people normally use their computers: loading a website, rendering a PDF, and adding filters to an image.

Like a lot of other benchmarks, Geekbench headline scores are split into single-core and multi-core scores. However, just due to the nature of the benchmark, Geekbench scores can vary a lot from run to run. Setting `force_turbo=1` in `config.txt` and avoiding things like software updates somewhat reduces the noise between runs, but the best strategy is to average multiple runs. Which is what we've done.

SINGLE CORE

Single-core scores measure the processing power of one CPU core and are more relevant for applications that are lightly threaded, meaning they rely mostly on a single core to process instructions.

Over one hundred runs of Geekbench 6 we saw an average score of 764 ± 6 for Raspberry Pi 5 using a 4KB page size, and 774 ± 6 using a 16KB page size. That's a $\times 2.4$ speed increase over Raspberry Pi 4.



SINGLE CORE	SCORE
Raspberry Pi 5 (16KB page size)	774
Raspberry Pi 5 (4KB page size)	764
Raspberry Pi 4	340

If you're running the 64-bit distribution of Raspberry Pi OS, it ships with 16KB page size enabled, which gives slightly improved performance at the expense of compatibility with older ARMv7 32-bit binaries.

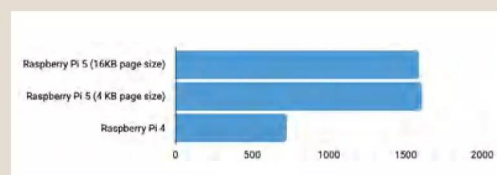
So we've also gone ahead and tested a Raspberry Pi 5 computer using both 16KB and 4KB page size. To do that we've used an internal build of Geekbench 6.2 which is patched to support 16KB page sizes (the off-the-rack release won't work with 16KB page size, and needs you to reboot and use a 4KB page size).

For Geekbench benchmarks, bigger numbers are better!

MULTI CORE

Multi-core scores attempt to measure the more realistic case where jobs are being distributed across all the cores of the board's CPU. These sorts of scores are more relevant for heavily threaded applications such as web browsers, which generally try to offload individual tabs into separate process threads.

Over one hundred runs of Geekbench 6 we saw an average score of $1,604 \pm 22$ for Raspberry Pi 5 using a 4KB page size, and 1588 ± 63 using a 16KB page size. That's a $\times 2.2$ speed increase over Raspberry Pi 4.



MULTI CORE	SCORE
Raspberry Pi 5 (16KB page size)	1588
Raspberry Pi 5 (4KB page size)	1604
Raspberry Pi 4	723



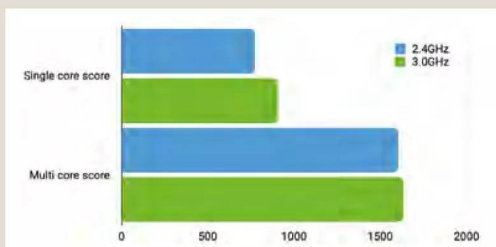
OVERCLOCKING TO 3.0GHZ

Overclocking Raspberry Pi 5 is something you do at your own risk, and Raspberry Pi doesn't necessarily recommend it. But, that said, we have a tutorial on Overclocking this issue for folks that want to live dangerously (see page 40).

So if you want to take Raspberry Pi 5 up in speed, here is what kind of performance increase you can look forward to.

Overclocking the CPU from 2.4GHz to 3.0GHz,

and the GPU from 800MHz to 1GHz, we see a corresponding increase in performance with a $\times 1.2$ increase in score on the single-core tests. Interestingly, we don't see a similar increase in performance in the multi-core tests. This discrepancy is probably down to constraints on memory bandwidth, but it's hard to tell for sure.



	2.4GHZ	3.0GHZ
Single-core score	774	906
Multi-core score	1604	1634

OTHER PEOPLE, OTHER BENCHMARKS

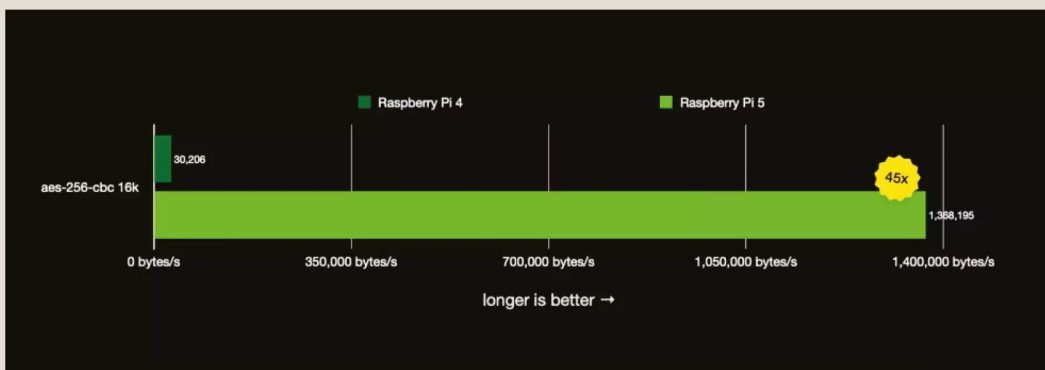
While that was my benchmark, you shouldn't just take my word for it. You should definitely go look at what other people have done around benchmarking and performance testing on our new hardware.

Because, as you'd expect, after perhaps looking at thermals (see page 30), the first thing most people did when they got their hands on a preview Raspberry Pi 5 was to run some benchmarks, which gives us lots of numbers and graphs.

As always, Jeff Geerling has performance

numbers (magpi.cc/jeff5benchmarks) for everything from Ethernet and wireless throughput, to memory throughput, to cryptographic performance – where Raspberry Pi 5 is 45 times faster than Raspberry Pi 4 thanks to the BCM2712 processor bringing Arm's Cryptographic Extension to Raspberry Pi (magpi.cc/a76cryptographic).

The benchmarks I found the most interesting are the ones from Seed Studio, who have gone out and benchmarked Raspberry Pi 5 (magpi.cc/

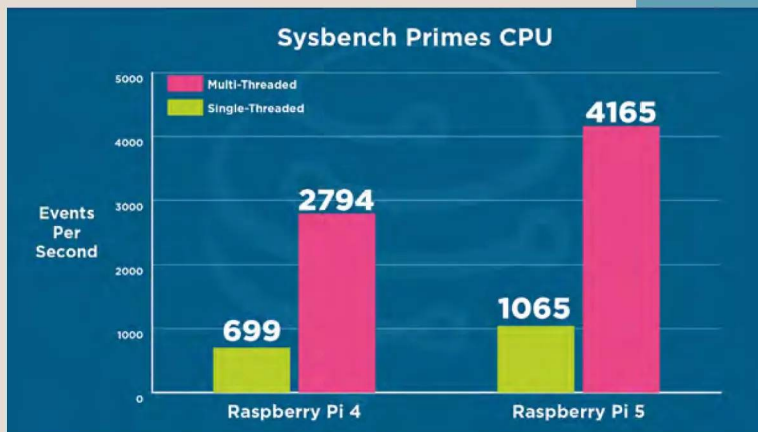


► Cryptographic benchmarks for Raspberry Pi 4 and 5 by Jeff Geerling

seed5benchmarks) using the ncnn framework (magpi.cc/ncnn).

The ncnn benchmark (magpi.cc/ncnnframework) is a deep-learning inference framework that supports various neural network models – such as PyTorch and TensorFlow – and a range of hardware. Designed with mobile deployment in mind, it offers GPU acceleration via the Vulkan API, and it's been really interesting to see the sort of performance increases we're seeing between Raspberry Pi 4 and 5.

Remember though, unlike my own Geekbench benchmarks, here smaller numbers are better, and they were getting some really nice numbers from Raspberry Pi 5. Testing the YOLOv8n (yolov8.com) model on Raspberry Pi 5 with a 640x640 pixel video stream, gave an inference rate of around 12fps.



WRAPPING UP

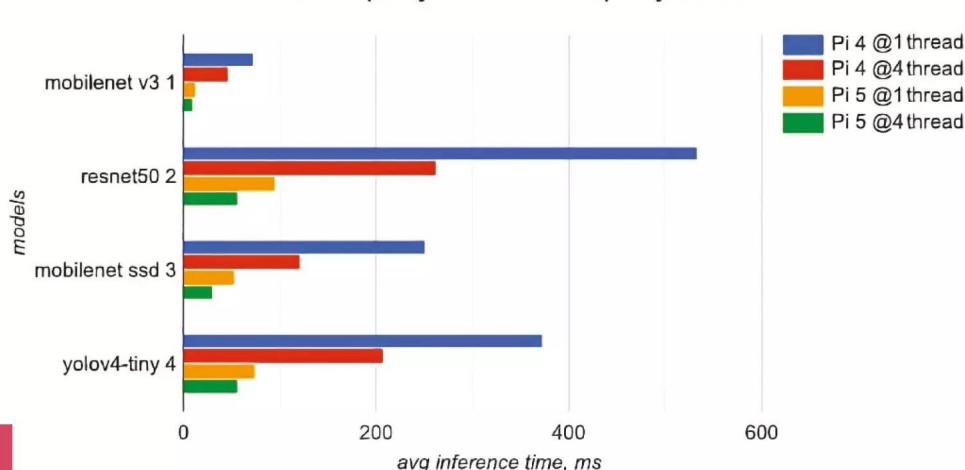
When we launched Raspberry Pi 4 back in 2019, with its quad-core Arm Cortex-A72 processor clocked at 1.5GHz, it was around 40 times faster than the original Raspberry Pi model from 2012. But Raspberry Pi 5, with its quad-core Arm Cortex-A76 processor clocked at 2.4GHz, means we now have between two and three times the CPU and GPU performance again; roughly twice the memory and I/O bandwidth; and for the first time we have Raspberry Pi silicon on a flagship Raspberry Pi device.

WE'RE PRETTY PROUD OF THE PERFORMANCE IMPROVEMENTS WE'VE BEEN ABLE TO MAKE

We're pretty proud of the performance improvements we've been able to make. But best of all, is seeing folks out in the wild finding out exactly how far we've come in the last few years.

▲ Benchmarks results showing the performance improvements from Core Electronics from (magpi.cc/core5benchmarks)

Benchmark on Raspberry Pi 4 8GB and Raspberry Pi 5 8GB



◀ Some ncnn framework benchmarks from Seed Studio

PicoVision (Pico W Aboard)

► Pimoroni ► magpi.cc/picovision ► £35 / \$37

SPECS

FEATURES:

Pico W, separate RP2040 (GPU), 2 × 8MB PSRAM, PCM5100A DAC for audio, 3 × user buttons, status LED, reset button, microSD card slot

CONNECTIONS:

HDMI video out, 3.5mm audio line out, breakout header, Qwiic/ST port, micro USB (on Pico W)

DIMENSIONS:

87 × 38 × 9mm

This pocket-sized digital video stick brings Pico games and more to the big screen. By **Phil King**

Aimed mainly at fans of retro gaming and old-school programming, PicoVision is a surprisingly capable digital video stick with a pre-soldered Raspberry Pi Pico W on board.

Using Pico W's RP2040 microcontroller chip as the CPU, it adds a second RP2040 to act as a GPU to help handle the big-screen visuals. Two extra 8MB RAM chips are used to swap frame buffers between the two RP2040s, enabling PicoVision to output reasonably high-res graphics (for a microcontroller) to a monitor via a standard HDMI output.

The only slight drawback is that it's not able to output audio via HDMI. For this, there's a 3.5mm stereo line out jack to connect a pair of PC speakers or headphones.

Power is connected using Pico W's micro USB port – as usual, you can hook this up to a computer

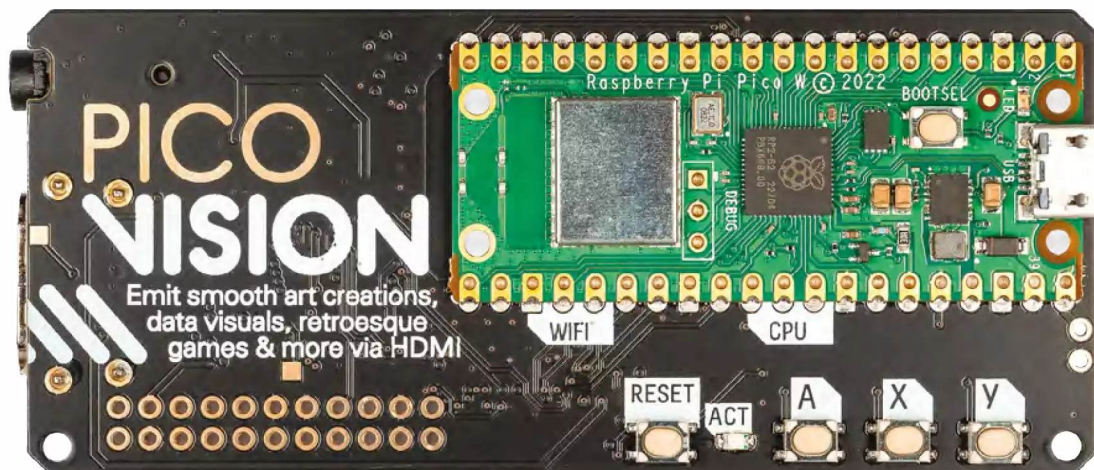
for programming in MicroPython or C/C++, or flashing Pico W with new firmware.

Alternatively, PicoVision can be powered via two unpopulated pins. There's also a breakout header for CPU and GPU debugging, plus a few other GPIO pins. A single Qwiic/STEMMA QT port can be used to connect I2C breakouts or sensors.

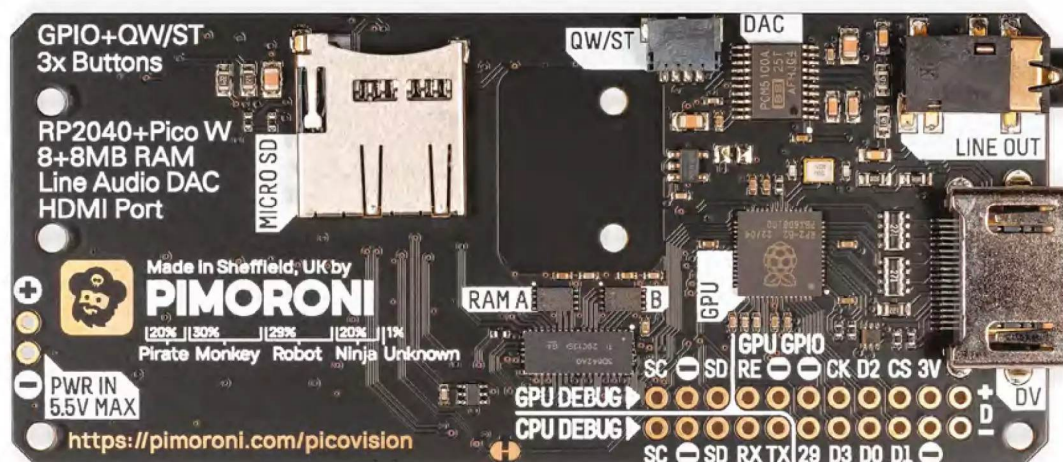
On the front of the board are three user buttons, an activity LED, and a handy reset button. At the rear is the bonus of a microSD card slot to provide extra storage for assets such as graphics.

Instant menu

Getting started is as easy as connecting PicoVision to an HDMI monitor or TV and powering it up. It comes pre-flashed with MicroPython firmware that auto-runs a menu of example programs. The



► The front of the PicoVision board features a pre-soldered Pico W and user buttons. The holes are an unpopulated breakout header



◀ On the rear are a microSD card slot, Qw/ST connector, audio line out and HDMI, plus the extra RP2040 'GPU' chip

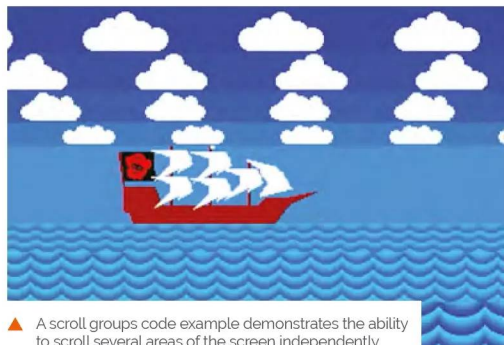
A and X user buttons are used to move up and down through the menu options; and a press of the Y button will select one.

Among the example programs are a fun *Flappy Bird*-style game, magic mirror dashboard, and Teletext-style headlines – the last two require connection to a Wi-Fi network.

“ Among the example programs are a fun Flappy Bird-style game, magic mirror dashboard, and Teletext-style headlines ”

Several graphical demos showcase PicoVision's capabilities. These include a classic 3D starfield effect (remember Windows 3.1's screensaver?), a screen modes menu with numerous resolutions (some require GPU overclocking), and a parallax scrolling animation that makes use of PicoVision's scroll modes – up to seven groups of scanlines can be scrolled separately.

Up to 80 sprites can be shown on screen simultaneously (or 32 with the special widescreen



▲ A scroll groups code example demonstrates the ability to scroll several areas of the screen independently

version of the firmware), with up to ten per any single scanline.

Start coding

When it comes to programming PicoVision yourself, a comprehensive GitHub repo (magpi.cc/picovisiongh) provides lots of useful info, along with MicroPython and C/C++ libraries and code examples from which to learn. The most basic are Hello World programs to print text to the screen. Graphics are handled using Pimoroni's standard PicoGraphics module, while a PicoSynth sound module is used to play simple tunes and noises using I2S audio from the board's DAC chip.

You'll also want to try playing classic *Doom* on PicoVision – which works really well. Just flash the *Doom* firmware UF2 to PicoVision and add the game's WHX file to a microSD card. For control, you'll need to connect a USB keyboard via a splitter cable (as supplied in the optional accessory kit). [M](#)

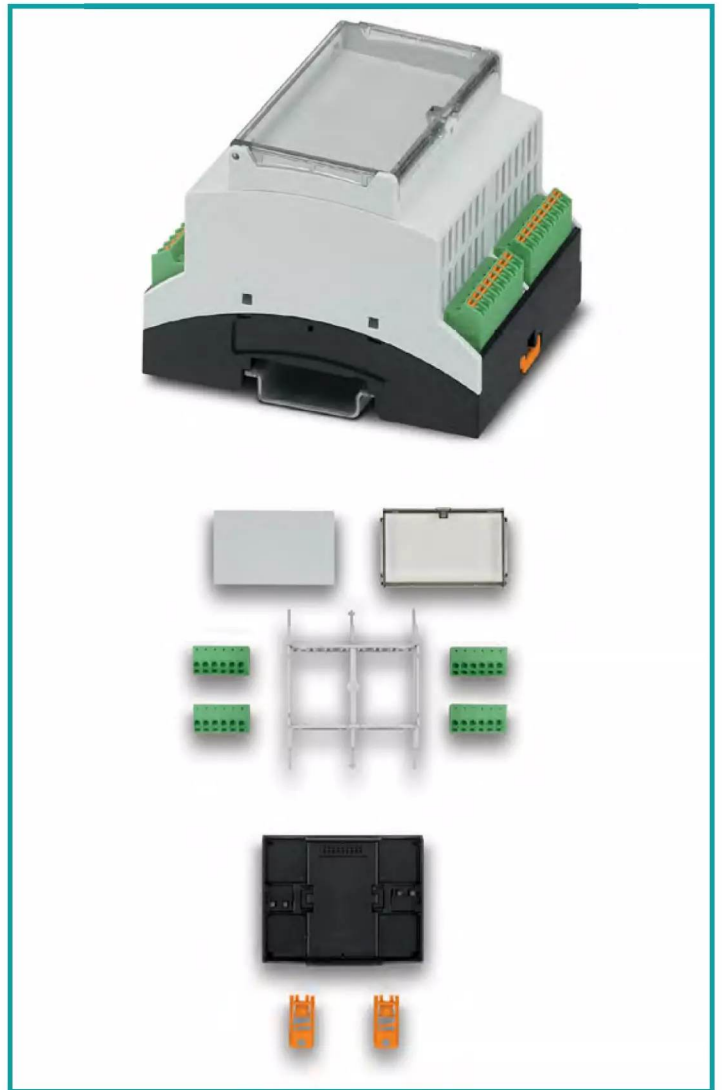


▲ Yes, you can play *Doom* on PicoVision! Use a splitter cable to plug a USB keyboard into Pico W

Verdict

With a second RP2040 and extra RAM to boost Pico W's capabilities, PicoVision offers plenty of possibilities for retro gaming and coding projects.

9/10



Electronics housings... as a complete kit.


Available exclusively from our global distribution and wholesale partners you will find the right housing for every application.

With wall, desk & DIN rail mounting options our kits make it easier for you to select the next housing solution for your electronics design.

For additional information call 01952 681700 or visit <https://phoe.co/housings-kits-uk>

10 Amazing: 3D prints

Print your own custom project pieces for Raspberry Pi

3 D printing and making go hand-in-hand, and these amazing prints and projects involving Raspberry Pi are fantastic ways to upgrade, or even realise your dream project. After many years, there's a huge range to choose from – here's just ten we like. 



▲ RaspBMO

Friendly console

While this BMO will be able to emulate games, it's also just a nice thing to have sitting on your desk.

magpi.cc/raspbmo



▲ simplyRetro Z5

Comfy and large

Some 3D printed handheld consoles are very small and portable, but sometimes a bigger screen is more comfy. The Z5 offers this.

magpi.cc/retroz5



◀ Rapiro

Bipedal robot friend

Rapiro was a very cool kit from several years ago which always had open source STL files so you could print your own.

magpi.cc/rapirostl



▲ ZeroBot

Practical and small

This cool looking robot for Raspberry Pi Zero is great for remote controlling and looking where you're going with computer vision.

magpi.cc/zerobot

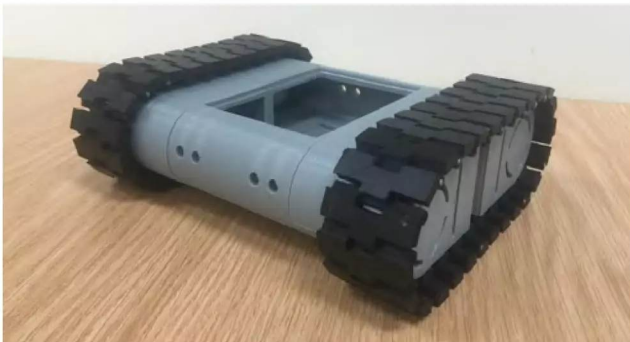


▲ Pipboy 3000 MKIV

Perfect cosplay prop

We've seen many iterations of a *Fallout*-inspired Raspberry Pi-powered Pipboy over the years, and this is one of the best.

magpi.cc/piboy3000



▲ Tank Bot

Treads for grip

There are many robots on Thingiverse – we quite like this one as it can be used for rough terrain better than normal wheels

magpi.cc/tankbot



◀ Retro desktop tower

Nostalgic protection

More powerful than a '90s PC, and a neat way to present Raspberry Pi.

magpi.cc/retrotower



▲ NinTastic

Retro gaming case

If you want to give your retro gaming Raspberry Pi a more suitable home, check out this great case.

magpi.cc/nintastic

► Gameboy NANO

Small and portable

This very tiny handheld console fits Raspberry Pi Zero – upgrade to Zero 2 and watch it really fly.

magpi.cc/gameboynano



◀ Case with camera mount

Goose neck mount

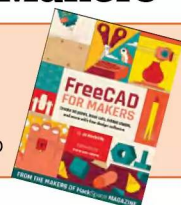
Watching your 3D printer with a 3D-printed Raspberry Pi stand may be a little meta, but it's fun and useful.

magpi.cc/casewithcam



FreeCAD for Makers

Fancy making your own 3D prints? Our colleagues at *HackSpace* magazine put together a great resource for designing 3D models in FreeCAD

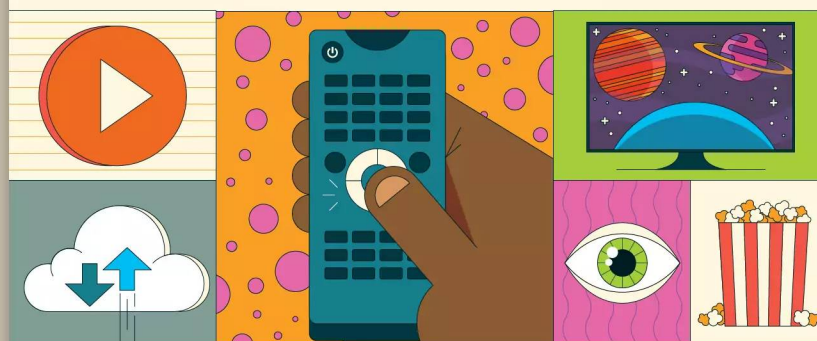


Your FREE guide to making a smart TV



BUILD A RASPBERRY PI MEDIA PLAYER

Power up your TV and music system



FROM THE MAKERS OF *The MagPi* THE OFFICIAL RASPBERRY PI MAGAZINE

magpi.cc/mediaplayer

Learn Scratch with Raspberry Pi

Resources to help you start coding with the block-based language. By **Phil King**

Code Club Scratch Modules

AUTHOR Raspberry Pi Foundation

Price: Free

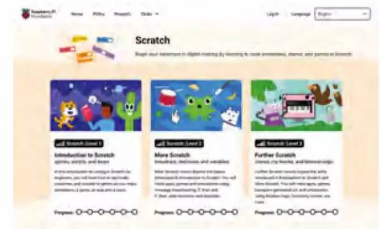
magpi.cc/codeclubscratch

Part of the Raspberry Pi Foundation, Code Club is all about getting young people to learn to program, whether by taking part in free coding clubs around the globe or from home using a host of online resources – including a set of modules based on Scratch.

Created by the MIT Media Lab, this block-based programming language enables users to drag and drop various blocks (arranged into colour-coded

categories) and stack them together to build programs to control interactive sprites on a ‘stage’. So it’s ideal for complete beginners to learn the key concepts of coding.

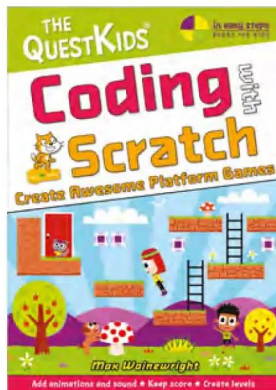
Code Club’s three Scratch modules progress in difficulty and take you through aspects such as sprites, loops, variables, conditionals, and Boolean logic. Each module comprises six projects to create, arranged



into sections: Explore (get new skills), Design (make design decisions), and Invent (create a new solution). It’s a fun and effective way to learn. **M**

Reading material

Books to help you learn Scratch coding



CODING WITH SCRATCH

Part of the QuestKids series aimed at youngsters, this colourful 80-page guidebook sees the cartoon gang show you how to create platform games. There’s also a second Scratch book that covers driving games. [▶ magpi.cc/qkscratch](http://magpi.cc/qkscratch)

OFFICIAL SCRATCH CODING CARDS

This set of 76 flash cards breaks simple projects down into easy-to-follow steps. There are ten sets of

sturdy cards covering various projects – such as Pong Game and Virtual Pet – with instructions to make them.

[▶ nostarch.com/scratchcards3](http://nostarch.com/scratchcards3)

CODING FOR BEGINNERS USING SCRATCH

This appealing spiral-bound Usborne book helps you to get started with Scratch and create a range of projects including animations and simple games. There are also handy tips and links to working projects.

[▶ magpi.cc/usbornescratch](http://magpi.cc/usbornescratch)

Scratch Ideas


AUTHOR

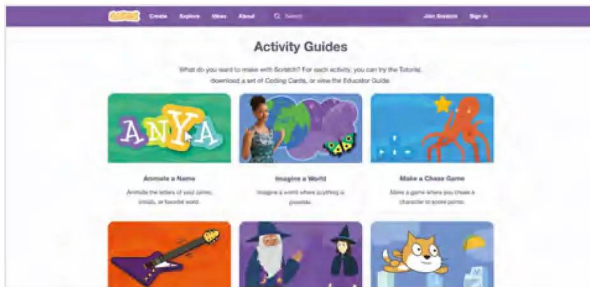
MIT Media LabPrice:
Freescratch.mit.edu/ideas

A popular section of the official Scratch site, Scratch Ideas is where you'll find a bunch of helpful activity guides, including a Getting Started one for newbies.

Each guide offers a set of coding cards to download, along

with an educator guide if you're helping someone to learn. Upon clicking 'Try the tutorial', you're presented with the usual online Scratch interface with a floating window. The latter plays a short introductory video before proceeding to a multi-step guide that shows you everything you need to complete the project. It's an effective way to learn how to use the numerous blocks and features in Scratch.

There's also a set of starter projects to remix, in five categories – animation, games, interactive art, music and dance, stories – that showcase the versatility of Scratch. 



Scratch Team YouTube Channel


AUTHOR

Scratch TeamPrice:
Free[youtube.com/
@ScratchTeam](https://youtube.com/@ScratchTeam)

If you like to learn by watching videos, the official Scratch Team channel on YouTube is an ideal destination. Tutorials include how to remix the built-in Scratch sprites, create new ones, import your own drawings, plus add and record sounds.

There are also guides to making different types of projects, including a *Pong* game and virtual town.

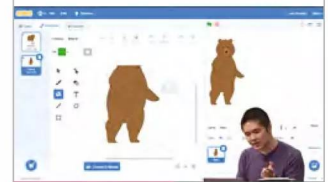
The Tips & Tricks selection covers useful pointers in two categories. 'Unlock the Block' videos help you to 'unlock the magic' behind some of the key Scratch blocks, while 'Scratching the Surface' is a series of one-minute videos that explain intermediate-level coding concepts.

You'll also find some Create-Along videos of livestreams in which the Scratch Team create a project in an hour, along with Scratcher Stories and video updates from the team. 



Online courses

Free Scratch courses to join



EDX CS50 SCRATCH

Created by Harvard, this self-paced three-week introduction to Scratch forms part of the university's renowned CS50 computer science course. Concepts covered include functions, events, conditions loops, and variables.

► magpi.cc/cs50scratch

TOWER DEFENCE GAME

Aimed at beginners and intermediate coders, this video series shows how to set up a basic tower defence game, add game logic and rules, design UI elements, incorporate sound and music, then test and debug it.

► magpi.cc/scratchtdgame

5 PROJECTS IN 5 DAYS

Subtitled 'Scratch Game Programming for Kids', this beginner-level course by the Little Apple Academy shows how to create five simple interactive animations and games. Video-based, it's fun to follow.

► magpi.cc/5projects



Jorvon Moss

A maker and roboticist known commonly as Odd Jayy, Jorvon creates cool stuff for Digi-Key

- Name **Jorvon Moss** | ➤ Occupation **Technical Content Creator**
- Community role **Maker** | ➤ URL magpi.cc/oddjay

If you've been to major Maker Faires in the US recently you've likely see Jorvon, aka Odd Jayy, and his amazing steampunk creations. Sometimes robots, sometimes wearables, and always very cool. He describes himself as an oddity, hence the name Odd Jayy.

"I have an art background, BFA in Illustration, and self-

“ He describes himself as an oddity, hence the name Odd Jayy ”

taught electronics, but mainly focus on character based robotics,” Jorvon tells us. “I just decided to start [making] one day in college as a hobby but then realized it was a passion of mine. I started learning back in 2014, but it has only been my full time job for the last two years.”

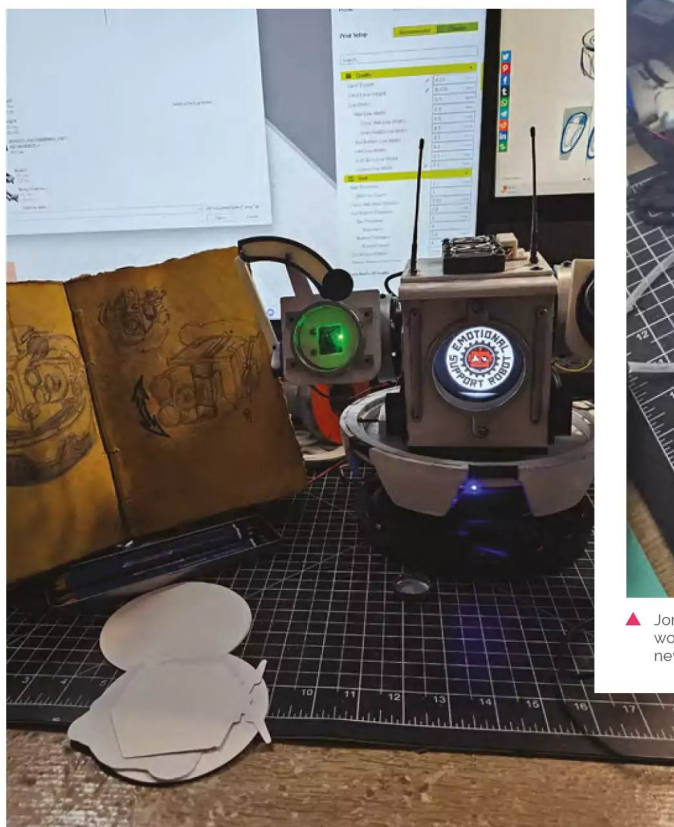
When did you learn about Raspberry Pi?

I learned about Raspberry Pi back in 2014, and to be honest



▲ Binary 2.0 is a cute robot that Jorvon hopes to bring to events

▼ Meet M3RVN, the AI-powered, shoulder-perched robot



▲ Jorvon is always working on new robots

I broke my first Raspberry Pi by trying to solder directly onto the board, but I have learned since then and Raspberry Pi is now one of my personal favourite boards.

How do you like to use Raspberry Pi stuff (including Pico)?

I like to use Raspberry Pi in a lot of my projects and I love Pico. I use Pico a lot now, being a very strong and easily accessible microcontroller. But when I want a robot with a lot of computing power I grab a Raspberry Pi 4 and now 5.

What's your favourite creation you've made with Raspberry Pi?

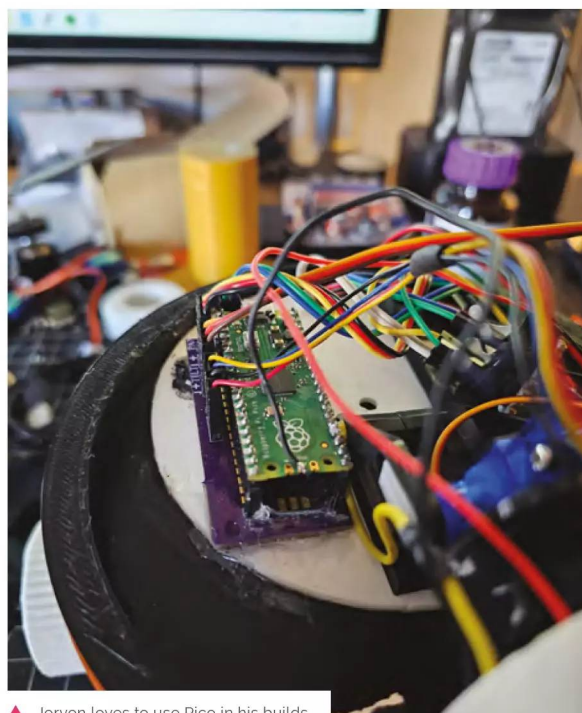
My favourite thing made with Raspberry Pi so far has been my AI robot M3RVN and my bubble-blowing robot Binary.

What inspires your creations?

Sci-fi inspires my work, movies, and books. I love making fun robotic designs and imagining a world they live in.

Where can we see your stuff?

You can find most if not all my work on [instagram.com/odd_jay](https://www.instagram.com/odd_jay) and my documentation on magpi.cc/oddjay



▲ Jorvon loves to use Pico in his builds

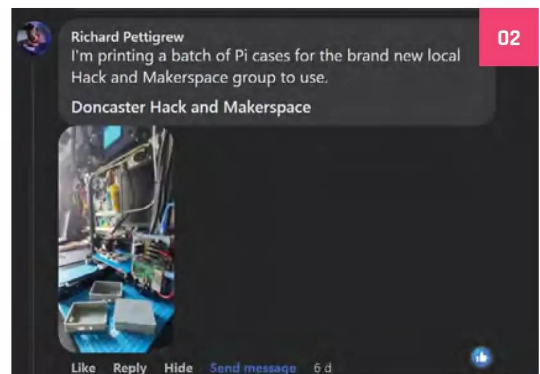
MagPi Monday

Amazing projects direct from social media!

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's a selection of some of the awesome things we got sent this month – and remember to follow along at the hashtag #MagPiMonday! 📷

01. We like a nice custom home sensor suite
02. 3D printing cases is a great way to keep your Raspberry Pi safe
03. A nice live light is useful whether you're working from home or streaming on Twitch
04. This is cool and we love how low-tech and cheap this solution is
05. The Digital Maker badge is something we wish we had when we were kids
06. Wonder what the scale of this is compared to the original
07. This is a cool little project – check out *HackSpace 55* and Andy's Hackaday for more info
08. Kev's robots invaded Italy recently, with some Raspberry Pi 5 power ups
09. Sound catcher or sound shooter? Do you need a sight for sounds? We have many questions...

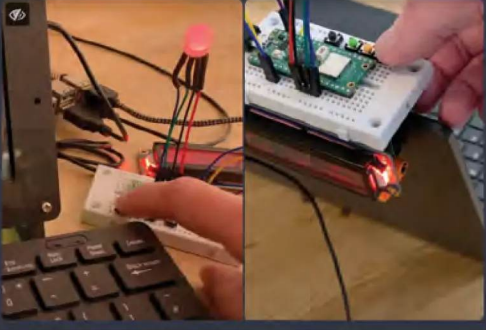


Pater Practicus
@paterpracticus@mastodon.world

03

@themagpi

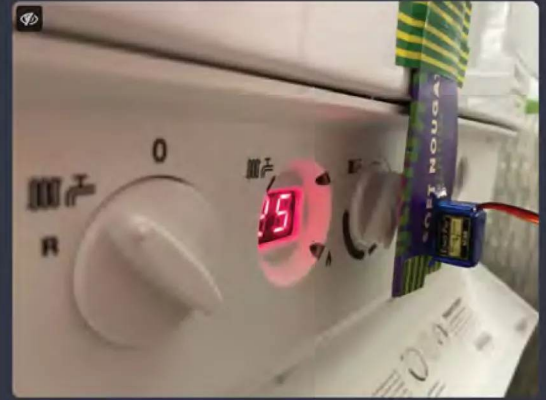
The Zoom call status indicator prototype is finally finished - along with my 'how-to' video playlist - taking the project all the way from learning about PWM for the LED colours and programming the buttons, to adding the text display and building a laptop screen mount: youtube.com/playlist?list=PLIN...



Simon
@staustellsimon@infosec.exchange

04

@themagpi it's only basic and in PoC stage, but I have a servo motor altering the boiler flow temperature to automate weather compensation for radiator temperatures and always have a higher temperature for DHW runs. Next is to make the script accessible via URL so I can integrate to #HomeAssistant once happy it works I'll need a more secure fixing 😊




PenguinTutor (Stewart Watkiss)
@penguintutor@fosstodon.org

05

@themagpi I ran a Raspberry Pi workshop for the Cub Scouts Digital Maker badge.

Scratch and physical computing

youtu.be/AbT-5UjpmVU?si=Y1BPqS...



YouTube

Scratch 3 on a Raspberry Pi - includes Raspberry Pi 5 and Bookworm OS version

By Penguin Tutor

Extreme Electronics
@Extelc@mastodon.social

06

@themagpi

Pi Pico Z80 emulation of a 96k Computers Lynx.

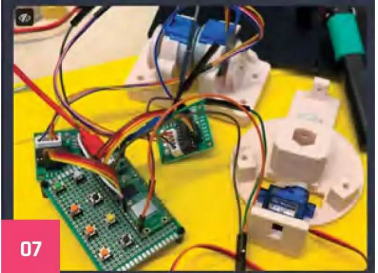
With a working keyboard and monitor. Also battery powered.



Andy Piper
@andypiper@imacaw.social

07

@themagpi today is the day (I think) that I should get a servo extender wire so that I can try out the wall-hanging plotter powered by Pico (from HackSpace #55 but with modifications and updates) hackaday.io/project/193338-mn



kevinmcaleer

2 08


Happy #MagPiMonday - this weekend I was at Maker Faire Rome showcasing Bubo which contains the new Raspberry Pi 5 for faster Python and AI image recognition



davidwilsonclarke

09

Well, there is this sound catcher I'm working on.

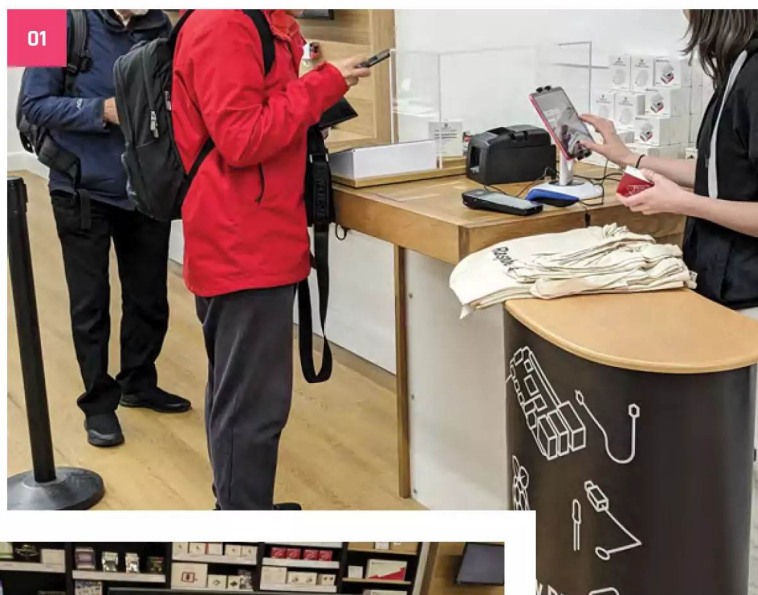


Events in pictures: Raspberry Pi 5 launch at the Raspberry Pi Store

Raspberry Pi 5 is here! We caught up with folks buying their boards on day one

In the Grand Arcade at Cambridge, the Raspberry Pi Store usually opens at a reasonable 10am but on Monday 23 October we were there at a pre-dawn 7:30 am to witness the release of Raspberry Pi 5 at 8am. While there weren't the huge lines you'd see at a pop up, a handful of die-hard Raspberry Pi fans were ready to get their new everything computer.

01. People were able to reserve their Raspberry Pi before hand or just buy one if they fancied on the day
02. Boards, PSUs, cases, and coolers were available to grab
03. Experience Raspberry Pi 5 at the store before buying one
04. The calm before the storm with Features Editor Rob



Crowfund this!

Crowdfunding projects for Raspberry Pi accessories

MULTI-IO HAT

"The MULTI-IO HAT combines two of each of the most needed interfaces used in industrial automation: digital and analogue inputs and outputs, RTD measurement and relays. Expandable to eight layers, it can perform from the simplest to the most complex industrial and home automation tasks."



► kck.st/3QcioER

CrowVision

We reviewed CrowVision a few issues ago and thought it was a great display for many kinds of Raspberry Pi projects. It's now up for crowdfunding and already passed its goal. The Crow line has always been good, so we're looking forward to seeing more displays in this vein too.



► magpi.cc/crowfund



A wireless hydrometer and thermometer that connects to your Raspberry Pi

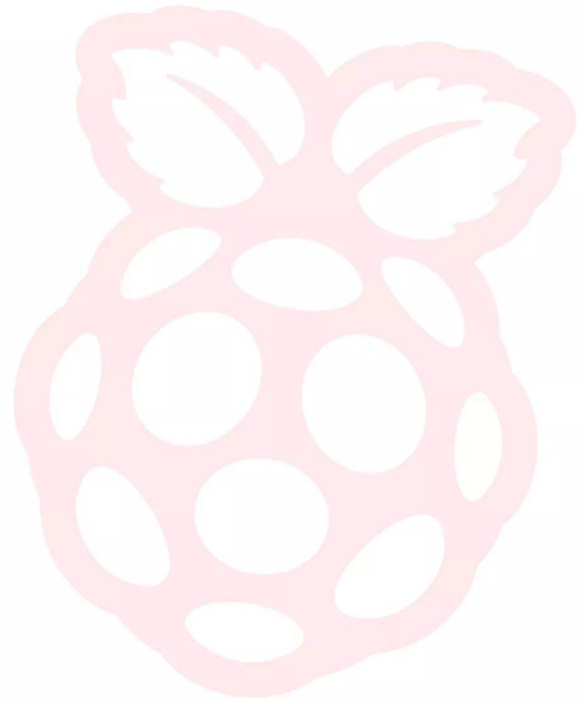
Free international shipping with checkout code:

SHIP-FREE-INTL

shop now at tilthydrometer.com



Your Letters



Priority boarding period

I have been on holiday and found out about the priority boarding codes long after the magazine came out – is it too late for me to get one of these codes?

Susan via email

Priority Boarding is something we'll be honouring for new print subscribers for the near future – go and check magpi.cc/priorityboarding for info on this. Codes can take up to two working days to be sent out.

If you're an existing subscriber and still don't have your code, please make sure to email us at magpi@raspberrypi.com and we can get it to you.

Raspberry Pi 5

PRIORITY BOARDING

We've reserved Raspberry Pi 5 boards
for *The MagPi* subscribers

GET YOUR RASPBERRY PI 5 **FIRST!**

magpi.cc/priorityboarding

▲ You still have a while yet to get your own Priority Boarding code to obtain Raspberry Pi 5 faster



▲ Head to Victoria Gate in Leeds to see the Raspberry Pi seasonal pop-up

Leeds pop-up store

I was surprised to see that there was a Raspberry Pi Pi pop-up store in Leeds the other day – I don't remember this being announced? Did I miss a blog or email about it?

Harry via email

The Leeds Raspberry Pi Store was supposed to be a big surprise and it's not a weekend pop-up like the rest of the pop-up shops this year – this one will be open until the end of December. It's a nice big store as well, with many of the features of the one in Cambridge!

Check out raspberrypi.com for more info on the shop, and make sure to drop by and buy several copies of *The MagPi* and our books while you're there.

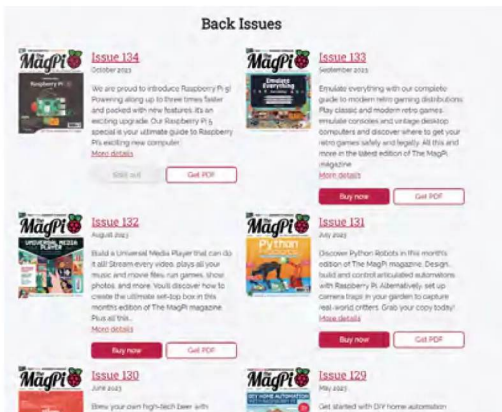
USA SPECIAL! 6 ISSUES FOR \$43

Back issues

Do you have any copies of issue 123 that I can order, or are you planning to print any more? I regrettably missed it and am now looking to complete my collection.

Az via Facebook

Unfortunately we don't reprint magazine issues so when they're gone, they're gone. However, after three weeks each issue becomes freely available as a PDF on our website (magpi.cc/issues) and you're welcome to print it for yourself if the PDF is not enough for you.

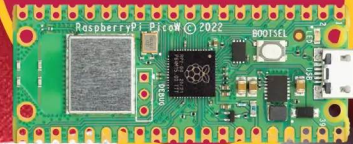


▲ All our issues are available as PDFs from magpi.cc/issues

Contact us!

- ▶ Mastodon magpi.cc/mastodon
- ▶ Threads @themagpimag
- ▶ Facebook magpi.cc/facebook
- ▶ Email magpi@raspberrypi.com
- ▶ Online forums.raspberrypi.com

FREE RASPBERRY PI PICO W



Subscribe online:
magpi.cc/subscribe

Continuous credit card orders will auto-renew at the same price unless cancelled. A free Pico W is included with all subscriptions. This is a limited offer. Not included with renewals. Offer subject to change or withdrawal at any time.



Community Events Calendar

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

01. Melbourne Raspberry Pi Meetup

- 📅 Sunday 3 December
- 📍 Docklands Makerspace and Library, Melbourne, Australia
- ▶ magpi.cc/mrpm136

This meet-up is open to everyone with an interest in electronics, robotics, home automation, 3D printing, laser cutting, amateur radio, high-altitude balloons, space tech, etc. Makers are invited to bring along their projects and project ideas, and come connect with other makers. Get your questions answered, show off the work you are doing, and get support to resolve nagging issues.



02. Coventry Raspberry Jam: Show and tell

- 📅 Thursday 7 December
- 📍 Blue Coat C of E School, Coventry, UK
- ▶ magpi.cc/crj136

Blue Coat C of E School is hosting a Raspberry Jam show-and-tell event in the school theatre. The aim for this event is for people to bring along the Raspberry Pi-based projects that they have created and share them with others. There is plenty of seating available, and also a projector connected to a desktop computer.

03. Cornwall Tech Jam

- 📅 Saturday 9 December
- 📍 Fraddon Village Hall, Saint Columb, UK
- ▶ magpi.cc/ctj136

Cornwall Tech Jams are run by volunteers working in IT and education throughout Cornwall. They are supported by Software Cornwall, its members and other local businesses. Our volunteers give their own time and expertise to plan each Cornwall Tech Jam and to the regular maintenance of all our equipment.

04. Mombasa Raspberry Pi Jam Experience Raspberry Pi 5!

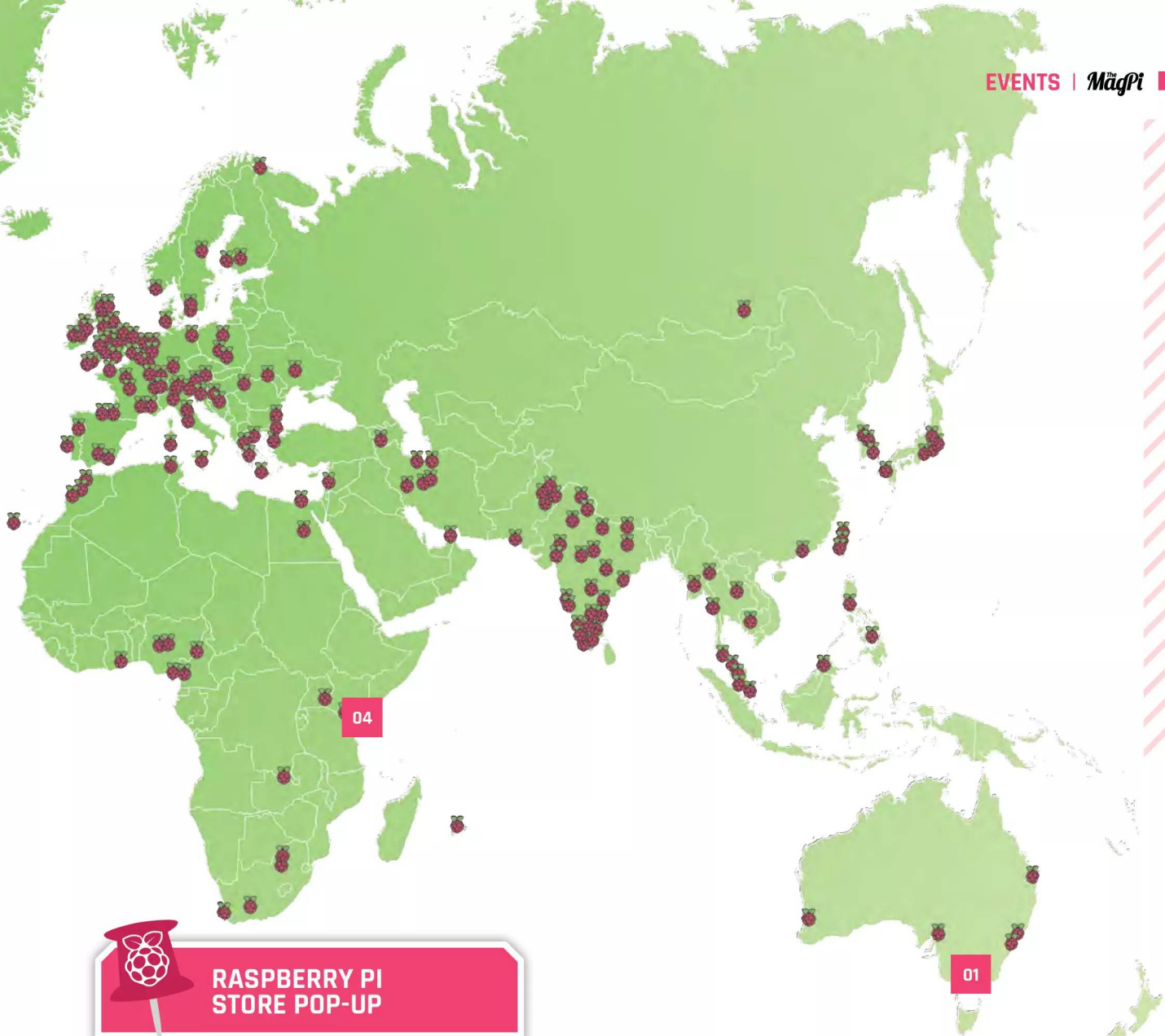
- 📅 Monday 11 December
- 📍 Tech Kidz Africa, Mombasa, Kenya
- ▶ magpi.cc/rpjm136

Raspberry Pi Jam Mombasa is a one-day event for students, teachers, and enthusiasts of robotics and AI. The Jam will feature workshops, demonstrations, and hands-on activities.



FULL CALENDAR

Get a full list of upcoming community events here:
magpi.cc/events



**RASPBERRY PI
STORE POP-UP**



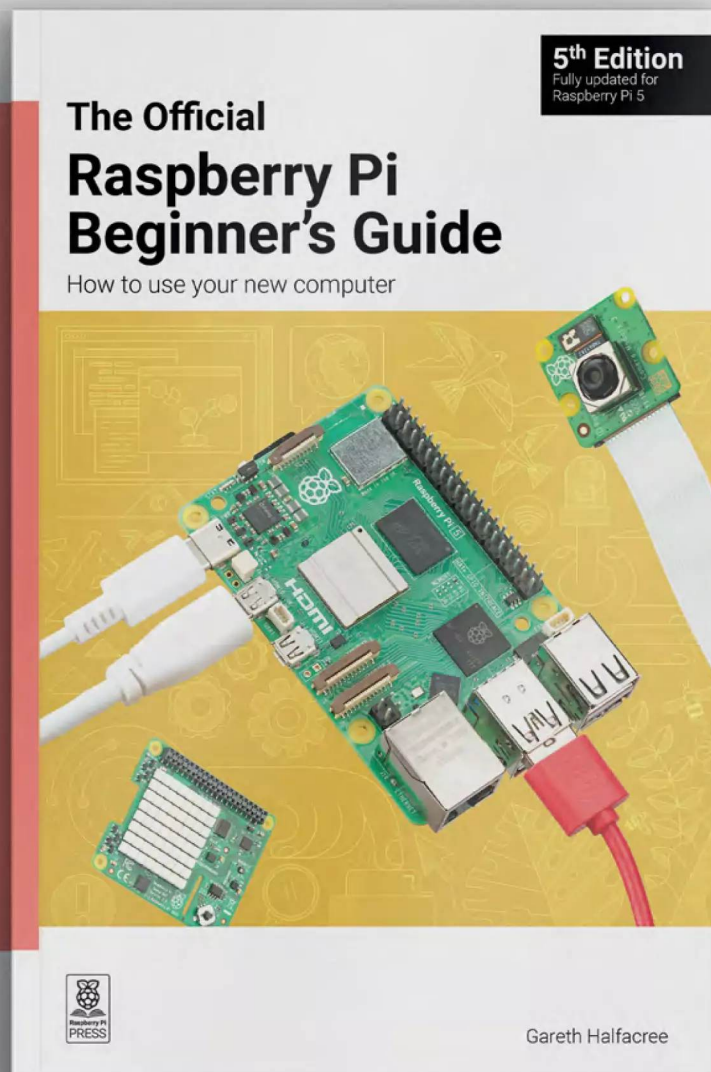
Raspberry Pi pop-up Leeds – Experience Raspberry Pi 5

- ▶ Where **Victoria Gate, Leeds, UK**
- ▶ When **Saturday 4 November to Saturday 30 December**

Get hands-on with the all-new **Raspberry Pi 5** in our holiday pop-up store on the ground floor of **Victoria Gate in Leeds, England**. Experience Raspberry Pi 5's blazing performance and check out the newest accessories. The store will have the full line of Raspberry Pi products for purchase.

magpi.cc/popupleeds

- Learn coding ▪
- Discover how computers work ▪
- Build amazing things! ▪

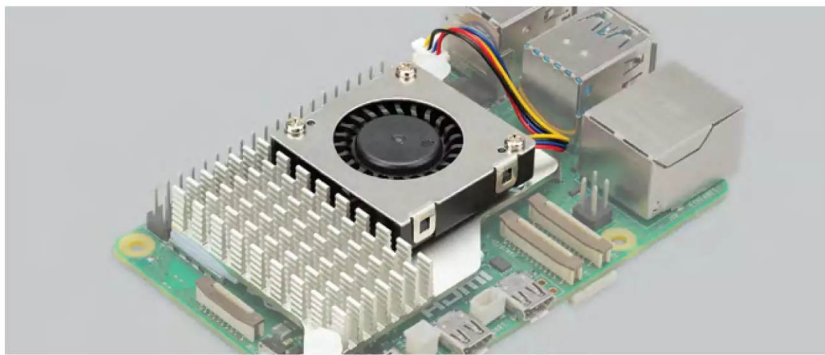


magpi.cc/beginnersguide

WIN ONE OF TEN

RASPBERRY PI 5 ACCESSORY PACKS!

Enjoying your new Raspberry Pi 5? Make it even better with this accessory pack, including the new 27W USB-C PSU, a display and camera module adapter cable, and an Active Cooler as well! We have ten packs to give away.



Head here to enter: magpi.cc/win | Learn more: raspberrypi.com/products

Terms & Conditions

Competition opens on **22 November 2023** and closes on **14 December 2023**. Prize is offered to participants worldwide aged 13 or over, except employees of Raspberry Pi Ltd, 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, Facebook, Twitter or any other companies used to promote the service.

HackSpace

TECHNOLOGY IN YOUR HANDS

THE **MAGAZINE**
FOR THE **MODERN MAKER**



SUBSCRIBE AND
SAVE UP TO
35%
on the cover price



ISSUE #73

OUT NOW

hsmag.cc



Start your Raspberry Pi

ADVENTURE

A beginner's guide
to mastering
single-board computing



THE MAGPI #137
ON SALE 14 DECEMBER

Plus!

Build a Beepy handheld PC

MyCobot 280 professional robotic arm

Recreate the Amiga with floppy drive

DON'T MISS OUT! magpi.cc/subscribe

MASTODON magpi.cc/mastodon

THREADS [@themagpimag](https://twitter.com/themagpimag)

FACEBOOK magpi.cc/facebook

EMAIL magpi@raspberrypi.com

ONLINE forums.raspberrypi.com

EDITORIAL

Editor

Lucy Hattersley
lucy@raspberrypi.com

Features Editor

Rob Zwetsloot
rob@raspberrypi.com

Sub Editor

Ian Evenden

ADVERTISING

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

DESIGN

criticalmedia.co.uk

Head of Design

Lee Allen

Designer

Sam Ribbits

Illustrator

Sam Alder

CONTRIBUTORS

Alasdair Allan, David Crookes, Pj Evans, Rosie Hattersley, Phil King, Stewart Watkiss, Ashley Whittaker

PUBLISHING

Publishing Director

Brian Jepson
brian.jepson@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.

The MagPi magazine is published by Raspberry Pi Ltd, Maurice Wilkes Building, St. John's Innovation Park, Cowley Road, Cambridge, CB4 0DS. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products, or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0). ISSN: 2051-9982.



Raspberry Pi
PRESS
3.0 Unported
(CC BY-NC-SA 3.0).
ISSN: 2051-9982.



Christmas with Raspberry Pi 5

Writing about Christmas in early November is weird, even for **Rob Zwetsloot**

These Final Word columns tend to be one of the last things we write for the magazine, meaning that even with print lead times the context of these columns is still fairly fresh. As I'm writing this, I realise we don't have one of my seasonal features in this issue for Christmas – my favourite winter holiday! I'm Mr Mince Pie according to my friends, after all.

(Don't tell anyone, but I might enjoy writing the annual Halloween feature a little more as I get to embrace my inner crypt keeper and alliterate more than is advisable.)

Lucy, *MagPi* Editor, firmly believes that Christmas can't start until after Bonfire Night [I'm right –Ed], and while I understand and respect that, I'm full on Christmas mode the moment the clocks turn midnight on Halloween. Which, if we were to write a Christmas projects feature, would have been roughly the deadline for it. That's weird, right? I know TV shows tend to film their Christmas specials in July and *It's a Wonderful Life* was shot during an unusually hot summer with everyone in big coats being pelted by newly-invented fake snow, but it's still a little odd to be writing

about turkey dinners and stockings on a rainy October evening.

Raspberry Pi-powered Christmas

Many of the Christmas tutorials that have appeared in the magazine over the years are based on projects I've

“ Many of the Christmas tutorials that have appeared in the magazine over the years are based on projects I've built myself ”

built myself as Christmas decorations, although I unfortunately broke the 3D-printed star I made many years ago. I'm trying to figure out what to do this year, and I think I'm long overdue a Pico overhaul of some of my lights around my home.

Maybe it's time to make a new star ornament? I do need to go back and make the star itself a lot lighter, as I

had trouble balancing it on my fake tree. I should also get them all automated with voice commands. Anyway, keep an eye on the *MagPi* socials for anything I decide is worth posting.

Under the tree

Hopefully you'll have made use of our Priority Boarding offer to get a Raspberry Pi 5 for yourself or someone else who will be opening it on Christmas morning. If you're looking for stocking stuffers, the *Official Handbook 2024* is a great gift, along with the newly updated *Beginners Guide*. We work hard on those to make them extra special, and they're a great way for new users to pick up Raspberry Pi.

Anyway, have a great holiday and a happy new year. And yes, we do have another issue out between now and then. The print industry is odd. 🍷

Rob Zwetsloot

AUTHOR

By the end of October, Rob has consumed more mince pies than most people eat in a year. He's currently considering making a Christmas cake.

magpi.cc

HiPi.io

HIGHPI PRO

————— The new case from the HiPi.io team —————



- Rapid tool-free assembly and disassembly
- Large internal volume for HATs
- Compatible with Pi 2/3/4
- Multiple lid options
- Passive & Active Cooling options
- Secure microSD card cover
- VESA mount support
- Molding-configurable output ports customizable for volume orders
- Printed logo for your branding

Available at these great Pi stores:

 PiShop.us  PiShop.ca  PiShop.mx  Pi Hut  Pi-Shop.ch  Welectron.

Contact your favorite Pi store if it's not listed here

PiKVM

Manage your servers or workstations remotely

A **cost-effective** solution for data-centers, IT departments or remote machines!



PiKVM HAT
for DIY and custom projects



Pre-Assembled version

- Real-time clock with rechargeable super capacitor
- OLED Display
- Bootable virtual CD-ROM & flash drive
- Serial console
- Open-source API & integration
- Open-source software

Available at the main Raspberry Pi resellers



Reseller suggestions and inquiries:
wholesale@hipi.io