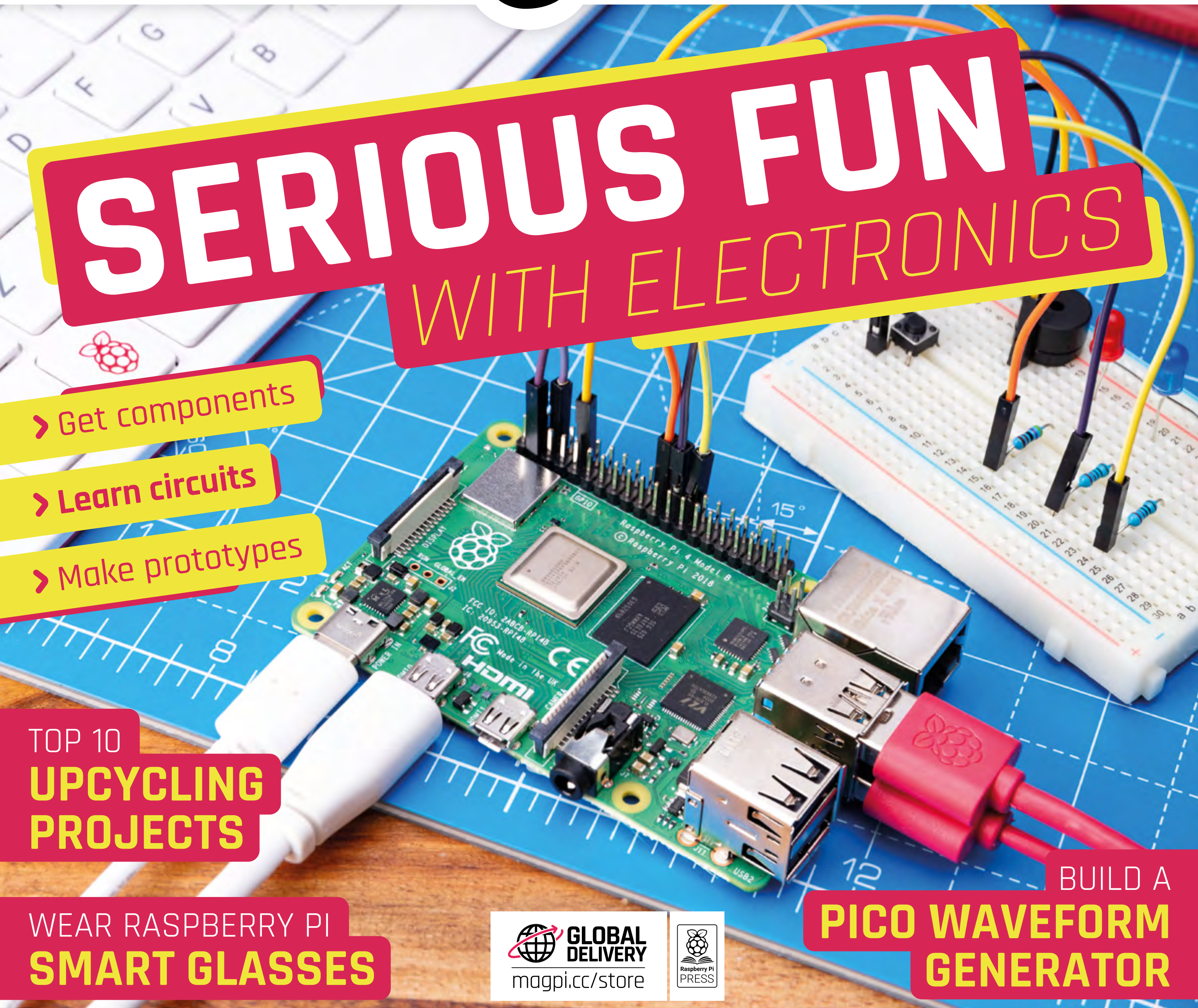
The MagPi

Issue 115 | March 2022 | magpi.cc

The official Raspberry Pi magazine



SERIOUS FUN WITH ELECTRONICS

- › Get components
- › **Learn circuits**
- › Make prototypes

TOP 10 **UPCYCLING PROJECTS**

WEAR RASPBERRY PI **SMART GLASSES**

BUILD A **PICO WAVEFORM GENERATOR**

WIN! MINI PUPPER ROBOT DOG KIT!

G-MASTER™
#MONITORS4GAMERS
by iiyama

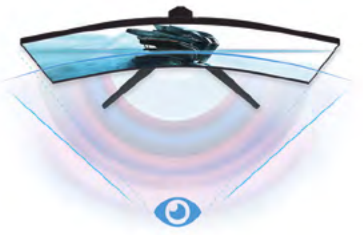
RED EAGLE™

QHD or Full HD?
Go CURVED!

G-Master GB3266QSU
Red Eagle

G-Master G2466HSU
Red Eagle

Immerse yourself in the game with the new Red Eagle #monitors4gamers.
Choose the height adjustable GB3266QSU with a 2560x1440 resolution or the G2466HSU
with 165Hz refresh rate and enjoy stutter-free gameplay on these 1500R curved VA panels.



1ms | HDMI | DISPLAY PORT | USB HUB | 🎧 | 🔊

Find your match at gmaster.iiyama.com

iiyama

WELCOME
to The MagPi 115

Every March we run our yearly #MonthOfMaking special (page 66). This is where we all get together and finally build that project we've been dreaming of. Get involved and you can join hundreds of other makers to share and inspire, and get a little advice along the way.

Raspberry Pi is where electronics and code smash together to produce (hopefully) something rather wonderful.

The metal pins on Raspberry Pi (known as 'GPIO') are used to wire up your little computer to lights, buttons, speakers, and even larger appliances around the home.

Code is the language you use to control a computer. Code is cool, but electronics is even cooler. It's one step beyond. You use code to control a computer to build and control real-world electronic appliances.

Getting started is a real challenge and that's why we've asked electronics expert Simon Monk to help. Simon's Raspberry Pi Electronics Special (page 34) takes you from knowing absolutely nothing up to attaching and controlling gadgets and gizmos with Raspberry Pi.

So what are you waiting for? Get learning about electronics and make something cool with Raspberry Pi. Then share it with us using #MonthOfMaking as the hashtag.

Lucy Hattersley Editor

EDITOR

Lucy Hattersley

Lucy is editor of *The MagPi* and her Dad's a spark, so she grew up holding fuses and prodding sockets. Still not dead.

@LucyHattersley

YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
The official Raspberry Pi magazine
Issue 115
March 2022
magpi.cc
SERIOUS FUN
WITH ELECTRONICS
Get components
Learn circuits
Make prototypes
TOP 10 UPCYCLING PROJECTS
WEAR RASPBERRY PI SMART GLASSES
BUILD A TIME MACHINE RADIO
WIN! MINI PUPPER ROBOT DOG KIT!

15.6"
2.8"
4"
4.3"
7"
5"
7"
13.3"
7.9"
5.5"

Contents

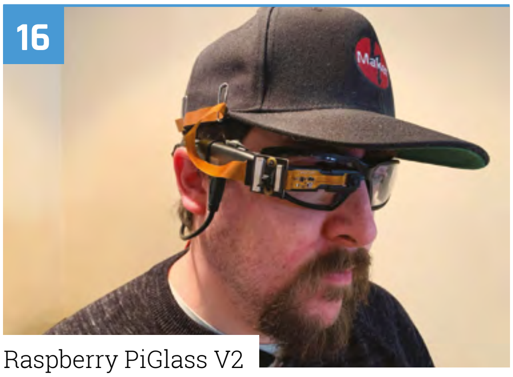
Issue 115 March 2022

Cover Feature

Regulars

Project Showcases

34

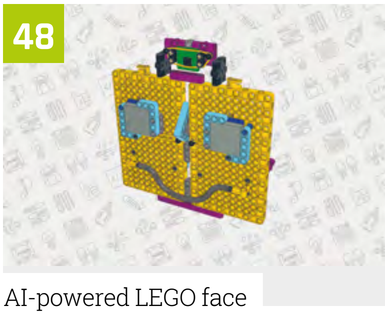

16

Raspberry PiGlass V2

20

Fancy Octopus Arcades

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.


48
AI-powered LEGO face

60
RetroPie in a joypad

The Big Feature

66
#MonthOfMaking 2022

76
Trilobot

86
Liz Upton interview

WIN A MINI PUPPER

95

Raspberry Pi OS 64-bit now widely available

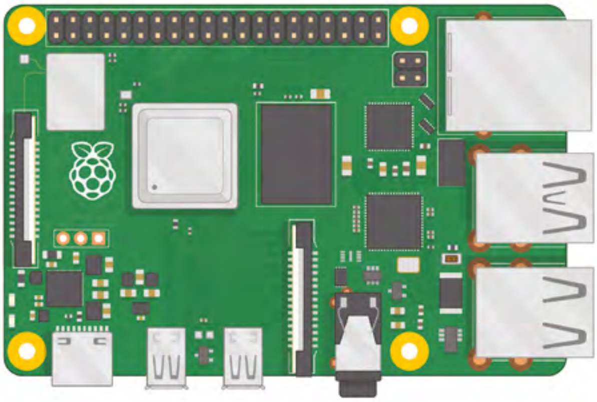
64-bit edition of operating system heads out of beta and into Raspberry Pi Imager

Raspberry Pi OS (64-bit) is now available in Raspberry Pi Imager, having been previously only available in a beta trial.

"Over the past year, we've been trialling a beta of Raspberry Pi OS in glorious 64-bit," writes Gordon Hollingworth, Raspberry Pi's Director of Software Engineering. "Now it's time to open it up to a wider audience."

Some Raspberry Pi computers have had the potential to run 64-bit operating systems for some time. Gordon explains on the Raspberry Pi blog (**magpi.cc/64bitblog**): "The ARMv8-A architecture, which encompasses the 64-bit AArch64 architecture and associated A64 instruction set, was first introduced into the Raspberry Pi line with Raspberry Pi 3 in 2016.

"From that point on, it has been possible to run a full 64-bit operating system on our flagship products, and many third-party operating systems are available."



Raspberry Pi 4 (8GB) is able to take full advantage of Raspberry Pi OS (64-bit)

Cores for horses

Not all Raspberry Pi computers can run Raspberry Pi OS (64-bit), as you can see from the 'Architecture width' column in **Table 1**. One advantage of Raspberry Pi OS (32-bit) is that it works on all Raspberry Pi computers.

"Using arm6hf (Raspbian's derivative of armhf with ARMv7-only instructions removed but floating-point instructions retained) provides us with an operating system which will run on every device we have ever manufactured, all the way back to 2011," explains Gordon.

However, the world moves on and its clear that many users are ready for a 64-bit switch-over.

"We've come to realise that there are reasons to choose a 64-bit operating system over a 32-bit one," says Gordon. "Compatibility is a key concern: many closed-source applications are only available for arm64, and open-source ones aren't fully optimised for the armhf port. Beyond that there are some performance benefits intrinsic to the A64 instruction set: today, these are most visible in benchmarks, but the assumption is that

> Over the past year, we've been trialling a beta of Raspberry Pi OS in glorious 64-bit

these will feed through into real-world application performance in the future.

Moving to the 64-bit version of the OS also enables owners of Raspberry Pi 4 (8GB) model some extra headroom. "32-bit pointers only allow you to address 4GB of memory," explains Gordon. "On Raspberry Pi 4, we use the ARM Large Physical Address Extension (LPAE) to access up to 8GB of memory, subject to the constraint that any process is limited to accessing 3GB (we reserve the top 1GB of the virtual address space for the kernel). Very few processes require more memory than this: happily, Chromium, which is probably the most memory-intensive application in Raspberry Pi OS, spawns a process per tab. But some use cases will benefit from being able to allocate the entire memory of an 8GB Raspberry Pi 4 from a single process."

So, head to Raspberry Pi's downloads page (**magpi.cc/downloads**) and grab your copy of 64-bit Raspberry Pi OS today. Do let us know how you get on and if your use case benefits (or suffers!) from the move to 64-bit.

Chromium media

The 64-bit version of Chromium that comes with Raspberry Pi OS (64-bit) has no version of the WidevineCDM library and therefore, it is not possible to play streaming media such as Netflix or Disney+. To instead choose the 32-bit version, just do the following within a Terminal window:

```
 sudo apt install chromium-browser:armhf
libwidevinecdm0
```

To return to the 64-bit version:

```
 sudo apt install chromium-browser:arm64
libwidevinecdm0-
```

Table 1: Raspberry Pi processor and architecture

Product	Processor	ARM core	Debian/Raspbian ARM port (maximum)	Architecture width
Raspberry Pi 1	BCM2835	ARM 1176	arm6hf	32-bit
Raspberry Pi 2	BCM2836	Cortex-A7	armhf	32-bit
Raspberry Pi Zero	BCM2835	ARM 1176	arm6hf	32-bit
Raspberry Pi Zero 2	BCM2710	Cortex-A53	arm64	64-bit
Raspberry Pi 3	BCM2710	Cortex-A53	arm64	64-bit
Raspberry Pi 4	BCM2711	Cortex-A72	arm64	64-bit

Sol-20 Terminal Computer

A fleeting glance at a gorgeous limited edition computer 45 years ago resulted in a marvellous Raspberry Pi 4 build, **Rosie Hattersley** learns

MAKER

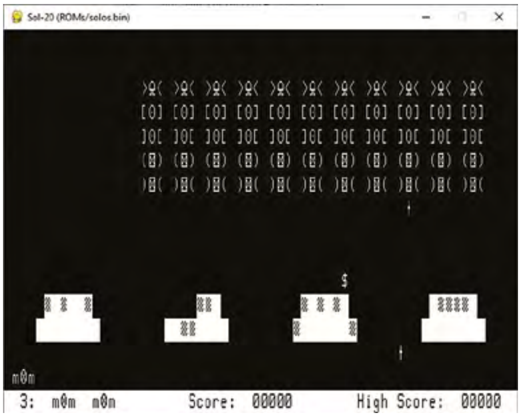
Michael Gardi

Michael is a retired software developer from Ontario, Canada who enjoys spending his free time on retro computer builds and whatever else the heck he feels like.

mikesmakes.ca

Laying claim to be the first fully-assembled microcomputer with a built-in keyboard and television output, the Sol-20 was launched back in 1976. "It had more in common with the Altair 8800s and IMSAI 8080s of the day, than it did with the Apple and Commodore computers that were soon to follow, despite looking more like the latter", says Michael Gardi who, a full 45 years later, built his own version, powered by Raspberry Pi, since a version made using only vintage components would be prohibitively expensive.

Michael's home is full of Raspberry Pi computers performing tasks from NAS to OctoPrint server and RetroPie games arcade, so it seemed obvious to use our favourite single board computer for his Sol-20 project. He chose a Raspberry Pi 4 because he needed "the horsepower it provides to run the emulator at speeds comparable to the original Sol-20". He also realised the GPIO ports and option to use a variety of HATs – particularly the voltage shifter module – made Raspberry Pi a great choice for this build.



```
m0m
3:  m0m  m0n        Score:  00000        High Score:  00000
```

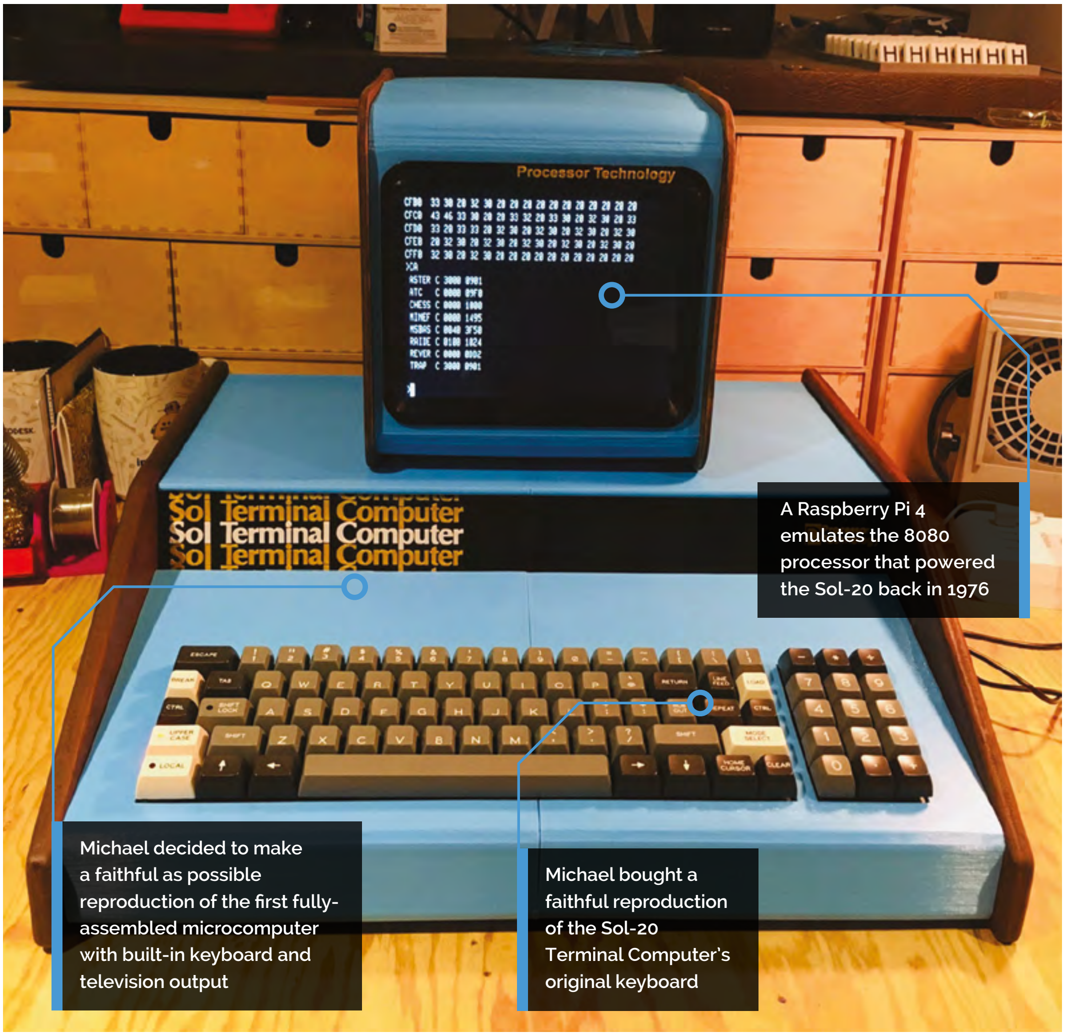
Other Sol-20 emulators exist, but Michael wanted to create one specifically to run on Raspberry Pi

In the end, the project took around four months to complete, and cost roughly US $550.

Computer love

Michael wanted his Sol-20 reproduction to be "as authentic-looking as possible, with the beautiful walnut sides for sure". He'd always admired the original computer's striking steel blue sides too, but chose to 3D-print the case for his version, pointing out that also makes it easier and cheaper for anyone else following in his retro build footsteps.

Raspberry Pi came into its own when it came to the software written for Sol-20. Recreating the motherboard to do this and populating it with vintage parts would have been prohibitively expensive, Michael explains. Raspberry Pi runs code that emulates the 8080 CPU at the heart of the Sol-20. "On top of this, it emulates the original hardware providing a display via the



A Raspberry Pi 4 emulates the 8080 processor that powered the Sol-20 back in 1976

Michael decided to make a faithful as possible reproduction of the first fully-assembled microcomputer with built-in keyboard and television output

Michael bought a faithful reproduction of the Sol-20 Terminal Computer's original keyboard

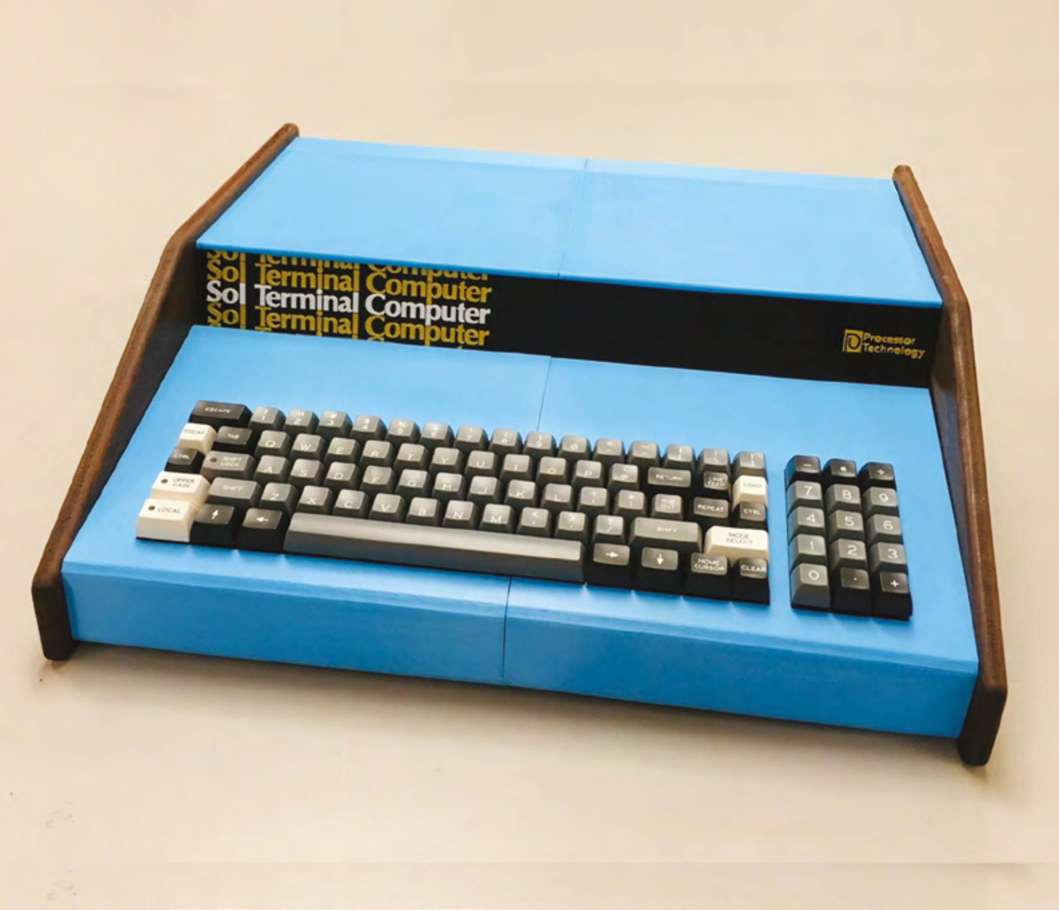
Quick FACTS

- Only 12,000 Sol-20 Terminal Computers were produced
- Original models sell for $5000 or more
- Michael spotted one in a shop in Toronto in 1976
- And has hankered after owning one ever since
- His Sol-20 emulator runs on any Raspberry Pi

> If you're thinking of your own reproduction of a classic, Michael warns you need to be prepared to be in it for the long haul

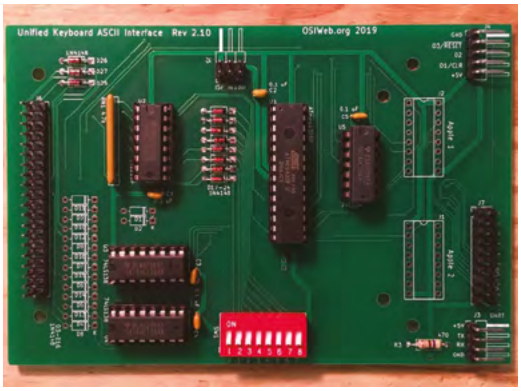
HDMI port and a virtual cassette interface for loading and saving programs." Michael used Python to write the Sol-20 hardware emulation software on top of an existing 8080 emulator that he found on GitHub (**magpi.cc/py8080**).

Faithfully recreating the 1976 Sol-20 Terminal Computer involved learning to use a CNC machine in order to make the walnut sides



▲ Michael 3D-printed the reproduction Sol-20 case, but learned to use a CNC cutter to make walnut wood sides to match the original design

▶ Fellow retro computer builder Osiweb had designed an ASCII keyboard interface board for which Michael adapted code for Raspberry Pi and Sol 20



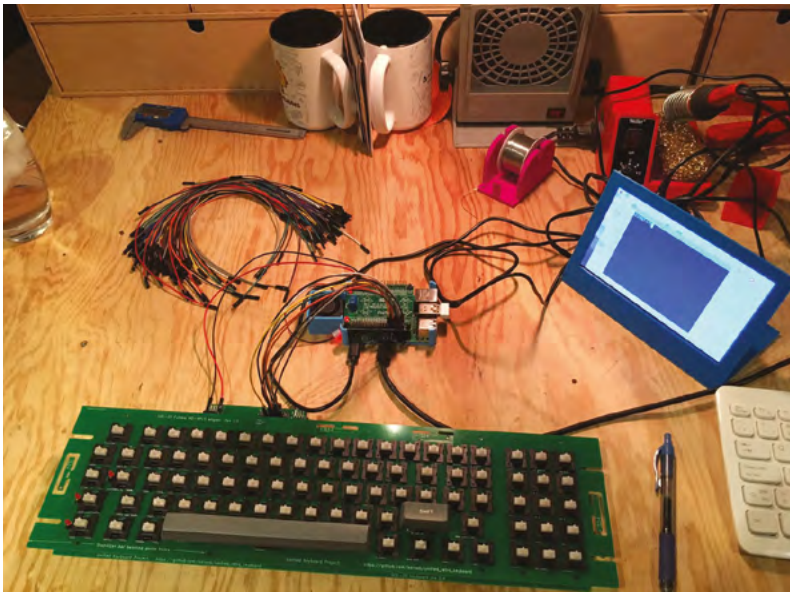
His Sol-20 emulator (**magpi.cc/sol20reproduction**) can be run on Raspberry Pi 4 with a standard keyboard and display, but another Sol-20 fan had recreated a keytronic version of the original keyboard (**osiweb.org**), and Michael couldn't resist!

Screenwise, he bought a Pimoroni PIM372 8in XGA display. The full build details and hardware list can be found at **magpi.cc/sol20bom**.

New tricks

With detailed documentation about the computer he was trying to recreate available online, Michael says the Sol-20 project went smoothly. As a retro computer build veteran, he was able to apply lessons learned from previous projects such as the KENBAK-2/5 and DEC VT100 Terminal, both of which we've featured in *The MagPi*. However, he also challenged himself, learning how to use a CNC machine at his local maker space in order to recreate the Sol-20's "gorgeous" walnut side panels.

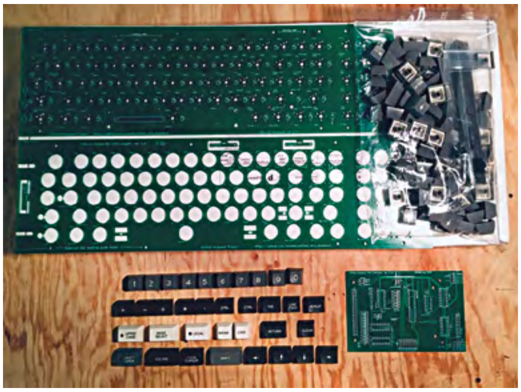
If you're thinking of your own reproduction of a classic, Michael warns you need to be prepared to be in it for the long haul. "Don't start unless you are passionate about the idea," he cautions,



▲ The reproduction Sol 20 keyboard looks great, but involves assigning key controls very carefully

before enthusing: "Break the project up into smaller pieces that can be finished in a relatively short time. Keep plugging away at it. Have fun!"

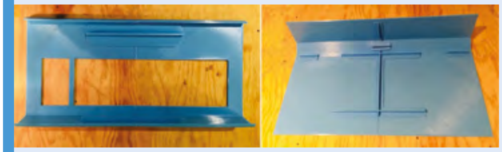
For further information on Sol-20 Terminal Computer, Michael recommends visiting **sol20.org**, where you'll also find Solace (SolAnachronistic Computer Emulation), a Windows version of Sol's interface.



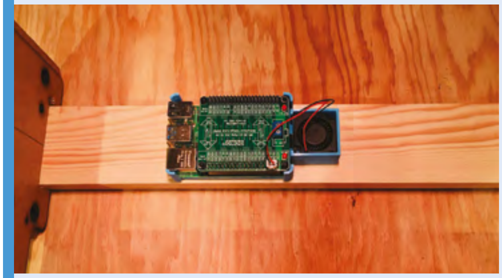
◀ Although a facsimile keyboard exists, the keyboard layout is not an exact match

Recreate Sol-20

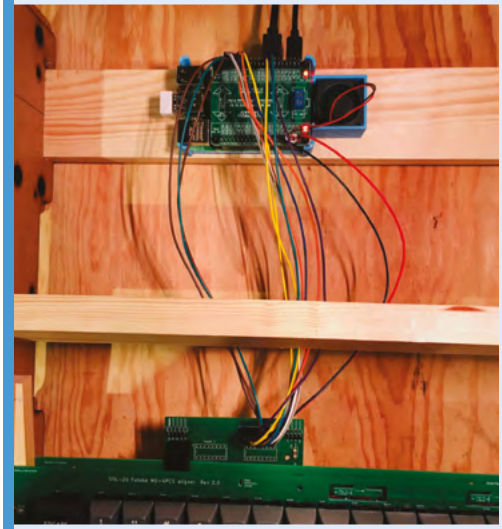
To build your own Sol-20 you'll need a Raspberry Pi 4, a 3D-printed or self-drafted computer case, an 8in 1024 × 768 × GA screen, 3.3V to 5V bi-directional voltage level adapter, USB to serial, and TTL adapters and, ideally, a keytronic keyboard.



01 Build or 3D-print the Sol 20 frame. Download the emulator software from **magpi.cc/sol20reproduction** and install it on Raspberry Pi 4 or another Python environment. Clone the repository and run python main.py.



02 Attach the 3.3V to 5V voltage adapter to Raspberry Pi 4, along with other hardware including power supply and fan.



03 Assemble the keyboard and keyboard emulator as per instructions at **magpi.cc/sol20bom**, then connect and test the keytronic keyboard and the LCD XGA screen.

Giant Raspberry Pi 3

Why have a standard-sized Raspberry Pi, when you can have a super-sized one? **Nicola King** admires a king-sized make

MAKER

Zach Hipps

Zach Hipps is an electrical engineer and maker. On his YouTube channel, Byte Sized Engineering, he builds ridiculous DIY projects with the goal to inspire people to unleash their inner maker.

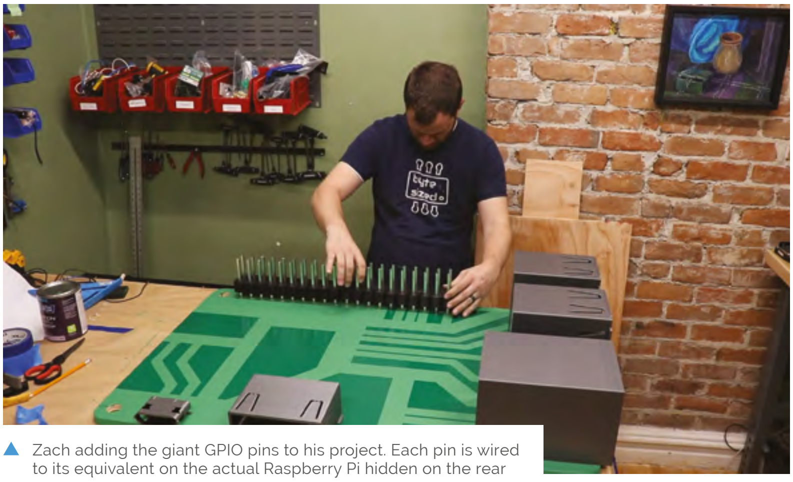
magpi.cc/bytesized

Zach Hipps doesn't like to do anything by halves; some of his past makes have included a giant Arduino board (magpi.cc/giantarduino) and what he describes as the "world's most over-engineered water rocket!" (magpi.cc/waterrocket). One of his latest creations arguably takes Raspberry Pi to a new level – he has made a giant version of the credit card-sized computer, twelve times larger than the standard, and yes… it is actually fully functional!

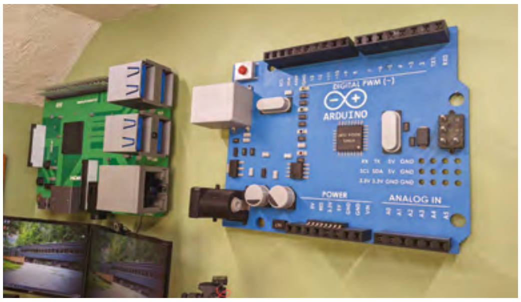
A monumental make

"I have always loved working with microcontrollers and single-board computers," Zach tells us. "At the end of 2020 I built a large-scale Arduino board and posted a video about it on my YouTube channel. I had lots of suggestions and requests that I build large-scale versions of other DIY electronics components. It wasn't a far stretch for me to want to do another similar project, and that's where the giant Raspberry Pi board was born."

Zach says he learnt a great deal from his first giant project build and used his knowledge from that in the latest construction. "The most time-consuming part was printing the large-scale components, which took over 150 hours to complete. I used over 5 kg of PLA filament on this project alone," he reveals.



Zach adding the giant GPIO pins to his project. Each pin is wired to its equivalent on the actual Raspberry Pi hidden on the rear



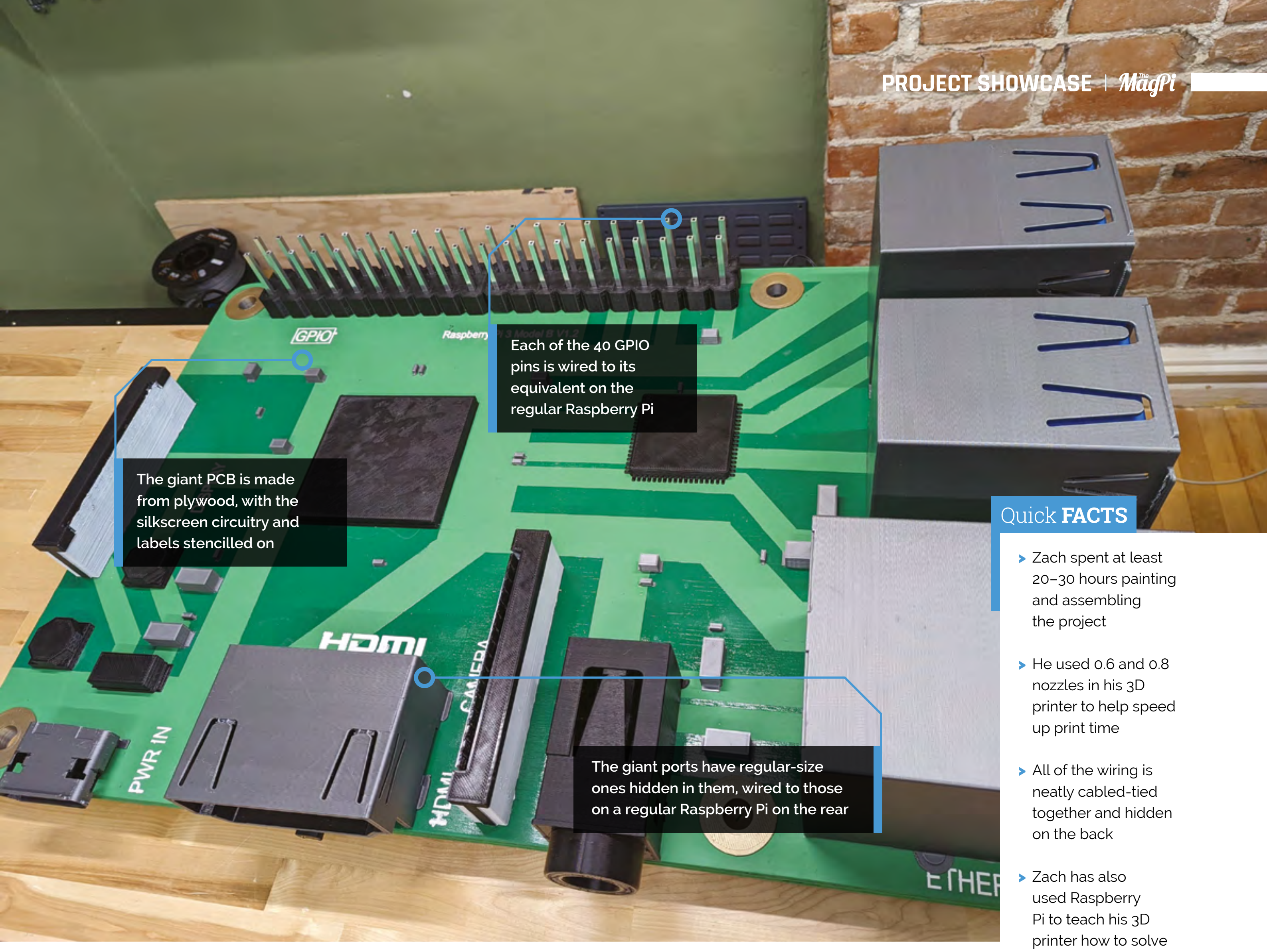
Two giant boards – Raspberry Pi and Arduino – mounted on Zach's wall, ready if needed

He used a ¾-inch-thick piece of plywood for the PCB, which he painted, stencilling on the appropriate lettering for an authentic look. A standard Raspberry Pi 3 kit, with a power supply and case, was hidden on the back of the board. "I connected my Raspberry Pi to all the large-scale connectors with extension cables. I plugged in a monitor and keyboard and everything fired right up! In the video I made (**magpi.cc/giantraspberrypi**), you can see the giant-sized smile (very appropriate for this project) on my face when I saw everything boot up for the first time."

One of the issues he faced was how to connect the GPIO pins of the real Raspberry Pi to the large-scale GPIO pins, but he worked around this neatly. "I found that ¼″ (6 mm) square aluminium tubing works perfectly for the 12× scale of this project. I used a 40-pin GPIO ribbon cable and soldered a long extension wire to each pin. Then I crimped a ring terminal on the end of each extension which was then screwed and secured to the bottom of each GPIO pin."

On the wall

Zach currently has both his giant Arduino Uno and giant Raspberry Pi 3 hanging up on the wall above his desk, with their connectors easily accessible in case inspiration strikes and he wants to prototype an idea. However, he's keen that other makers get to see them soon: "Once we start having public meet-ups again, I would love to bring these giant-sized boards to maker events. The whole goal with



Each of the 40 GPIO pins is wired to its equivalent on the regular Raspberry Pi

The giant PCB is made from plywood, with the silkscreen circuitry and labels stencilled on

The giant ports have regular-size ones hidden in them, wired to those on a regular Raspberry Pi on the rear

Quick FACTS

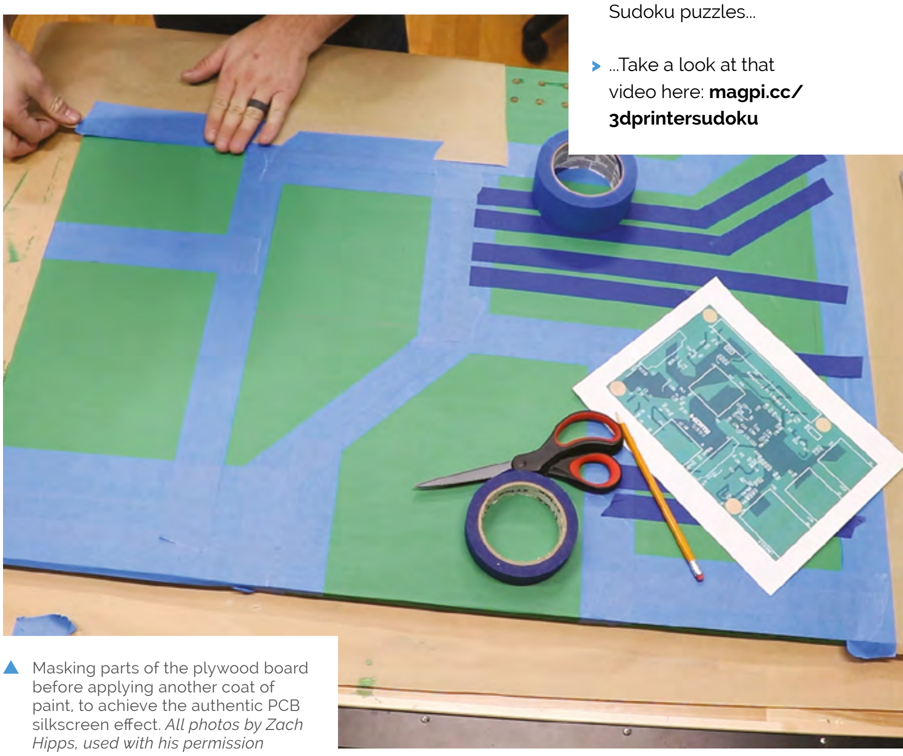
- Zach spent at least 20–30 hours painting and assembling the project
- He used 0.6 and 0.8 nozzles in his 3D printer to help speed up print time
- All of the wiring is neatly cabled-tied together and hidden on the back
- Zach has also used Raspberry Pi to teach his 3D printer how to solve Sudoku puzzles…
- …Take a look at that video here: **magpi.cc/3dprintersudoku**

> “ The whole goal with my projects and YouTube channel is to get people excited about making things ”

my projects and YouTube channel is to get people excited about making things. I want to build crazy and ridiculous things to inspire others to unleash their inner maker.”

So, any more gargantuan makes on the horizon? “I had lots of great suggestions that came pouring into the comment section of the video I posted,” he responds. “I know my channel name is Byte Sized Engineering, and the irony is not lost on me. But this won’t be the last giant-sized project you see me build.”

Zach possesses creativity and inventiveness in bucket-loads and his aim is clearly to inspire others, so take a look at his library of fascinating and entertaining YouTube videos, and get creating!



▲ Masking parts of the plywood board before applying another coat of paint, to achieve the authentic PCB silkscreen effect. *All photos by Zach Hipps, used with his permission*

PiGlass V2

A new Raspberry Pi Zero means a new take on PiGlass, the smart glasses powered by Raspberry Pi. **Rob Zwetsloot** takes a look

MAKER

Matt Desmaris

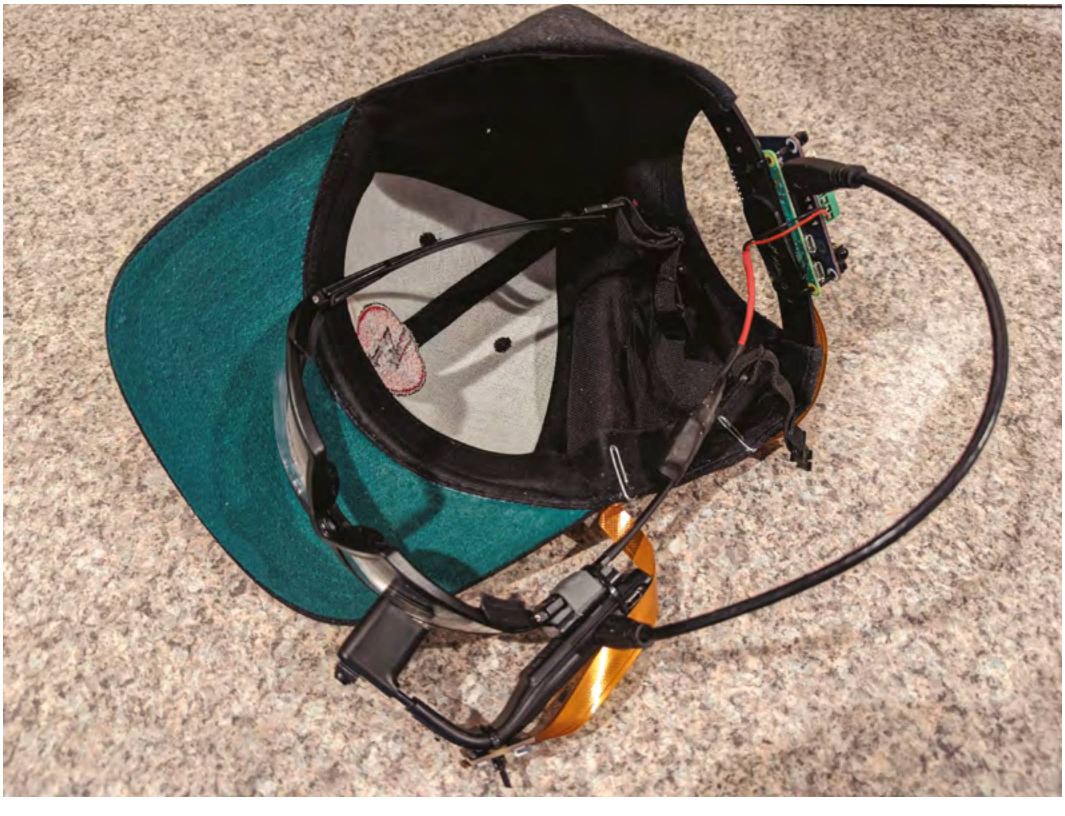
Matt is a volunteer at his local food pantry as the IT department, and has been using Raspberry Pi for ten years.

mrdcreations.org

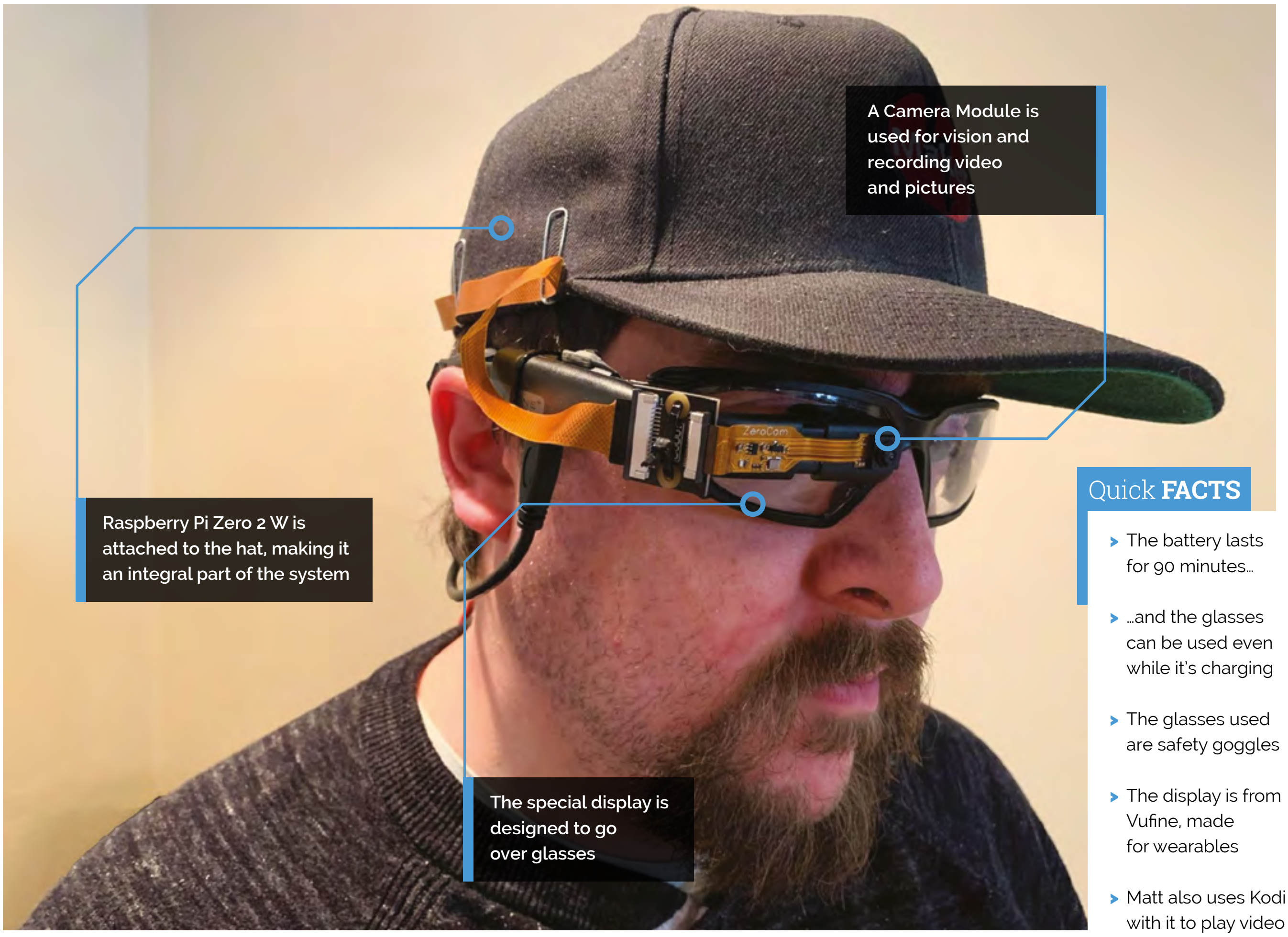
Do you remember Google Glass? They were smart glasses with a bit of augmented reality thrown in. Unfortunately, they never became a full product; however, with advancements in computer vision and miniaturisation of computers, custom smart glass builds never really went away.

One maker, Matt Desmaris, recently revisited the idea. "I made PiGlass V1 in 2018," he explains. "I wanted to try to make a heads-up display wearable and see how far I could take it. I kept running into performance issues with Raspberry Pi Zero when I was trying to add more features, and I made a note to revisit the project when Raspberry Pi Zero 2 came out."

PiGlass V2 has some extra features, like bone conduction earphones. It also makes use of a wearer's hat.



A better view of the wiring of the system



A Camera Module is used for vision and recording video and pictures

Raspberry Pi Zero 2 W is attached to the hat, making it an integral part of the system

The special display is designed to go over glasses

Quick FACTS

- The battery lasts for 90 minutes…
- …and the glasses can be used even while it's charging
- The glasses used are safety goggles
- The display is from Vufine, made for wearables
- Matt also uses Kodi with it to play video

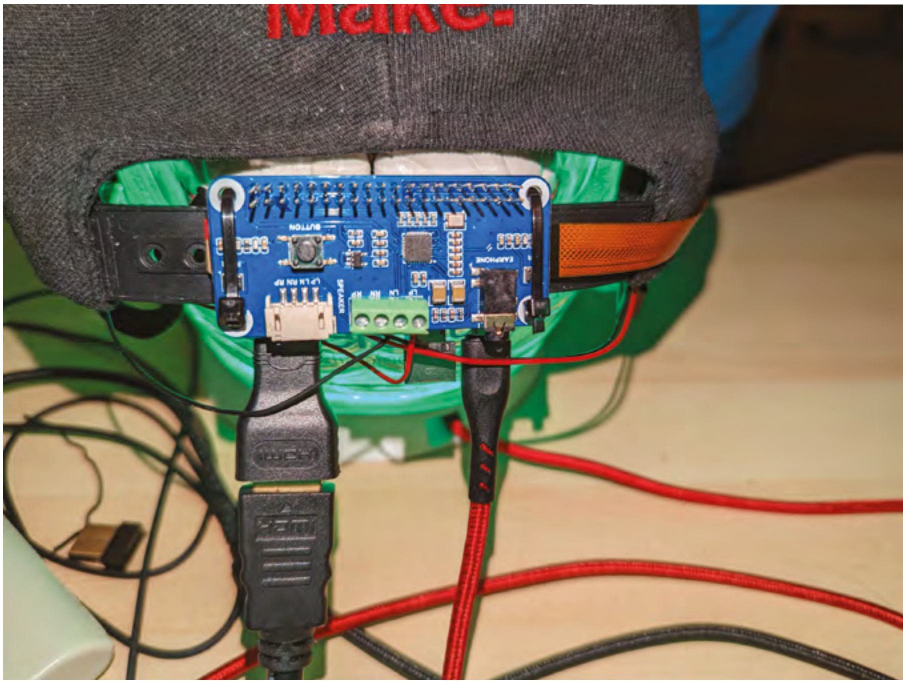
> The whole system is controlled with a gamepad, which makes for an interesting sight

Virtual vision

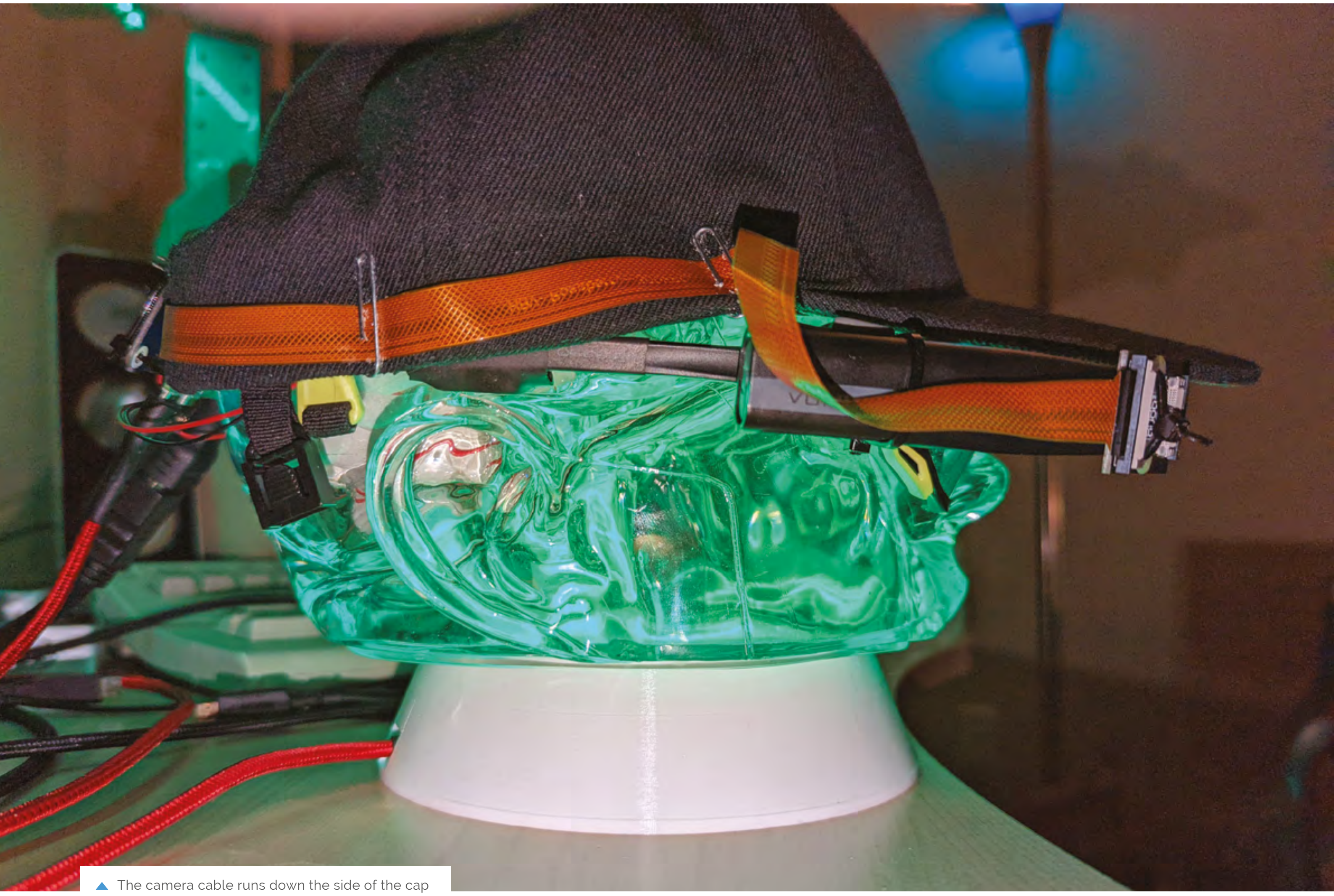
Matt was in contact with us throughout the process of making this and the build, actually, was quite quick, as he elaborates: "Construction was straightforward – a soldering iron, heat gun, and small flat-head [screwdriver] are all that's required. Construction took place over a couple weeks as I progressively got more and more parts in. Everything is secured with zip ties or heat shrink."

The whole system is controlled with a gamepad, which makes for an interesting sight.

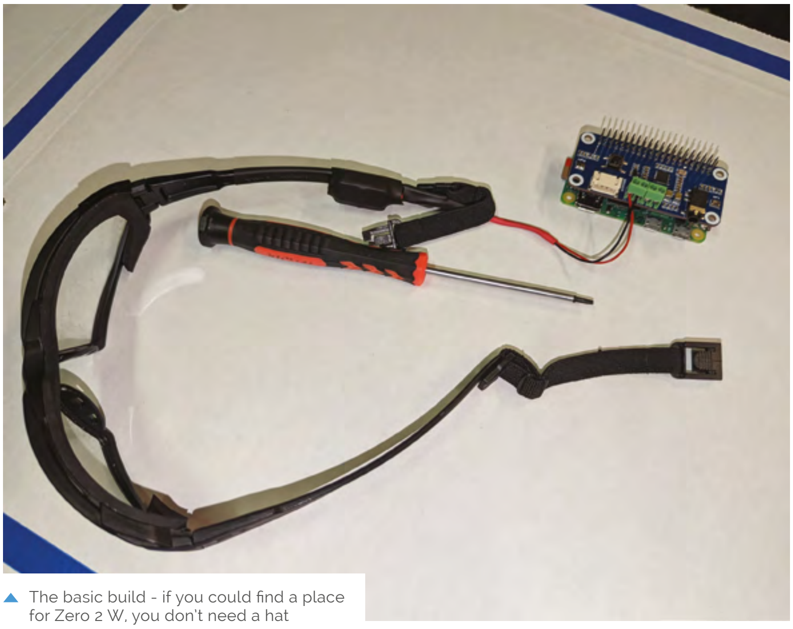
"There is a button on the audio hat on the back of my head," Matt reveals. "It is the start/stop button. Start the menu program



▲ The main system is mounted on the rear of the hat, feeding to the display



The camera cable runs down the side of the cap



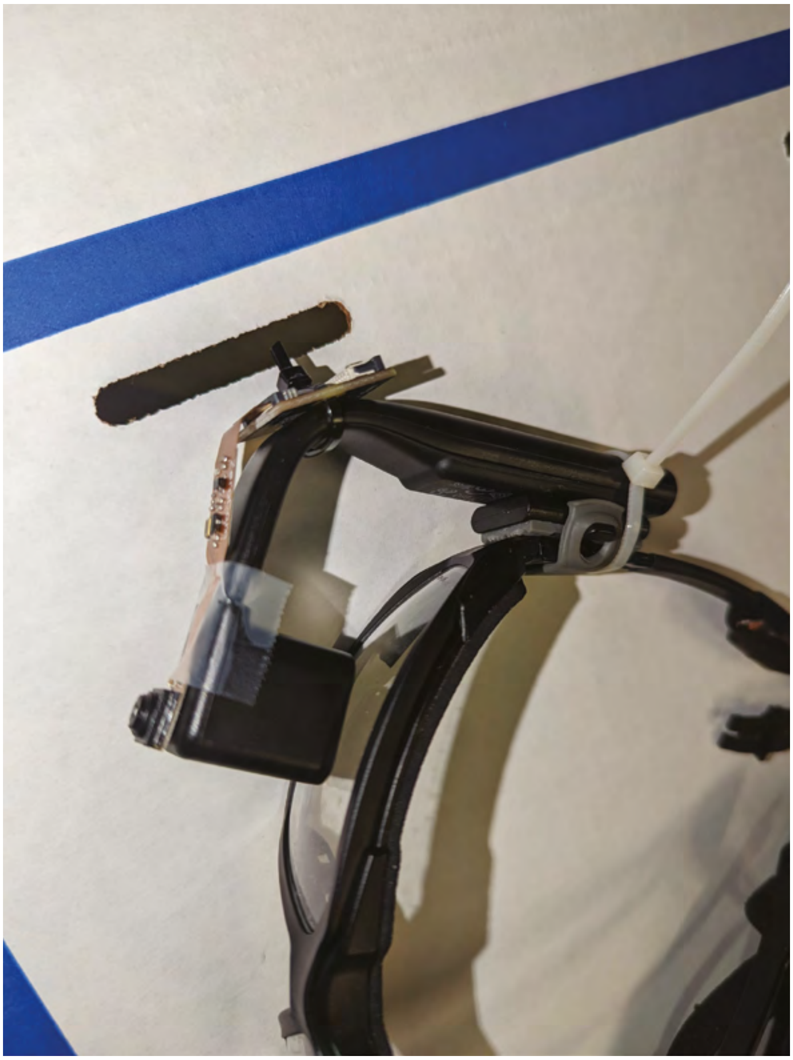
The basic build - if you could find a place for Zero 2 W, you don't need a hat

or kill every program that could be running. The menu program uses the [picamera] API which allows it to be recorded, including the text overlays. The menu program has a few options: camera (from PiGlass V1) at 1080p, record video with audio at 1080 p, stream YouTube at 720p, Emulation Station, Kodi at 720p, and Steam Link [with controller issues]."

While Matt developed it hooked up to a monitor, he's tested it thoroughly by watching streaming video and playing some retro games – something we heartily approve of.

Seeing cyber

"Within the menu/related programs it works very well; some things take a few seconds to load," Matt tells us. "The 720p display looks really good; all the text in menus and captions are easily readable. The camera program allows you to take 1080p images/silent videos, and video with audio records at 1080p, 25 fps, with 44kB audio.



YouTube stream will livestream to YouTube in 720p, 25 fps, with 44kB audio."

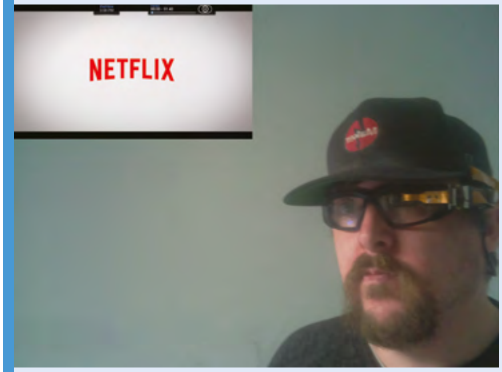
With the project now at an advanced stage, Matt has been thinking of future improvements as well: "I want to add voice commands and I have a few ideas. The microphones are located on the back of my head and they can pick up my voice at normal speaking levels. I want to add real-time object detection. I have tested demo code and it looks like [Raspberry Pi] Zero 2 W can do it. I want to make a program that uses the gamepad to be able to select which type of object is being detected."

We very much look forward to the cyberpunk cyberglass future.

▲ The display is designed for this kind of use

Using smart glasses

01 Using the controller, you can choose what function you'd like to use. These are displayed as words over the feed image.



02 Watching streaming services will allow you to sit back and relax on a commute, or while you wait for something.

03 For gaming, you're able to use the controller you use for the rest of navigation, making the whole process more streamlined.

Fancy Octopus Arcades

Inspiring the next generation of gamers led to a business creating custom arcade housings, discovers **Rosie Hattersley**

MAKER

Shonee Strother

Shonee swapped a role as creative director and fulfilled a childhood dream to build gaming arcades, with assistance from his "highly talented" six-year-old son.

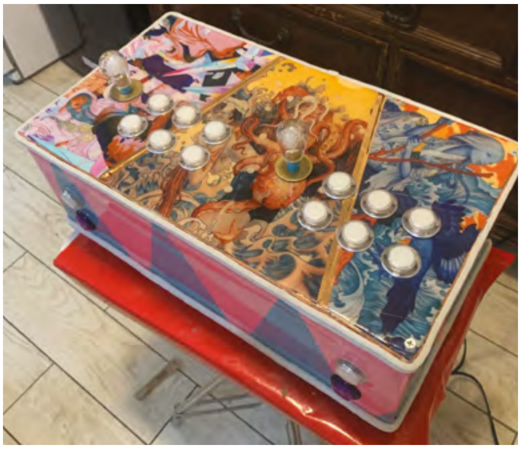
magpi.cc/fancyoctopus

"**We're in the golden age of emulation," declares Shonee Strother.** He doesn't have a standard technical background, but he's been a gamer all his life and, when the pandemic put his role as a creative director on hold, he began looking for new hobbies. Shonee and his six-year-old son Wolf started their Raspberry Pi adventures "by modding an arcade 1up with a Raspberry Pi 3B." Wolf is extremely interested in gaming, like his dad, who felt that his son's young age meant "there's so much gaming history he's missed. Giving him a slice of that was super important." When Shonee documented their work on Instagram, people began to take notice. Their bespoke gaming arcade business, Fancy Octopus Arcades, "just sort of took off from there."

Bijou is beautiful

Brooklyn resident Shonee started following a few Reddit and Facebook pages where he "saw folks making huge gaming rooms, loading them up with arcade cabinets, and realised that for some of us in smaller apartments, that wasn't a possibility." Shonee was also frustrated that people were capitalising on the scarcity of arcade cabinets, "snagging all the 1up machines they could find and price gouging them to death." He decided to address pent-up demand for games arcades, focusing on designs that would fit a New York-sized apartment.

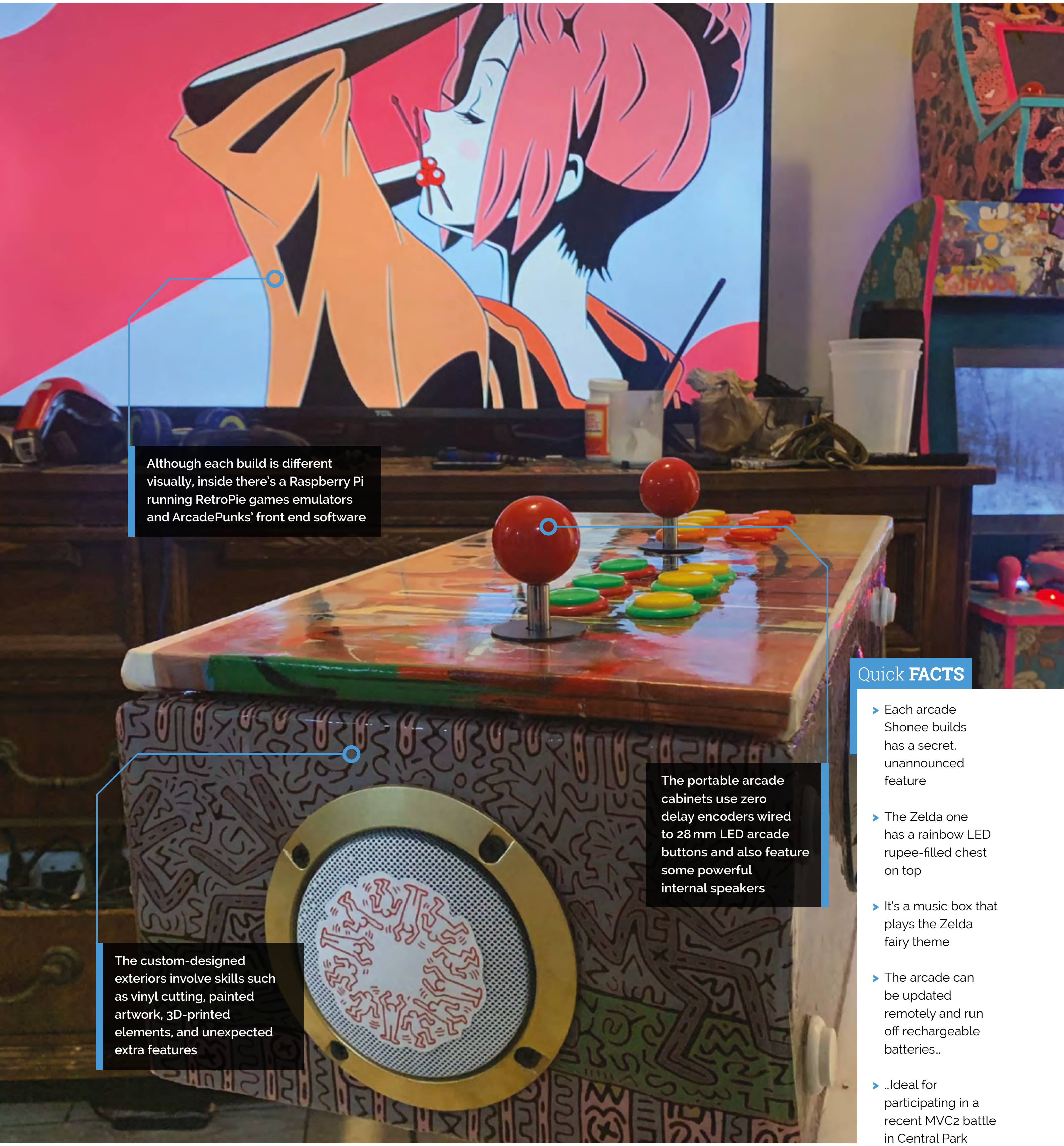
The idea was to make small, mini cabinets, packed to the gills with games, easily plugged via standard HDMI for video audio output. Shonee also wanted them to be highly personalised art pieces "that a person could be proud to put on a display shelf or coffee table." Raspberry Pi's small profile and extremely robust performance helped him "pack a lot of oomph into a very small package," he confirms. He also loves being able to easily flash images directly onto the SD card with little or no manual coding changes. As a result, he can focus on each cabinet's artistry.



The unique retro arcade builds have proved extremely popular, with handheld designs planned soon

Daring designs

Although homemade retro games cabinets are a popular build, Shonee's stand out because he designs and creates all the parts himself. The 3D printing, silkscreen, vinyl wrap, and decoupage work is all done in-house, depending on the build. Shonee works closely with clients, creating a design around their vision. "No two builds are ever alike, and no two designs are ever repeated," he says of what is now a two-year-old business. "I've had so many folks ask me 'can you build me a copy of that Donkey Kong deck?' My bank account absolutely hates that I have to say no." The unique elements that bring his designs to life include gear sticks masquerading as swords for his Samurai



Although each build is different visually, inside there's a Raspberry Pi running RetroPie games emulators and ArcadePunks' front end software

The portable arcade cabinets use zero delay encoders wired to 28 mm LED arcade buttons and also feature some powerful internal speakers

The custom-designed exteriors involve skills such as vinyl cutting, painted artwork, 3D-printed elements, and unexpected extra features

Quick FACTS

- Each arcade Shonee builds has a secret, unannounced feature
- The Zelda one has a rainbow LED rupee-filled chest on top
- It's a music box that plays the Zelda fairy theme
- The arcade can be updated remotely and run off rechargeable batteries…
- …Ideal for participating in a recent MVC2 battle in Central Park

▲ Arcade cabinet designs feature elements from the client's favourite games

▲ Following the success of the bespoke arcades, Shonee is keen to branch out into custom handheld consoles

◀ This RetroPie arcade features a barrel-shaped planter from a garden store

The handle for this Haring vs Basquiat build is from a 1980s Panasonic boom box that Shonee bought at a NY flea market

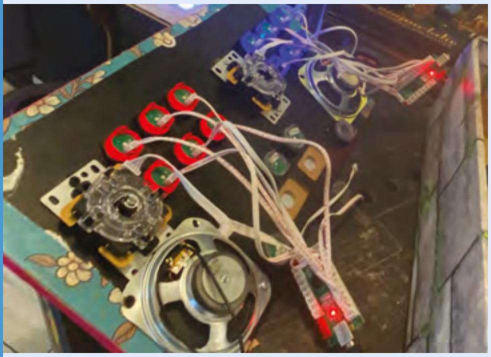
Jack build, the handle from a Panasonic boom box found at a Chinatown flea market and used in his Basquiat/Keith Haring build, and a Donkey Kong barrel refashioned from a garden centre planter. "Basically, I use whatever I can." More often than not, he uses software from **Arcadepunks.com**. Their front ends are "stable, responsive, and rarely cause problems down the line," Shonee says.

"Some folks really want to focus on retro arcade gaming; some folks are more focused on having a game preservation library of console games from their past. There are always some adjustments that need to be made. Luckily, with the versatility of Raspberry Pi, it's pretty easy." Each Fancy Octopus Arcade typically takes about a month to complete, with costs varying depending on the brief.

"Without Raspberry Pi, this would just be a box with buttons on it," Shonee concludes. "The size, power, versatility, and ease of use have given me the ability to help the dream of making my hobby a career come to fruition."

Pimp your arcade

Build the arcade cabinet, either by downloading and printing designs from a site like Thingiverse or creating a custom wood build. You'll need a Raspberry Pi, zero delay encoders for buttons and joysticks, speakers, a mini amp, and some creativity.



01 Wire up the unit

Connect zero delay encoders to 28 mm LED arcade buttons. For internal audio, add a mini amp and stereo speakers mounted to the control deck.

02 Allow for updates

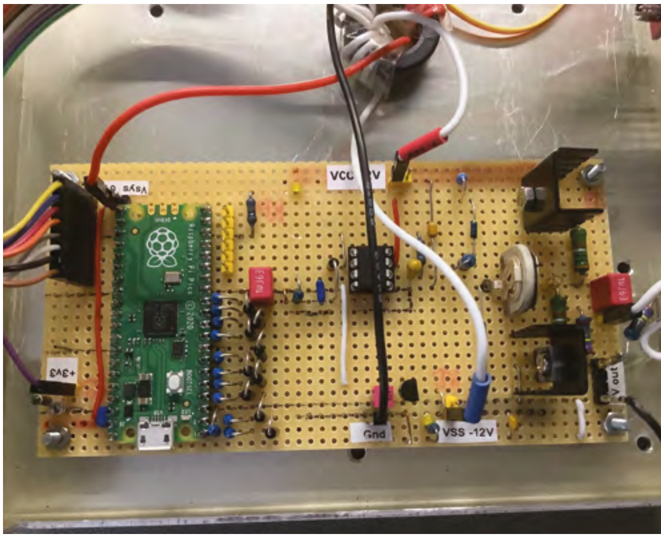
Connect Raspberry Pi to an external Ethernet port for any updates or system changes. Install a USB hub to power the LEDs, and attach a mini keyboard for any edits that may be needed.

03 Testing, testing

Flash-update RetroPie and run it to make sure the front end works with the library you've installed. Depending on the display you're using, check for scan lines, and adjust frame rates and the aspect ratio.

Arbitrary waveform generator

Waveforms can be used to develop and test electronic devices. Raspberry Pi Pico proves the ideal basis, learns **Rosie Hattersley**



Breadboarding the AWG project allowed for efficient testing, while the micro-gui he adapted worked straight away

MAKER

Wolfgang Strenzl

Electrical engineer Wolfgang enjoys projects that combine hardware and software and is a big fan of the Internet of Things.

magpi.cc/arbwaveform

An 8-bit DAC provided reliable signal amplification and low output impedance

Sharing the small projects he makes, often with an Internet of Things slant, is Wolfgang Strenzl's way of connecting with other makers and sharing his knowledge. He particularly enjoys combining hardware and software "to build little tools", noting that "many of his projects make use of software done by the open-source community." His Raspberry Pi Pico arbitrary waveform generator came about when he needed a tool to create a signal as an input while repairing some equipment.

Wolfgang had made a hardware-based arbitrary waveform generator before, but having researched the capabilities of the RP2040 microcontroller and its support for DMA and PIO, he realised Raspberry Pi Pico would be ideal for amplifying waveform signals. MicroPython, which Pico natively supports, also happens to be Wolfgang's preferred programming language. He decided to add an output stage "to amplify the signal of the RP2040 and make it short circuit-proof," plus a display to make the results clearer. The amplifier was needed because the RP2040 can only drive small loads, he explains. Initial tests using a breadboard proved the concept worked well, while online research suggested other users were also searching for a device of this kind, and may well appreciate having a design they could use too.

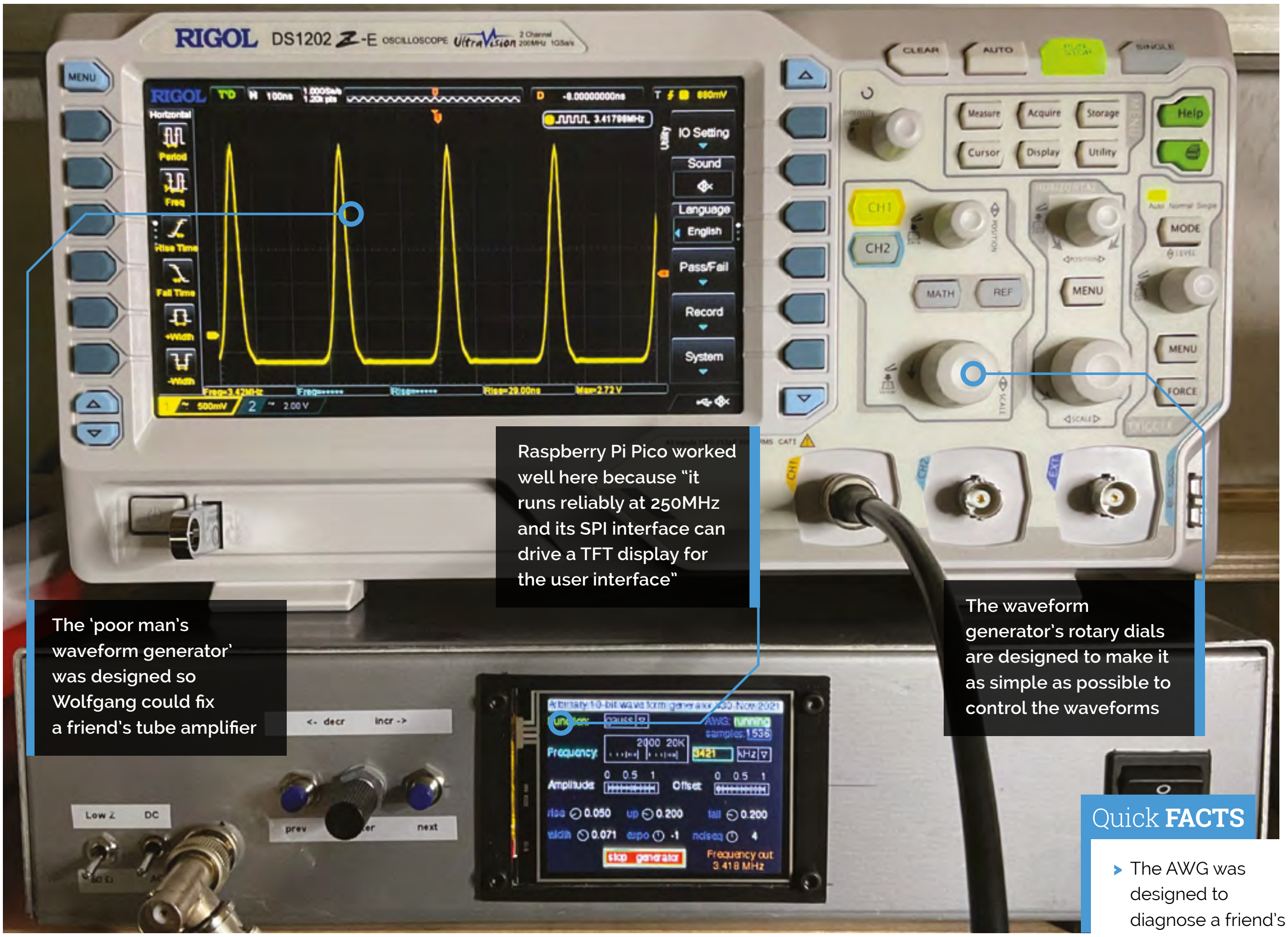
In praise of Pico

"Pico is the perfect microcontroller board for this project. It has plenty of RAM and flash memory. The RP2040's CPU clock is specified for 125MHz, but runs reliably at 250MHz, which for my project enables output frequencies well above 20MHz, while the SPI interface can drive a TFT display for the user interface," Wolfgang says of the Pico's impressive specifications.

It wasn't all plain sailing, though. "Ideally a function generator should have a clean output signal with no distortions or noise." However, using Pico to both generate the wave and power the TFT display via the SPI interface resulted in a noisy output signal because of cross-talk from the SPI interface to the PIO output pins, Wolfgang reports. Making some changes to his breadboard layout improved things, but didn't get rid of all the signal noise. Instead, he took charge of how the TFT updated. "Once the user has entered or adjusted the wave and started the generator output, the display refresh is turned off. This turns off the SPI, which reduces most of the noise," he explains.

Community input

A fan of rapid prototyping, he split the development into three phases beginning with the generator engine. Here, he added some control code that fellow maker Rcgo had written up as an Instructable (**magpi.cc/wavegenerator**) that enabled Raspberry Pi Pico to calculate a wave and program the DMAs and PIO. Wolfgang was able to modify this for the AWG core, adding a list of data elements the generator module would use to create the desired wave based on the user's input. Another community-contributed project, Peter Hinch's micro-gui (**magpi.cc/microgui**), proved "the perfect base to build a user interface for small



Raspberry Pi Pico worked well here because "it runs reliably at 250MHz and its SPI interface can drive a TFT display for the user interface"

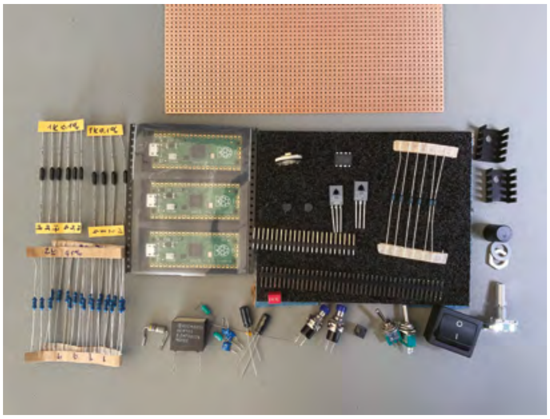
The 'poor man's waveform generator' was designed so Wolfgang could fix a friend's tube amplifier

The waveform generator's rotary dials are designed to make it as simple as possible to control the waveforms

> A few days after Wolfgang started designing it, the so-called poor man's AWG was used to diagnose and fix his friend's tube amplifier

embedded systems." Peter helped Wolfgang put together a knob widget to go with the two push-buttons and rotary encoder he'd set up to work alongside micro-gui and the TFT controls. These dials were critical, since they govern changes to the waveform created.

The class AB output stage consists of discrete components including "a nice video driver IC that provides low output impedance," while compromising with an 8-bit DAC (rather than a better resolution 10-bit one) avoided timing problems at higher frequencies. "Four wave data samples are loaded every other clock cycle, so there is time for the DMAs to reconfigure, while the PIO still has data to shift out. Running the CPU at 250MHz, waves can be generated well above 20MHz," Wolfgang reports. Just a few days after he started designing it, the so-called poor man's AWG was used to diagnose and fix his friend's tube amplifier. "One more happy friend," says Wolfgang.



The Raspberry Pi Pico-based project cost just $60 and took a few days to complete

Quick FACTS

- The AWG was designed to diagnose a friend's tube amplifier
- But RP2040's DMA and PIO do the real work here
- Wolfgang's waveform project was completed in just a few days
- He's previously built Raspberry Pi backyard wildlife cameras…
- …and has a Raspberry Pi 4 family recipe server!

Pica and Dot

Kaisar has been thinking big in his efforts to create small robots but he's not yet finished, as **David Crookes** finds out

MAKER

Kaisar

Kaisar is an embedded engineer and an avid maker who is keen on robotics, computer vision, and AI.

github.com/kaisarh

Dot is the latest creation by maker Kaisar who has been progressively creating ever tinier robots for the best part of a year. He began by introducing CutieBot using a Raspberry Pi Zero computer in June 2021. He followed it up last November with Pica, based around a Raspberry Pi Pico development board.

Pica was born almost by accident. "I'd bought a few 6 mm geared motors about a year ago and, when I was sorting some parts on a Sunday morning, I found a Raspberry Pi Pico board and the motors were on the table at the same time," Kaisar says. "Some combination of neurons fired in my brain and the idea of Pica was born."

This effort used two motors, a DRV8833 motor driver, and a 70 mAh lithium-polymer battery. It moved along on three 8 mm wooden beads acting as wheels. To make it as small as possible, Kaisar decided to saw away Raspberry Pi Pico's lower section.

"I had a number of these boards, so I took the risk to trim one, up to the crystal," he explains. "Trimming a Raspberry Pi Pico board is not something I recommend, and it's also not required for the functional operation of this robot."

Joining the dots

Having taken to Twitter to showcase Pica, he wanted to go even smaller. This time he turned his attention to Pimoroni's Tiny 2040 – a development board that, as with Raspberry Pi Pico, includes the RP2040 microcontroller chip.

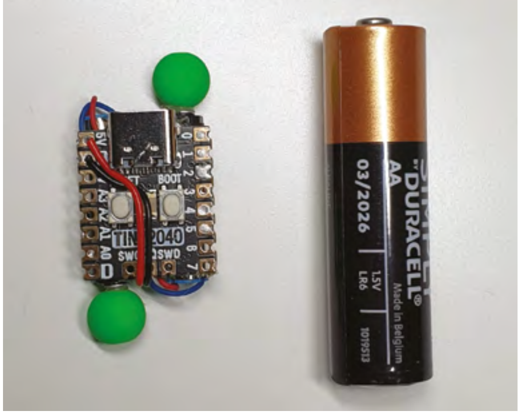
"I just wanted to make a smaller version of Pica utilising the Tiny 2040 board," he says. Given the Tiny 2040 is roughly the size of a postage stamp, he could confidently claim on social media to have made the smallest Raspberry Pi robot in the world. This was Dot.

"The initial plan was to have the same DRV8833 motor driver and 6 mm motors," he reveals. "But because of the much smaller size of the Tiny 2040, it was a little tricky to find room for the motor driver and still maintain the low profile and cuteness of the robot. Then I had the idea of measuring the motor current to see if they can be used without a motor driver."

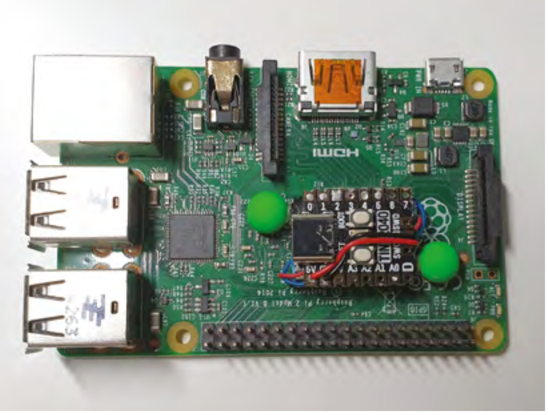
He found that they could, so the two 3 V geared DC motors were connected directly to the Tiny 2040 board's GPIO pins. Kaisar admits there is a risk of damage – "running even these small motors off the GPIO should be avoided because of the potential spikes and high stall currents," he says – but it allowed the robot to be as small and cute as possible.

Wooden wheels

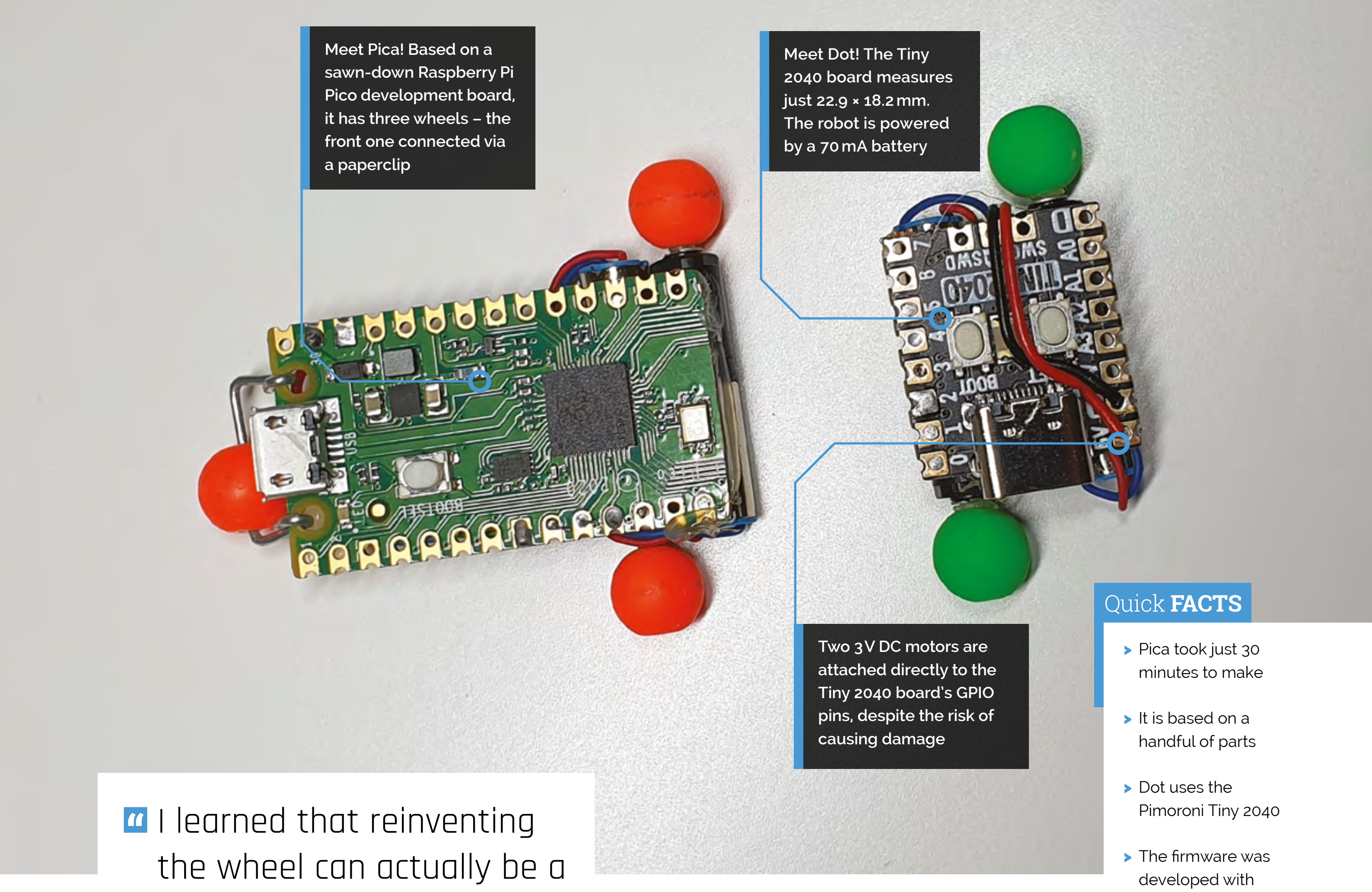
This time, Kaisar used just two wooden bead wheels, with Dot having a zero-degree turning circle. "I think they look awesome," Kaisar says. "They also get their uniqueness from the fact that the wheels are not on the same axle, so I learned



▲ The Dot is so small, thanks to it being barely larger than the Pimoroni Tiny 2040 board, that it is almost dwarfed by an AA battery



▶ Here's a little game. Can you see a robot in this image of a Raspberry Pi 2 Model B computer?



Meet Pica! Based on a sawn-down Raspberry Pi Pico development board, it has three wheels – the front one connected via a paperclip

Meet Dot! The Tiny 2040 board measures just 22.9 × 18.2 mm. The robot is powered by a 70 mA battery

Two 3 V DC motors are attached directly to the Tiny 2040 board's GPIO pins, despite the risk of causing damage

> I learned that reinventing the wheel can actually be a satisfactory experiment

that reinventing the wheel can actually be a satisfactory experiment."

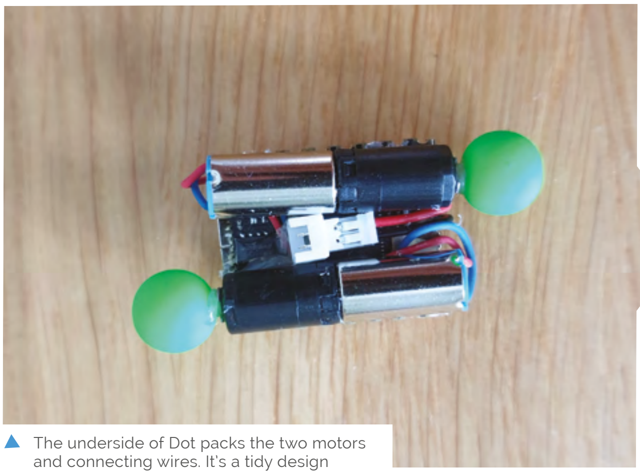
Dot is powered by a 70 mAh LiPo battery and, as with Pica, it has been programmed using CircuitPython. But there are limitations. "The robots have very minimal functionality so far – the moves are hard-coded," Kaisar continues. "They don't have remote controlling over wireless LAN, Bluetooth, or Zigbee either because that doesn't excite me too much.

Even so, Kaisar is looking to improve both robots. He wants Dot to benefit from voice control and Pica to be able to draw. "Work is also in progress for machine-learning-based autonomous controls developed with the C++ SDK which will be made open-source," Kaisar says, thinking big.

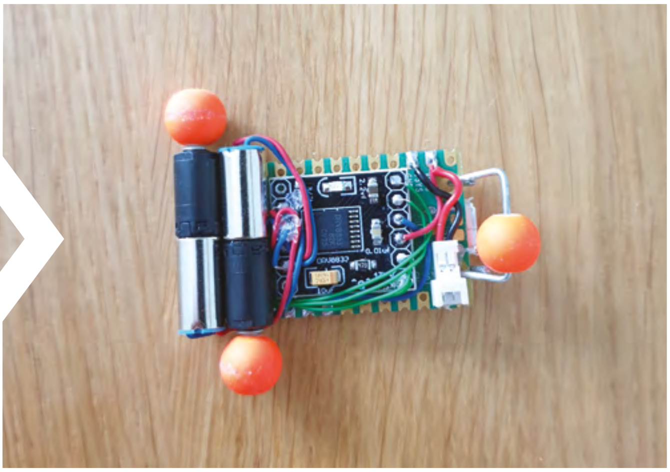
Quick FACTS

- Pica took just 30 minutes to make
- It is based on a handful of parts
- Dot uses the Pimoroni Tiny 2040
- The firmware was developed with CircuitPython
- Kaisar wants to use C++ code

▼ Pica is a larger robot but not by much. Two motors are connected at the back with a third wheel for stability



▲ The underside of Dot packs the two motors and connecting wires. It's a tidy design



Speech-to-text badge

Aimed at people who are struggling to hear someone speak, this project by Kevin Lewis has got people talking, as **David Crookes** discovers

MAKER

Kevin Lewis

Kevin is a Developer Advocate at Deepgram, which means he gets to help people build cool projects with voice.

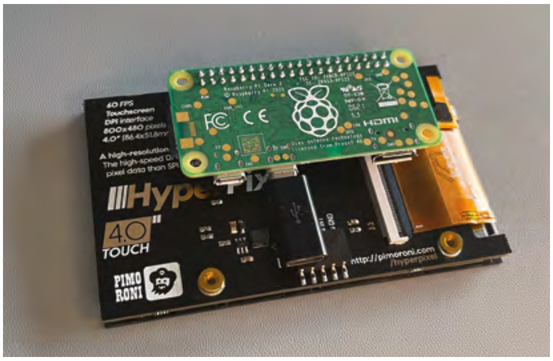
magpi.cc/s2tbadge

Excited about attending his first physical event in two years, Kevin Lewis was looking forward to chatting to lots of people. But he was also worried that he wouldn't be properly understood because he intended to wear a mask. "I won't be removing my mask under any circumstances," he said, prior to the event. "But I also find it hard to understand people when they wear their mask. I thought this would be a good chance to build something useful."

Kevin's idea was to transcribe everything he said in real-time and display it on a screen that could be worn as a badge. Since he works at Deepgram – which offers a deep learning speech recognition system – he already knew what software he was going to use. One of Kevin's friends, Bevis, then suggested he pick up a Raspberry Pi Zero computer for the project. "I wanted to create a wearable and I needed the smallest and lightest option," he explains.

Listening intently

In essence, the project is a full-screen browser application. It runs on Raspberry Pi Zero which is connected to a hi-res HyperPixel 4.0 display and the setup is powered via a battery pack. A generic USB lapel microphone is also plugged in, and this is intended to sit on the wearer's shoulder so that it picks up the person's speech.



Raspberry Pi Zero will connect directly to the back of the HyperPixel 4.0 four-inch display

From there, it was a case of incorporating the Deepgram API into a program. "At its core, the whole Deepgram feature is just twelve lines of code. It asks for the microphone, accesses raw data, opens a persistent WebSocket connection with Deepgram, and sends mic data every 250 milliseconds," Kevin says. "It then logs the results when they come back. Every spoken word is transcribed onto the screen."

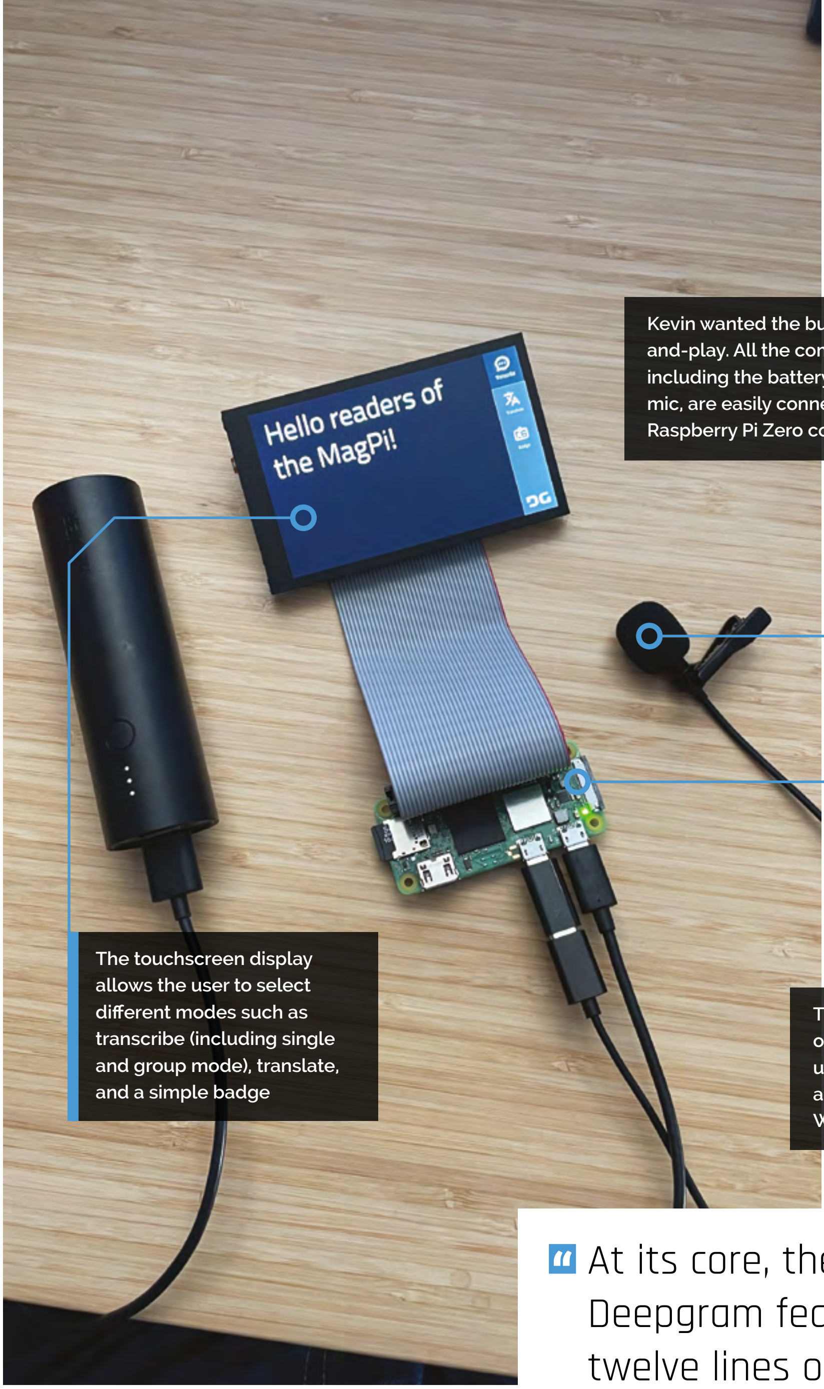
As a coder, Kevin had few problems. "The software development side of the project was perfectly within my wheelhouse," he says. "But as the project grew in complexity, so did the software, from plain HTML and JavaScript to Vue.js, and finally implementing a back-end with Node.js to handle translation."

Look who's talking

Initially, the intention was to only transcribe his own voice, but the project soon evolved. Since Deepgram works by detecting everyone who is speaking (and can transcribe many languages), Kevin initially ensured that only the first voice would be displayed on the screen – this meant he could talk and know that only his words were being transcribed.

"But because the data was there, it was extremely simple to include a mode that would also show all of the user voices," Kevin says. The spoken words of each person in range of the microphone are shown on the screen in a different colour. "A feature called 'diarisation' detects different speakers and returns data about who is speaking."

After showcasing the project on Twitter, one of the most common requests was for a translation option so that words could appear in a different language. "Deepgram doesn't do this," Kevin says. "So I found a lovely, simple translation API called



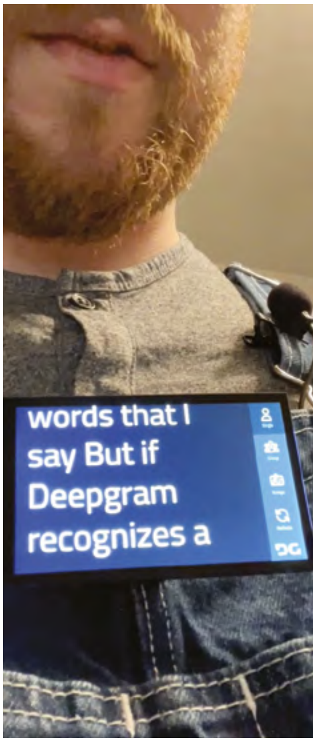
Kevin wanted the build to be plug-and-play. All the components, including the battery pack and mic, are easily connected to the Raspberry Pi Zero computer

The touchscreen display allows the user to select different modes such as transcribe (including single and group mode), translate, and a simple badge

The software installed on Raspberry Pi Zero uses the Deepgram API and opens a persistent WebSocket connection

Quick FACTS

- Speech is displayed in real-time
- It can transcribe and translate
- The first version took two hours to make
- Kevin wants to create a custom case
- It cost around £60 in parts

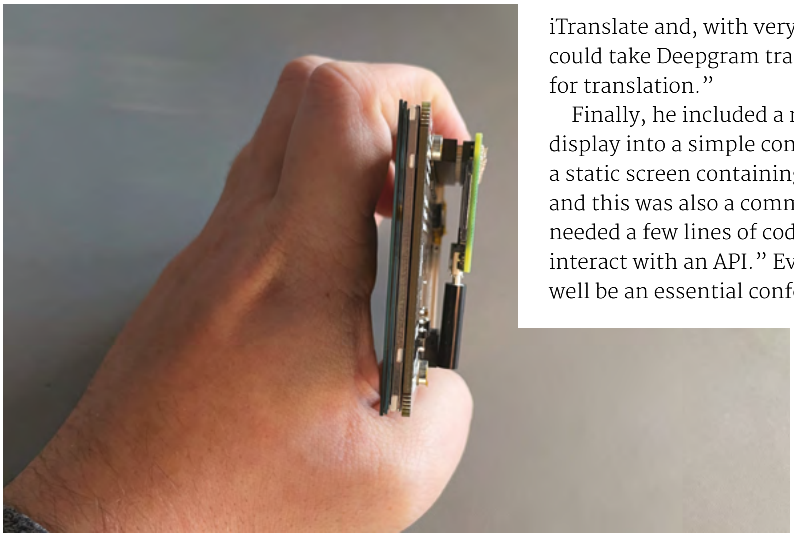


The whole device is designed so that it can fit comfortably on the body

> At its core, the whole Deepgram feature is just twelve lines of code

iTranslate and, with very minimal extra work, I could take Deepgram transcripts and pass them on for translation."

Finally, he included a mode which would turn the display into a simple conference badge. "It's just a static screen containing personal information and this was also a community suggestion. It only needed a few lines of code and it didn't need to interact with an API." Even so, this project could well be an essential conference accessory.



The project works so well because it's relatively slimline and light

CASE STUDY

XOGO digital signage

XOGO's digital signage software works across almost any computer platform and has a simple and user-friendly interface

At XOGO's inception in 2015, digital signage was an industry aimed decisively at existing premium brands, with a price tag and complexity to suit. XOGO (**xogo.io**) wanted to make digital signage easy and accessible, and saw a huge opportunity to offer improved setup and user experience, while also lowering the barrier to entry with a tiered pricing structure.

XOGO successfully launched its first bespoke media hardware – the XOGO Mini – in 2019. For its follow-up, XOGO founder and CMO Justin Miller knew price would be critical, especially as the company was aiming for growth along with the thriving digital signage market. He says that the choice of Raspberry Pi 4 for the company's own media player device was clear.

THE CHALLENGE

XOGO needed to offer their customers a 4K-capable turnkey hardware platform for its digital media player, at a very low price point: its solution must retail for under $100 (US).

THE SOLUTION

The XOGO Mini 2 is based on a Raspberry Pi 4 running a custom XOGO Player/Linux image. XOGO's custom software turns it into a media player that pushes content to a display. The Mini 2 is wireless-enabled, so the user can control and play content stored on its SD card using XOGO's Player app on any smartphone. Access is restricted by a security code; entering this lets the user send playlists securely from their app to the Mini 2. The XOGO Mini 2 is presented in XOGO's custom silent, fanless enclosure, and ships with an HDMI cable and an international power adapter to give customers a plug-and-play setup experience.

WHY RASPBERRY PI?

XOGO's CMO, Justin Miller, puts it succinctly: "No other hardware platform in this price range is as powerful or flexible.

"Raspberry Pi 4 was the perfect media player platform for us because it's 4K-capable, it allowed us to build a customised disk image running XOGO Player/Linux, and we can even offer disk images to our customers around the world so they can build their own DIY Raspberry Pi media players," he adds. "Competing solutions are more expensive, do not offer customisations such as international

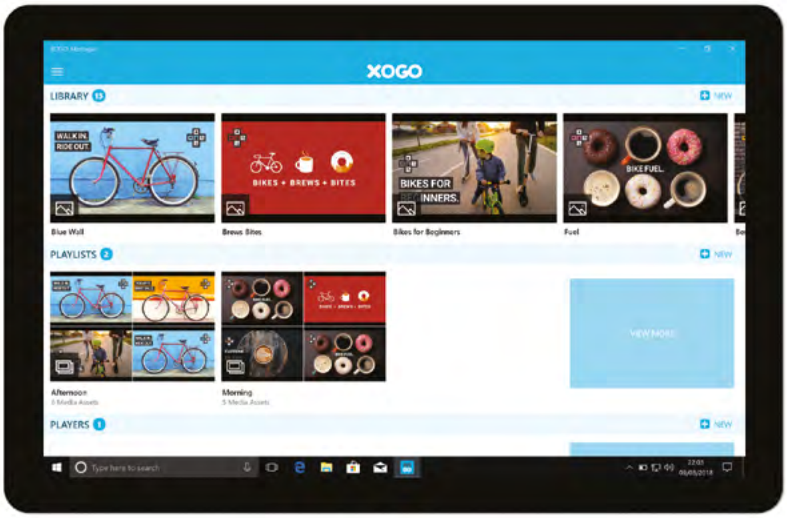
power supplies and custom cases, and are not as widely available worldwide."

XOGO was understandably cautious about moving away from the hardware that had proven successful with its first XOGO Mini. Raspberry Pi proved convincing: XOGO's experts were persuaded by its superior power and video/URL playback capabilities and its lower cost, together with what they describe as the "ultimate flexibility" it offers them to design their own player device around it, right down to the case and power supply. "We absolutely love Raspberry Pi 4."

> No other hardware near this price is as powerful

THE RESULTS

Sales of the Raspberry Pi 4-based XOGO Mini 2 are strong in both the US and the UK, and the device is proving extremely reliable in the hands of customers. XOGO believes that "Raspberry Pi is the platform for digital signage," and its software is now integrated with Sharp/NEC's Raspberry Pi Compute Module-driven displays. The two organisations are working together on sensor integration plans using Raspberry Pi hardware at the cutting edge of digital signage technology.



JOIN FOR 12 MONTHS AND GET A

FREE Raspberry Pi Zero 2 W

WITH YOUR FIRST
12-MONTH SUBSCRIPTION

Subscribe in print today and get a FREE computer!

WORTH $15

- A full Raspberry Pi desktop computer
- Learn to code and build your own projects
- Make your own retro games console, media player, magic mirror and much, much more

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

Buy now: magpi.cc/subscribe

YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
PRACTICAL PROGRAMMING
Create your own apps, games & projects!
Maker tools for all ages
Monitor your plants
Recreate classic TV Teletext
WIN! HUMANOID ROBOT UP FOR GRABS
The MagPi
NEW USER GUIDE 2022
RASPBERRY PI GAMING SPECIAL!

SUBSCRIBE on app stores

From £2.29

Available on the App Store

GET IT ON Google Play

SERIOUS FUN WITH ELECTRONICS

Discover the sheer joy of circuits, components, and physical computing. By **Lucy Hattersley**

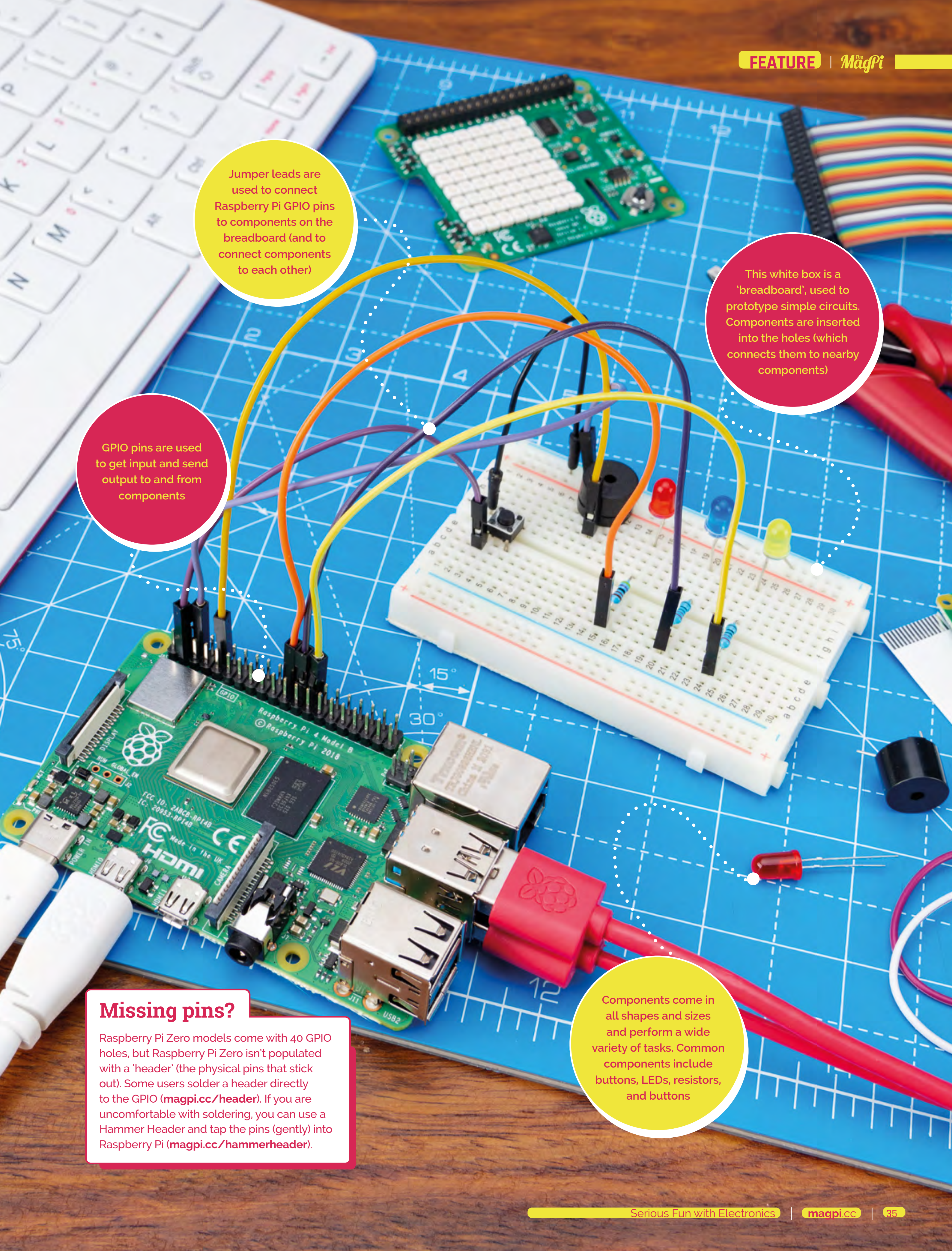
With any luck, you've managed to get a wonderful Raspberry Pi computer. So, the question now is what to make with it?

Raspberry Pi isn't like other computers. In the computers of yore, all the components were highly visible and this made computing easier to understand.

Modern devices are made from glass and glue with all the interesting innards kept away from prying hands behind security screws and 'warranty void' warnings. Good for keeping little hands out; lousy for learning.

On every Raspberry Pi sit GPIO (general purpose input/output) connections. These pins enable Raspberry Pi to connect physically with electronics. Here is the real joy of Raspberry Pi. There are thousands of components you can use with your computer, from buttons and buzzers to small screens and sensors. You can recreate just about any gadget you own and bring your own ideas to life.

This feature is for those who have a Raspberry Pi and want to have some serious fun with electronics. We'll show you how to hook up wires, connect HATs, and get started on a wonderful electrical journey.



Jumper leads are used to connect Raspberry Pi GPIO pins to components on the breadboard (and to connect components to each other)

This white box is a 'breadboard', used to prototype simple circuits. Components are inserted into the holes (which connects them to nearby components)

GPIO pins are used to get input and send output to and from components

Components come in all shapes and sizes and perform a wide variety of tasks. Common components include buttons, LEDs, resistors, and buttons

Missing pins?

Raspberry Pi Zero models come with 40 GPIO holes, but Raspberry Pi Zero isn't populated with a 'header' (the physical pins that stick out). Some users solder a header directly to the GPIO (**magpi.cc/header**). If you are uncomfortable with soldering, you can use a Hammer Header and tap the pins (gently) into Raspberry Pi (**magpi.cc/hammerheader**).

KIT AND COMPONENTS

The kit you can use to learn electronics

BREADBOARD

Experienced makers can get away with soldering components together with wires, but it's a faff and it's much faster to prototype circuits using a handy piece of plastic called a 'breadboard'.

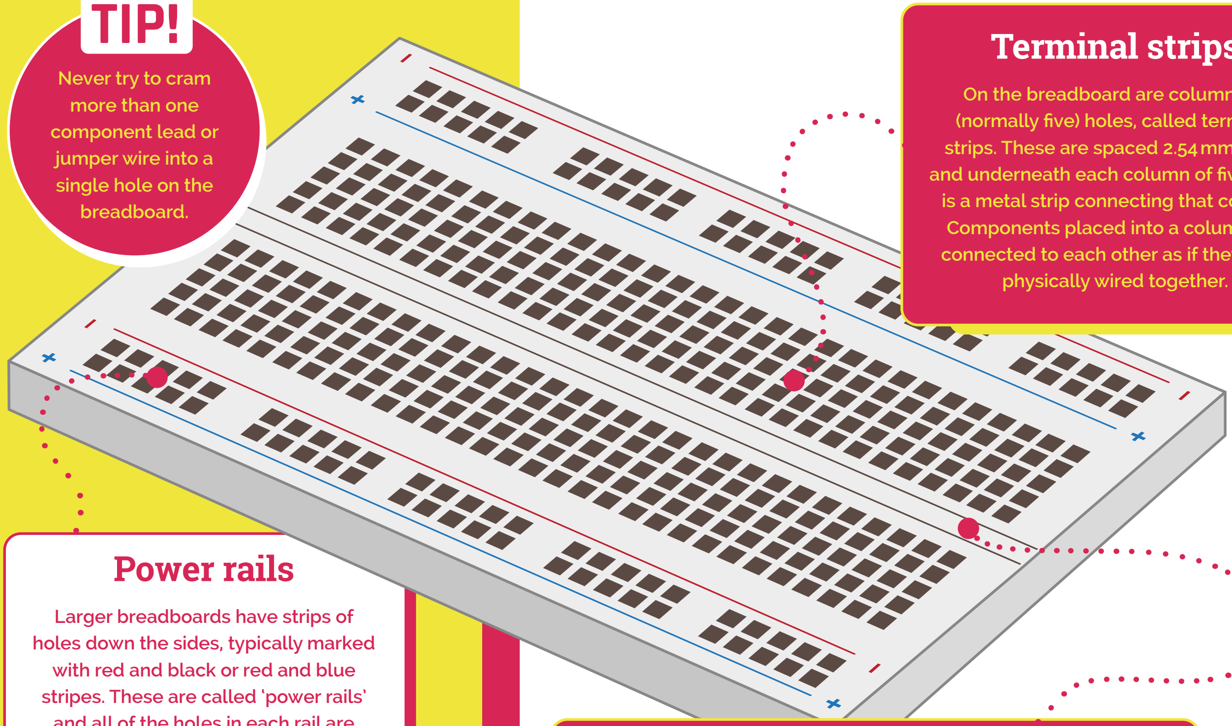
Unlike its food namesake, an electronics breadboard is a plastic slab with a bunch of holes in it. At first glance, it looks pretty unfathomable, but it soon becomes easy to understand. See our How To Use a Breadboard tutorial (**magpi.cc/breadboard**). In case you're wondering about the name, the first breadboards were wooden boards with rows of nails instead of holes.

TIP!

Remember: holes are connected in columns, aside from the split in the middle, so a component lead in A1 is electrically connected to anything you add to B1, C1, D1, and E1.

TIP!

Never try to cram more than one component lead or jumper wire into a single hole on the breadboard.



Terminal strips

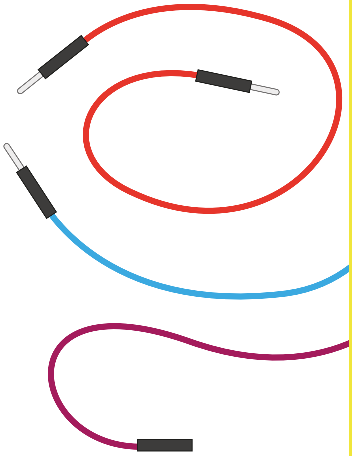
On the breadboard are columns of (normally five) holes, called terminal strips. These are spaced 2.54 mm apart and underneath each column of five holes is a metal strip connecting that column. Components placed into a column are connected to each other as if they were physically wired together.

Power rails

Larger breadboards have strips of holes down the sides, typically marked with red and black or red and blue stripes. These are called 'power rails' and all of the holes in each rail are connected. These are used to provide common ground and power for a project. By connecting one ground hole to a ground pin on Raspberry Pi, all holes in the rail will act as ground. You can do something similar using a power pin if a circuit needs 3.3 V or 5 V power.

DIP spacing

In the centre of a breadboard, between the two columns of terminal strips, is a gap. This is usually the exact size to place a DIP (dual in-line package) chip. These IC chips can straddle the central division with a row of pins falling into holes on either side. Four-pin buttons are also the right size to straddle the gap.

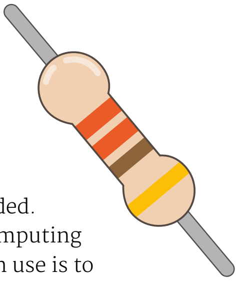


JUMPERS →

Jumper wires, also known as 'jumper leads' or 'jumper cables', connect components to each other, and Raspberry Pi's GPIO pins to the breadboard. Different types are used to connect to holes and pins.

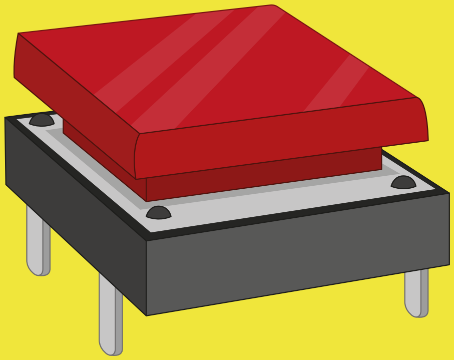
- Male-to-female (M2F), which you'll need to connect a breadboard to the GPIO pins;
- Female-to-female (F2F), which can be used to connect individual components together if you're not using a breadboard; and
- Male-to-male (M2M), which is used to make connections from one part of a breadboard to another.

You'll need all three for more complicated projects. And you can pick up a pack like from most Raspberry Pi resellers (like The Pi Hut's Jumper Bumper pack, **magpi.cc/jumperbumper**).



RESISTORS →

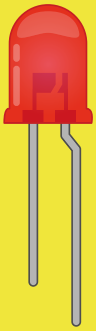
Resistors control the flow of electrical current and are available in different values measured using a unit called ohms (Ω). The more ohms, the more resistance is provided. For Raspberry Pi physical computing projects, their most common use is to protect LEDs from drawing too much current and damaging themselves or your Raspberry Pi; for this, you'll want resistors rated at around 330 Ω, though many electrical suppliers sell handy packs containing a number of different commonly used values to give you more flexibility. The more powerful the resistor, the dimmer the LED – be careful not to use one too strong or the LED light might not be visible.



BUTTONS →

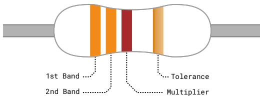
A push-button switch is used as an input device. You tell a program to watch out for it being pushed and then perform a task (button is pushed; turn on light, for example). They are commonly available with two or four legs – either type will work with Raspberry Pi.

It's much faster to prototype circuits using a handy piece of plastic called a breadboard



LEDS →

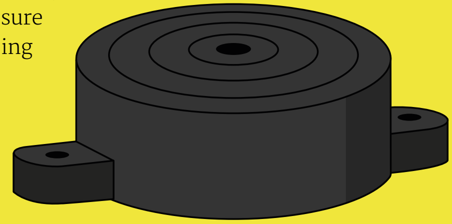
A light-emitting diode (LED) is an output device, a small light for your circuit that can be turned on and off via code. LED lights are found in many electronic gadgets, such as the light on a washing machine to let you know it's turned on. LEDs are available in a wide range of shapes, colours, and sizes, but not all are suitable for use with Raspberry Pi: avoid any which say they are designed for 5 V or 12 V power supplies. Instead pick lower voltages such as 1.2 V, 3.8 V, or the 2 V LEDs found in this pack: **magpi.cc/ledpack**. The 'diode' part of LED means it can only be used one way around, so be sure to check the direction when using LEDs in your circuit.



	1st/2nd Band	Multiplier	Tolerance
Black	0	×10^0	-
Brown	1	×10^1	±1%
Red	2	×10^2	±2%
Orange	3	×10^3	-
Yellow	4	×10^4	-
Green	5	×10^5	±0.5%
Blue	6	×10^6	±0.25%
Violet	7	×10^7	±0.1%
Grey	8	×10^8	±0.05%
White	9	×10^9	-
Gold	-	×10^{-1}	±5%
Silver	-	×10^{-2}	±10%
None	-	-	±20%

Reading resistor colour codes

Resistors come in a wide range of values, from zero-resistance versions, which are effectively just pieces of wire, to high-resistance versions the size of your leg. Very few of these resistors have their values printed on them in numbers, though: instead, they use a special code printed as coloured stripes or bands around the body of the resistor. RS Components has a great guide to reading resistors (**magpi.cc/resistorsguide**).



BUZZER ↑

As you'd expect, a buzzer produces a buzzing noise. Inside is a pair of metal plates that vibrate against each other to make the sound. There are two types of buzzers: active and passive. Be sure to get an active buzzer, such as this one from Pi Hut, **magpi.cc/buzzer5v**, as these are the simplest to use.

UNDERSTANDING WIRING

Start putting together your components and learn to code

Electronic circuits are daunting for complete beginners. Fortunately, help is widely available, and it soon becomes a lot simpler than you imagine.

Once you have the breadboard and components, you should start wiring them up by following a simple tutorial (like the one at the end of this feature).

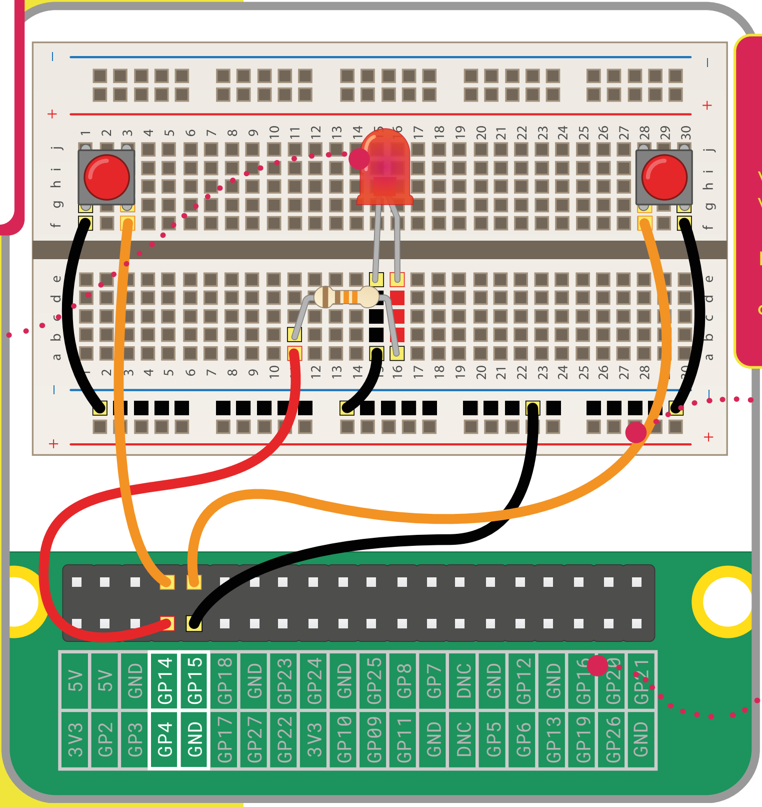
There's absolutely no shortage of electronic projects out there, and most of them guide you through attaching components to your breadboard, then using software such as the GPIO Zero Python library (**magpi.cc/gpiozero**).

There are a range of easy-to-follow electronics tutorials on the Raspberry Pi Foundation's website (**magpi.cc/electroniccomponents**).

Most tutorials include a wiring diagram. See the wiring diagram below. This provides a visual guide to how circuit is built. You'll find wiring diagrams like this throughout *The MagPi*, and similar ones used by the Raspberry Pi Foundation and other resources. If you're interested in creating your own, you use a program called Fritzing to make them (**fritzing.org**).

Components

The components are visually similar to real-life counterparts. Some, like the LED, visually demonstrate which way around they should go, so pay attention to the position of legs on components.



1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
a b c d e f g h i j
5V 5V GND GP14 GP15 GP18 GND GP23 GP24 GND GP25 GP8 GP7 DNC GND GP12 GND GP16 GP20 GP21
3V3 GP2 GP3 GP4 GND GP17 GP27 GP22 3V3 GP10 GP09 GP11 GND DNC GP5 GP6 GP13 GP19 GP26 GND

Wires

Jumper leads are represented by these coloured wires. The wires work the same, no matter which colour you use, but you'll often see red used for the power, and black connected to ground. The other colours are often used to represent various GPIO pin connections.

Pinout

Sometimes the diagram will have GPIO pin labels on the board, like this one. But these aren't printed on Raspberry Pi and many diagrams include a pinout separately. If you ever need a guide to the pins, you can find one here: **magpi.cc/pinout**.

THE OFFICIAL Raspberry Pi Beginner's Guide

How to use your new computer

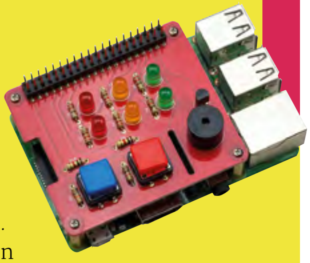
Simple Electronics with GPIO Zero

For more information on using GPIO Zero and learning to wire up circuits from wiring diagrams, take a look at our *Raspberry Pi Beginner's Guide*. **magpi.cc/BGbook**

PUT A HAT ON IT

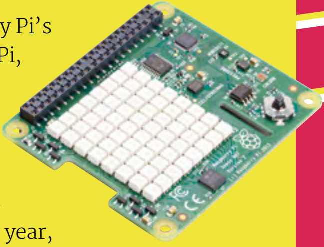
Take the hassle out of circuit building with HATs

HATs (Hardware Attached on Top) are pre-built circuit boards with components and parts. They are designed to connect to the 40-pin GPIO header on Raspberry Pi, and are easy to set up and get started. Each HAT has an EEPROM on board with the software installation needed for the HAT to work, so all you need to do is plug it in and start using the electronics on board. Here are some fun HATs to try out. You'll also see pHATs, which are designed for Raspberry Pi Zero.



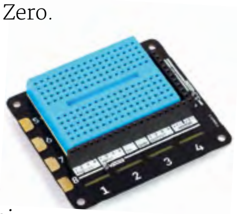
JAM HAT →

Designed to make starter electronics easier in the classroom, JAM HAT is packed with LEDs, buttons, and a buzzer. These components can be used with a range of beginner projects, like building a traffic light system with a button crossing. Students can focus on the code without spending the whole class wiring up the components. Of course, that's half the fun for us!
magpi.cc/jamhat



SENSEHAT →

Designed for Raspberry Pi's space program, Astro Pi, Sense HAT allows Raspberry Pi to sense the world around it. Two have been on-board the ISS since 2017. And, every year, students around the world use them to perform experiments in space. Down to earth, the Sense HAT has orientation, pressure, humidity, and temperature sensors, along with an LED Matrix and joystick control. Best of all, it is backed by hundreds of documented experiments and our Sense Hat Experiments book (**magpi.cc/sensehatbook**), with projects such as a Gravity Simulator, Magic 8 Ball, Pixel Pet, and Data Logger.
magpi.cc/sensehat



EXPLORER HAT PRO →

Put a breadboard on top of your Raspberry Pi with Pimoroni's Explorer HAT Pro (£20/$27). Alongside the mini breadboard are a range of inputs and outputs, including capacitive touchpads (that can be used as buttons), coloured LEDs, analogue inputs, and motor drivers. What's great about the Explorer HAT is that it's been around for years and is backed up with a custom Python code library and a bunch of examples on Pimoroni's learn page (**learn.pimoroni.com**).
magpi.cc/explorerhatpro

There's absolutely no shortage of electronic projects out there

ZEROSEG →

One of the more popular HAT projects combines two four-digit LED unit displays with two buttons. The result is a small HAT that can be used to create scrolling tickers, such as news displays, stock market results, and short moving messages.
magpi.cc/zeroseg

INKY WHAT ↑

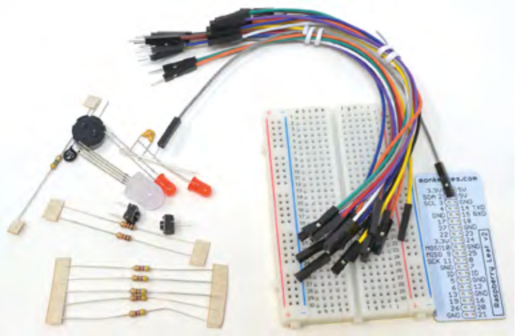
Electronic ink displays, of the kind found in e-readers, are a great attachment to Raspberry Pi. Pimronoi's Inky wHAT (£52/$70) is a cut above with a large 400 × 300 three-colour screen (with a choice of red or yellow accent alongside the black and white). Take a look at our starter tutorial (**magpi.cc/inkyhello**) on using Inky wHAT.
magpi.cc/inkywhat

WEATHER HAT ↓

This is a neat solution for attaching climate and environmental sensors to Raspberry Pi. Wind/rain sensors are attached to the RJ11 connectors. Information is displayed on the 1.54 in LCD screen. You can get a whole kit, including HAT and a wind vane, anemometer (wind speed), and rain gauge. Pimoroni has a guide for getting started (**magpi.cc/weatherhatstarter**).
magpi.cc/weatherhat

GET STARTED

While you can buy components separately, it is often easier to pick up a starter kit



MONK MAKES →

Simon Monk has been putting together starter kits and tutorial guides since 2013, and his Raspberry Pi collection is one of the finest. You'll find a tutorial by Simon overleaf based upon his Project Box for Raspberry Pi (£12/$9). Inside you'll find a breadboard, jumper wires, LEDs, resistors, push-buttons, buzzer, and both a thermistor and phototransistor. There's also a booklet with ten products. We also like his 'Leaf', a plastic guide to GPIO pins that fits over Raspberry Pi and makes it easier to identify which wire goes where. **monkmakes.com**

The CamJam EduKit is a stalwart classic of Raspberry Pi electronics

RASPBERRY PI 4 ULTIMATE KIT ↓

If you are starting from scratch, and want everything you need to be delivered in one place, then CanaKit's Ultimate Kit (£95/$129) is a great option. Inside the box are a Raspberry Pi 4 Model B, power supply, case, microSD card, cables, and a breadboard with electronics parts. **magpi.cc/canakitultimate**

Affordable electronic projects for your Raspberry Pi
CamJam EduKit
PiHut
camjam.me/edukit
Kit Contents:
CamJam EduKit
Download the EduKit worksheets from camjam.me/edukit
PiHut

CAMJAM EDUKIT ↑

The CamJam EduKit (£5/$9) is a stalwart classic of Raspberry Pi electronics, and many a maker cut their teeth here. Inside the tin is a breadboard, resistors, LEDs, button, buzzer, and jumper leads. Good parts, and good value. **magpi.cc/edukit**

Get started with electronics with Raspberry Pi

Raspberry Pi is a great way to start learning about electronics. Here's how to take your first steps in this fascinating subject

WRITER

Simon Monk

Simon divides his time between writing and designing products for MonkMakes Ltd. His books include *Programming Raspberry Pi* (TAB) and *The Raspberry Pi Cookbook* (O'Reilly). He has sold over 700,000 books in ten different languages.

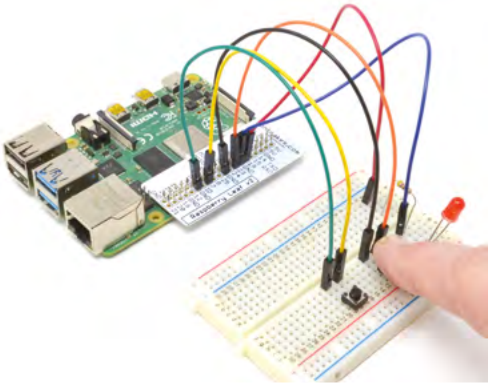
@simonmonk2

You'll Need

- Raspberry Pi with GPIO header
- Breadboard
- Male-to-female jumper leads, LED, button **magpi.cc/projectbox1**

One of the great things about Raspberry Pi, is the inclusion of the double row of pins called the GPIO connector. This enables you to connect external electronics to your Raspberry Pi and use code to control things like LEDs and buzzers, as well as reading values from sensors or detecting when switches are pressed.

In this tutorial, you will use the popular MonkMakes Project Box 1 for Raspberry Pi (**monkmakes.com**) to get started with electronics. This kit contains the LED, jumper leads, breadboard, and buttons we will use to explore electronics. You can pick up these components separately, and other kits are available.



▲ The Leaf template makes it easier to identify GPIO pins

No soldering will be required, as you will be using a solderless breadboard to make your electronic circuits, and then connect them to your Raspberry Pi using jumper wires.

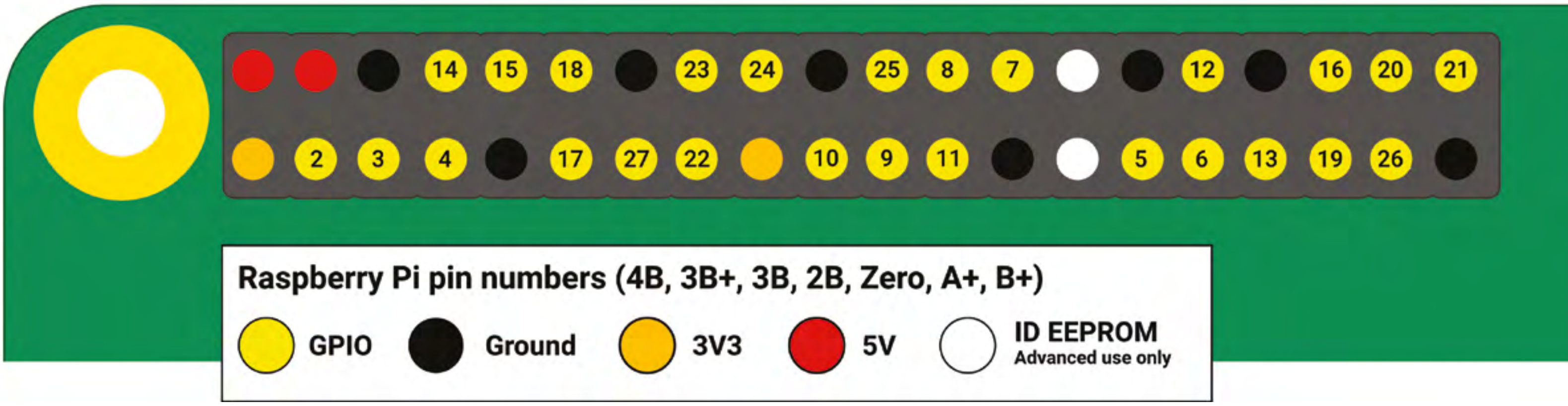
01 LED and resistor on breadboard

Place an LED, resistor, and switch onto the breadboard as shown in **Figure 1** (overleaf). You can identify which resistor to use by its stripes (RS Components has a good guide to resistors, **magpi.cc/resistors**). Use one of the 470 Ω resistors that have yellow, purple, and brown stripes.

One of the leads of the LED is slightly longer than the other. This is the positive lead and should be on row 3 of the breadboard, as shown. It does not matter which way around the resistor or switch go, but make sure that the switch pins run top to bottom.

02 Fit the Leaf

Place the Raspberry Leaf GPIO template from the kit over the GPIO header pins, so that the text on the Leaf that says 'Raspberry Leaf' is to the outside of Raspberry Pi. This allows you to easily identify which GPIO pin is which.



▲ Use this pinout guide to identify the GPIO pins. The yellow circles pins are used in code using the corresponding BCM number, black pins are for ground, red pins provide constant 5 V power, while the orange pin provides 3 V power. The white pins are reserved for add-on hardware

03 Connecting to Raspberry Pi

Use two female-to-male jumper leads (**magpi.cc/mfjumpers**) to make the connections from your Raspberry Pi to the breadboard. It does not matter what colour leads you use, but a convention is to use red for plus volts and black or blue for 0 V (GND/ground).

Connect from 5 V on Raspberry Pi to row 2 on the left side of the breadboard; from GND on Raspberry Pi's pins to row 5 on the right side of the breadboard, which will connect to the negative side of the LED.

There are eight pins on the Raspberry Pi's GPIO connector labelled GND and it does not matter which you use. Similarly, there are two 5 V pins and either can be used.

04 Try it!

This simple circuit only uses Raspberry Pi to provide power to the circuit. The 5 V supply from Raspberry Pi connects to one end of the push switch. If the switch is pressed, then electrical current can continue flowing and will flow first through the resistor and then through the LED before returning back to Raspberry Pi's GND connection. When the button is pressed and the current flows, the LED will light.

The resistor has the job of restricting the flow of current, as otherwise too much current would flow through the LED and it would burn out.

05 Two switches

As it stands, we are not getting anything from Raspberry Pi that we couldn't from a battery. So, let's include Raspberry Pi in the action so that it can monitor two switches and when they are pressed, use them to alter the brightness of the LED.

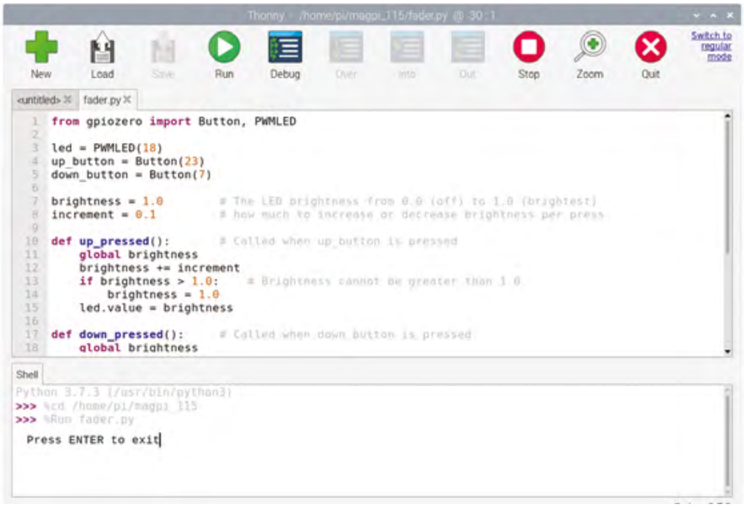
Pull all the jumper leads out from Raspberry Pi and breadboard, being careful that none of the metal ends of the jumper wires touch.

Remake the breadboard so that there are now two switches as well as the LED and resistor connected to the breadboard. This will involve moving the first switch and adding a new one.

06 Raspberry Pi in control

Use six female-to-male jumper wires, as shown in **Figure 2**, to connect the LED (and its resistor) and switches. Notice that now the LED and two switches are each connected separately to Raspberry Pi GPIO pins and GND pins. The LED is connected to GPIO 18, which will act as an output that we can use to control the LED's brightness.

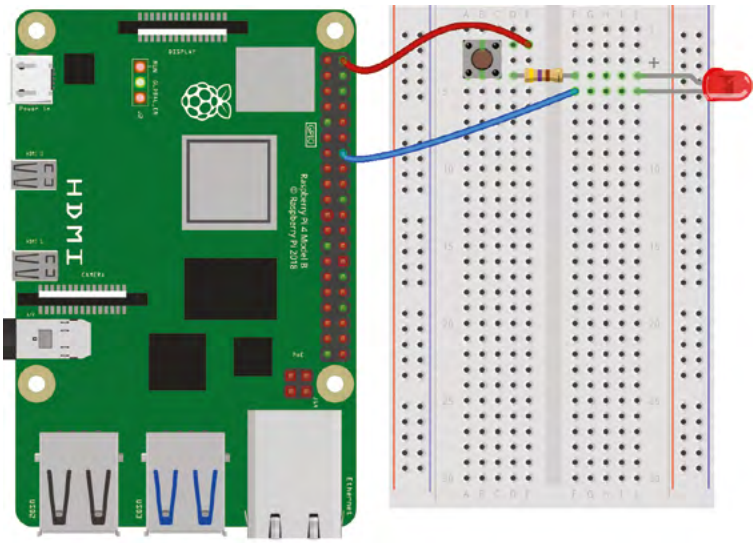
▼ You can edit the program and run it using Thonny



```
from gpiozero import Button, PWMLED

led = PWMLED(18)
up_button = Button(23)
down_button = Button(7)

brightness = 1.0        # The LED brightness from 0.0 (off) to 1.0 (brightest)
increment = 0.1         # how much to increase or decrease brightness per press

def up_pressed():       # Called when up button is pressed
    global brightness
    brightness += increment
    if brightness > 1.0:    # Brightness cannot be greater than 1.0
        brightness = 1.0
    led.value = brightness

def down_pressed():     # Called when down button is pressed
    global brightness
```

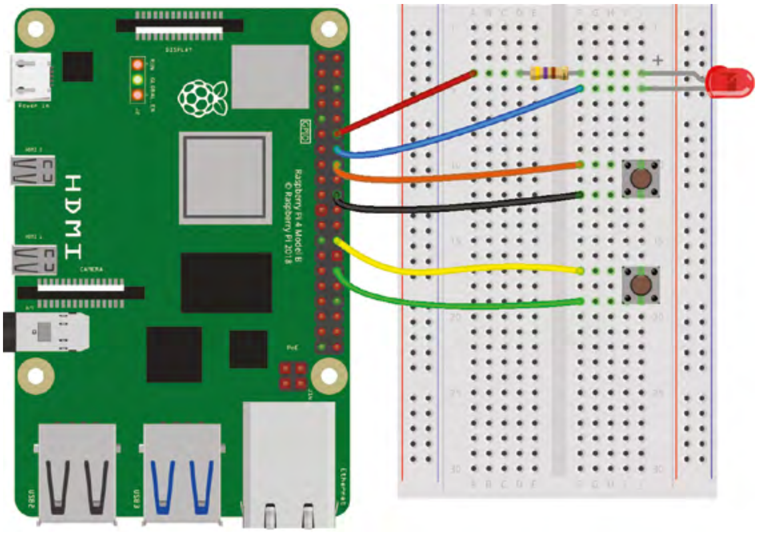
```
Python 3.7.3 (/usr/bin/python3)
>>> %cd /home/pi/magpi_115
>>> %Run fader.py

 Press ENTER to exit
```

Figure 1



▲ **Figure 1** Here we use Raspberry Pi like a battery, and supply 5 V to provide power to an LED (via a resistor to protect it)

Figure 2



▲ **Figure 2** Wiring the LED fader project. Make sure you get the LED the right way round and the wires in the right place

The two switches are connected to GPIO pins 23 and 7. These GPIO pins will be acting as inputs, so that our code can detect when the switches are pressed and carry out some action.

07 Get the code!

As it stands, nothing will happen when we press the buttons because we need a program to be running on our Raspberry Pi that can monitor the buttons and do things when they are pressed. To fetch that code from GitHub, first open a Terminal window. This can be found in the Accessories section of Raspberry Pi OS's start menu.
Make sure you have a connection to the internet and run the command:

```
 git clone https://github.com/simonmonk/
magpi_115.git
```

This will fetch the program into a directory called **magpi_115**. You will also find the **fader.py** code listing in this article.

08 Run the code

To run the code, first change directory and then run the program by typing the commands below into the Terminal:

```
cd /home/pi/magpi_115
python3 fader.py
```

If everything is OK, you should see the message 'Press ENTER to exit'. If you get error messages, go back to Step 7 and make sure the code downloaded OK.

09 Try it out!

When the program first runs, the LED should be at maximum brightness. Press the lower

fader.py

Language: **Python**

DOWNLOAD THE FULL CODE:

magpi.cc/faderpy

```python
from gpiozero import Button, PWMLED

led = PWMLED(18)        # variable brightness LED using pin 18
up_button = Button(23)
down_button = Button(7)

brightness = 1.0        # The LED brightness from 0.0 (off) to 1.0 (brightest)
increment = 0.1         # how much to increase or decrease brightness per press

def up_pressed():           # Called when up_button is pressed
    global brightness
    brightness += increment
    if brightness > 1.0:    # Brightness cannot be greater than 1.0
        brightness = 1.0
    led.value = brightness

def down_pressed():     # Called when down_button is pressed
    global brightness
    brightness -= increment
    if brightness < 0.0:    # Brightness cannot be less than 0.0
        brightness = 0.0
    led.value = brightness

up_button.when_pressed = up_pressed # link up_pressed to up_button
down_button.when_pressed = down_pressed

led.value = brightness          # so that the LED is lit even if buttons not pressed

input("Press ENTER to exit")    # avoid the program finishing as soon as it starts

```

of the two buttons on the breadboard and you should see the LED start to dim. Press it some more times and the LED will go off altogether. Pressing the other button will increase the brightness.

When you are ready to quit the program, just press **ENTER** on your keyboard.

10 Looking at the code

If you want to inspect or change the code for this project, run Thonny from Raspberry Pi OS's menu (in the Programming section) and click the Load button, navigate to **fader.py**, and open it.

You can also run the program from Thonny by clicking on the Run button.

Try changing the value of the variable `increment` from 0.1 to 0.2. You will notice that the brightness changes in larger increments.

11 What next?

We have used a few components from Project Box 1 for Raspberry Pi. This kit contains the parts and instructions for lots of other interesting projects to get you started with coding and electronics.

Top Tips

Unplug Raspberry Pi

It's easy to accidentally connect wires that shouldn't be connected and this could damage your Raspberry Pi (although it's unlikely). So, it's a good idea to unplug your Raspberry Pi when moving components about.

LED polarity

If the LED doesn't light when it should, then check that it is the right way around.

FROM THE MAKERS OF The MagPi THE OFFICIAL RASPBERRY PI MAGAZINE
NEW 2022 UPDATE
RETRO GAMING WITH RASPBERRY PI
2ND EDITION
164 PAGES OF VIDEO GAME PROJECTS
+ BUILD AN ARCADE MACHINE
RETRO GAMING WITH RASPBERRY PI 2ND EDITION
PLAY & CODE GAMES!

Experimental Pi
PiBoy DMG
SELECT
START
Z
Y
X
C
B
A

Make an AI LEGO® face

Your robot face will be able to recognise and react to a range of objects

MAKER

James Robinson

Senior Learning Manager, Pedagogy and Training at Raspberry Pi. Ex-Teacher, spaceenthusiast.

raspberrypi.org

MAKER

Mark Scott

Marc is a former computer science teacher, and manages informal learning content for the Raspberry Pi Foundation.

raspberrypi.org

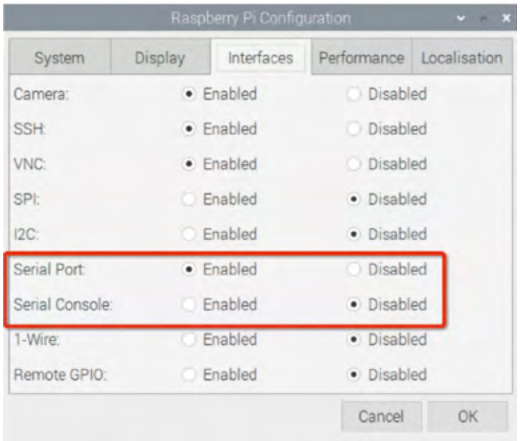
Over the course of this tutorial, you will construct a robot face from a combination of LEGO and electronic components. You'll then use an existing machine learning model to help your face recognise different objects and react to them.

You should build your robot face using whatever LEGO and other materials that you have available. We have used the LEGO SPIKE™ Prime kit (**magpi.cc/spikeprime**). This tutorial can also found online on the Raspberry Pi Foundation's website (**magpi.cc/legorobotface**).

01 Get started

Before you begin, you will need to have set up your Raspberry Pi computer and attached your Build HAT. Follow the instructions for Make a Build HAT Game Controller in *The MagPi* magazine, issue #113 (**magpi.cc/buildhatgamecontroller**).

As you are connecting the Build HAT to Raspberry Pi, don't forget to attach the Camera cable and feed the cable through the slots on both your Raspberry Pi and the Build HAT.



Camera:	● Enabled	○ Disabled
SSH:	● Enabled	○ Disabled
VNC:	● Enabled	○ Disabled
SPI:	○ Enabled	● Disabled
I2C:	○ Enabled	● Disabled
Serial Port:	● Enabled	○ Disabled
Serial Console:	○ Enabled	● Disabled
1-Wire:	○ Enabled	● Disabled
Remote GPIO:	○ Enabled	● Disabled

Figure 1 The Raspberry Pi OS Configuration settings used in this project

Once your Raspberry Pi has booted with the Camera Module attached, open Raspberry Pi's Configuration tool by clicking on the menu button and then selecting 'Preferences' and then 'Raspberry Pi Configuration'.

Click on the 'Interfaces' tab and set Camera, SSH, VNC, Serial Port to Enabled, and Serial Console to Disabled (as shown in **Figure 1** – Configuration Settings).

You can find instructions on how to set up the camera guide here (**magpi.cc/getstartedcam**). If using Raspberry Pi OS 'bullseye' enable Legacy Camera mode (**magpi.cc/legacy**).

02 Test the machine learning model

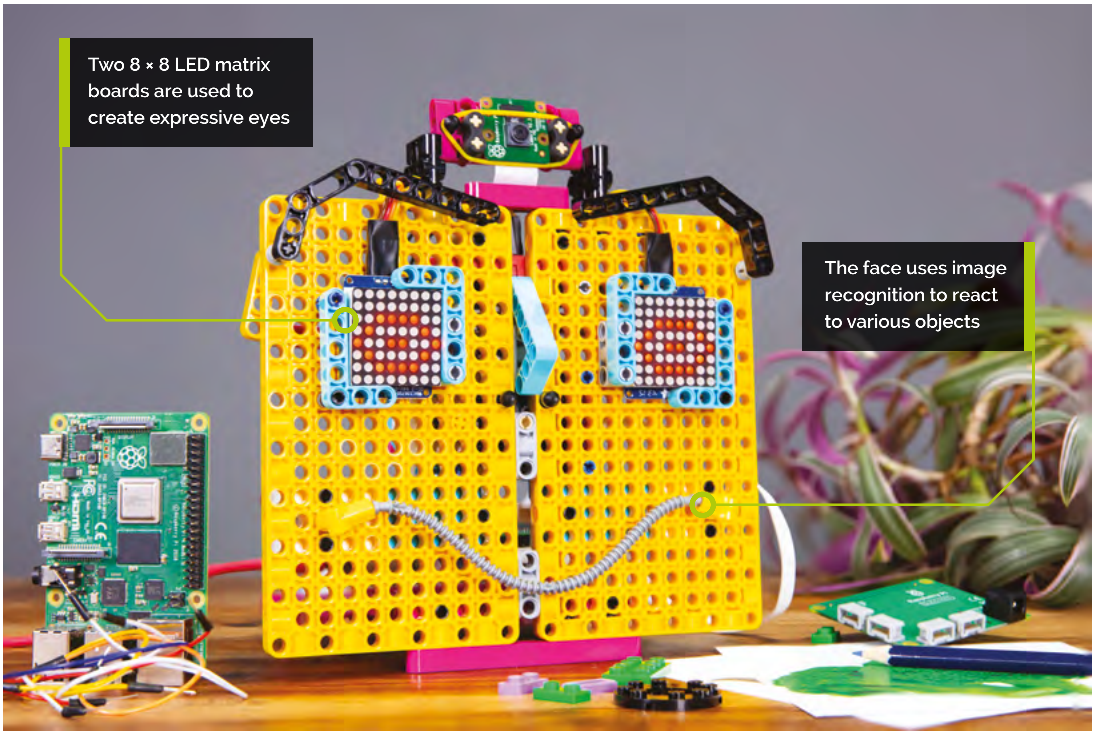
Your first step is to understand and test how you can use a machine learning model to recognise objects. For this project, you won't be creating and training your own model, but using an example model that can recognise a range of objects.

Computers do not have a natural ability to learn. Most things that computers do have been directly programmed by a human. This makes them great for tasks that have a few clearly defined rules, but they struggle with more human-like tasks such as recognising different objects. Using machine learning, a computer can be shown thousands and thousands of images, each of which has been labelled. Gradually, the program can learn the characteristics of a group of images and then give them the correct label. The end result of this process is called a model. Once trained, models can be used in the real world to perform tasks.

To get started, download the resources for this project to your Raspberry Pi from GitHub (**magpi.cc/legorobotfaceresources**).

You'll find a range of files that will be useful for the project, but for this step, you're going to use:

- **model.tflite** – The machine learning model file



Two 8 × 8 LED matrix boards are used to create expressive eyes

The face uses image recognition to react to various objects

- **labels.txt** – Labels for each object the model can recognise
- **classifer.py** – A Python program to test the model

Open Thonny, which is under the Programming category in your Raspberry Pi applications menu.

Open and Run the **classifier.py** program. Your Raspberry Pi will display what the camera is 'seeing' and the name of the main object in view that it recognises. Try presenting the camera with different objects, and investigate which ones it can recognise with confidence.

Find at least four objects (or images) that your camera can recognise reliably – you'll need them for your machine learning model.

> Once trained, models can be used in the real world to perform tasks

03 Use emoji for your robot face

The goal is to build a robotic face that can respond to objects that it recognises. If you break that down into smaller steps, you might say that your robot face will:

1. Use the Raspberry Pi camera to look for objects
2. If an object is detected, use that object to change the face
3. Match the object detected to a reaction or emotion
4. Change the look of the face to represent a reaction
5. Return to step 1 to look for the next object to react to

For the project to work, it's going to need a selection of reactions that it can display using simple facial expressions. Emojis are a great example of this.

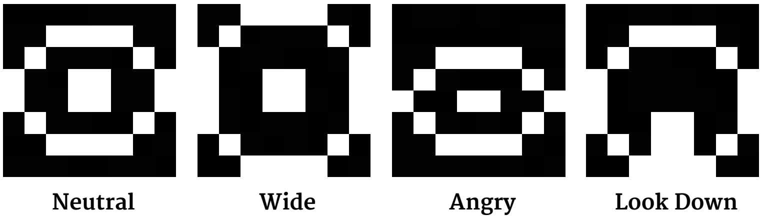
An emoji is an example of an abstraction, a simplified representation of a real face. All of the complexity has been removed and limited to the simple key parts of the face. In this project, you can use four emojis to represent eyes: Neutral, Wide, Angry, Look Down (see **Figure 2**, overleaf).

04 Connect objects to the expressions

From your experiments in the previous step, you will have identified at least four objects that your camera and machine learning model can reliably detect.

You'll Need

- A Raspberry Pi computer
- A Raspberry Pi Build HAT **magpi.cc/buildhat**
- A Raspberry Pi Camera Module
- 2× Small LEGO Technic motors
- 1 × Large LEGO Technic motor
- 2 × Adafruit 8×8 LED matrices **magpi.cc/88ledmatrix**
- Assortment of LEGO (we used a selection from the LEGO SPIKE Prime kit)
- Jumper leads, mini breadboard, and soldering kit



▲ **Figure 2** The four emojis used to represent eyes

Top Tip

Install Build HAT

Open a Terminal window on your Raspberry Pi by pressing **CTRL+ALT+T**. At the prompt, type:

```
sudo pip3 install buildhat
```

Press **ENTER** and wait for the 'installation completed' message.

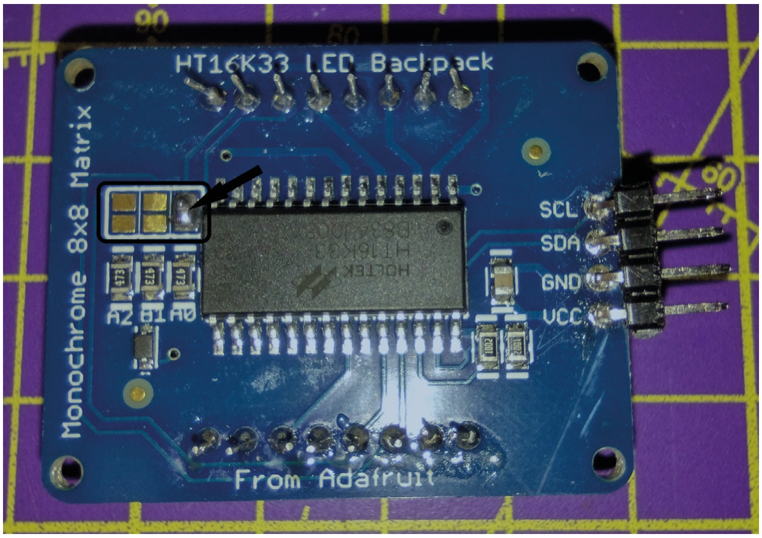
Choose which objects will trigger which reactions in your robot. Each expression should have a reaction associated with it. For our example, we used the following:

Object	Reaction
Broccoli	Neutral
Teapot	Wide
Snake	Angry
Hotdog	Look Down

05 Making a face

Now that you know which objects will trigger which responses, you need to build a robotic face that can be animated to show the different expressions.

To build this robot face, we used parts from the SPIKE Prime and SPIKE Expansion sets. However, most of the parts can be found in many existing LEGO sets. To construct this model, or inspire your own, download Raspberry Pi's handy build guide (**magpi.cc/robotfacebuildguide**). This Sketchfab model is what your face will look like (**magpi.cc/sketchfabface**).



▼ **Figure 3** Soldering the Adafruit LED Backpacks

06 Add the eyes

The LED matrices used in the example face are connected to Raspberry Pi's I2C interface. Devices using I2C are connected using a specific number called an address. As you are using two matrices, each will need its own address.

Before you connect them up, you need to follow the relevant assembly instructions (**magpi.cc/matrixassembly**). Assembly of the LED arrays requires some soldering, so get permission from an adult before you use any tools. You can follow our soldering guide at **magpi.cc/soldering**.

The matrices used in this project all come with the same address, meaning that for two to work together, one of them needs a new address. For this, some more soldering is needed.

Using your soldering kit, close the A0 connection of only one of your matrices (see **Figure 3**).

Place the eyes into the square sockets on your robot face; use elastic bands to secure them and make sure the pins are at the top (see **Figure 4**).

Now that the basic construction of the robot face is complete, you need to add your Raspberry Pi and connect your components to it.

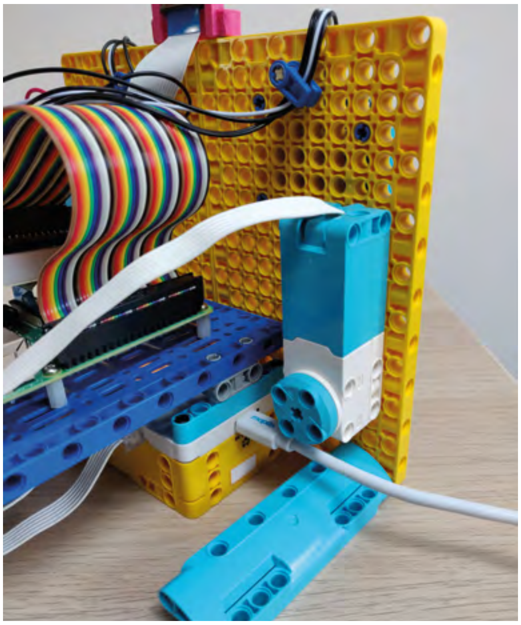
> “Most of the parts can be found in many existing LEGO sets”

07 Add Raspberry Pi

For this project you'll ideally want to use the Maker Plate element to mount your Raspberry Pi and Build HAT. To see a visual guide with photos of this step take a look at the build instructions online (**magpi.cc/robotface5**).

Mount your Raspberry Pi onto the Maker Plate using M2 bolts and nuts, making sure Raspberry Pi is on the flat side.

Before adding the Build HAT, you'll first need to attach the camera ribbon cable to the Raspberry Pi and thread it through the hole in the Build HAT. If you haven't already connected the camera board to your Raspberry Pi, you can do so by following the instructions at 'Getting started with the Camera



LEGO Technic motors are used to adjust the eyes and mouth

Module' (**magpi.cc/getstartedcam**). Leave the camera ribbon connected to the Raspberry Pi, but remove the camera board from the loose end of the ribbon by pushing up the small black clip and sliding the ribbon out. Poke the ribbon through the underside of the Build HAT and out through the top, making sure the ribbon isn't twisted.

Connect the Maker Plate to the back of your robot face using some black studs. Next, connect up your small LEGO Technic motors to ports A and B, ready to control the mouth.

Connect up your large LEGO Technic motor to port C, ready to control the eyebrows.

To connect the pair of eyes to Raspberry Pi's GPIO, they first need to be connected together using a breadboard, and then to the GPIO pins from the breadboard.

Use eight male-female jumper wires to connect the four pins from each eye together on the breadboard. Make sure that both VCC pins are in the same row of the breadboard, both GND pins are in the same row, and so on. Then connect to the 3V3, GND, SDA, and SCL pins on the Raspberry Pi, as shown in **Figure 5**.

Your robot face is now built, connected, and ready to be programmed!

08 Motorise the mouth

Open Thonny from the Programming menu and save a new file called **robot_face.py**, in the same directory as **classifier.py**, **labels.txt**, and the 8×8 pixel art images. Now enter the Python code from the **robot_face.py** listing.

robot_face.py

Language: **Python 3**

DOWNLOAD THE FULL CODE:

magpi.cc/robotfacepy

```
from buildhat import Motor
import board
from adafruit_ht16k33.matrix import Matrix8x8
from PIL import Image
from classifier import Classifier
from time import sleep

## Set up the motors
mouth_r = Motor('A')
mouth_l = Motor('B')
eyebrows = Motor('C')

## Move the motors to 0 position
mouth_r.run_to_position(0)
mouth_l.run_to_position(0)
eyebrows.run_to_position(0)

## Set up the eyes
i2c = board.I2C()
left_eye = Matrix8x8(i2c, address=0x70)
right_eye = Matrix8x8(i2c, address=0x71)

## Link names of expressions to images in the Resources
directory for the eyes to display
neutral = Image.open("neutral.png").rotate(90)
wide = Image.open("wide.png").rotate(90)
angry = Image.open("angry.png").rotate(90)
look_down = Image.open("look_down.png").rotate(90)

## Link names of expressions to motor movement and to eye
display in a dictionary
faces = {
    "neutral":{"mouth":0, "right_eye":neutral,
"left_eye":neutral, "eyebrows":0},
    "happy":{"mouth":45, "right_eye":wide,
"left_eye":wide, "eyebrows":-150},
    "angry":{"mouth":-20, "right_eye":angry,
"left_eye":angry, "eyebrows":150},
    "sad":{"mouth":-45, "right_eye":look_down,
"left_eye":look_down, "eyebrows":-40}
    }

## Use the classifier.py to recognise different images
(file is in resources directory)
seen_items = Classifier(label_file="labels.txt",
model_file="model.tflite",threshold=0.5)

```

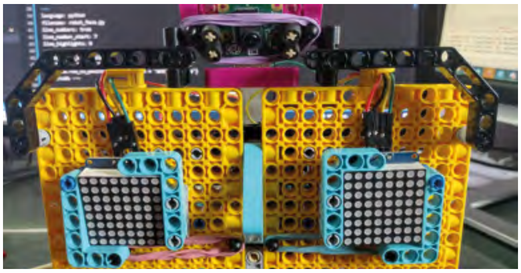


Figure 4 Rubber bands are used to hold the eyes in place

It starts with the import you will need to control the LEGO Technic motors (**robot_face.py** line 1).

Create two new objects for the left and right motors. In this example, the right motor is connected to port A and the left to port B (line 9 and 10).

Both motors should start in the 0 position when the program starts.The `move_mouth` function (line 43) will move the mouth motors. They need to turn in opposite directions, so the left motor will turn to a negative value and the right motor to a positive value. Adding `blocking=False` will make both motors turn at the same time.

> “The third motor is used to move the face's eyebrows”

09 Program the eyebrows

The third motor is used to move the face's eyebrows. The `eyebrows.run_to_position(0)` object on line 16 is used for the eyebrows' motor.

Make sure that your large motor is positioned so that the lollipop and the circle are aligned, and that your face's eyebrows are set horizontally. If they are not, you may need to adjust your build a little.

You set the motor to turn to the 0 position when the program starts.

There are three eyebrow positions shown here, but you can create more.

- 0 will make the eyebrows appear horizontal
- 150 will lower the eyebrows
- −150 will raise the eyebrows

The `move_eyebrows` function (line 49) gets the current eyebrow position, and if the position it is supposed to move to is less than the current one, it will move anticlockwise; otherwise it will move clockwise.

robot_face.py (cont.)

Language: **Python 3**

```
040. ## Set reactions for different objects that are
     recognised
041. reactions = {"broccoli":"neutral", "teapot":"happy",
     "Indian cobra":"angry", "hotdog":"happy"}
042.
043. def move_mouth (position):
044.     '''Move the mouth to value of position parameter'''
045.     mouth_l.run_to_position(position * -1,
     blocking=False)
046.     mouth_r.run_to_position(position, blocking=False)
047.
048.
049. def move_eyebrows (position):
050.     '''Move the eyebrows to value of position
     parameter'''
051.     current_position = eyebrows.get_aposition()
052.     if position < current_position:
053.         rotation = 'anticlockwise'
054.     else:
055.         rotation = 'clockwise'
056.     eyebrows.run_to_position(
     position, direction = rotation)
057.
058. def change_eyes(left, right):
059.     '''display the PIL objects on the left and right
     eye'''
060.     left_eye.image(left)
061.     right_eye.image(right)
062.
063. def set_face (face):
064.     '''call all functions that change the expression,
     according to the face from the faces dictionary'''
065.     change_eyes(face["right_eye"],face["left_eye"])
066.     move_mouth(face["mouth"])
067.     move_eyebrows(face["eyebrows"])
068.
069. ## Loop forever and check the list of seen items and set
     the correct face if the object has been seen
070. while True:
071.     sleep(1)
072.     if seen_items.item != seen_items.last_item:
073.         item = seen_items.item
074.         if item in reactions.keys():
075.             set_face(faces[reactions[item]])
076.     sleep(1)
```

Top Tip

Raspberry Picamera

The Python Picamera module is currently not, by default, compatible with the latest version of Raspberry Pi OS (called Bullseye). To use the Picamera module, you will need to enable legacy support for the camera Open a Terminal window and type the following command:

```
sudo raspi-config
```

Set Legacy Camera to Enable under Interface Options. See **magpi.cc/legacy** for more information.

10 Program the eyes

The LED matrices can show 8×8 pixel images on their displays. These can be used to display different motions of the eyes.

The three libraries on lines 2, 3, and 4 enable you to display images on the LED displays.

```
import board
from adafruit_ht16k33.matrix import Matrix8x8
from PIL import Image
```

The `i2c`, `left_eye`, and `right_eye` objects (lines 19, 20, and 21) control the eyes. For now, the images on each eye will be the same, but you can adjust your code later, if you want to use different images on the different displays, depending on which one you soldered the A0 pads on.

Using the PIL library, the images are opened and stored (starting line 23).

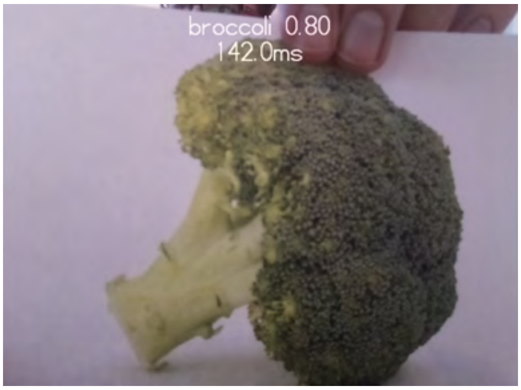
```
neutral = Image.open("neutral.png").rotate(90)
wide = Image.open("wide.png").rotate(90)
angry = Image.open("angry.png").rotate(90)
look_down = Image.open("look_down.png").
rotate(90)
```

The `change_eyes(life, right)` function changes the eyes that are displayed on the LEDs.

11 Changing faces

Now it's time to bring all your different functions together to change the whole face. The `faces` dictionary is used to store the different facial expressions you want to use. This will give values for the mouth motors, the eyebrow motor, and the eyes (lines 30 to 35).

And the `set_face(face)` function (line 63) is used to set the mouth, eyebrows, and eyes.



The classifier.py program detecting broccoli

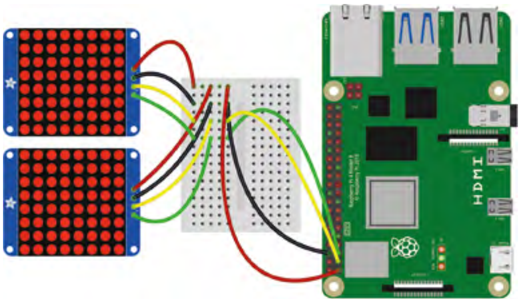


Figure 5 Wiring up Raspberry Pi

12 Emotional responses to objects

To finish off the project, you can make the robot face display different expressions depending on what the camera can see.

Start by importing the `Classifier` class from the file that you used earlier, along with the `sleep` function (lines 5 and 6).

Then create a `seen_items` list of objects that the classifier can recognise (line 38). You can adjust the threshold later to make the classifier more or less accurate with its recognition.

The `reactions` dictionary (line 41) links objects to different emotions. You can choose your own objects and emotions depending on what you have around you, or images that you have printed out.

Lastly, the `while` loop (line 70) checks what is in the `seen_items` list every two seconds, and then display the facial expression according to your `reactions` dictionary.

> "Add more expressions by altering the rotation of the motors"

13 What next?

There are lots of ways to extend your robot face project. You can add more objects that the face can recognise, and add more expressions by altering the rotation of the motors and the images displayed on the LED matrices.

There is also the possibility of adding sound to your project, using Pygame to play laughs, screams, and sighs when the face recognises different objects.

Or, how about taking all you have learned in this path and creating something original and new!

If you've completed all the LEGO Spark projects and want to have more fun exploring Python, then you could try out any of these projects.

English is not your mother tongue?

MagPi is also available in German!

elektor
DAS MAGAZIN FÜR DIE RASPBERRY PI COMMUNITY
MagPi
Das offizielle Raspberry Pi Magazin
Januar/Februar 2022 Nr. 1
magpi.de
NEU!
RASPBERRY PI ZERO 2
✓ Better & faster
✓ Detailed specs
✓ Engineering interview
✓ Project ideas
20 RASPBERRY PI PICO Projekte
Erkunden Sie die Welt der Sensoren
Experimentieren Sie mit einem völlig neuen System
64 SEITEN PROJEKTE UND PRAXIS
#29 Juli/August 202
Raspberry Pi Pico Grundlagen
48 SEITEN

Subscribe to the German edition of MagPi and get a Raspberry Pi Pico with headers and a cool welcome box **FOR FREE!**

Use the coupon code 115PicoDE

on www.magpi.de/115

50 HACKS & HINTS
RASPBERRY PI OS
EASY Raspberry Pi Pico PROJECTS

THE Official RASPBERRY PI HANDBOOK
2022
50 Raspberry Pi Hacks & Hints
iPod Classic Spotify Player
BeccaCam Digital Camera
Raspberry Pi Pico projects
Build a Home Assistant
Raspberry Pi Amiga 600
Walking Bipedal Robot
Updated! QUICKSTART GUIDE
Start incredible projects with Raspberry Pi
FROM THE MAKERS OF MagPi THE OFFICIAL RASPBERRY PI MAGAZINE

Build a Raspberry Pi Quiz System

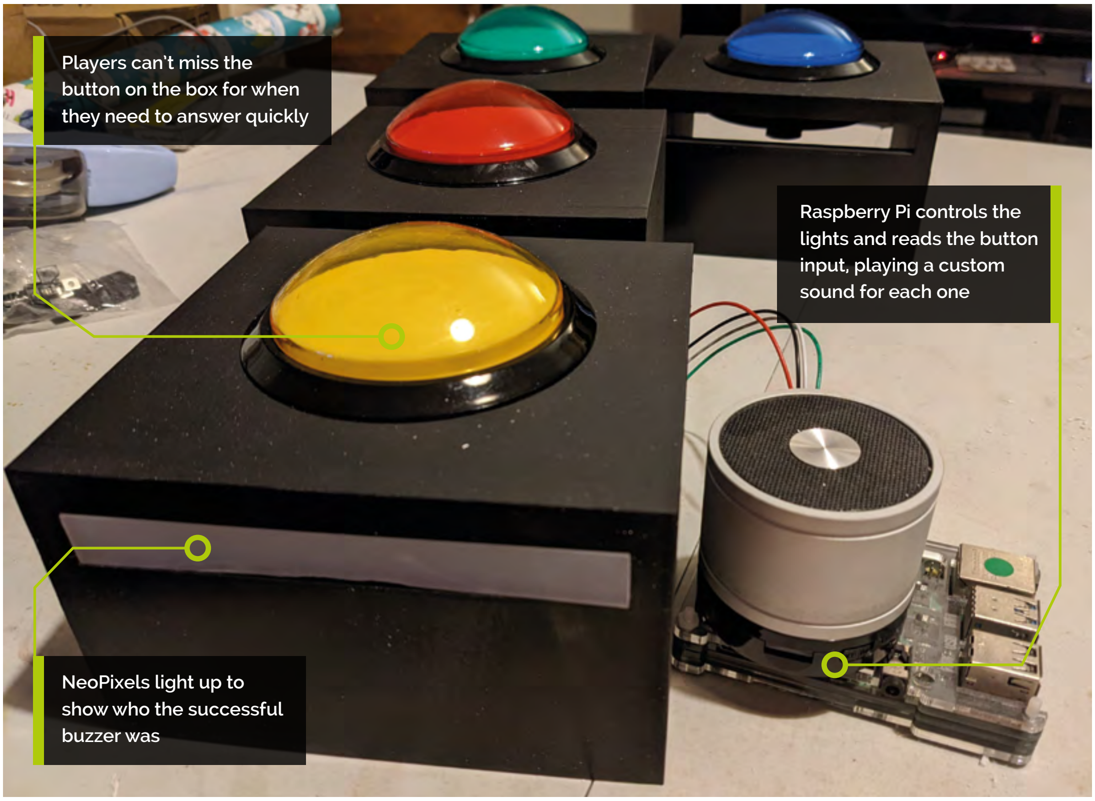
Create your own customisable buzzers to put on your own comedy panel show or pub quiz. Questions not supplied

MAKER

Rob Zwetsloot

Rob puts on yearly quiz panel comedy shows at anime conventions for some reason. At least, he thinks they're funny.

magpi.cc



Players can't miss the button on the box for when they need to answer quickly

Raspberry Pi controls the lights and reads the button input, playing a custom sound for each one

NeoPixels light up to show who the successful buzzer was

Warning! Power Tools

This tutorial uses power tools and sharp hand tools, so be careful if you plan to replicate it

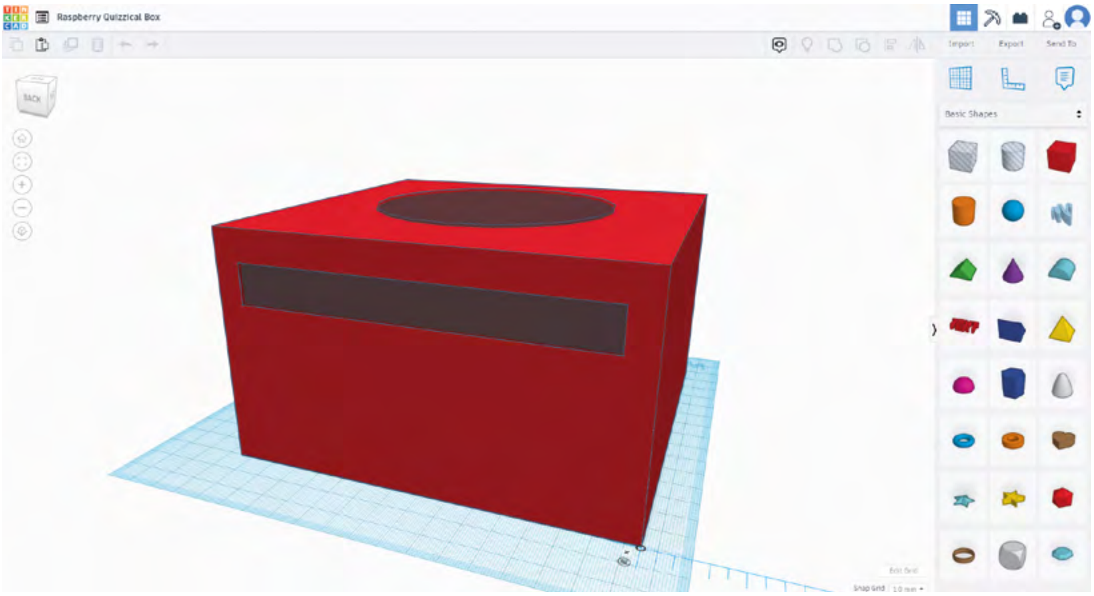
magpi.cc/drillsafety

Buzzing in is a quintessential part of many quiz experiences. Unfortunately, there is no real product you can just buy off the shelf or online that lets you have a series of connected buzzers. The best you can do is some barnyard buzzers, and a keen ear to listen out for what went first.

Thanks to advancements in quiz technology/maker components though, it's actually not too hard to make one yourself, and use minimal soldering as well. Now our first question: how quickly can we start?

01 Design your box

For our quiz buttons, we need a way to actually hold the button that looks somewhat professional. While the buttons themselves do have lights, we can make more interesting ones with NeoPixels, so we also need room for LEDs as well. Luckily, we know the dimensions of the button and NeoPixel strips, so the box can be made with the correct size hole on top, enough depth to contain it, and a large enough slot at the front for the lights to shine through. We also made it 10 mm thick. You can grab the STL we



▲ We designed a simple case for the quiz buttons on Tinkercad, as it allows for precise mm measurements

made from at **magpi.cc/magpiquiz**, or make your own on **tinkercad.com**.

02 Box printing

If you're lucky enough to own or have access to a 3D printer, then you can get straight to printing. Orientate the box with the front/side with letterbox opening down on the print bed, and have supports for the holes. The boxes need to be sturdy enough that pressing the button does not break them and, due to the 10 mm walls and general shape, a 5% infill is strong enough. You'll need to know this for printing yourself or finding someone else to print them. The boxes are not small, so the print may take a while.

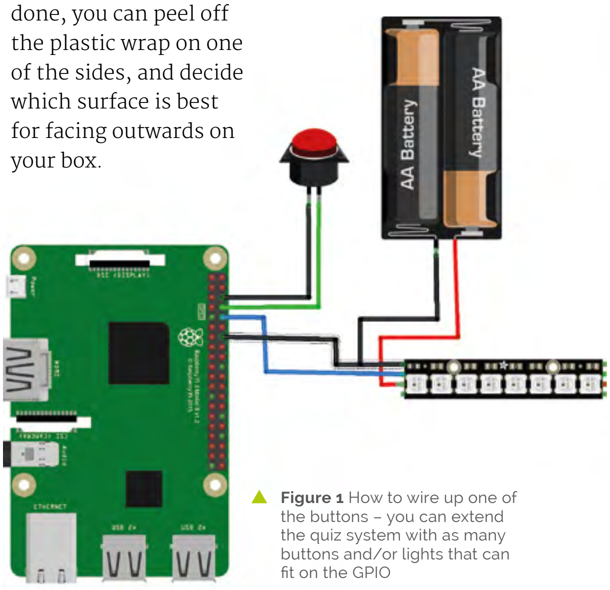
> " The boxes are not small, so the print may take a while "

03 Cutting acrylic

If you want frosted acrylic cut to size, you'll need to make the letterbox opening a bit bigger – a lot of places won't cut acrylic to less than 50 mm wide, and the smallest we saw were 25 mm. As our opening is 15 mm, we opted to cut it ourselves. If you have access to a laser cutter, that's the best way to get it cut. However, we used a mixture of handsaw and rotary tool to cut out the strips after carefully measuring them on the plastic wrap side. You could also use a jig-saw if you have one; however, frosted acrylic usually comes too thick to cut with a craft knife.

04 Trim to fit

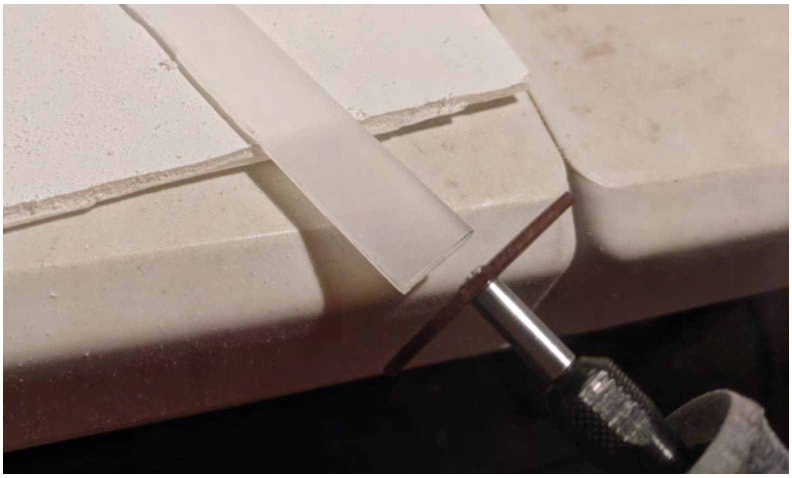
We made our strips a little bigger than the holes so we could sand them down to size. Again, we used a rotary tool for this, checking for fit as we went; however, sandpaper also works well for this. We suggest getting the width right first, and then making sure it fits lengthways. Once that's done, you can peel off the plastic wrap on one of the sides, and decide which surface is best for facing outwards on your box.



▲ **Figure 1** How to wire up one of the buttons – you can extend the quiz system with as many buttons and/or lights that can fit on the GPIO

You'll Need

- 4 × massive arcade buttons **magpi.cc/bigbutton**
- 1 m flexible NeoPixel LED strip **magpi.cc/neopixels**
- Arcade button quick connect wires **magpi.cc/buttonwire**
- 3-pin LED strip connector **magpi.cc/ledconnect**
- 4-pin JST connectors with male and female cable ends
- Wire
- Shrink-wrap
- Speaker
- Frosted acrylic
- 4 × rechargeable AA batteries
- Battery holder for 4 × AA batteries
- 3D-printed button box **magpi.cc/magpiquiz**



Hand-cutting acrylic is not always precise. Make use of a rotary tool or sandpaper to trim it to fit

05 Prepare the NeoPixels

The flexible NeoPixel strips can be cut between LEDs on a specific cut-line on the pads. Measure out how many you would need to fill the gap on front of the box, and snip out that length of LEDs you need to fit. Our NeoPixel strip came in a little rubber case around them, which you should slip the LEDs out of, and on one side attach the LED strip connector. This makes it easier to connect to the wires we're going to use to run control back to Raspberry Pi, without any soldering to the NeoPixel strip required.

Top Tip

Quiz tips 1

Sometimes you need to write questions where a specific answer is expected, even if it's not always correct.

06 Create the circuit

Our diagram, **Figure 1**, shows a Fritzing diagram for wiring up one of the boxes. The buttons have built-in resistors, so you do not need to add another one to the circuit. For prototyping this circuit, we stuck the ends of wires into the JST connectors and used prototyping wire to connect the two JST connectors together – for a portable system that is easy to dismantle, we will have male-to-male JST connectors with a long, shrinkwrapped cable to run some distance.

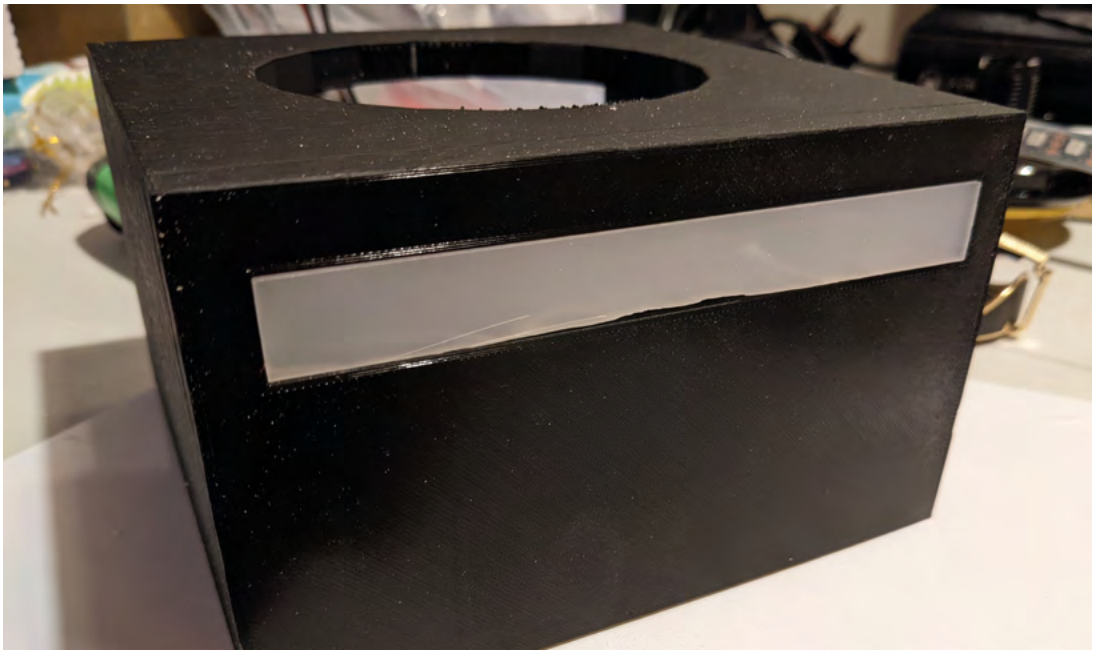
It's important to note that the ground connection of each NeoPixel strip needs to go to ground on the battery pack and Raspberry Pi. Feasibly, Raspberry Pi 4 can power all the NeoPixels, but it's safer and easier with rechargeable AAs.

07 Set up Raspberry Pi

Flash a fresh microSD card using Raspberry Pi Imager (**magpi.cc/imager**). It's easier to set up and add extra control later by using the standard install. Make sure it's all up-to-date, and then open the Terminal and install the extra libraries we'll need:

```
 sudo pip3 install rpi_ws281x adafruit-
circuitpython-neopixel
```

If you've not updated the password on Raspberry Pi OS as part of the setup, do so now, and then turn on SSH in case you need to dial into the system while a monitor is not available. If you have a



Use hot glue to affix the LED strip to the back of the acrylic, and then hot-glue the whole piece into the slot from the back of the box

Bluetooth speaker you plan to use, install it now and make sure you can play audio through it.

08 Get the code

You can download the code for this build from **magpi.cc/magpiquiz**, the same location as the STL for the box from earlier. We've also included a test audio file for all the buzzers to use. Download **quiz.py** and **buzzer.mp3** to a new folder – you can call it whatever you want, but we went with 'quiz'.

The code needs to be run with sudo, so for editing and debugging purposes, head to the Terminal and type `sudo thonny` to open up the Thonny IDE. Open **quiz.py** in it, and you'll be able to run it, update it, or add to it as you see fit for your system.

09 Always be testing

It's now time to test the circuit and code together, to see if there are any issues before installing everything into the box. This is how it should work.

With the script running, all the LEDs on the front of the box will be lit in white to indicate they're working. Pressing one of the buttons will activate a sound from Raspberry Pi, turn off the lights on all the other boxes, and have a pulsing series of lights on the box that was pressed. The other buttons cannot be pressed at this time. On the display, you'll see a graphical button labelled reset that will bring us back to the start.

> “It's easier to set up and add extra control later by using the standard install”

10 Install the components

It's now time to install everything into the boxes, carefully. Put the front side of the box on a flat surface so that you can lay the acrylic flat through the opening. With a glue gun, add some glue to the four corners and wait for it to set. After that, put a dab of hot glue on some of the pads on the NeoPixels and push it down onto the back of the acrylic. After those have set, put the top of the button through the hole, with the large washer underneath. Jiggle in the button/LED house, and tighten it all up, making sure the main JST connector runs freely out the back of the box.

11 Choose your audio

In the code, you'll find a line like:

```
buzzer1audio = "buzzer.mp3"
```

...with other buzzer audio lines after that. To change the audio for the buzzers, put an MP3 file in the same folder as **quiz.py** and then change the name to that in the code. It makes it a bit easier to rename it to a file name with a single word, e.g. **policesiren.mp3** or **horseracing.mp3**.

12 Improvements

As the code uses Pygame and creates an interface, you can easily expand that to get an indication from the host side for who's pressed the button. You can also use it to play a correct jingle, or a fail horn for if they're wrong. Depending on the format of the show, a hidden button on the back of the quiz box could be used by the contestant to signal they have something to share, allowing for a more natural conversation flow.

Now all you need to do is write that quiz. Good luck! M

Top Tip

Quiz tips 2

Write a quiz where you can teach people about an interesting subject, with each answer adding a little more to the story.

▼ We used different colour buttons to differentiate each box

Make a complete RetroPie console in a joypad

Make a compact device capable of playing all kinds of retro games – all packed into one controller

AUTHOR
RYAN LAMBIE

Ryan is the editor of Wireframe magazine, the author of *The Geek's Guide to SF Cinema*, and owner of far too many Sega Mega Drive games.

The likes of Nintendo and Sega may have got our nostalgia synapses humming with their miniature plug-and-play consoles, but how about an entire RetroPie device that fits snugly into a SNES-style controller? Follow the instructions here, and you'll have a complete, self-contained system capable of playing a wealth of retro games. It'll connect to your TV via HDMI, will charge using a micro USB port, and can be made for a relatively paltry sum.

A version of this consolised controller previously appeared in our sister publication, The MagPi – issue 42, to be precise – but here we're updating things a bit by adding the new Raspberry Pi Zero 2 W into the mix. The new Raspberry Pi Zero 2 provides a major upgrade over the one used in that earlier guide: not only is it more powerful, allowing you to emulate a greater range of systems and games, but its wireless connectivity allows you to add ROMs remotely. This is pretty handy since it saves having to open up the controller to get at the SD card.

Our project will make use of an affordable SNES-style USB controller, which, with a bit of modification, is large enough inside to fit in all the tech we need to run RetroPie: the Raspberry Pi Zero 2, a PowerBoost 500C power supply board, and a 500mAh LiPo battery.

The latter means you can simply charge the device up with a standard 5 V charger. For a full list of the bits you'll need, check out the box on the opposite page.

SET UP YOUR RASPBERRY PI

Before we start soldering, the first job is to install RetroPie on the Raspberry Pi Zero 2. To do this, you'll need to download a copy of RetroPie from **retropie.org.uk** – see the box on page 64 for details. The Raspberry Pi Zero 2 W doesn't have

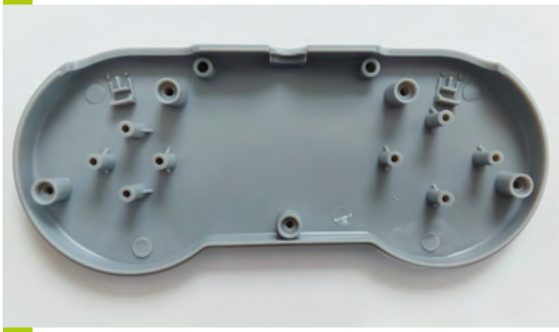


Figure 1: Most of the struts and screw posts you can see here will need to be removed so we can fit everything inside the joypad.

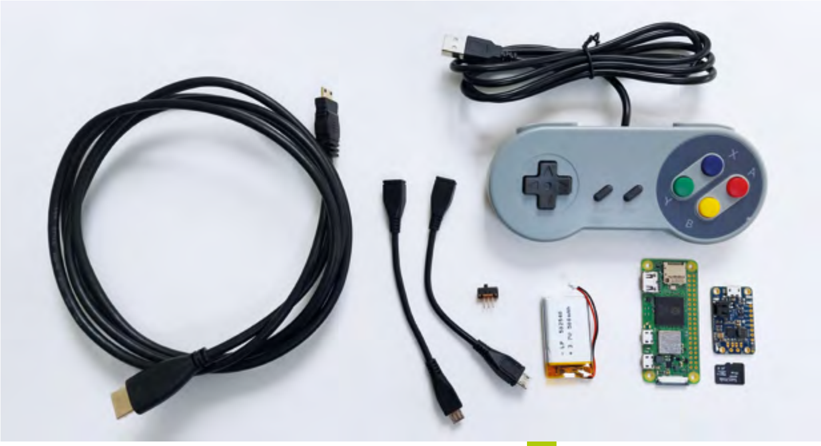
Wireframe

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

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

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

The parts you'll need for this project.



a specific version of RetroPie ready at the time of writing, but that may have changed by the time this issue hits newsstands – keep an eye on RetroPie's website for updates.

Once RetroPie's installed, you'll then need to plug a keyboard into the Raspberry Pi and connect the device to your wireless network. To do this, power up the Raspberry Pi Zero 2, let RetroPie boot up, then go to the RASPI-CONFIG option in the main menu. From there, select System Options and S1 Wireless LAN, choose your country of residence, then input your wireless network connection's SSID and password.

With that set up, you now can add a few ROMs to the Raspberry Pi Zero 2 ready for testing later. You can find a list of where to get some great homebrew and legal ROMs from **magpi.cc/legalroms**. ROMs can then be added wirelessly by first ensuring your Raspberry Pi Zero 2 is turned on, opening up a window on your PC, and typing \\RETROPIE into the Quick Access bar. (Find more detailed instructions for adding ROMs wirelessly at **retropie.org.uk/docs**, where you'll also find directions for Mac, Linux, and PC.)

MOD THE CONTROLLER

There's a surprising amount of space inside your typical SNES USB controller, though you'll still find that fitting everything we need into its shell will require some careful positioning – and a few nips and tucks here and there.

To begin with, we'll need to take the controller to bits by removing the screws in the back of the case. If you flip the backplate over, you'll see a number of plastic struts that will eventually need to be snipped away with a sharp pair of pliers or a craft knife (see **Figure 1**). Once the Raspberry Pi and other components start going in, you'll have a better idea of what will need to be trimmed out of the backplate, so put it to one side for now.

> **"A complete, self-contained system capable of playing a wealth of retro games"**

WIRE THE CONNECTOR

The beauty of our consolised joypad is that it's entirely self-contained. The HDMI cable will emerge straight from the controller itself, taking the place of the existing USB cable, while power will be delivered by that dinky LiPo battery – →

YOU'LL NEED:

Unless otherwise specified, all the items below were acquired from **thepihut.com**:

- Raspberry Pi Zero 2 W
- MicroSD card
- USB SNES controller
- PowerBoost 500C
- SPDT switch
- Mini HDMI-to-HDMI cable (2 m)
- 3.7 V 500mAh LiPo battery: *from Amazon.co.uk*
- 3 × jumper cables
- Velcro Command Strips *from Amazon.co.uk*
- 1 × micro USB-to-USB cable *(optional)*

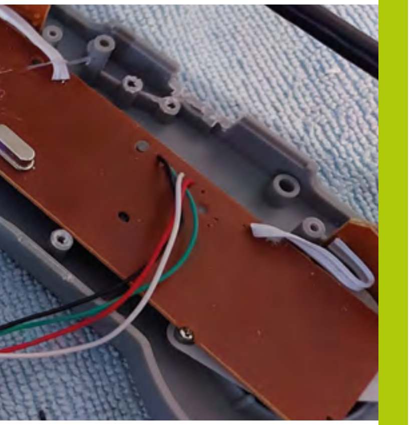
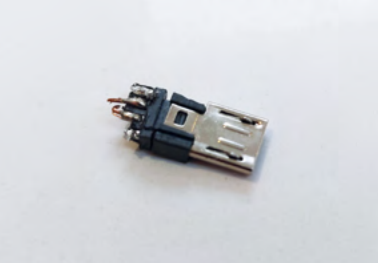


Figure 2: When removing the existing USB cable, be sure to leave about 5 cm of the coloured wires behind.

Figure 3: A micro USB connector stripped and ready for use.



which in turn can be charged via the micro USB port on the PowerBoost 500C board.

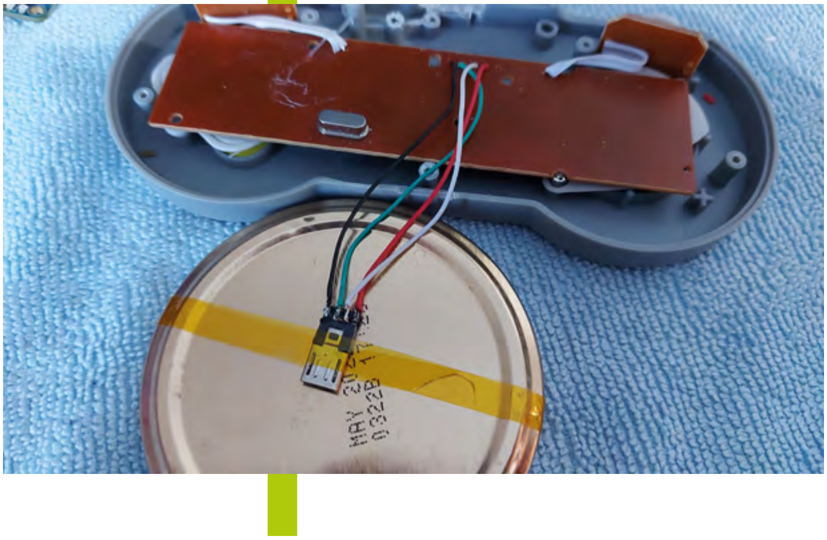
For button inputs, we're going to connect the controller's existing USB cables to the Raspberry Pi Zero 2 using the stripped connector from a micro USB cable. To do this, we'll need to cut the USB cable from the controller while ensuring we have enough remaining wire to connect to the Raspberry Pi – around 5 cm should be plenty for now (**Figure 2**). If your controller's anything like ours, bear in mind that the wires soldered onto the board are quite delicate and have a tendency to detach if they're left to wiggle around too much. A bit of electrical or Kapton tape will help keep them from moving around as we get on with our project.

Next, take a micro USB cable, and with a pair of pliers, gently strip away the plastic sheath until you've revealed the metal innards of the male connector at the end – once you're finished, it should look something like **Figure 3**. With the old wires from the cable removed from the connector, you can now solder the wires from the controller in their place – the colour coding of the wires should be the same, with the order being black, green, white, and red from left to right (see **Figure 4**).

Once the wires have been soldered to the connector, you can give your joints some protection (and prevent any electrical shorts) by neatly wrapping them with a bit of electrical or Kapton tape (**Figure 5**).

"Your mileage may vary depending on which sort of joypad you're working with"

Figure 4: The wires from the controller need to be soldered to the connector like this. If you don't have a set of helping hands, taping the connector to a lid like this helps keep it still.



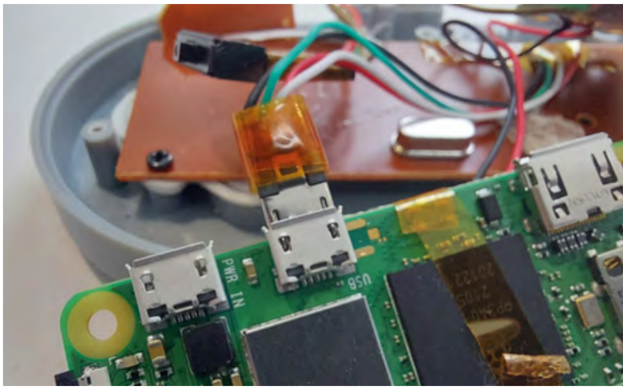


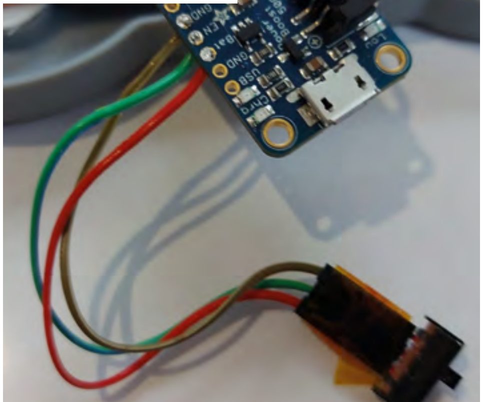
Figure 5: Once your USB wires are soldered in place, wrap a bit of Kapton tape around them for protection.

WIRE UP THE SWITCH

Our console-controller hybrid requires 5 volts to run, which will be delivered by the PowerBoost 500C – a tiny bit of tech that will take the power from our 3.7 V LiPo battery and uprate it to the correct voltage. What we'll need to add, though, is an on-off switch: we've used an SPDT (single pole, double throw) switch, which we'll connect to the PowerBoost with three colour-coded jumper cables. The female ends of the cables slide neatly onto the switch; the male ends can then be snipped off and soldered to the GND, EN, and Bat pins on the PowerBoost (**Figure 6**).

We then need to connect the PowerBoost to our Raspberry Pi Zero 2. There are two ways you can do this: you could use the stripped connector from a second micro USB cable, solder its 5 V and GND wires to the PowerBoost, and plug the connector end into the Raspberry Pi Zero 2's USB input. In practice, though, we found that

Figure 6: The jumper wires are soldered to the PowerBoost like this.



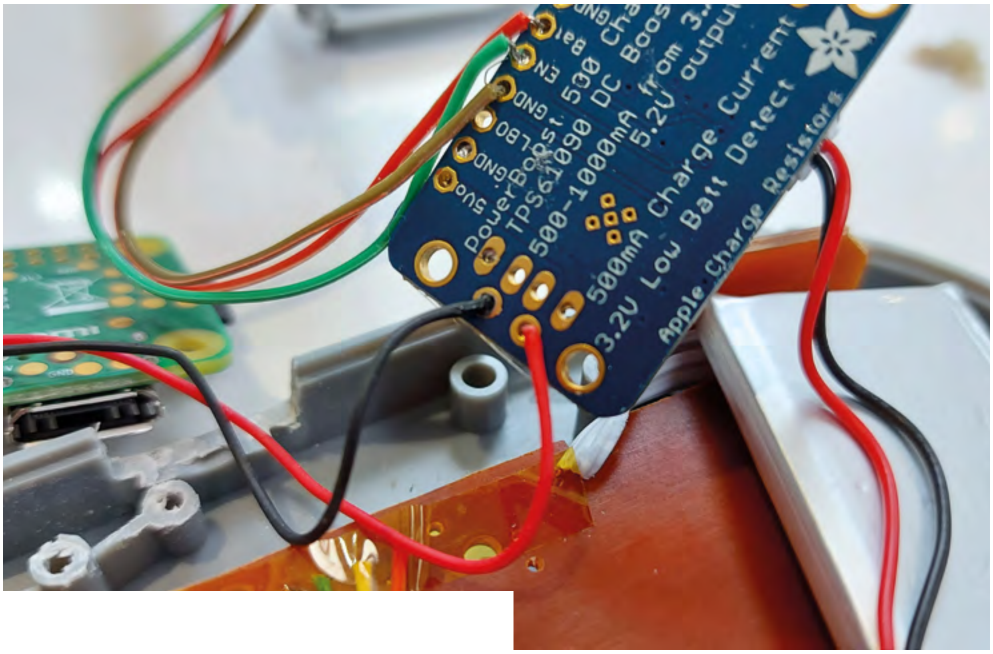


Figure 7: 5 V (red) and ground (black) wires soldered to the respective pins on the PowerBoost 500C.

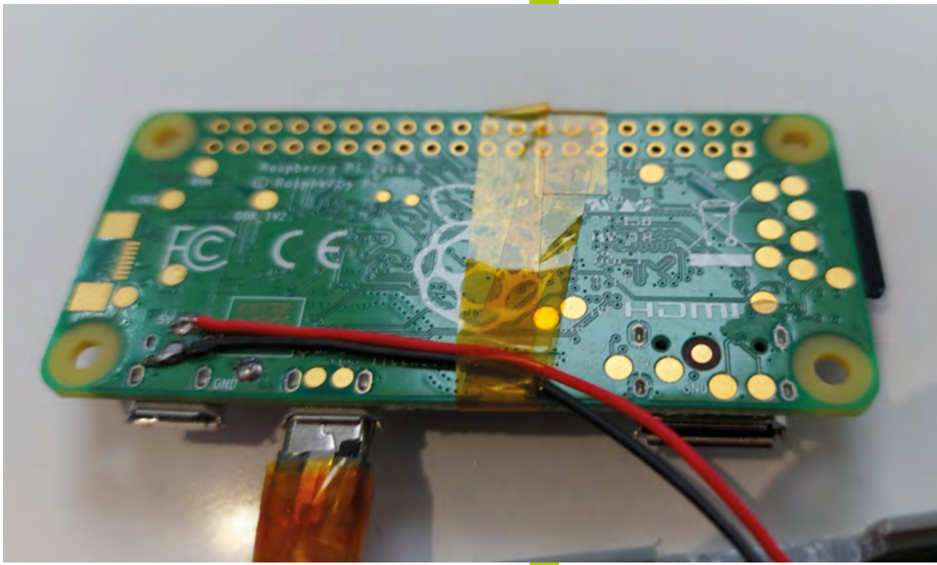


Figure 8: The 5 V and ground wires can now be soldered to the respective pads on the Raspberry Pi Zero 2.

there simply wasn't quite enough room for the Raspberry Pi Zero 2 to fit inside the controller's case with the USB connector sticking out of it; your mileage may vary depending on which sort of joypad you're working with. Instead, we took two wires and soldered them to the 5 V and ground pins on the PowerBoost (see **Figure 7**), before soldering the other ends to the 5 V and ground pins on the underside of the Raspberry Pi Zero 2. We used another bit of Kapton tape to keep them neatly attached to the back of the Raspberry Pi Zero 2 (**Figure 8**).

CONNECT THE POWERBOOST

In terms of soldering, we're essentially finished – we can now connect the battery to the PowerBoost and see how things are working. A quick note here, though: these LiPo batteries sometimes have their positive (red) and negative (black) cables wired up the opposite way from the PowerBoost, so ensure you have them the right way around before plugging them in; if you get them reversed, you could run the risk of damaging the PowerBoost board (see **Figure 9**).

The LiPo battery probably isn't charged up, either, so have a 5 V micro USB charger handy, just in case. All being well, flicking your switch to the 'On' position will result in a blue light on your PowerBoost board, and a flickering green light on the Raspberry Pi Zero 2. If you want, you can connect your HDMI cable to the Raspberry Pi Zero 2, stick the other end in your TV or monitor, and carefully check that your Raspberry Pi boots up and that the buttons on your controller are behaving correctly. →

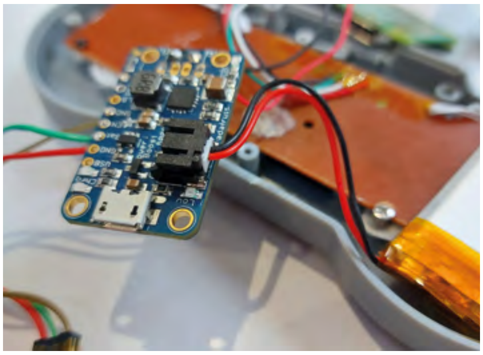


Figure 9: Ensure the positive (red) and negative (black) wires are the correct way around when you plug your LiPo battery into the PowerBoost.

TOOLS OF THE TRADE

- Soldering iron
- Small screwdriver
- Leaded solder
- Wire strippers
- Sharp-nosed pliers
- Needle file
- Sharp craft knife

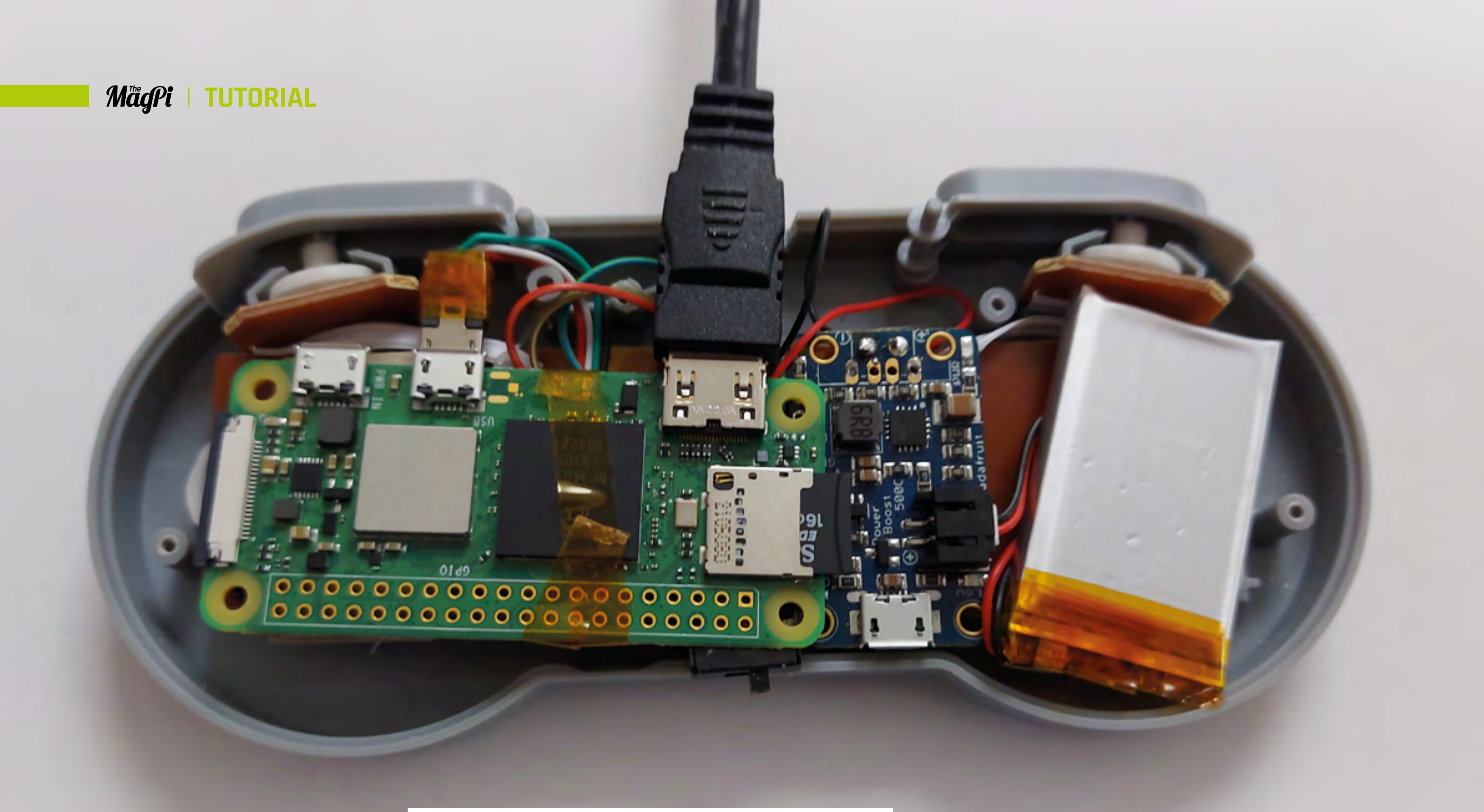


Figure 10: Here's how all the parts need to fit inside the controller.

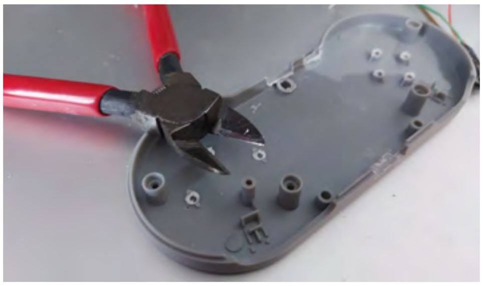


Figure 11: We can now start to trim away the unwanted pieces of plastic from the two halves of the case.

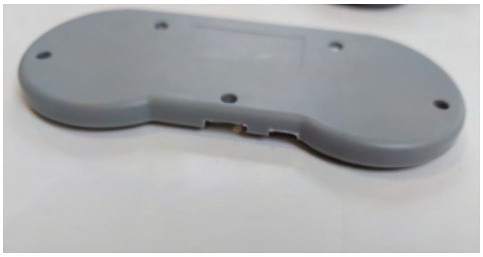


Figure 12: Be sure to cut good-sized holes for the on-off switch and the PowerBoost's micro USB port.

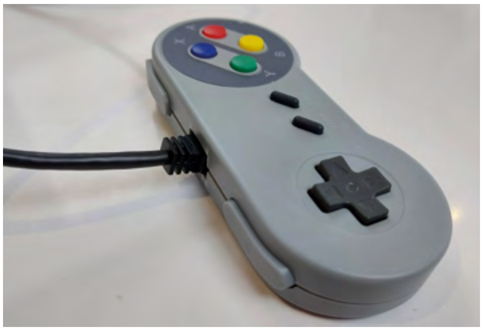


Figure 13: You'll need to cut a fairly large aperture to allow the mini HDMI connector to pass into the controller.

ASSEMBLE THE CONTROLLER

With all the soldering and connecting done, the next job is to fit everything inside the joypad. **Figure 10** shows how we've laid everything out: the Raspberry Pi Zero 2 needs to sit towards the bottom right of the controller case so that the HDMI cable can go straight through the middle of the aperture, where the USB cable sat. The on-off switch will go underneath the Raspberry Pi Zero 2 at the bottom; the PowerBoost 500C will sit directly to the right of it, while our battery just about sits in the bit of space remaining on the far right.

It's now that you should see how much plastic will need to be carved out of the case to get everything to fit. We found that we had to cut most of the struts from the backplate to make enough room, with just the left and right supports for the screws left, so we can still secure the two halves of the case back together when we're finished (**Figure 11**). You'll also need to cut holes out of the case to allow for the HDMI cable, the switch, and the micro USB port on the PowerBoost – this is positioned as close as possible to the edge of the case so that when it's all reassembled, we can plug a charger in and feed the LiPo battery now and again (**Figure 12**).

Carving the case can be fiddly, but take your time, and use a needle file to smooth any rough edges. If you like, you can pare down the plastic housing on the mini HDMI connector so that you don't need to cut such a large hole at the top of the controller case; we eventually opted to leave

RETROPIE ON RASPBERRY PI ZERO 2 W

At the time of writing, there isn't a specific version of RetroPie ready for the Raspberry Pi Zero 2 W. The good news is that the Raspberry Pi Zero 2 uses some of the same underlying tech as the Raspberry Pi 3B+, so that computer's image is sufficiently compatible. To download it, you'll need the latest version from RetroPie's Weekly Builds page at **wfmag.cc/RPWB**. Once this image is installed on your Raspberry Pi Zero 2, you'll need to connect it to the internet and ensure its drivers are all updated to the latest versions.

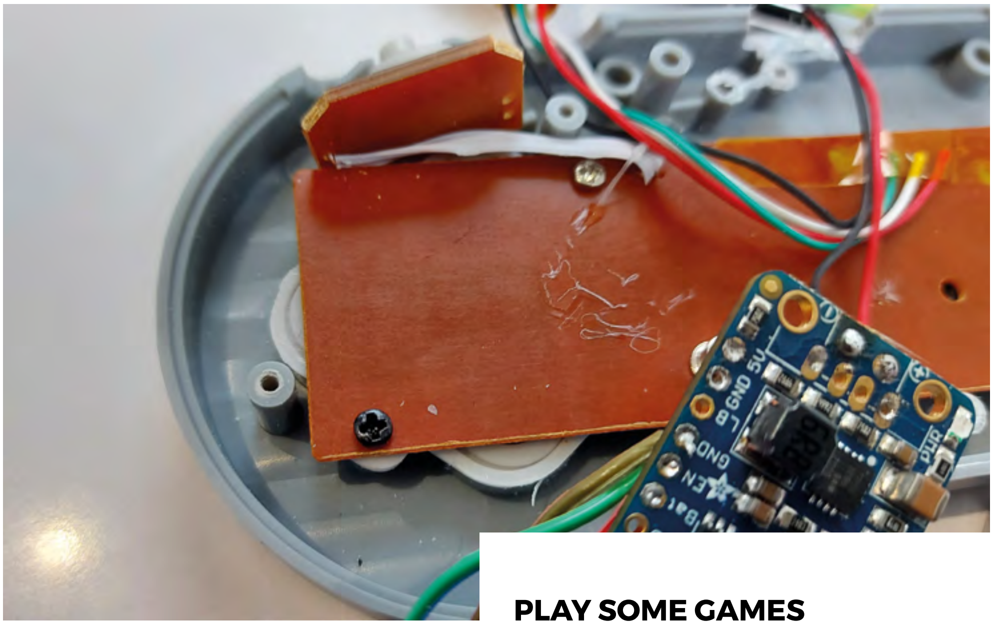


Figure 14: To keep our buttons feeling nice and firm, we added a couple of small screws to the controller's board.

the connector alone and just whittle away at the controller, but it's entirely up to you (**Figure 13**).

While we were hacking away at the case, we also took a couple of spare screws and used them to secure the controller's board to the front plate; this is worth doing, since we've cut away the supports that once kept the board pressed firmly in place. Without these screws (see **Figure 14**), we found that the buttons felt a little spongy and hard to press, even with all those new components crammed in tightly behind them.

When it comes to fitting everything into the case, you can either use hot glue to mount the components to the controller's board, or take our approach: use Command strips instead. The latter option keeps everything securely in place, but it's also repositionable so you can make adjustments as you go along. We found this approach also helps the Raspberry Pi sit neatly on top of any components that may be sticking up from the back of the controller's board (**Figure 15**).

As you're fitting everything together, pay attention to how you route the various wires between components: these need to be positioned so that they aren't snagged by the screws or the shoulder buttons as the controller's reassembled. You'll also want to add some insulating tape between the PowerBoost and the underside of the Raspberry Pi to avoid any shorts.

PLAY SOME GAMES

With the controller screwed back together, you're essentially finished. You can now connect your device to a screen and start playing some games. If you haven't already, you'll need to configure the controller's buttons in RetroPie; it may give you a warning about not setting up a hot key if you run out of buttons to define, but don't worry – just press Select and Start at the same time to quit out of a game at any point. The built-in Wireless LAN means you can add ROMs remotely, as well as scrape the web for thumbnail images for your games.

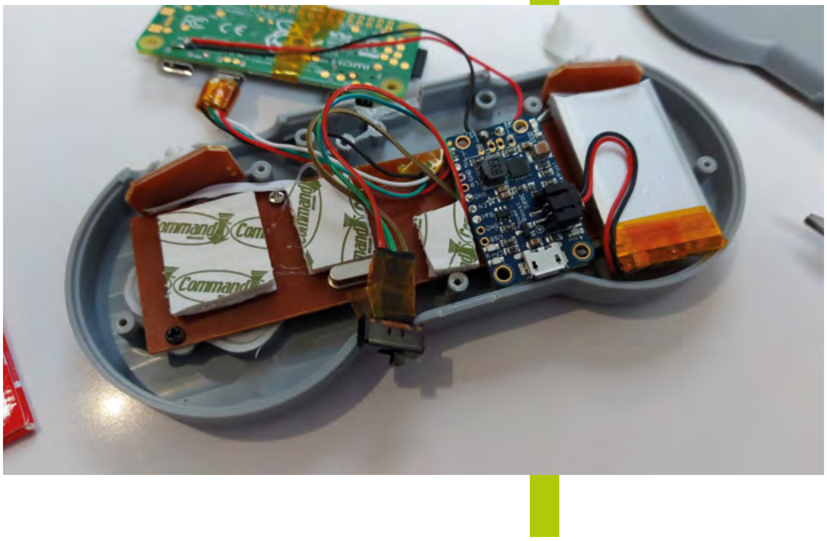
There we have it: one compact, low-cost console in a joypad. Happy gaming!

"The built-in wireless LAN means you can add ROMs remotely"

ALTERNATIVE CONTROLLERS

We've used a SNES replica as our controller for this build; within reason, you could use just about any joypad you like, assuming there's enough space inside to mount all your components. An original Xbox 'Duke' controller, for example, is roughly the size of a Fray Bentos pie tin, and should have ample room for this project.

Figure 15: Command tape is strong enough to hold everything in place, while allowing us to reposition things if we need to.



#MonthOfMaking 2022 – Microcontroller Mayhem

As March draws near, we'd like to invite folks, once again, to join in with the community collaborative build event: #MonthOfMaking

During the month, we want you to start that project you've been putting off. With the help of the community, we can get these all built and have something amazing to show for it!

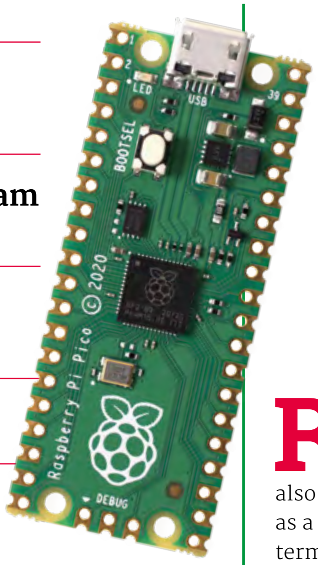
You can build whatever you want, but in this feature we'd like to highlight new and different ways to make. This year, we want to highlight Raspberry Pi Pico, which has only been out for around a year but has made a huge splash on makerspaces. With Raspberry Pi's own microcontroller, you can do a lot of stuff that Raspberry Pi can and cannot do, but with an even tinier footprint.

Grab your Pico, and let's get making!

The Rules of #MonthOfMaking

- Work on a project, new or old
- Take photos of your progress and completed projects
- Share it to Twitter and/or Instagram with a helpful description
- Make sure to include the #MonthOfMaking hashtag
- That's it!

If you don't have social media, you can always email your projects to **magpi@raspberrypi.com**



Pico buyers' guide

Raspberry Pi Pico is the amazing microcontroller made by Raspberry Pi, using the custom chip RP2040 that was also created by Raspberry Pi. It's not as powerful as a standard Raspberry Pi board in raw computing terms, but it's better suited for some smaller-scale projects (and much cheaper).

Essential accessories

Pico Explorer Base | £22

A great way to experiment with all the amazing functions of Raspberry Pi Pico.

magpi.cc/picoexplorer

Pico Wireless Pack | £12

Need a network connection for your Pico project? This little add-on adds that and a microSD card slot.

magpi.cc/picowireless

Essential Raspberry Pi Pico Kit | £6

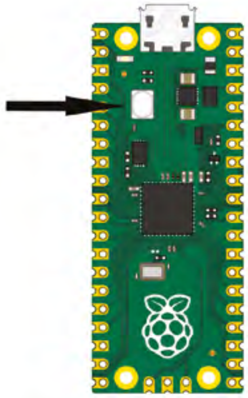
It's got essential in the name for a reason – you can add headers, a little USB shim, and even feet to your Pico with this.

magpi.cc/essentialpico

Raspberry Pi Pico projects

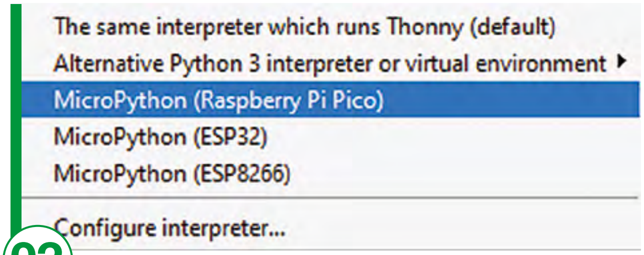
Make something cool with a Pico

Set up your Pico



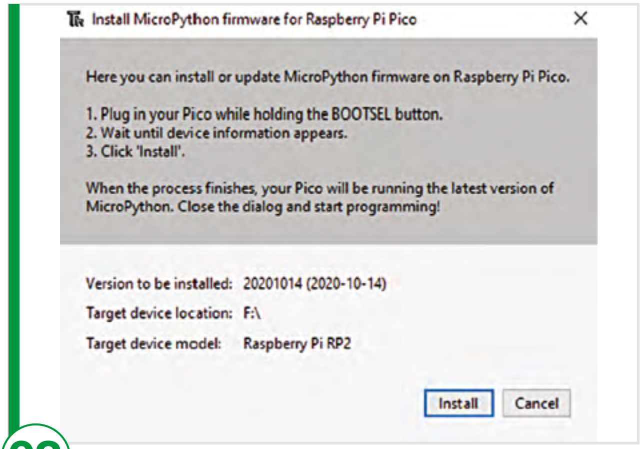
01 Updating the firmware

To use MicroPython, you need to install new firmware to your Pico. Plug your USB cable into your computer or Raspberry Pi of choice, hold down the BOOTSEL button on top of Pico, and plug it into the end of the USB cable. You can now access mass storage device mode.


The same interpreter which runs Thonny (default)
Alternative Python 3 interpreter or virtual environment
MicroPython (Raspberry Pi Pico)
MicroPython (ESP32)
MicroPython (ESP8266)
Configure interpreter...

02 Get Thonny

If you're plugging into a Raspberry Pi running Raspberry Pi OS, you don't need to worry about installing Thonny, a Python IDE. If you don't have it on your computer, head to **thonny.org** and install it. Open it up and click on the Python version in the bottom right corner, then select 'MicroPython (Raspberry Pi Pico)'.



Install MicroPython firmware for Raspberry Pi Pico

Here you can install or update MicroPython firmware on Raspberry Pi Pico.

1. Plug in your Pico while holding the BOOTSEL button.
2. Wait until device information appears.
3. Click 'Install'.

When the process finishes, your Pico will be running the latest version of MicroPython. Close the dialog and start programming!

Version to be installed: 20201014 (2020-10-14)
Target device location: F:\
Target device model: Raspberry Pi RP2

03 Confirm install

A dialog box will ask if you want to install the firmware to your Pico – click Install and it will be sent over to it. Click on Close once it's done, and you're ready to code.

Getting a Raspberry Pi

If you're having trouble getting a new Raspberry Pi for a project you want to make, we suggest signing up for email notifications from Approved Resellers so you can be the first to know when stock is available.

shop.pimoroni.com

thepihut.com

okdo.com

Amazing Pico projects

Hot Wheels Drag Race

Robert Hamilton-MacLaren

This LEGO™/Hot Wheels/Pico hybrid is a lot more complex than you might think. It has light sensors, accurate starting lights, 3D-printed parts, and is designed to be easily taken apart and moved around as well with liberal use of connectors. It can run on a battery too, so you don't need to plug it into the wall when moving it around.

magpi.cc/picodrag

Raspberry Pi Pico solar system display

Dmytro Panin

This simpler build is no less impressive to look at. Using a smart-looking Pico Display, Dmytro was able to create a clock that also acts as a solar system simulator – displaying the orbits of the planets at the time set. This means you can set it to the future or past, and get a fairly accurate representation of the positions of all eight planets.

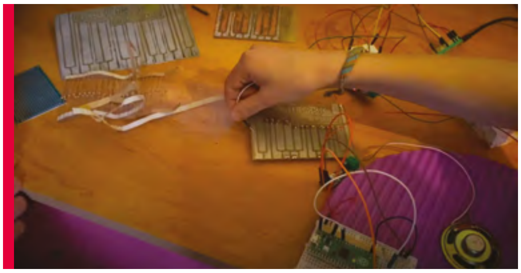
magpi.cc/picosolar

Raspberry Pi Pico piano

GurgleApps family

A very cool and DIY piano that allows you to touch keys to activate notes – touching a key will complete the circuit, and the different keys are further along a path of resistors which will create a different voltage reading on Pico. Once analysed, the correct note can be played. The GurgleApps siblings tried out loads of different materials before settling on the final version.

magpi.cc/picopiano



Midi Fighter

Liz Clark aka Blitz City DIY

This MIDI controller uses a series of arcade buttons which are perfect for mashing, which is why it's a Midi Fighter. While you can't throw hadoukens with it, you can lay down some sick beats in this stylishly 3D-printed case. You can also use it with a digital audio workstation over USB if that's more your thing.

magpi.cc/midifighter

Resources for Pico

Raspberry Pi Pico Documentation

The official documentation for Pico is a comprehensive guide on just about everything you'll need to know about the microcontroller. It even has info on how to use MicroPython and C on Pico, so it's a good place to start.

magpi.cc/picodocs

Get Started with MicroPython on Raspberry Pi Pico

From our sibling publication HackSpace Magazine comes this book that is all about how MicroPython can be used on Pico. It has a load of great projects from which to learn, such as a reaction game, traffic light controllers, and even some tips on how to use I2C and SPI.

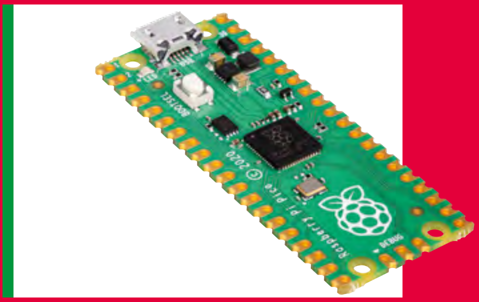
magpi.cc/picobook

Raspberry Pi and Raspberry Pi Pico projects

How to use Pico to augment your Raspberry Pi

Raspberry Pi 4 vs Pico

CPU: Quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
RAM: 1/2/4/8 GB
Connectivity: 2 × USB 3.0 ports, 2 × USB 2.0 ports, 40-pin GPIO, wireless LAN, Bluetooth 5.0, Gigabit Ethernet, 2 × micro-HDMI ports, CSI camera port, DSI display port, 3.5 mm audio and composite video jack
Storage: MicroSD card, minimum 4GB
Operating system: Yes
Analogue support on GPIO: No

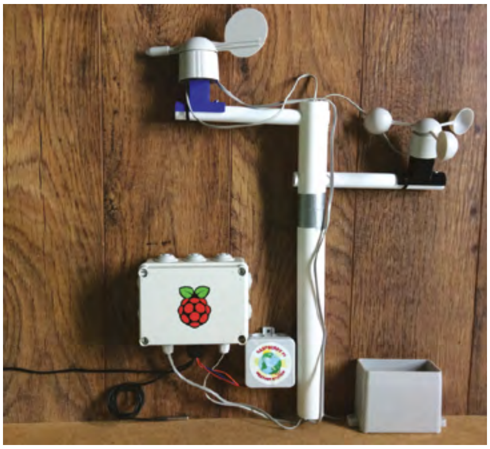


CPU: Dual-core ARM Cortex-M0+ processor @ 133 MHz
RAM: 264kB SRAM
Connectivity: 26-pin GPIO with 3 × analogue inputs, 2 × UART, 2 × SPI controllers, 2 × I2C controllers, 16 × PWM channels, 8 × Programmable I/O (PIO), micro-USB port (power and data)
Storage: 2MB flash
Operating system: No
Analogue support on GPIO: Yes

Connecting Pico to Raspberry Pi

Pico is great at a wide variety of electronics tasks that Raspberry Pi can't handle without some extra help. Pico can be that extra help. The simplest method is GPIO to GPIO, with Raspberry Pi acting as buttons for Pico to read, and maybe translating analogue signals into digital (ADC) for Raspberry Pi to read. With a USB connection, you can have Raspberry Pi update the programming on Pico, and Pico can send some data back. You can also create a serial connection between the two over the USB and Tx/Rx pins, as discussed in this great tutorial: **magpi.cc/picoserial**.

Amazing Raspberry Pi projects (that could use Pico)



Raspberry Pi Weather Station

Raspberry Pi

Weather stations are a great way to do a bit of low-key science and also know whether or not you should put a coat on before you head to the shops. While these Raspberry Pi ones work very well, the majority of the sensors take analogue readings which Raspberry Pi cannot handle directly. With Pico, you can use cheaper and simpler sensors to still get a good result which can be fed back to Raspberry Pi to display or make available over a network. Caroline Dunn wrote a good tutorial on how to make a Pico Weather Station here: **magpi.cc/picoweather**.

magpi.cc/weatherstation

Looking for more inspiration for using Raspberry Pi and Raspberry Pi Pico together? Check out our Arduino and Raspberry Pi feature from issue 61 for how to use microcontrollers with Raspberry Pi: **magpi.cc/61**.

Christmas lights

Rob Zwetsloot

Our own features editor created NeoPixel lights for his Christmas tree one year. Raspberry Pi can control them; however, Raspberry Pi Pico is more suited to the job. You just need to add the specific CircuitPython libraries (a slight offshoot of MicroPython), wire it up correctly, and you're away. Take a look at the Adafruit guide on NeoPixels and Pico: **magpi.cc/picopixels**.

magpi.cc/xmaslights

P19 robot

Team PiDrogen

The robots from the PiDrogen team are Pi Wars champions, with loads of features added on, such as a series of sensors for automation. A lot of these sensors could be controlled via Pico – such as getting distance from lasers and ultrasonic sensors, proximity switches, and even the cool lights used throughout the chassis. People have already been making entire robots out of Pico; however, computer vision is something only Raspberry Pi can do for now.

pidrogen.com

Getting ideas and inspiration

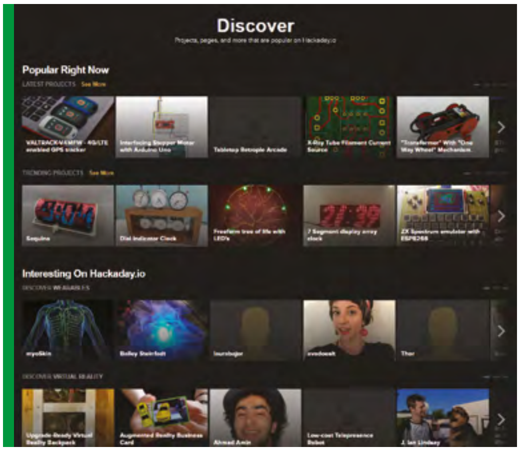
What do you want to build?

We've all been there. Sat down at a workbench or desk, staring at some components and thinking… what can I make with this? What would I like to make? Like any other creative pursuit, you'll need some inspiration. If the projects in the magazine haven't inspired you, then here are some website suggestions we have…

Tools for the job

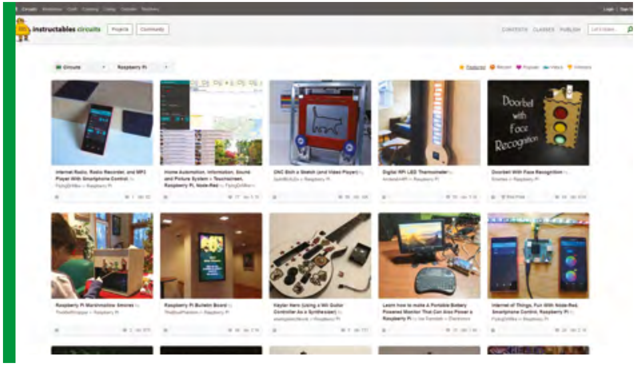
Not everyone has a full workshop of tools and gizmos, especially if this is your first foray into making. Check out issue 114 of *The MagPi* (**magpi.cc/114**) for our feature on maker tools for all ages.

Project websites



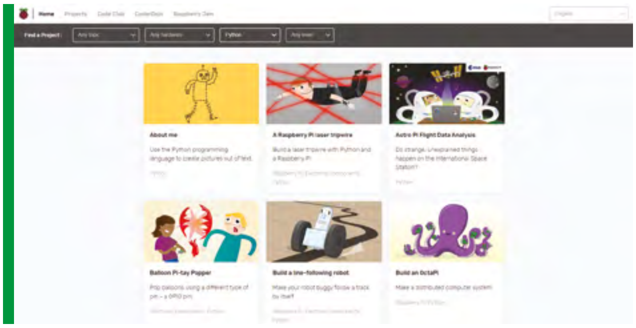
Hackaday

For more serious hacks for more advanced makers, Hackaday has some great projects that really take a deep dive into a project. If you're curious as to the limits of electronics and programming, this may be the place to look. Equally, if you want to do something huge with a lot of computer power, this should be your first stop.
hackaday.io



Instructables

Instructables is one of the oldest sites out there for finding amazing project guides and ideas, and we've been fans of it for years. The best part is you can search by specific project types as well, including Raspberry Pi if you'd like to keep it on-brand. They've recently added more arts and crafts stuff if you fancy trying your hand at knitting.
instructables.com


Find a Project
Python
About me
A Raspberry Pi laser tripwire
Astro Pi Flight Data Analysis
Balloon Pi-tay Popper
Build a line-following robot
Build an OctaPi

Raspberry Pi Projects

There's so many amazing things on the Raspberry Pi Projects site that can help you with your first steps in just about any field of making. It's also home to loads of great and simple homegrown projects that are perfect for young makers and older makers alike.
magpi.cc/projects

Sharing your projects

Show off your achievements online!

The point of #MonthOfMaking is to share what you're building! Here are our tips on how to post your projects online.

Posting your projects

#MagPiMonday

Top Latest People Photos Videos

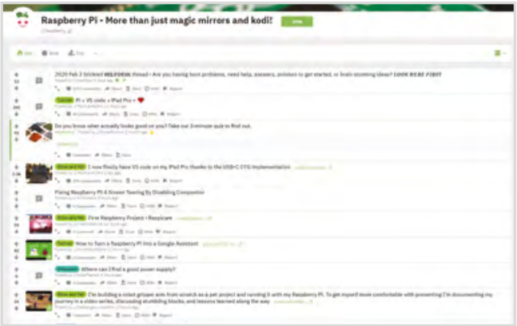
Dr Footleg · Robotem @drfootleg · 3h
Replying to @TheMagPi
Finished my first revision of my Robotics board for the Raspberry Pi. Fixed the main issue of wrong side footprints for all the resistor arrays, but also addressed many minor niggles. Sent the order on Sunday afternoon to get a batch of boards made to this design. #MagPiMonday

phil hardcastle @phil_hc · Jan 6
#MagPiMonday @TheMagPi
My Pi polytunnel monitor goes out again. Pi + temp/humidity sensor, & relays with linear actuators to open/close ventilation doors. Also, because it's off the web it's got a RT clock and an SMS dongle for reporting and control. 18 months of tinkering!

Orchard Audio @Orchardaudio · Dec 6, 2019
Replying to @TheMagPi
#magpimonday
orchardaudio.com/pocket

Twitter

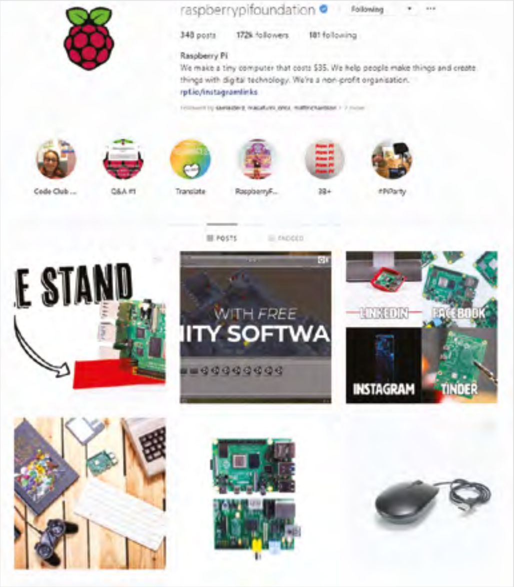
Posting on Twitter is the best place to share your project. A brief description, pictures, and any relevant links if you have a blog for the build, along with the #MonthOfMaking and #MyLatestBuild. You can also tag in **@TheMagPi**, and we'll make sure to share it ourselves, and maybe feature it in the magazine!



Raspberry Pi - More than just magic mirrors and kodi!

Reddit

There are many relevant subreddits for different project types – although the Raspberry Pi subreddit is a good place to go for any Raspberry Pi-related projects you want to post. Upload some images to Imgur when you've finished your project, mark your post as 'Show and Tell', and it's ready to show! We like to keep an eye on Reddit, so we'll look out for any #MonthOfMaking posts!



raspberrypifoundation Following

340 posts 172k followers 181 following

Raspberry Pi
We make a tiny computer that costs $35. We help people make things and create things with digital technology. We're a non-profit organisation.
rpf.io/instagramlinks

Code Club ... Q&A #1 Translate RaspberryF... 28+ #PiParty

POSTS TAGGED

Instagram

We love using Instagram in our personal lives, and posting amazing project builds on there is a great way to get them seen by people, and to find other amazing folks who are posting build photos. Using the #MonthOfMaking hashtag will get it seen by folks looking at that tag, and if you want to post more photos of your build, you can always make it into a story using the story feature!

Send it to us!

If you'd prefer to send us your builds, send us an email at **magpi@raspberrypi.com** with the subject line #MonthOfMaking!

Be in The MagPi!

We'll be keeping an eye on all the amazing stuff you make over the #MonthOfMaking, and we'll be sure to feature all the great stuff we see in the magazine in some way. You may even end up as a project showcase…

Argon **EON**

▶ Argon40 ▶ **argon40.com** ▶ From £95 / $128

Argon adds to its respected Raspberry Pi cases with a striking new NAS enclosure. **PJ Evans** takes it for a spin

SPECS

DIMENSIONS: 210 mm × 165 mm (per side)

STORAGE CAPACITY: 2 × 3.5″ drives, 2 × 2.5″ drives, M.2 via internal USB adapter

TRANSFER SPEEDS: HDD 7200 rpm: 150 MBps, SSD 7200 rpm: 280 MBps, Max: 500 MBps

PERIPHERALS: IR receiver, OLED screen

Sleek lines, tailored styling, and impressive build quality hallmarked the original Argon40 case. Now Argon is back with a follow-up, the EON. Having raised over £100,000 on Kickstarter, the project is nearly ready to ship, and we've been lucky enough to road-test a prototype for a couple of months.

The EON is a full network-attached storage (NAS) enclosure for your Raspberry Pi 4. Whereas previous NAS projects often involve a USB hub and cables to external drives, the EON fits up to four drives in an enclosure with a single power source. This allows you to build your own multi-drive storage solution for a fraction of the price of a PC-based equivalent.

▶ Stunning looks, solid build, and lots of space for storage

The first thing that strikes you about the EON is the enclosure itself. No boring black box here – instead we appear to have a pyramidical version of the monolith from *2001: A Space Odyssey*. The frame is built from "space-grade" aluminium, and feels reassuringly heavy. If dropped, we suspect it's the floor that's getting damaged, not the internals. Two magnetically-held opaque plates give a sleek black finish. The rear is solid with plenty of mounting holes for your drives. Raspberry Pi 4 sits at the base of the unit.

Inside is a layered affair with Raspberry Pi 4 at the bottom, then two further PCBs, the top exposing four SATA ports for your drives. These connect via USB 3.0 directly to your Raspberry Pi 4 using a supplied external connector. The SATA ports support four drives, mounted vertically. The unusual shape of the EON means that the two outermost connectors only support 2.5″ drives, but the inner two can be full-size 3.5″ HDD or SSD.

Versatile

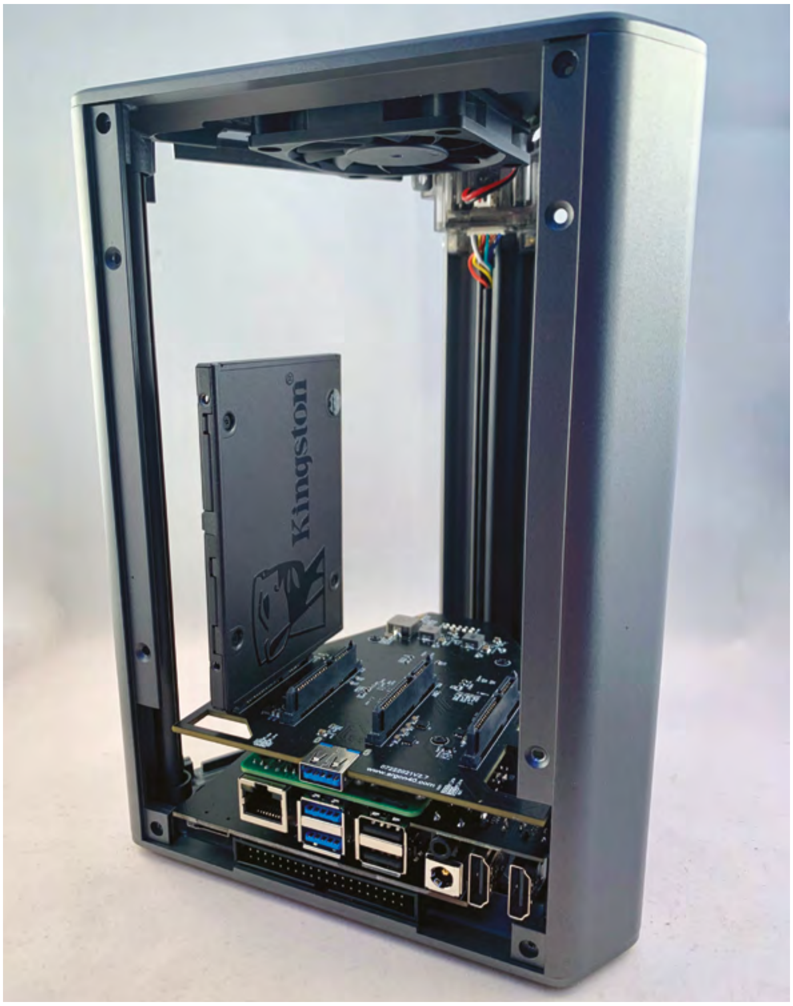
Argon has also baked versatility into the design with a built-IR port that works with the firm's remote control (reviewed last month), perfect for a home theatre setup. A fan provides cooling for the entire unit, and the top-mounted power switch is a small OLED screen that can be customised to display anything, such as IP address or CPU temperature.

Installing drives was straightforward and the supplied screws and tools were a welcome bonus. Our Raspberry Pi 4 immediately recognised the new devices, and soon we had a RAID system up and running. Argon plans to add an internal USB port to support an M.2 SSD, so you can separate the operating system and RAID drives.

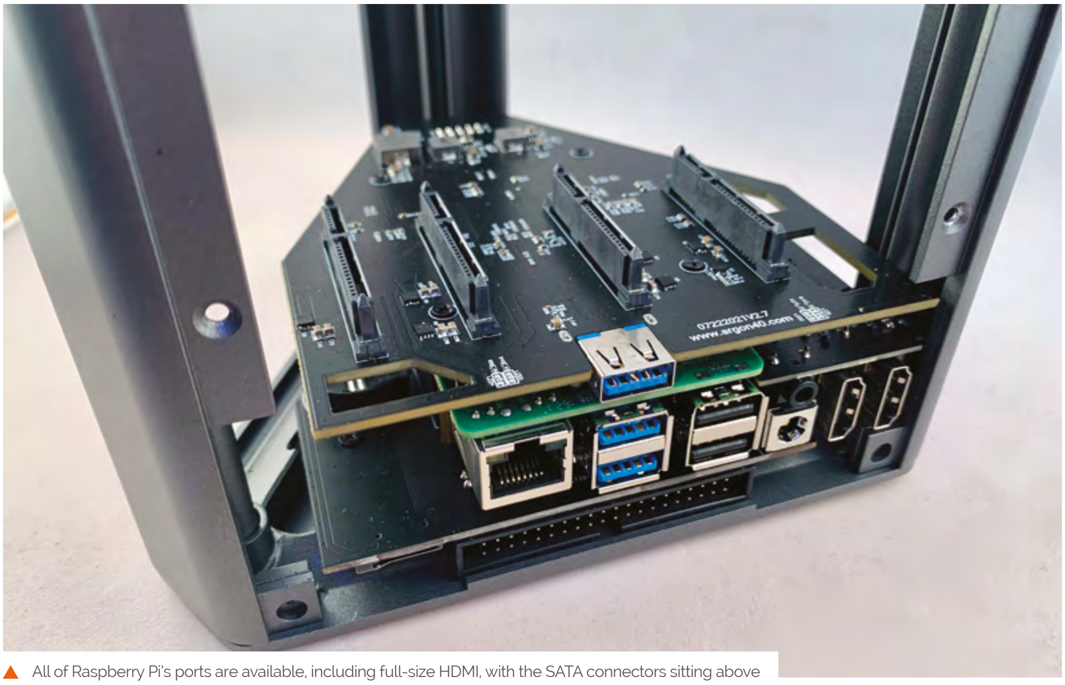
We found it hard to come up with anything negative about this case. The positioning of the power switch makes it very easy to hit it when handling the device, which can punish the impatient when moving things around. Also, despite the rock star looks, the shape of the device does cheat you out of 4 × 3.5˝ drives, which could be seen as form over function.

> “We appear to have a pyramidical version of the monolith from 2001: A Space Odyssey”

These minor nitpicks aside, this is one of the most impressive home NAS setups we've seen. If you want your home server not only capable of huge storage and Raspberry Pi performance, but also as something great to look at, this is for you.



▲ Shown here with a single 2.5˝ SSD drive. Note the adaptive cooling fan at the top



▲ All of Raspberry Pi's ports are available, including full-size HDMI, with the SATA connectors sitting above

Verdict

This is simply a stunning piece of engineering. You can build a NAS that competes with much more expensive options, and it looks like something from a sci-fi movie.

9/10

Trilobot

▶ Pimoroni ▶ **magpi.cc/trilobot** ▶ From £48 / $54

A two-wheeled robot with style and plentiful options for expansion. By **Phil King**

SPECS

FEATURES: 2 × FR-4 PCBs, mounts for ultrasonic distance sensor (supplied) and camera, six-zone RGB underlighting, 4 × push-buttons with status LEDs

MOTORS: 110:1 metal-gear motors with pre-soldered shims; DRV8833PWP motor controller integrated into PCB

EXPANSION: 2 × Qwiic / STEMMA QT ports, 1 × servo header, 1 × I2C header, 5 × Breakout Garden headers (all unpopulated)

At first sight, it may look like most other two-wheeled Raspberry Pi robots, but Trilobot has a lot of neat tricks up its sleeve, including some very cool LED underlighting. Described by Pimoroni as a 'mid-level' robot, it's designed to be simple for newcomers to get started with, while offering plenty of possibilities for adding extra functionality via its ports and numerous unpopulated headers.

The standard Base Kit includes everything you need, apart from a Raspberry Pi 4, microSD card, USB-C power bank, and optional Raspberry Pi Camera Module v2. Note, however, that you could use any full-size Raspberry Pi model and/or power bank by substituting the supplied USB-C cable with one for the connections you need.

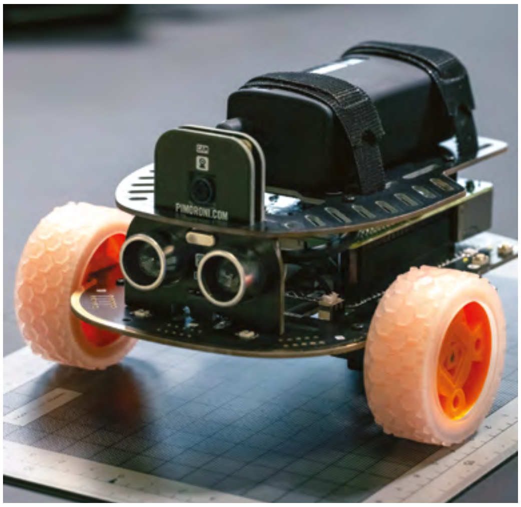
Easy assembly

We found the kit very straightforward to assemble, aided by the well-illustrated step-by-step online guide. The only real difficulty we encountered was getting the tiny nuts into the plastic brackets for the motors – the fit is tight, so we needed to use the blade of a flat-headed screwdriver to push them into place.

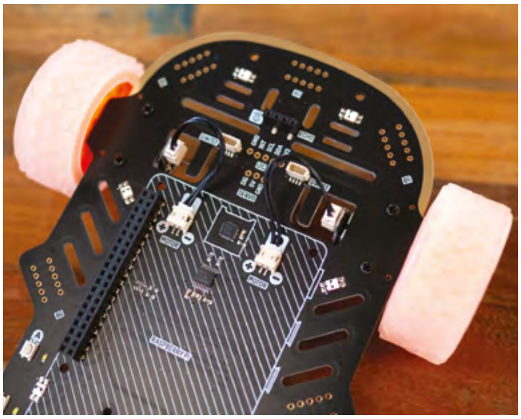
The robot's chassis comprises two FR-4 PCBs. Rather neatly, a DRV8833PWP dual H-bridge motor controller is integrated seamlessly into the main PCB, with two mini JST sockets to connect the supplied short cables to the pre-soldered shims of the metal-gear motors – no soldering or screwing required. Two moon buggy wheels are supplied, along with a standard metal ball castor to be attached to the rear of the board (with an acrylic spacer).

At this point, you can add an optional Raspberry Pi Camera Module v2 using the kit's camera mount before adding the supplied HC-SR04 ultrasonic distance sensor in its own mount in front, with a hole for the camera lens to poke through as the two are sandwiched together. The distance sensor's pins fit into a port on the PCB.

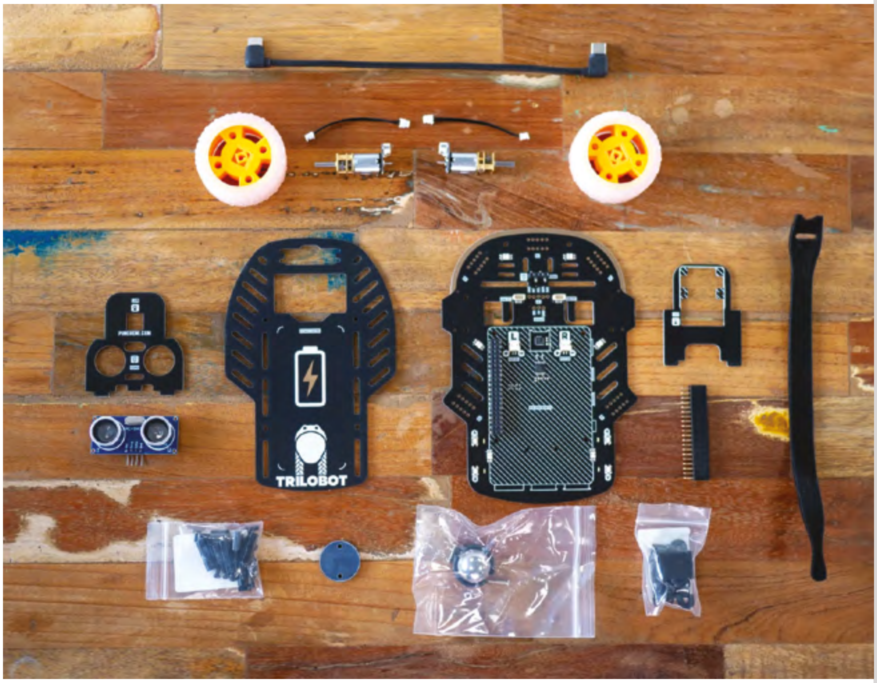
▲ Some spectacular underglow effects are possible using the six RGB LEDs on the underside of the main PCB



▶ The supplied ultrasonic distance sensor and optional Camera Module are mounted at the front of the robot



Wiring up motors has never been easier: just plug each cable into the mini JST connectors on the motor and PCB port



The components supplied in the Trilobot Base Kit. Just add a Raspberry Pi, microSD card, power bank, and Camera Module (optional)

Your Raspberry Pi is then mounted upside down on the main PCB's female GPIO header (via a booster). Once attached using the standoffs, the top PCB serves as a platform for the power bank – ours was a fair bit wider than the one featured in Pimoroni's assembly guide, but the Velcro straps secured it well enough.

Robot running

With the robot assembled, you just need to install the software in Raspberry Pi OS using three Terminal commands. A comprehensive Python library offers a large range of useful functions and comes with several code examples to help you get started. These include some impressive underlighting effects using the six RGB LEDs located on the bottom of the main PCB, reading the four push-buttons located near the rear, and avoiding walls using the distance sensor.

One of the unpopulated Breakout Garden headers is right at the front, although we're not quite sure how practical it is as it's so near to the ultrasonic distance sensor

Another code example enables you to remote-control the Trilobot wirelessly using a Bluetooth Xbox, PlayStation, or 8BitDo gamepad. There aren't yet any examples for using the camera, although it should be fairly straightforward to use it to take photos and stream video; you could even use OpenCV to add object or face detection.

> " Some impressive underlighting effects using the six RGB LEDs located on the bottom of the main PCB "

While it's not the speediest robot we've ever seen, the 110:1 metal-gear motors provide a good level of torque while the moon buggy wheels offer plenty of grip, even on a hard surface.

There are plentiful possibilities for expanding the robot. As well as two Qwiic / STEMMA QT ports, the main PCB has several unpopulated headers: one for a 5V servo (or NeoPixel strips if you prefer), another for I2C, along with five more for adding Pimoroni's Breakout Garden sockets to use the wide range of sensors available. So, for instance, you could add a motion sensor or mini LCD.

Verdict

With a solid chassis, detailed software library, and abundant options for expansion, it all adds up to an excellent robotics platform for beginners and more experienced enthusiasts alike.

9/10

8BitDo Pro 2 Controller

▶ 8BitDo ▶ magpi.cc/pro2 ▶ £40 / $50

Professional-grade video game controller that works a treat with Raspberry Pi. **Lucy Hattersley** flexes her thumbs

SPECS

POWER: 1000 mAh lithium-ion battery or 2 × AA batteries

CONNECTIVITY: USB-C, Bluetooth 4.0

DIMENSIONS: 153.6 × 100.6 × 64.5 mm, 228 g

8BitDo is a company that's been making a name for itself in the retro gaming sphere, supplying quality game controllers and conversion kits at a good price.

The firm recently sent us a box of interesting things to look at, and we decided to start here, with the Pro 2 controller.

Reminiscent of a PlayStation DualShock, the Pro 2 controller has two analogue sticks, a D-pad, four buttons, four triggers, two Pro-level back buttons, Select, Start, Star, Heart, and Profile button. It's certainly not short of a button or two.

If all that wasn't enough, there is an 'SADX' Mode switch underneath that swaps between four different modes: Switch, Apple, Android, and Windows.

It comes with a long USB-C cable and 1000 mAh lithium-ion battery with "20 hours of battery life."

In terms of value, £40 is not particularly low-cost in the world of Raspberry Pi, but it is good value when stacked up against its immediate rivals: a Sony DualShock will cost you £50 and an Xbox One Controller starts at £55 (without a rechargeable battery). So, this is cheaper than either. But is it better?

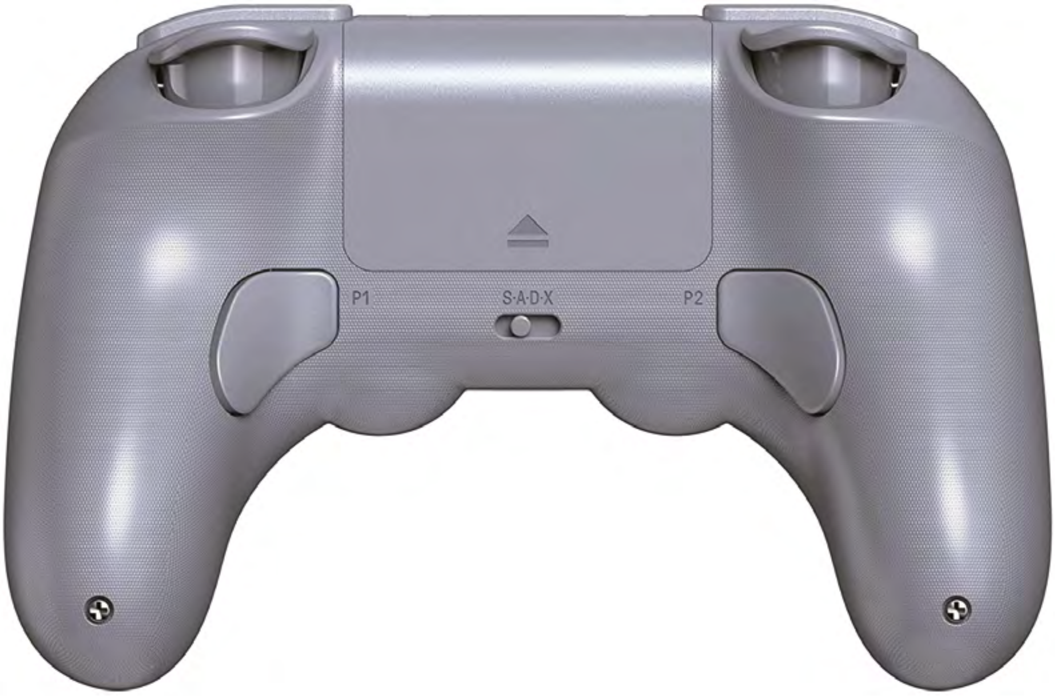
Setup was a breeze. We used the Windows (X) setting on the back and started with a direct USB-C connection. We then held down the Pair button and synced it up with 'Add Device' in the Bluetooth settings in Raspberry Pi OS.

Support in Raspberry Pi OS is game-dependent, although we had a blast in Super Tux Kart and Doom.

We moved onto classic games with Batocera.linux (**batocera.org**) which is a new retro gaming distribution that we'll be talking more about in future. Setup was even easier there, requiring us only to plug in via the USB-C and hold the Pair button. RetroPie was equally easy to set up, mapping the buttons on the controller during the setup process.

For a more modern experience, we tested it out with Xbox Cloud gaming (**xbox.com/play**). This enabled us to use all the analogue sticks and

▼ The Pro 2 controller features all the buttons from a modern console gamepad



The SADX Mode Switch enables you to adjust the controller mapping to fit a range of consoles, devices, and computers

“It integrates neatly with Raspberry Pi”

triggers with some of the latest 3D masterpieces. Again, we had no problems.

Button combinations can be mapped to the two Pro buttons on the rear; sadly, the software to control them is only available for Windows, macOS, Android, and iOS. It’s a shame you can’t do the setup via a web or Linux app.

There are a few quirks. You switch off the controller by holding down the Start button for three seconds, and you might find using Windows (X) not immediately obvious over the other settings. But really there’s nothing here that a read of the supplied instruction manual won’t clear up.

Holding its own

Build quality of the Pro 2 controller is superb. It’s easy to grip and buttons have a nice responsive click with no sponginess. The analogue sticks are weighted well and spring cleanly back to the centre. It’s certainly a step above the usual fare for a third-party controller.

The Pro 2 integrates neatly with Raspberry Pi, and the Mode Switch means you can quickly transfer it to any other consoles or computers that you might be using.

We really have no hesitation in recommending this one.

Two Pro buttons, found underneath the gamepad, can be mapped to button combos using separate (non-Linux) software

Verdict

A fantastic controller for a good price that works across a range of Raspberry Pi games, apps, and distributions. Easy to set up and use. Shame the Ultimate Software isn’t available in Linux, though.

9/10

reTerminal

▶ Seeed Studio ▶ magpi.cc/reterminal ▶ £144 / $195

A tablet-like device powered by a Raspberry Pi Compute Module is aimed more at business than consumers. **Rob Zwetsloot** checks it out

SPECS

DIMENSIONS: 140 mm × 95 mm × 21 mm

STORAGE: 32GB eMMC

DISPLAY: 5" IPS LCD 1280 × 720 touchscreen

SENSORS AND INPUTS: Accelerometer, light sensor, capacitive touchscreen, four programmable buttons, CSI port on rear

CLOCK: In-built RTC (NXP Semiconductors PCF8563T)

At first glance, reTerminal is a sort of small Raspberry Pi tablet computer with an interesting selection of physical buttons and labelled lights. That's what we were fooled into thinking when it landed on the desk, but in actuality it's more of a system display screen, or indeed a terminal much like the name implies. The design that first seemed a little different now makes complete sense, and makes it an even more interesting prospect due to these added features.

When we say small, we mean small – it's 140 mm wide and 95 mm high, putting it in the lower end of the size range for a Raspberry Pi attached to a touchscreen. It's a little bit thick though at 21 mm. However, this is not all because of the internal components – it's not a full Raspberry Pi inside, but a special PCB with a Compute Module 4 installed, giving it the kind of power you would normally see in a Raspberry Pi 4. This makes it much slimmer than what you'd get if it was sitting on a Raspberry Pi 4.

The construction feels solid though, with a tough plastic case that includes plenty of ways to mount itself on walls, and brackets on both the rear and sides of the case, and even includes a standard camera tripod thread on the bottom of the case.

▲ There are several mounting options for reTerminal

Practically minded

The buttons on the front of the case are fully programmable, as well as the STA- and USR-labelled LEDs. Seeed gives some simple instructions on how to program them on their website, as well as simple guides on building graphical interfaces with the touchscreen.

Each side of the machine is used for some of the standard Raspberry Pi connections such as an Ethernet port, a HDMI port, and two USB 2.0. It also includes a full 40-pin GPIO in the same arrangement as any Raspberry Pi since the Raspberry Pi B+ redesign. As well as any extra components you can connect, it will also support standard HATs and an expansion system to add further functionality.

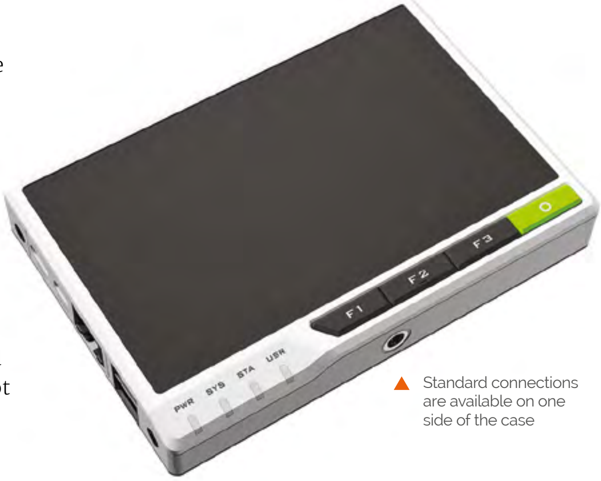
As this is a kind of display that is not built to be portable, it does lack a battery. It's powered by a USB C cable, which could easily be built into wherever it's installed, but also means it can make use of a standard mobile phone charger with a decent amp output. It's not a negative, but it is something that we initially assumed it came with.

> “ As you'd expect from hardware running on Raspberry Pi 4 chips, it runs extremely well ”

The four face buttons have no specific function and are fully programmable

Full power

As you'd expect from hardware running on Raspberry Pi 4 chips, it runs extremely well. The touchscreen is responsive even when running the full OS, and the 1080 × 720 screen is high enough resolution that it doesn't feel too awkward navigating the desktop and such. We've already got a few projects in mind that we want to use it for in a more permanent place, which is always a good thing after a review.



Standard connections are available on one side of the case

Verdict

Smaller than expected but a very cool, solid device for monitoring or controlling larger systems.

10/10

10 Amazing: Upcycling projects

Take your old tech and make something really cool with it

Upcycling is a proud tradition in making circles and, thanks to Raspberry Pi being so tiny, it's very easy to upcycle old tech… or even just antiques. You could even make it your #MonthOfMaking project if you so wish.

▲ Alexa Ruxpin

Smart teddy

Teddy Ruxpin used to tell stories via cassettes. Now, it will answer all your burning questions using Alexa.

magpi.cc/alexaruxpin

◀ iPod Spotify player

Old school streaming

The iPod is a legendary piece of old tech that, unfortunately, can't quite keep up with the modern streaming landscape. Unless, of course, you install a Raspberry Pi Zero W inside it and connect it to Spotify.

magpi.cc/ipodspotify

▲ Game Boy Zero

Improved handheld

Got an old Game Boy lying around? You could play Tetris with it, or completely gut it, drill into it, and reassemble it as a retro gaming powerhouse with Raspberry Pi.

magpi.cc/gbzero

▶ Desktop mirror

Retro cool

Reminiscent of speculative tech that you might see in 1960s television, a broken laptop and a pine frame were cobbled together to create this desktop computer that could easily hide on a vanity.

magpi.cc/desktopmirror

▲ Spotify tapes

Streaming cassettes

What if you could put a special tape in an old tape player and turn it into a Spotify media server? Well, in this case, you'd have to put a Raspberry Pi inside first.

magpi.cc/spotifytape

◀ Magic Dresser

Smart antique

Some might say it's sacrilege to modify an antique dresser's mirror to be a smart mirror. We think it doesn't matter what you do with your own stuff.

magpi.cc/magicdresser

▼ Digital dash

Upcycling automobiles

While we're not 100% sure how legal this is, we're very sure that you should take caution if you plan to replicate this very cool upgrade to your car's dashboard.

magpi.cc/digitaldash

▲ 1975 Hitachi Pi Info-TV

Alternate Pi-niverse

This portable television gets a new lease of life in a universe where it was actually powered by a Raspberry Pi.

magpi.cc/infotv

◀ Vintage film camera

35mm to 35K

Using an old camera housing is a very cool way to make use of a case that it is already fit for purpose, along with some proper lenses, if you can get them to work with a Camera Module.

magpi.cc/vintagecamera

▲ Raspberry Pi boom box

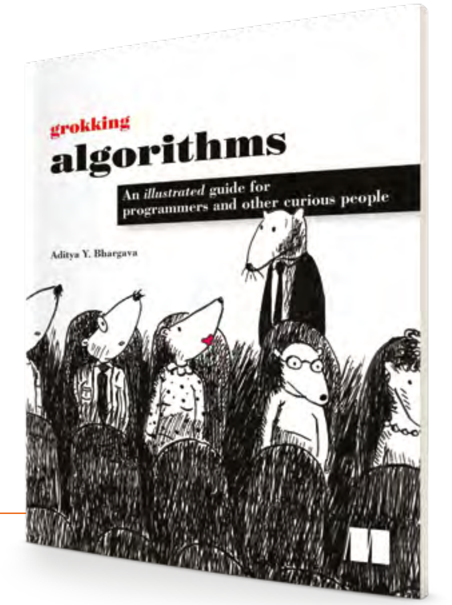
Portable dance party

Instead of streaming Spotify, this upcycled boom box is a portable audio media server using Volumio, but housed in a stylish 1980s facade.

magpi.cc/piboombox

Learn algorithms with Raspberry Pi

Explore the key computing concept of algorithms with these resources. By **Phil King**


grokking
algorithms
An illustrated guide for programmers and other curious people
Aditya Y. Bhargava

Grokking Algorithms

CREATOR Aditya Bhargava

Price: £28 / $45

magpi.cc/grokalgo

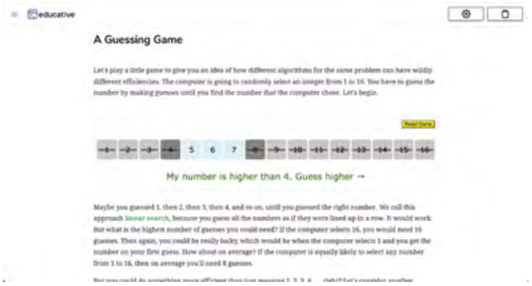
The concept of algorithms predates computers by several centuries. First postulated by 9th century Persian mathematician Muhammed al-Khwarizmi, from whose name the word 'algorithm' is derived, they involve creating a finite sequence of well-defined steps to perform a computation or solve a problem. Algorithms can be defined in written language (e.g. a recipe), flowcharts or, of course, computer programs.

Aimed at those with basic coding skills, *Grokking Algorithms* is an easy-to-follow illustrated guide that teaches you how to apply common algorithms to the sorts of practical problems you may face as a programmer. Starting off with the basics of binary search, recursion, and sorting, it proceeds to cover more complex topics such as data compression and AI. Along the way, it covers topics such as problem-solving techniques, hash tables, graph algorithms, K-nearest neighbours (KNN), and dynamic programming.

Each algorithm example is accompanied by diagrams and fully annotated Python code samples that can be downloaded.

Free online courses

Learn about algorithms with these free web courses



A Guessing Game

Let's play a little game to give you an idea of how different algorithms for the same problem can have wildly different efficiencies. The computer is going to randomly select an integer from 1 to 16. You have to guess the number by making guesses until you find the number that the computer chose. Let's begin.

~~1~~ ~~2~~ ~~3~~ ~~4~~ 5 6 7 ~~8~~ ~~9~~ ~~10~~ ~~11~~ ~~12~~ ~~13~~ ~~14~~ ~~15~~ ~~16~~

My number is higher than 4. Guess higher →

Maybe you guessed 1, then 2, then 3, then 4, and so on, until you guessed the right number. We call this approach linear search, because you guess all the numbers as if they were lined up in a row. It would work. But what is the highest number of guesses you could need? If the computer selects 16, you would need 16 guesses. Then again, you could be really lucky, which would be when the computer selects 1 and you get the number on your first guess. How about on average? If the computer is equally likely to select any number from 1 to 16, then on average you'll need 8 guesses.

But you could do something more efficient than just guessing 1, 2, 3, 4, … right? Let's consider another

A VISUAL INTRODUCTION TO ALGORITHMS

Packed with diagrams and illustrations, Educative's course takes around 14 hours to complete. Topics covered include binary search, asymptotic analysis, recursion, and a variety of sorting methods.

▶ magpi.cc/visualalgo

INTRODUCTION TO ALGORITHMS AND DATA STRUCTURES IN C++

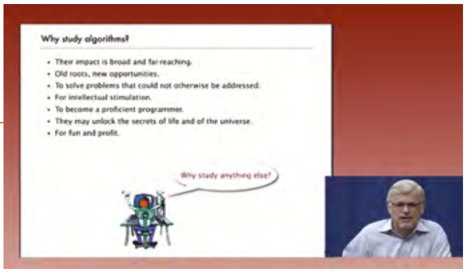
This Udemy video-based course is aimed at C++ programmers. Using plenty of diagrams, Andrei Margeloiu takes you through fundamental data structures and explains practical techniques to solve algorithmic problems.

▶ magpi.cc/udemyalgoc

DATA STRUCTURE – PART I

This is one of the best courses aimed at complete beginners in the field. Using Java code examples, it explores dynamic arrays, linked lists, stacks, queues, and tree and time complexity analysis.

▶ magpi.cc/udemydata



Why study algorithms?

- Their impact is broad and far-reaching.
- Old roots, new opportunities.
- To solve problems that could not otherwise be addressed.
- For intellectual stimulation.
- To become a proficient programmer.
- They may unlock the secrets of life and of the universe.
- For fun and profit.

Algorithms Part 1 & 2

CREATOR

Coursera

Price:
Free

magpi.cc/algocoursera

Created by Kevin Wayne and Robert Sedgewick from Princeton University, this two-part, intermediate-level survey course is available for free on Coursera. Covering essential information on algorithms and data structures for programmers, it has an emphasis on applications and scientific performance analysis of Java implementations. Part I covers elementary data structures, sorting, and searching algorithms, while part II focuses on graph- and string-processing algorithms.

Starting off with the union-find data type and analysis of algorithm performance, Part I explores elementary sorting, the merge-sort algorithm, priority queues, balanced search trees, and hash tables. Part II tackles undirected and directed graphs, shortest paths, substring search, and regular expressions. Modules comprise videos, written materials, and quizzes to test your new-found knowledge.

The Secret Rules of Modern Living: Algorithms

CREATOR

BBC

Price:
Free / $2.99

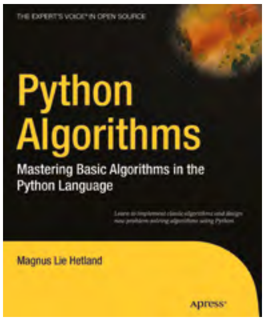
magpi.cc/algodocu

For those wanting a simple video-based introduction to algorithms and what they can do, this BBC TV documentary – available via Amazon Prime Video / IMDb TV – is ideal. Presenter and mathematician Marcus du Sautoy demystifies the concept, revealing the origins of algorithms, how they work, and how they can now even program themselves (i.e. machine learning).

What can be a dry subject in textbooks is brought alive in an engaging way. For instance, it explains a web page ranking system by looking at the number of passes (i.e. links) between soccer players (sites). Other topics covered include Euclid's algorithm (greatest common divisor), bubble and merge sorting, the Gale-Shapley algorithm (for stable matching), and the travelling salesman problem (shortest possible route).

Reading material

Books for exploring algorithms



THE EXPERT'S VOICE® IN OPEN SOURCE
Python Algorithms
Mastering Basic Algorithms in the Python Language
Magnus Lie Hetland
Apress®

PYTHON ALGORITHMS

By *Beginning Python* author Magnus Lie Hetland, this 352-page book gives a good overview of implementing algorithms in Python. It explores various types and each chapter ends with problems to solve.

▶ magpi.cc/pythonalgo

ALGORITHM DESIGN MANUAL

This 748-page tome is a comprehensive reference guide to computer algorithms. The first part covers methods for designing and analysing them, while the second is a catalogue of algorithmic resources and implementations.

▶ magpi.cc/algodesign

INTRODUCTION TO ALGORITHMS

MIT Press's classic guide is considered the 'bible' of algorithms. At over 1000 pages, it isn't the lightest read, but is ideal for those who want a comprehensive reference book to dip into during a course.

▶ magpi.cc/introalgo

Liz Upton

The head of community management at Raspberry Pi has been there since before the start

> Name **Liz Upton** | > Occupation **Chief Marketing and Communications Officer**
> Community role **Community leader** | > Twitter **@Raspberry_Pi**

If you've read a blog on a Raspberry Pi website, chances are it was written by Liz. If you've replied to a Twitter post from **@Raspberry_Pi**, chances are that Liz wrote it. For several years, at the start of Raspberry Pi as a computer and charitable Foundation, Liz Upton was the entirety of the team relaying official Raspberry Pi news. While she doesn't have to do everything any more, her influence is still there, despite that never really being the plan.

"I was a freelance copywriter and editor, working on food, travel, and fragrance beats; it was a lovely job," Liz explains. "I got to review the maiden voyages of cruise ships, travel around the world eating weird stuff, and feel accomplished (I was the main collaborator on a bestselling National Geographic book on food travel, and won a bunch of awards).

"I'd come on board to help out with Raspberry Pi on the understanding that I'd get press and community on an even keel, then hand off to someone else so I could go back to swanning about in nice hotels and writing about eating gelatinous cubes in Japan. Ten years later, I'm still here running our now-quite-hefty marketing and communications department, and I've got absolutely no intention of going anywhere else!"

▼ You'll find Liz sometimes at local Raspberry Jams

From your position, how did you think the original Raspberry Pi release would go?
We knew it'd be a success; it was pretty clear that there was a really engaged community out there, and I spent a lot of my time talking to those people in person at Maker Faires and other events, and online. The degree of success was completely unexpected. There's a concept called the Success Disaster, and honestly, we only just skirted around one of those; on day

▲ The original stock of Raspberry Pi seems quite paltry compared to the tens of millions that have sold since

one we had 2000 Raspberry Pis on a pallet and enough parts to make another 10,000 or so; but we got orders for more than 100,000 just on that first day. At one point on that day, we were getting 700 orders every second. I will admit to several moments of blind panic. Thankfully, we had some great manufacturing partners and a community that responded really well when we explained exactly what our pipelining issues looked like.

> To have got here and to still be steaming ahead still seems extraordinary to me

Even after that, I never imagined we'd be where we are today. By the end of last year we'd sold 45 million Raspberry Pis; I remember the day we sold a million and being blown away. To have got here and to still be steaming ahead still seems extraordinary to me.

What are your favourite things you've made with Raspberry Pi?

There are two different categories for me: the practical (the Pi-hole in the cupboard with the router and the NAS box) which I use every day; and the fun stuff. My most fun project was stupidly simple, but made me very happy: it's a big hand-painted box with arcade buttons on top that I made for my son so he could press them to play his favourite tunes. His big sister has a RetroPie setup that I painted too.

What early Raspberry Pi projects really 'wowed' you?

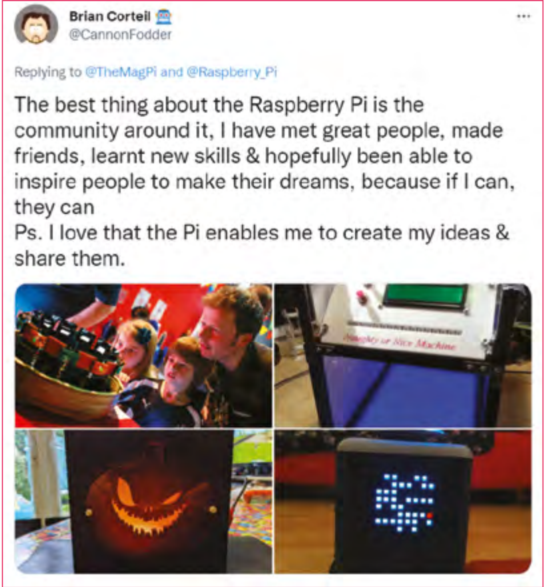
Oof: that's a hard one – there are so many. The earliest 'wow'? A couple of months after we launched, a fella called Dave Akerman started using Raspberry Pis to send live pictures and GPS data from hydrogen balloons he was launching into near space. He invited me and Eben to come to a launch, and it was just the best fun: watching these beautiful images of the curvature of the Earth and the darkness of space coming in in real time, then going on an off-road drive and hike to find the remains of the balloon (and the Raspberry Pi!) once it had popped at altitude. Dave's a superstar; I should catch up with him soon!

▲ These high-altitude balloons started a long tradition of Raspberry Pi computers going higher and higher

This Decade in Raspberry Pi

Ten years of Raspberry Pi memories from the community

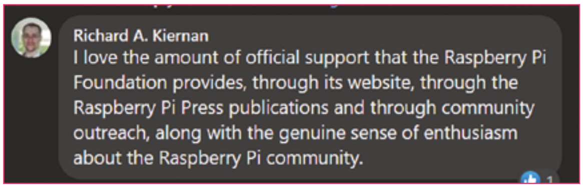
This March issue of *The MagPi* is released on 24 February 2022, a few days before whatever could be considered the official birthday of Raspberry Pi – it's a 29 February birthday, and 2022 is not a leap year. As well as our interview with Liz, we wanted to see what the community had to say about ten years of Raspberry Pi. We had a huge response – here's some of our favourites!



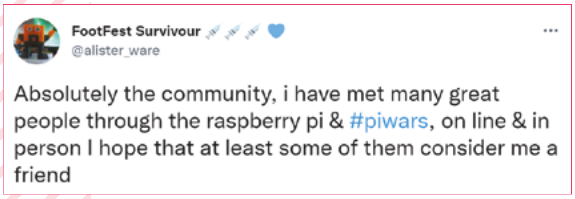
Brian Corteil 🤖
@CannonFodder

Replying to @TheMagPi and @Raspberry_Pi

The best thing about the Raspberry Pi is the community around it, I have met great people, made friends, learnt new skills & hopefully been able to inspire people to make their dreams, because if I can, they can
Ps. I love that the Pi enables me to create my ideas & share them.

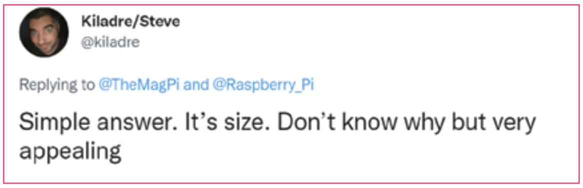


Richard A. Kiernan
I love the amount of official support that the Raspberry Pi Foundation provides, through its website, through the Raspberry Pi Press publications and through community outreach, along with the genuine sense of enthusiasm about the Raspberry Pi community.



FootFest Survivour 💉💉💉 💙
@alister_ware

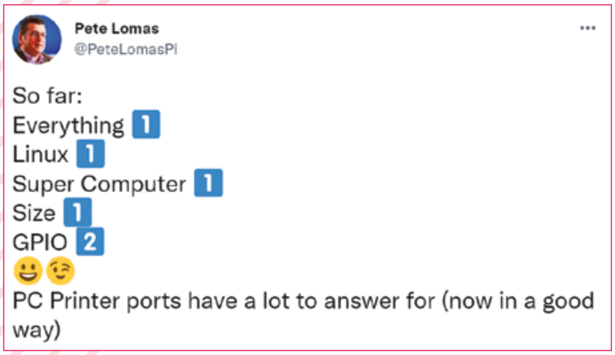
Absolutely the community, i have met many great people through the raspberry pi & #piwars, on line & in person I hope that at least some of them consider me a friend



Kiladre/Steve
@kiladre

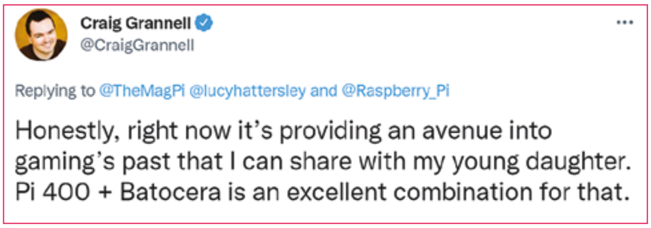
Replying to @TheMagPi and @Raspberry_Pi

Simple answer. It's size. Don't know why but very appealing



Pete Lomas
@PeteLomasPI

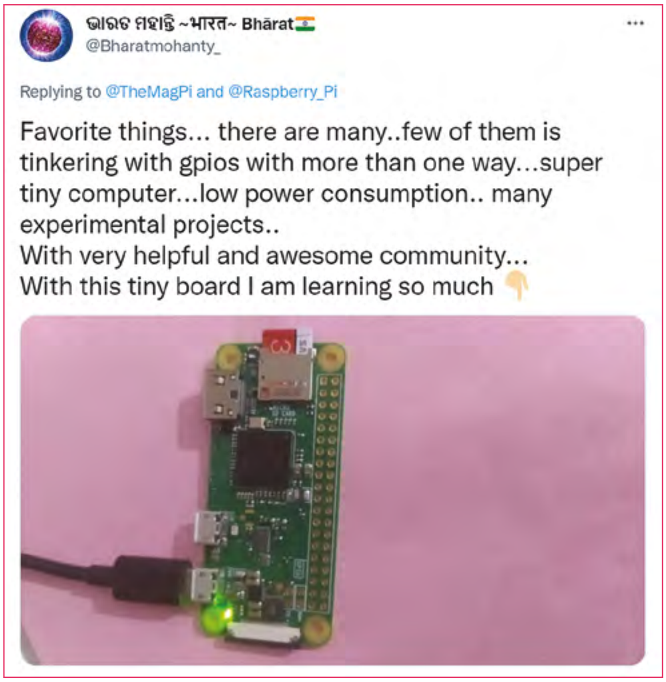
So far:
Everything 1️⃣
Linux 1️⃣
Super Computer 1️⃣
Size 1️⃣
GPIO 2️⃣
😀😉
PC Printer ports have a lot to answer for (now in a good way)



Craig Grannell ✓
@CraigGrannell

Replying to @TheMagPi @lucyhattersley and @Raspberry_Pi

Honestly, right now it's providing an avenue into gaming's past that I can share with my young daughter. Pi 400 + Batocera is an excellent combination for that.



ଭାରତ ମହାନ୍ତି ~भारत~ Bhārat🇮🇳
@Bharatmohanty_

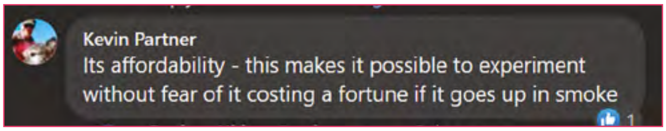
Replying to @TheMagPi and @Raspberry_Pi

Favorite things... there are many..few of them is tinkering with gpios with more than one way...super tiny computer...low power consumption.. many experimental projects..
With very helpful and awesome community...
With this tiny board I am learning so much 👇

Neil Hopkins
@thermalsatsuma

Replying to @TheMagPi and @Raspberry_Pi

I used in it my Open University degree final project to demonstrate genetic algorithms learning to play games (and got a first!). Thanks Raspberry Pi!

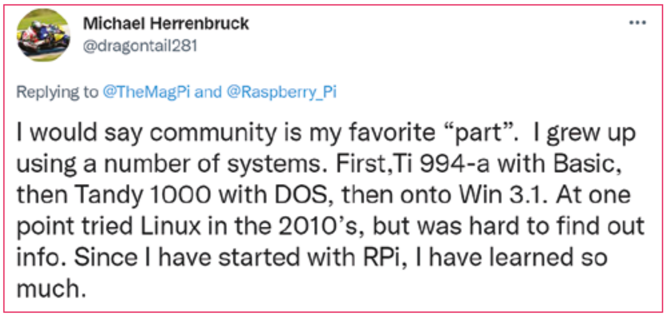


Kevin Partner
Its affordability - this makes it possible to experiment without fear of it costing a fortune if it goes up in smoke

Aula-J
@Aula_J2018

Replying to @TheMagPi and @Raspberry_Pi

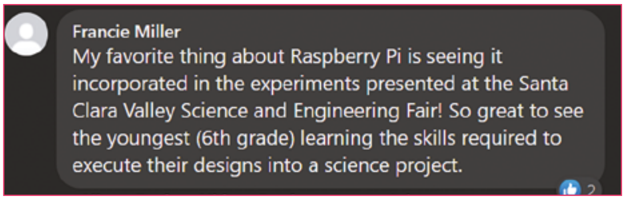
Raspberry Pi boards are great and there are new releases & improvements all the time, faster than expected.
I like using it in my projects & educational subjects because it makes learning more useful & fun and thanks to it I learned many new concepts and met



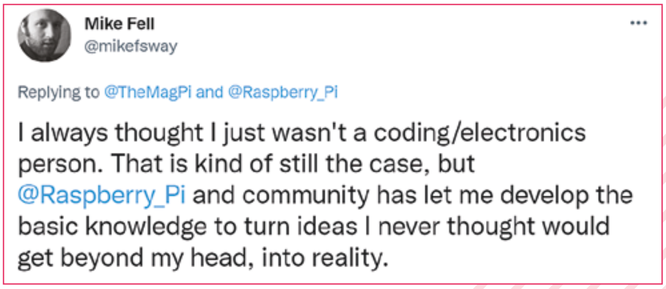
Michael Herrenbruck
@dragontail281

Replying to @TheMagPi and @Raspberry_Pi

I would say community is my favorite “part”. I grew up using a number of systems. First,Ti 994-a with Basic, then Tandy 1000 with DOS, then onto Win 3.1. At one point tried Linux in the 2010’s, but was hard to find out info. Since I have started with RPi, I have learned so much.



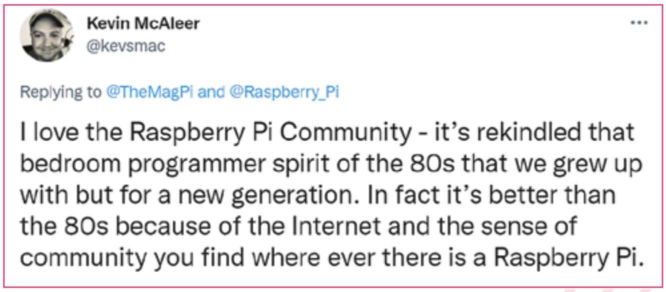
Francie Miller
My favorite thing about Raspberry Pi is seeing it incorporated in the experiments presented at the Santa Clara Valley Science and Engineering Fair! So great to see the youngest (6th grade) learning the skills required to execute their designs into a science project.



Mike Fell
@mikefsway

Replying to @TheMagPi and @Raspberry_Pi

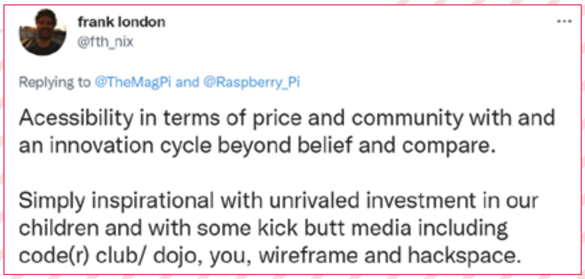
I always thought I just wasn't a coding/electronics person. That is kind of still the case, but @Raspberry_Pi and community has let me develop the basic knowledge to turn ideas I never thought would get beyond my head, into reality.



Kevin McAleer
@kevsmac

Replying to @TheMagPi and @Raspberry_Pi

I love the Raspberry Pi Community - it’s rekindled that bedroom programmer spirit of the 80s that we grew up with but for a new generation. In fact it’s better than the 80s because of the Internet and the sense of community you find where ever there is a Raspberry Pi.



frank london
@fth_nix

Replying to @TheMagPi and @Raspberry_Pi

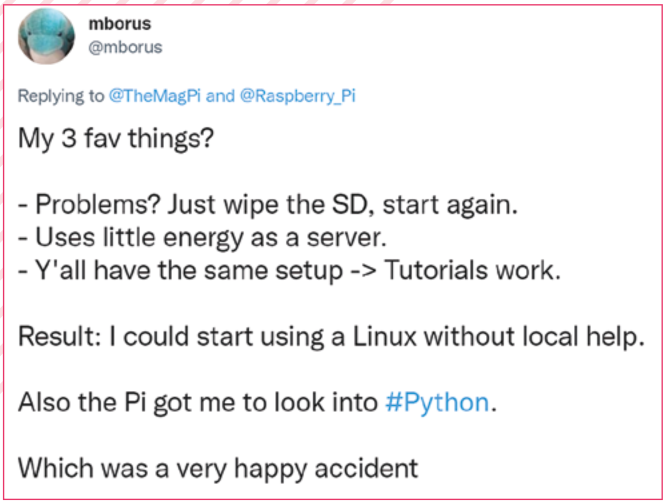
Acessibility in terms of price and community with and an innovation cycle beyond belief and compare.

Simply inspirational with unrivaled investment in our children and with some kick butt media including code(r) club/ dojo, you, wireframe and hackspace.

Scott Young
@thebotmakes

Replying to @TheMagPi and @Raspberry_Pi

What I love about the community is the broad mix of interests and abilities all brought together by the Pi, all willing to help out and cheer people on and share projects and code in the hope someone else might find it useful in building their own ideas. It’s really inspiring.



mborus
@mborus

Replying to @TheMagPi and @Raspberry_Pi

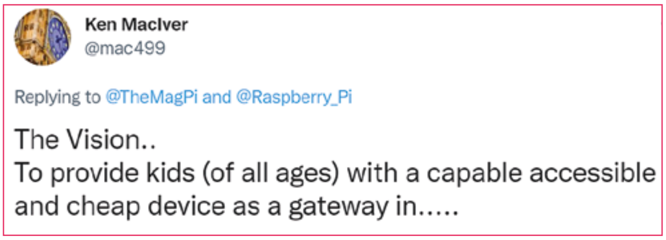
My 3 fav things?

- Problems? Just wipe the SD, start again.
- Uses little energy as a server.
- Y'all have the same setup -> Tutorials work.

Result: I could start using a Linux without local help.

Also the Pi got me to look into #Python.

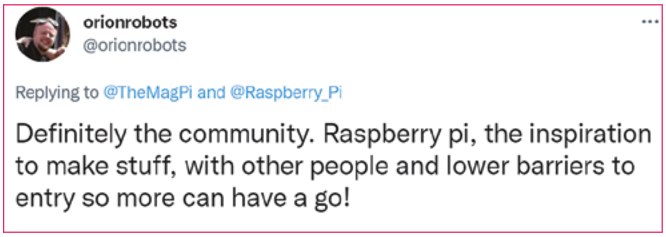
Which was a very happy accident



Ken MacIver
@mac499

Replying to @TheMagPi and @Raspberry_Pi

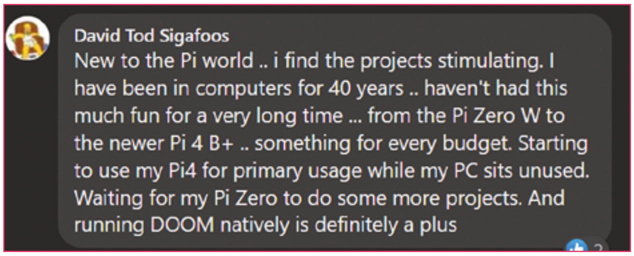
The Vision..
To provide kids (of all ages) with a capable accessible and cheap device as a gateway in.....



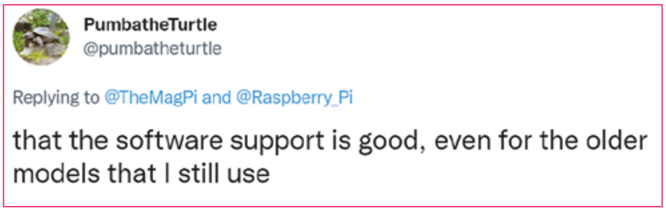
orionrobots
@orionrobots

Replying to @TheMagPi and @Raspberry_Pi

Definitely the community. Raspberry pi, the inspiration to make stuff, with other people and lower barriers to entry so more can have a go!



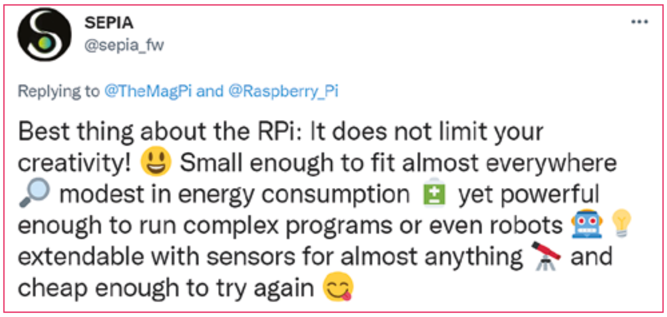
David Tod Sigafoos
New to the Pi world .. i find the projects stimulating. I have been in computers for 40 years .. haven't had this much fun for a very long time ... from the Pi Zero W to the newer Pi 4 B+ .. something for every budget. Starting to use my Pi4 for primary usage while my PC sits unused. Waiting for my Pi Zero to do some more projects. And running DOOM natively is definitely a plus



PumbatheTurtle
@pumbatheturtle

Replying to @TheMagPi and @Raspberry_Pi

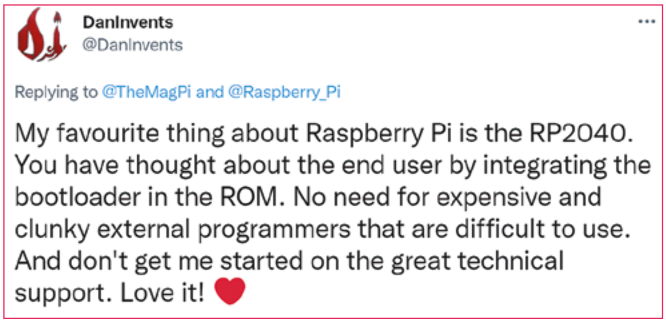
that the software support is good, even for the older models that I still use



SEPIA
@sepia_fw

Replying to @TheMagPi and @Raspberry_Pi

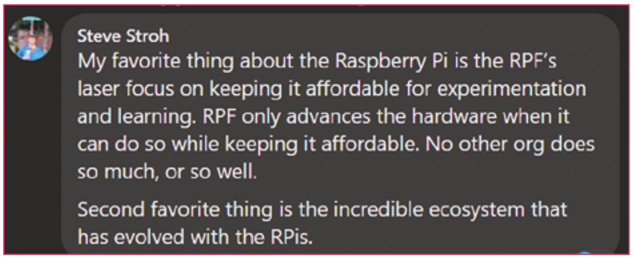
Best thing about the RPi: It does not limit your creativity! 😃 Small enough to fit almost everywhere 🔎 modest in energy consumption 🔋 yet powerful enough to run complex programs or even robots 🤖💡 extendable with sensors for almost anything 🔭 and cheap enough to try again 😋



DanInvents
@DanInvents

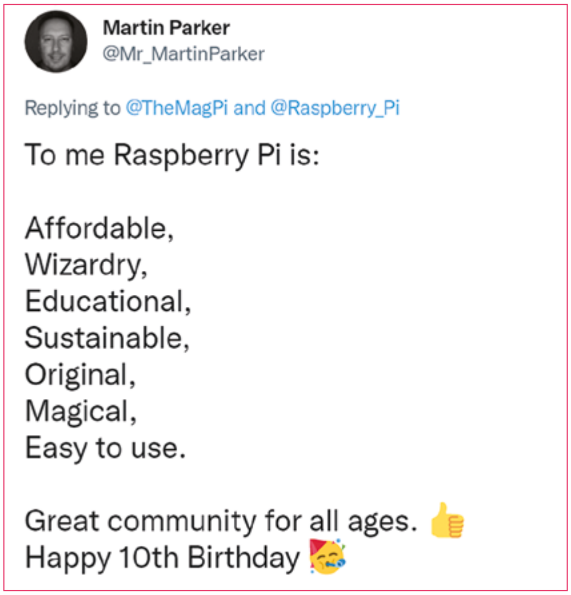
Replying to @TheMagPi and @Raspberry_Pi

My favourite thing about Raspberry Pi is the RP2040. You have thought about the end user by integrating the bootloader in the ROM. No need for expensive and clunky external programmers that are difficult to use. And don't get me started on the great technical support. Love it! ❤️



Steve Stroh
My favorite thing about the Raspberry Pi is the RPF's laser focus on keeping it affordable for experimentation and learning. RPF only advances the hardware when it can do so while keeping it affordable. No other org does so much, or so well.

Second favorite thing is the incredible ecosystem that has evolved with the RPis.



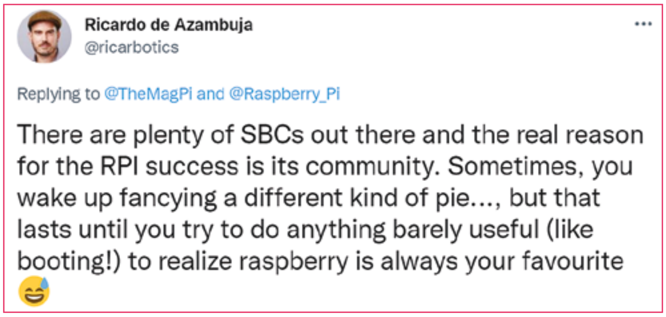
Martin Parker
@Mr_MartinParker

Replying to @TheMagPi and @Raspberry_Pi

To me Raspberry Pi is:

Affordable,
Wizardry,
Educational,
Sustainable,
Original,
Magical,
Easy to use.

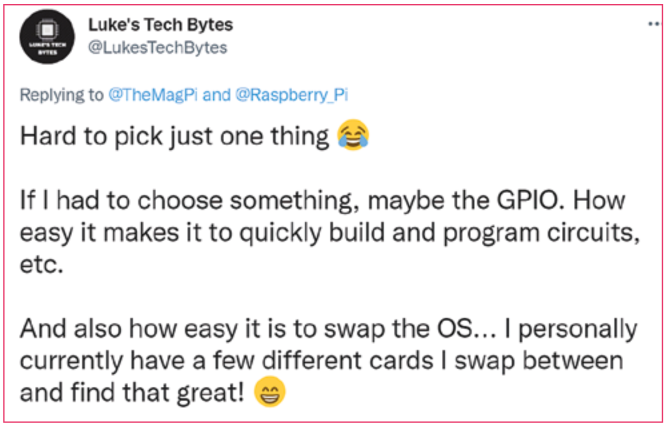
Great community for all ages. 👍
Happy 10th Birthday 🥳



Ricardo de Azambuja
@ricarbotics

Replying to @TheMagPi and @Raspberry_Pi

There are plenty of SBCs out there and the real reason for the RPI success is its community. Sometimes, you wake up fancying a different kind of pie..., but that lasts until you try to do anything barely useful (like booting!) to realize raspberry is always your favourite 😅



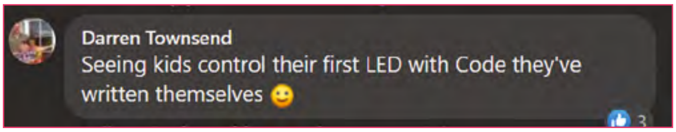
Luke's Tech Bytes
@LukesTechBytes

Replying to @TheMagPi and @Raspberry_Pi

Hard to pick just one thing 😂

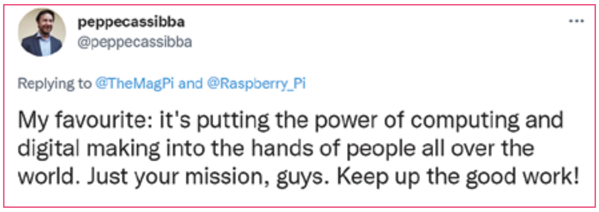
If I had to choose something, maybe the GPIO. How easy it makes it to quickly build and program circuits, etc.

And also how easy it is to swap the OS… I personally currently have a few different cards I swap between and find that great! 😁



Darren Townsend
Seeing kids control their first LED with Code they've written themselves 😊

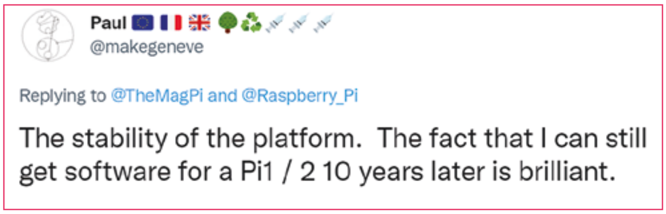
👍 3



peppecassibba
@peppecassibba

Replying to @TheMagPi and @Raspberry_Pi

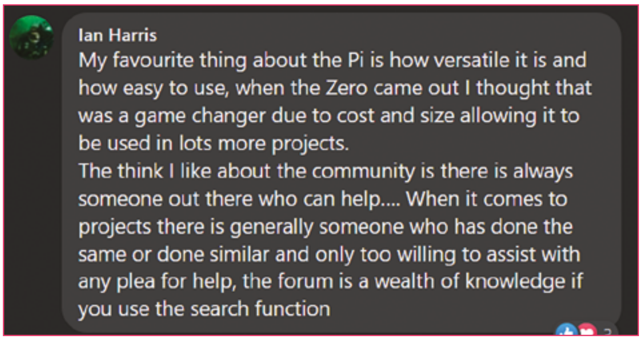
My favourite: it's putting the power of computing and digital making into the hands of people all over the world. Just your mission, guys. Keep up the good work!



Paul 🇪🇺🇫🇷🇬🇧🌳♻️💉💉💉
@makegeneve

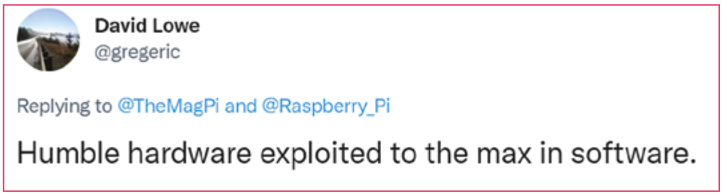
Replying to @TheMagPi and @Raspberry_Pi

The stability of the platform. The fact that I can still get software for a Pi1 / 2 10 years later is brilliant.



Ian Harris
My favourite thing about the Pi is how versatile it is and how easy to use, when the Zero came out I thought that was a game changer due to cost and size allowing it to be used in lots more projects.
The think I like about the community is there is always someone out there who can help.... When it comes to projects there is generally someone who has done the same or done similar and only too willing to assist with any plea for help, the forum is a wealth of knowledge if you use the search function

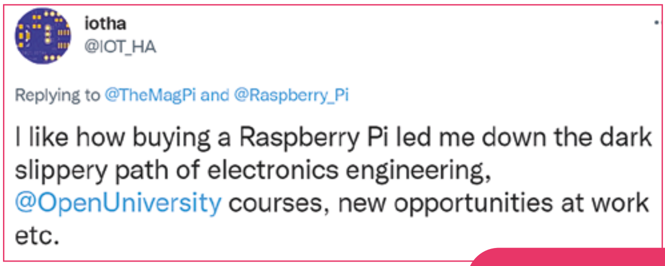
2



David Lowe
@gregeric

Replying to @TheMagPi and @Raspberry_Pi

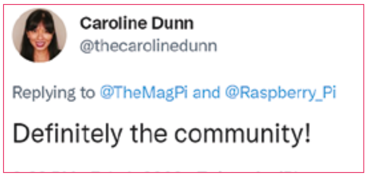
Humble hardware exploited to the max in software.



iotha
@IOT_HA

Replying to @TheMagPi and @Raspberry_Pi

I like how buying a Raspberry Pi led me down the dark slippery path of electronics engineering, @OpenUniversity courses, new opportunities at work etc.

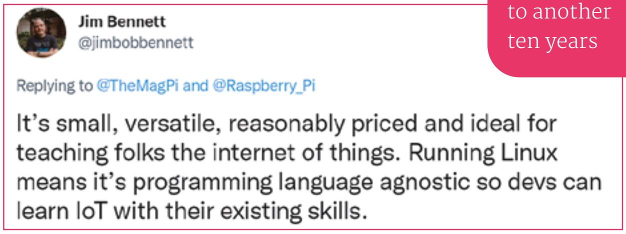


Caroline Dunn
@thecarolinedunn

Replying to @TheMagPi and @Raspberry_Pi

Definitely the community!

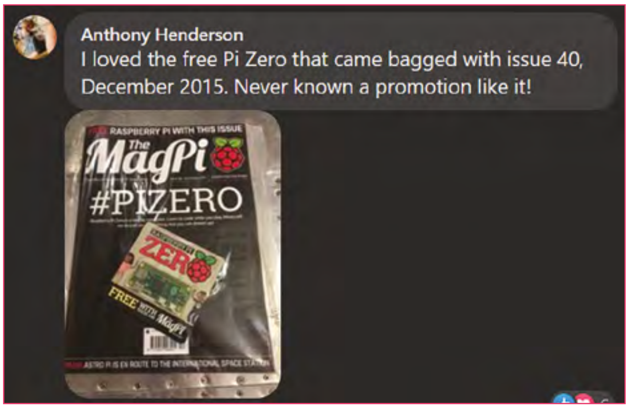
Thank you to everyone in the community that supports Raspberry Pi and *The MagPi*! Here's to another ten years



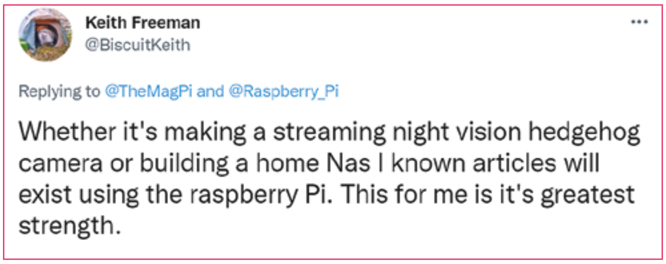
Jim Bennett
@jimbobbennett

Replying to @TheMagPi and @Raspberry_Pi

It's small, versatile, reasonably priced and ideal for teaching folks the internet of things. Running Linux means it's programming language agnostic so devs can learn IoT with their existing skills.



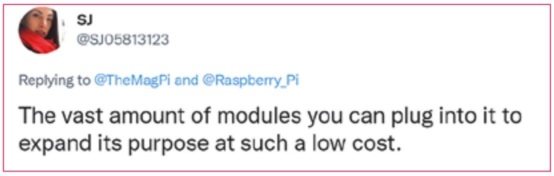
Anthony Henderson
I loved the free Pi Zero that came bagged with issue 40, December 2015. Never known a promotion like it!



Keith Freeman
@BiscuitKeith

Replying to @TheMagPi and @Raspberry_Pi

Whether it's making a streaming night vision hedgehog camera or building a home Nas I known articles will exist using the raspberry Pi. This for me is it's greatest strength.



SJ
@SJ05813123

Replying to @TheMagPi and @Raspberry_Pi

The vast amount of modules you can plug into it to expand its purpose at such a low cost.

Your Letters

OBD2 is a data connection on modern cars that allows you to get info from the engine

Carberry Pi

I am a research master in computer engineering. My project is to design a vehicle tracking system. I want to get the vehicle speed through the OBD2 device.

I want to connect the OBD2 with Raspberry Pi and read the vehicle's engine information on a Raspberry Pi display.

If you can help me I will be grateful and, if you can't, please point me to someone who can help me.

Saba via Facebook

Way back in issue 81 (**magpi.cc/81**), we did a feature on how to build your own car computer which connected to OBD2 stuff in a very similar way. There are several good ones you can find online too – we like this one from Instructables: **magpi.cc/obdpi**.

Be careful when doing such modifications to a vehicle though, and make sure anything you add does not obstruct your view.

Store updates

Will the London Raspberry Pi store come back? I unfortunately missed it the first time and would love to visit!

Chris via Twitter

If you'd asked us any other time we would not have been able to answer because well, we didn't know. However, three pop up stores have been announced for 2022. As well as the Cambridge shop staying where it is, you'll be able to visit pop-ups here:

Metro Centre, Newcastle –
Saturday 28 and Sunday 29 May

St James Quarter, Edinburgh –
Friday 29 and Saturday 30 July

Oxford Street, London –
Friday 29 and Saturday 30 October

You can only buy certain merch at these Raspberry Pi store pop ups

Printer problems

As a retirement project, I have acquired a Raspberry Pi 400 and subscribed to your excellent magazine. Though I have been a long-term user of PCs, iMacs, and iPads, I consider myself a beginner with regard to Raspberry Pi and Linux.

The documentation published by Raspberry Pi for beginners is very helpful and enabled me to set up my Raspberry Pi4 00 ready for use.

I have come to dead-end because I cannot find any formal documentation relating to WiFi printer set-up. My printer is a Canon Pixma 475 which works with my PC and iPads, and was very easy to configure. All I would like to do is link this printer to my Raspberry Pi 400 as well. Your advice would be much appreciated.

I have tried to solve the problem by referring to the forum pages of the Raspberry Pi website, but found them a bit too techie, though I deduced that printing with Linux was never simple without using CUPS, and even that required some customisation.

Richard via email

Printer connections is the closest thing to black magic that we have in technology circles. You are correct though, it is a bit of a known weakness in Linux to get a printer working, however huge strides have been made in the last decade or so.

Unfortunately, due to the wide array of printers and printer drivers, there may not always be a simple answer. However, Raspberry Pi OS is based on Debian, which is also the base for Ubuntu. We suggest looking for ways to get your specific printer working on those Linux distributions as well and it might give you a solution for Raspberry Pi.

Contact us!

› Twitter	@TheMagPi
› Facebook	magpi.cc/facebook
› Email	magpi@raspberrypi.com
› Online	forums.raspberrypi.com

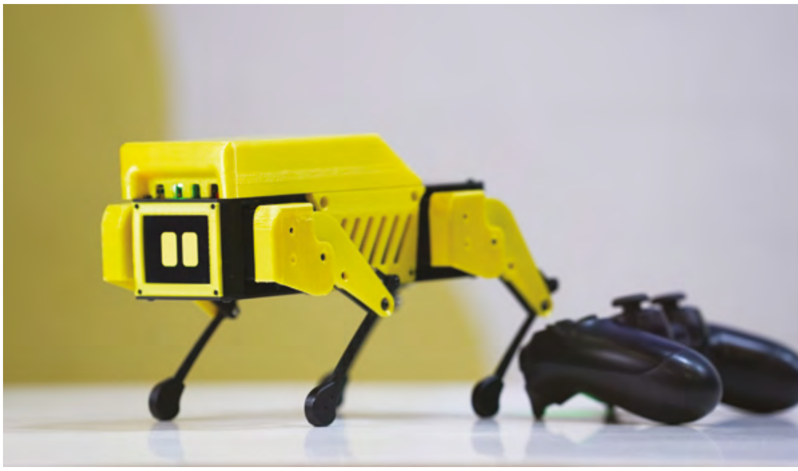
YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
SERIOUS FUN WITH ELECTRONICS
NEW USER GUIDE
START CODING WITH SCRATCH
BUILD A BINARY CLOCK
TOP 10 UPCYCLING PROJECTS
WEAR RASPBERRY PI SMART GLASSES
BUILD A TIME MACHINE RADIO
WIN! MINI PUPPER ROBOT DOG KIT!
TOP 10 LEGO PROJECTS
GET STARTED

Wireframe
LIFTING THE LID ON VIDEO GAMES
ALL FORMATS
116 PAGES MONTHLY
HARD MODE
Challenging traditional approaches to game difficulty
DARK SOULS
Make a messaging system in Unreal Engine 4
Issue 59 £6
wfmag.cc
FACE THE FUTURE
With Epic's revolutionary MetaHuman Creator
2022's MOST ANTICIPATED INDIE GAMES
DESIGNING DIGITAL CARD GAMES
The hit skate
RAINBOW ISLANDS
Make your own retro platformer in Python
NALS

WIN A MINI PUPPER

IN ASSOCIATION WITH MANGDANG

Successful Kickstarter Mini Pupper is an open-source, dog-like robot that can hop and trot around, show off expressions, and is designed with computer vision applications in mind.



Head here to enter: **magpi.cc/win** | **Learn more:** **www.mangdang.net**

Terms & Conditions

Competition opens on **23 February 2022** and closes on **31 March 2022**. 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 or Facebook.

ROBOT UP FOR GRABS
Monitor your plants
YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
The official Raspberry Pi magazine
BULLSEYE! RASPBERRY PI OS FAQ
RASPBERRY PI NEW USER GUIDE 2022
Installing the OS
Discover new features
Starter electronics
START CODING WITH SCRATCH
BUILD A BINARY CLOCK
RASPBERRY PI GAMING SPECIAL!
TOP 10 LEGO PROJECTS
MAKE & BUILD WITH ZERO 2
RASPBERRY PI CHRISTMAS PROJECTS
OVERCLOCK RASPBERRY PI ZERO 2 W
GET STARTED WITH THE BUILD HAT
Better & faster
Engineering interview
Detailed specs
Project ideas
Explore the world of sensors!
Experiment with a different operating system
BUILD A RASPBERRY PI PICO STREAM DECK
USE GPIO PINS
WIN! AIR QUALITY KITS UP FOR GRABS!
42 PAGES

YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
The official Raspberry Pi magazine
Develop games with Scratch & Python
Pi-top robot tested!
ROBOT V2 UP FOR GRABS!
RASPBERRY PI BIG BUILDS
PLUS! Summer Projects
Gigantic projects to make in your garage
Top 10 Media Players
Build a burglar alarm
Discover an amazing Drinks dispenser
50 PAGES OF PROJECTS & TUTORIALS
HOME OF THE FUTURE
Automate your home with Raspberry Pi
Upcycle iPod Classic with Spotify
Build an arcade machine
48 PAGES OF PROJECTS & TUTORIALS
RASPBERRY PI 50 HACKS & HINTS
Code a Pico reaction game
Discover amazing portable projects
SECRETS THAT ARE OUT OF THIS WORLD
The best of #MonthOfMaking
51 PAGES OF PROJECTS & TUTORIALS
Easy Pico Projects
Turbocharge Raspberry Pi 400 with an SSD drive
#MONTHOFMAKING
Discover new ways of making with Raspberry Pi
Build an on-air meeting light
Upcycle an Amiga 600
42 PAGES OF PROJECTS & TUTORIALS
NEW! RASPBERRY
BUILD MEDIA
40
Upcycling a Game Boy
Raspberry Pi 4 SID Synth
RASPBERRY PI PICO

RASPBERRY PI TIPS & TRICKS

THE MAGPI #116 ON SALE 31 MARCH

Plus!

Stargazing with Raspberry Pi

Create a digital art display

Learn ARM Assembly

DON'T MISS OUT!

magpi.cc/subscribe

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

EDITORIAL

Editor
Lucy Hattersley
lucy@raspberrypi.com

Features Editor
Rob Zwetsloot
rob@raspberrypi.com

Sub Editor
Nicola King

ADVERTISING

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

DESIGN

criticalmedia.co.uk

Head of Design
Lee Allen

Designers
Olivia Mitchell, Sam Ribbits, Ty Logan

Illustrator
Sam Alder

CONTRIBUTORS

David Crookes, PJ Evans, Rosemary Hattersley, Nicola King, Phil King, Ryan Lambie, Simon Monk, James Robinson, Marc Scott

PUBLISHING

Publishing Director
Russell Barnes
russell@raspberrypi.com

Director of Communications
Liz Upton

CEO
Eben Upton

DISTRIBUTION

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

SUBSCRIPTIONS

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

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.

Raspberry Pi PRESS

ISSN: 2051-9982.

When nerds collide

A chance encounter led to a chat of monolithic proportions

I was chatting in the pub – remember those? – the other day to a friend who has started work in a new coding job. Like many people I know, they picked up programming and, after a few courses, quickly found work in the area.

Coding certainly beats digging a ditch. I can't recommend highly enough learning to code or going on a boot camp.

The aforementioned friend started with a smattering of Python and, after some teaching work, now works for a prestigious, but somewhat contentious, institute that I shall not name here. They do incredible data science and will probably cure cancer.

Fortunately, in this instance, they did not have to fend off too many questions. I doubt if they have the 'Mulder and Scully' level of clearance just yet.

Another coding friend develops regression tests for a food delivery firm and has to explain constantly that: "no they don't deliver food; no they don't ride a bicycle; and no they can't personally help with that order that went wrong," and so on. We all have our crosses to bear.

Dry January being over, I'd had a few wines and therefore explained animatedly just how much I hate the R programming language with its first-indexed arrays. I get that this rant isn't quite normal, but starting arrays at one instead of zero. "Really, R, really!?"

They responded with "Java? So wordy." An ironically succinct answer. We agreed on Python. But then, doesn't everybody?

> “I went full Sheryl Sandberg and leaned in to the chat”

When two nerds chat, the world around them glazes over or rolls its eyes. And all our friends certainly did plenty of both. But, it's so rare I get to talk to people in the pub that know what a zero-indexed array is. Or a NAND gate. Or regression testing. So I went full Sheryl Sandberg and 'leaned in' to the chat. It stopped me from falling over at any rate.

Raspberry Pi delight

Like many a geek, they were delighted by Raspberry Pi, and my friend explained how one had been used to time-lapse the office construction. Time-lapse is one of my favourite projects because it shows a lot of what you can do with Raspberry Pi and the Camera Module. Raspberry Pi is small, reliable, and easy to set up and leave in the situation for a few days, weeks, or even months. Attach a battery and you can leave it going all day, snapping one shot after another.

For bonus points, my friend had used machine learning to select the daytime from night-time images and ditching the dull night shots.

Maybe that's why I spend time evangelising about learning to code, and why Raspberry Pi is the best way to go about it. I just want more people to chat to down the pub.

Let's hope 2022 is a more sociable year.

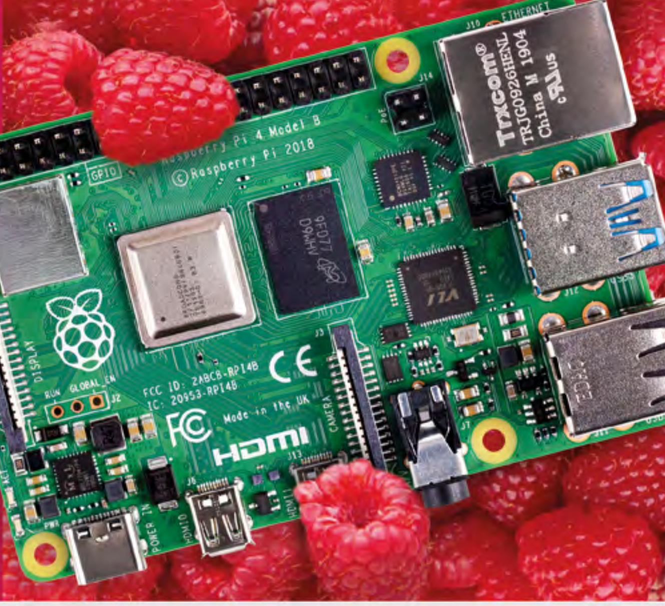
AUTHOR

Lucy Hattersley

Lucy Hattersley is editor of *The MagPi* and needs to stop buying second-hand computers. It's becoming a habit now.

magpi.cc

American **Raspberry Pi** Shop



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

Partner and official reseller for top Pi brands:

adafruit sparkfun ELECTRONICS PIMORONI HiFiBerry and many others!

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

USA PiShop.us

Canada PiShop.ca

Raspberry Pi APPROVED RESELLER

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 ThePiHut Pi-Shop.ch Welectron.

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