Astounding projects for every maker

Raspberry Pi
Official Annual
2027

# Raspberry Pi Official Annual 2027




Fruitbox Jukebox | Operation Pico | Vertical runner
Floating laboratory | Reflecting telescope
Ultimate media centre | Pixel pet programming
Robot arm hacking | Drum machine | Cyberdecks
Airborne projects | Garden projects

Astounding projects for every maker

Raspberry Pi Official Annual 2027

# Raspberry Pi Official Annual 2027



Fruitbox Jukebox | Operation Pico | Vertical runner
Floating laboratory | Reflecting telescope
Ultimate media centre | Pixel pet programming
Robot arm hacking | Drum machine | Cyberdecks
Airborne projects | Garden projects

OceanofPDF.com

SFERALABS

# Open Technology for the Industrial World

When Reliability Is Non-Negotiable

## Strato Pi Plus

### Edge Server with Multiple Isolated Serial Interfaces

- DIN rail edge server with up to 4 RS-485 interfaces and one CAN 2.0 / CAN FD bus, all individually isolated
- Hybrid architecture with dual-core RP2354 microcontroller
- All serial interfaces can be driven by the Raspberry Pi CPU or MCU
- Hardware watchdog
- PCIe SSD support (Raspberry 5 only)
- 10-50Vdc power supply
- 1 relay (NO/NC)





## Iono Pi v3

### High-Performance Versatile I/O Module

- DIN rail I/O module with protected analog and digital input/output lines
- Analog inputs: 2 x 0÷20mA and 2 x 0÷30V
- Up to 7 GPIO lines, with 1-Wire and Wiegand support
- 3 open collector outputs with overcurrent/short-circuit protection
- 4 relays (NO)
- 9-28Vdc power supply

### Both products include

- Raspberry Pi 4 or 5
- Secure Element
- Highly accurate RTC
- Ethernet, Wi-Fi, BLE, USB
- Optional fan
- CE | FCC | IC compliance

Powered by Raspberry Pi

Raspberry Pi Design Partner

www.sferalabs.cc

# **Welcome to** the Official Annual

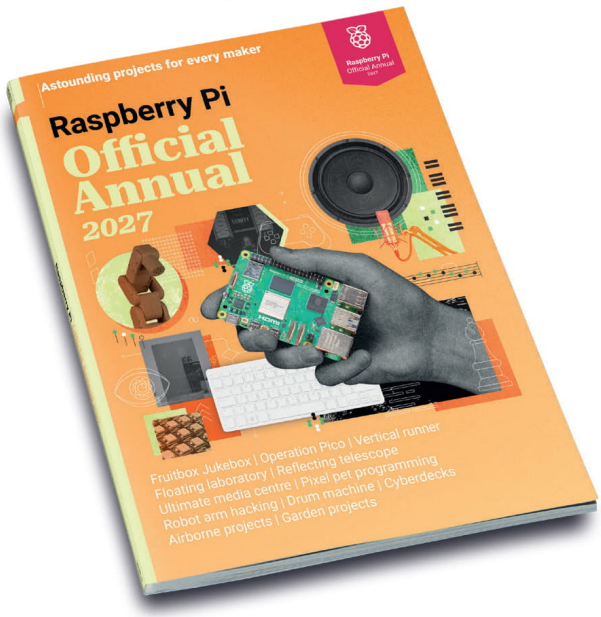


**You hold in your hands the very first Raspberry Pi Official Annual.** I hope this momentous occasion feels weighty, but not too weighty so that you can't comfortably read it.

There are 200 pages crammed into this annual with a huge selection of community projects from around the globe showing just some of the incredible things you can do with Raspberry Pi. Of course, there are also a load of excellent tutorials and features covering every kind of Raspberry Pi you can get your hands on, from Raspberry Pi 500 to Raspberry Pi Pico and everything inbetween.

I always find it tricky whittling down the articles from the past year to put into these books, however I hope there's a wide enough variety here to spark your imagination – no purple dragon required.

**Rob Zwetsloot – Editor**




[OceanofPDF.com](OceanofPDF.com)

## Editorial

**Editor**
Rob Zwetsloot
rob@raspberrypi.com

**Sub Editor**
Phil King

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

## Design

**Head of Design**
Jack Willis

**Designers**
Sara Parodi, Natalie Turner

**Illustrator**
Sam Alder

**Brand Manager**
Brian O Halloran

## Contributors

David Crookes, PJ Evans, Ben Everard, Andrew Gregory, Lucy Hattersley, Rosie Hattersley, Phil King, KG Orphanides

## Publishing

**Publishing Director**
Brian Jepson
brian.jepson@raspberrypi.com

**Director of Communications**
Helen Lynn

**CEO**
Eben Upton

## Distribution

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

## Subscriptions

The Maltings, West Street
Bourne, Lincolnshire, PE10 9PH
+44 (0) 1778 392472
rpimag.co/subscribe
raspberrypi@warnersgroup.co.uk







This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.
Raspberry Pi Official Magazine is published by Raspberry Pi Ltd, 194 Cambridge Science Park, Milton Road, Cambridge, England, CB4 0AB. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products, or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0).

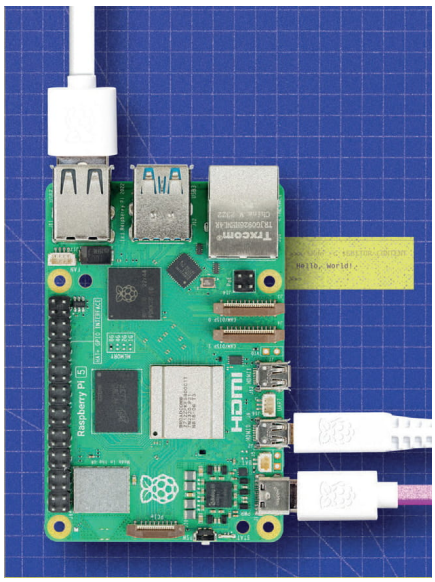



ISBN: 978-1-916868-57-1

OceanofPDF.com

# Contents


Get started with Raspberry Pi



Raspberry Pi 500+



# Raspberry Pi Problem Solving

## Project Showcase

**HexBoard**
A hexagonal MIDI controller

**Mini Air Hockey games table**
Heavily customised gaming

**Raspberry Pi Chessboard**
Test your cerebral might

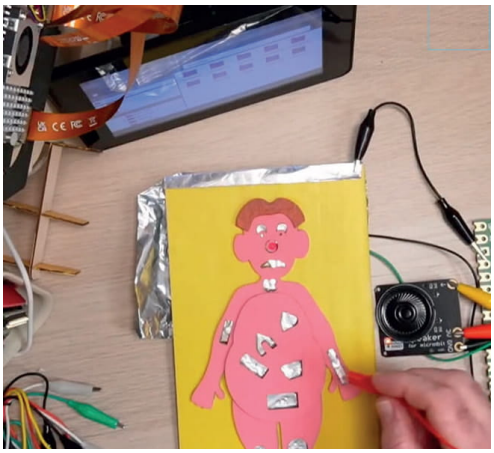
**Raspberry Pi Fruitbox Jukebox**
Dial up the grooves

**Alpakka v1.0 controller**
A modular controller that you print yourself

**Whispering Wires**
Poetry by phone

**Bipedal companion robot**
Immobile desk assistant

**Operation Pico**
As in the classic boardgame



**Zero-G Lab Floating Platform**
Simulating space robotics

**Pepper Pi**
Victorian illusion, futuristic computing



**Avanade Intelligent Garden**
Interactive smart flowers

**Embedded MIDI Synthesizer on Raspberry Pi**
An electric accordion

**Reflecting telescope**
Turning a pipe into a telescope

**DOOM SNES 2025**

Pico-powered retro revival

**Vertical Runner**
A functional movie prop

**Bog body theatre prop**
Spooky performance art

**Yashica IR camera**
Surreal IR photography

**Sfera Baltic boat**
Upcycling a 60-year-old tug boat with Raspberry Pi

**Electronic Drum**
Business Card Playable contact details

**Flokk herd management**
Canadian cattle tracking

**Amstrad PPC 640 Cybderdeck**
The classic computer made much cooler

## Maker Guides

**Build the ultimate media centre**
The perfect box for watching your faves



**Play Steam games using x86-64 emulation**
Modern games on Raspberry Pi

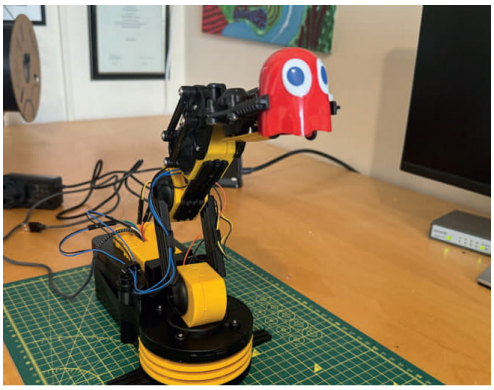
**The A to Z of Raspberry Pi**
26 essential things to know

**Create a Sense HAT interactive pixel pet**
Program your own pocket-sized monster

**Pico drum machine**
Percussion at your fingertips

**Home automation made easy**
Make a smarter home with Raspberry Pi

**Hack a robot arm**
Powering up a kid’s toy



**Extreme Raspberry Pi**
At the very limits of computing




## Reviews and Resources

**Argon ONE V5**
A solid case, literally

**Pironman 5 Mini**
Mini desktop case, fully featured

**RP2350 Pico W5**

5GHz Wi-Fi-enabled

**Dream HAT+**
Radar add-on for Raspberry Pi

**CrowPi 3**
Electronics learning lab

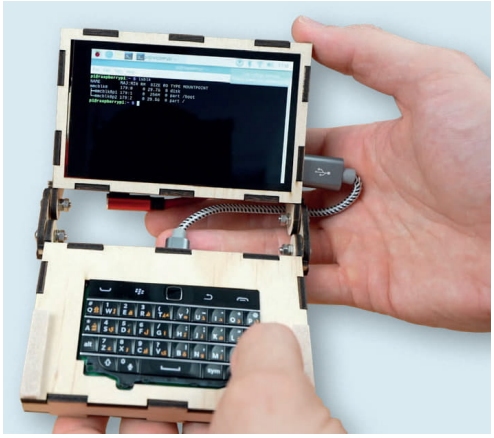
**Adafruit Fruit Jam**
Pico as a computer

**10 amazing airborne projects**
Take to the skies

**10 amazing astronomy projects**
Look to the stars

**10 amazing Raspberry Pi garden projects**
Down in the allotment

**10 amazing Raspberry Pi cyberdecks**
Hack the planet

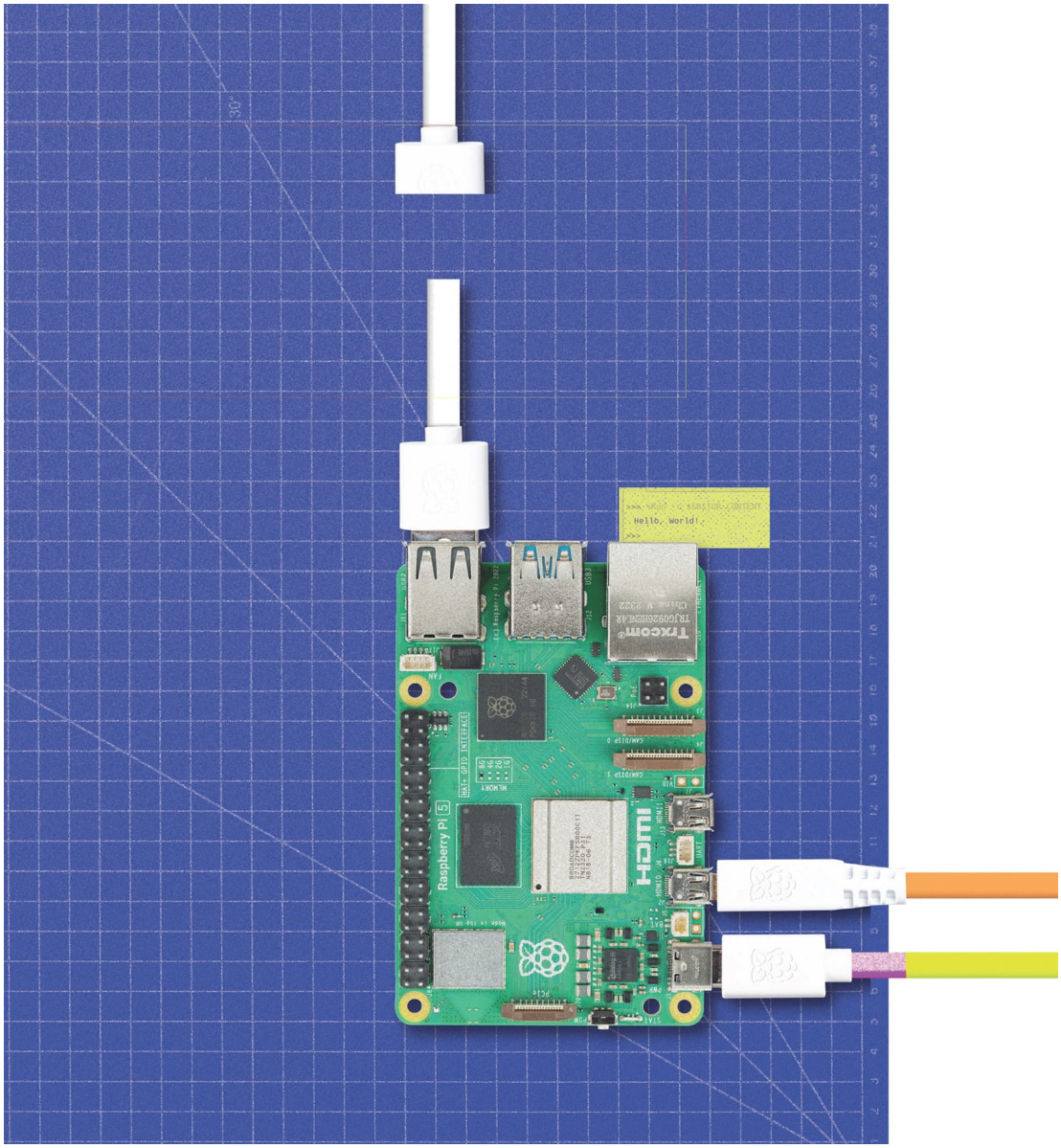


*OceanofPDF.com*

# GET STARTED WITH RASPBERRY PI

No matter which Raspberry Pi model you have, you're holding a powerful little computer that can help you learn to code, control electronic circuits through its GPIO pins, and tackle all kinds of home projects. Here's how to get started…

By Phil King




# Install Raspberry Pi OS

How to install the official operating system for your Raspberry Pi

*The official Raspberry Pi Imager tool makes installing an OS quick and simple*

**T**o start using your Raspberry Pi computer, you'll need an operating system installed on a microSD card (or SSD, but more on that later), which also acts as its storage. You can buy an official Raspberry Pi microSD card with the Raspberry Pi OS preloaded onto it from Raspberry Pi Approved Resellers. Otherwise, you can easily install an OS onto any microSD card (of sufficient capacity) using the Raspberry Pi Imager tool. Let's take you through the process…

---

## 01 Get a microSD card




A microSD card with 32GB or greater capacity is recommended for installing the desktop version of Raspberry Pi OS. You may well be able to use a lower-capacity card, but you'll have less space left for storing files and applications. For the non-desktop 'Lite' version of Raspberry Pi OS, a 16GB card is sufficient.

In addition, for optimal performance, you should look for a Class A2 microSD card such as the official cards available from Raspberry Pi. While some cards come with Raspberry Pi OS pre-installed, it may not be the latest version, in which case you may want to reinstall it following the steps here.

**YOU'LL NEED**

- Raspberry Pi
- microSD card
- Raspberry Pi Imager (**rpimag.co/software**)

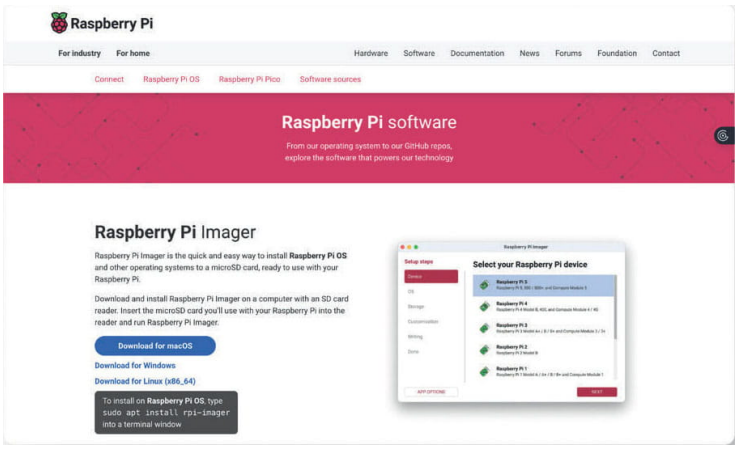
## 02 Install Imager

The official Raspberry Pi Imager tool makes installing an OS quick and simple. Visit **rpimag.co/software** and download it for your computer: it's available for Windows, macOS, and Debian/Ubuntu Linux.

On Windows or macOS, double-click the downloaded EXE or DMG file and follow the instructions. On macOS, you may need to change your Privacy & Security setting to allow apps downloaded from the App Store and identified developers to enable it to run.

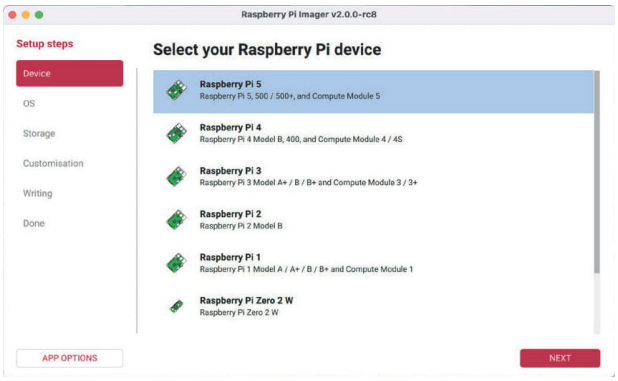
On Linux, right-click the downloaded AppImage file, select Properties, and switch to the Permissions tab. If you see a checkbox that allows you to make the file executable, click it; if you see an Execute drop-down menu, set it to Anyone. You can now double-click the AppImage file to run the installer.

Alternatively, if using a Raspberry Pi computer running Raspberry Pi OS, you should find the preinstalled Raspberry Pi Imager in the applications menu under the Accessories category. If not, you can install it from a Terminal window, by entering `sudo apt install rpi-imager`.




## 03 Choose model

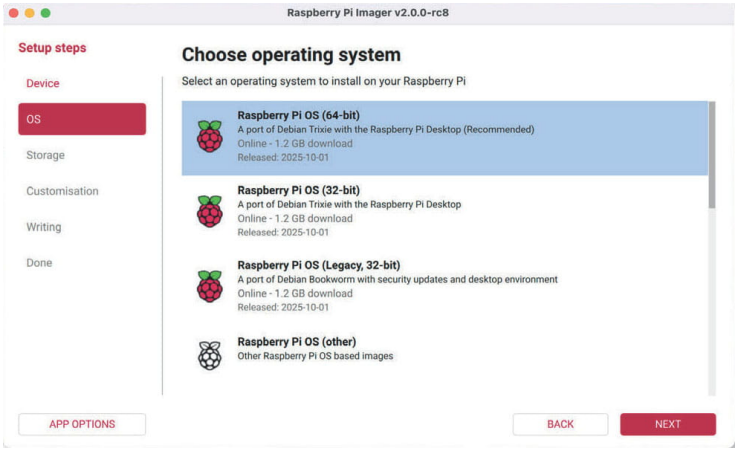
Once Raspberry Pi Imager has been installed on your computer, double-click its icon to launch it. The installation steps are listed in the left-hand pane of the window. The first step is Device, prompting you to select your Raspberry Pi model in the main pane. This will filter the selection of operating systems accordingly for the following step, so that it lists only those suitable for your model. Click the Next button to proceed to the next step.




## 04 Select OS

The OS step lists a range of operating systems available for your Raspberry Pi model. Versions of the official Raspberry Pi OS will appear at the top. In most cases, you'll want to select the top 'Recommended' option. For most

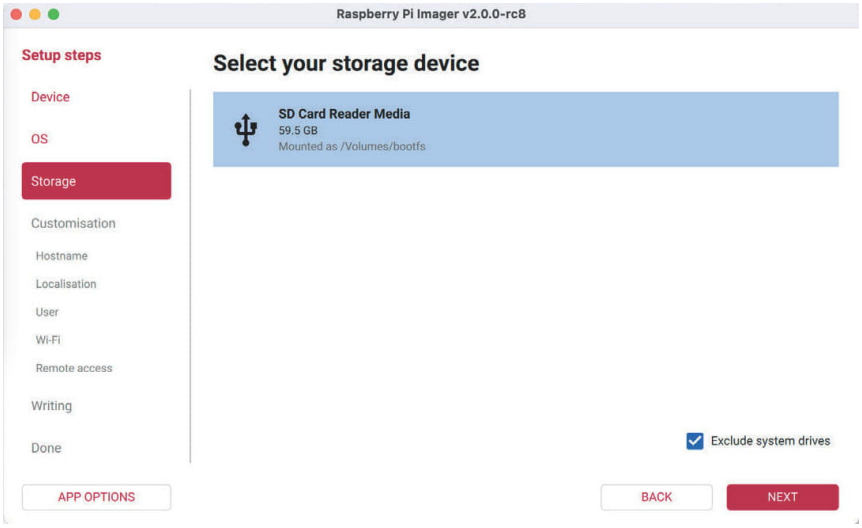
models, this will be the latest 64-bit version with Raspberry Pi OS Desktop; for earlier and Zero models, it will be 32-bit.




## 05 Choose storage

You'll need to connect your microSD card to your computer to write to it. Some laptops have suitable card slots. If not, you can use a USB adapter (you'll need one if you're running Imager on a Raspberry Pi, as it can't write to the microSD card in its slot).

In the Storage step, your attached microSD card should then appear. Typically, it'll be the only device listed. If not, be careful: you don't want to erase a different device during the write process, so disconnect all removable drives except your target microSD card, then open Imager again.

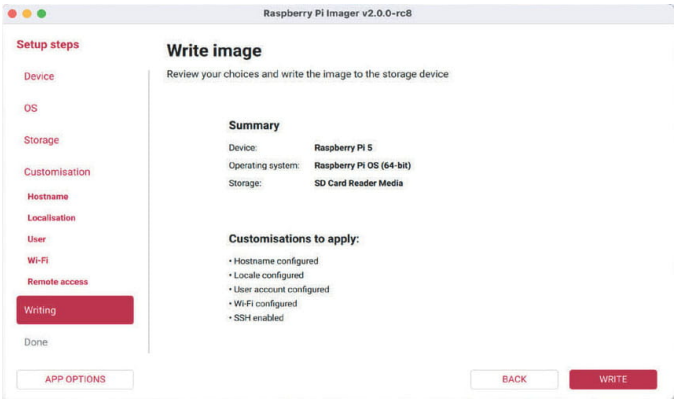


## 06 Customisation options

The optional Customisation step enables you to set up some things in advance, such as the Wi-Fi connection and remote access – especially useful if you'll be using your Raspberry Pi headless, without a monitor.

First, you get to set the hostname of your Raspberry Pi system to help identify it on the network. Then, you can select your locale, time zone, and keyboard layout. Next, you choose a username and password – don't forget the latter!

After that, you can enter your Wi-Fi router's name (SSID) and password to connect to it. Then you can enable SSH for remote login. There are two options for authentication: a password (which you set earlier) is simplest; or, for passwordless authentication, you can use an RSA key that you've generated on the computer you're using (with the `ssh-keygen` command). Finally, you can enable Raspberry Pi Connect for remote access using your Raspberry Pi ID; just follow the prompts.

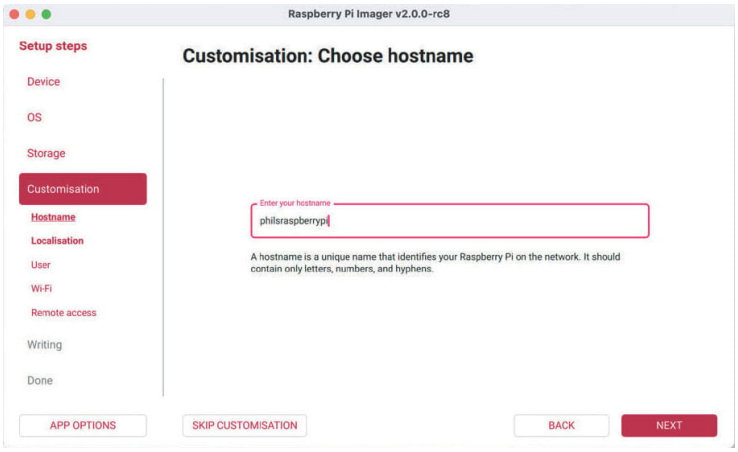



## 07 Write image

The final part of the process is to write the OS image to your microSD card. After checking the summary and list of customisation options you've set, click the Write button. Note the warning about all files being erased on the card, then click the 'I understand, erase and write' option. You may need to enter your local computer's password to proceed.

While Imager is writing the OS, don't disconnect the storage device or the write process will fail. It can take a few minutes to write the OS and

then verify it. Once finished, it will show a summary, with the option to write another card with the same options. You can now remove the card and insert it in the microSD card slot on your Raspberry Pi.




## Solid-state drive

Instead of booting Raspberry Pi from a microSD card, you can use a different medium such as a USB drive or solid-state drive (SSD) by writing the OS to it with Imager. Note that the SSD in Raspberry Pi 500+ is preloaded with Raspberry Pi OS, so you don’t need to do it manually.

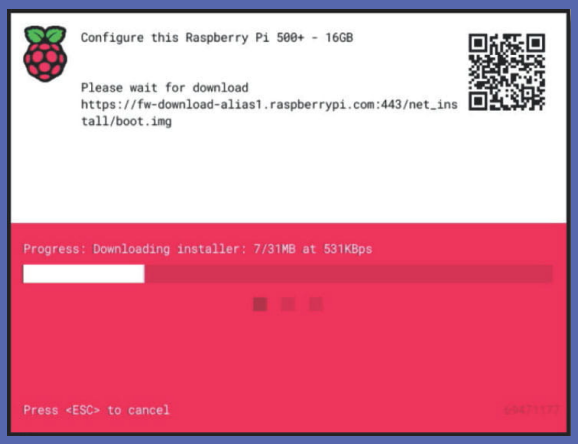
The best way to use an SSD with Raspberry Pi 5 is to buy an official SSD Kit, which includes a 256GB or 512GB NVMe SSD and connects to Raspberry Pi 5’s PCIe port.




You can write an OS to the SSD using Imager on the same Raspberry Pi having first booted from a microSD card, then change the boot order in the raspi-config tool: enter `sudo raspi-config` in a Terminal window, then select Advanced Options > Boot Order; specify an option that includes NVMe, then select Finish and reboot Raspberry Pi.

## Network Install

On a Raspberry Pi 4, 5, 400, 500, or 500+ with a monitor, mouse, and keyboard connected, you can instead install the OS over the network. To do so, insert a blank microSD card into your Raspberry Pi – or connect an SSD – then use an Ethernet cable to connect it to your router and power up Raspberry Pi. Hold the **SPACE** key as it boots; this will bring up the boot mode menu. Press **N** to start the network installer, which will download a special version of Raspberry Pi Imager and load it. As before, you'll then be able to use it to choose an operating system and install it onto your storage device.



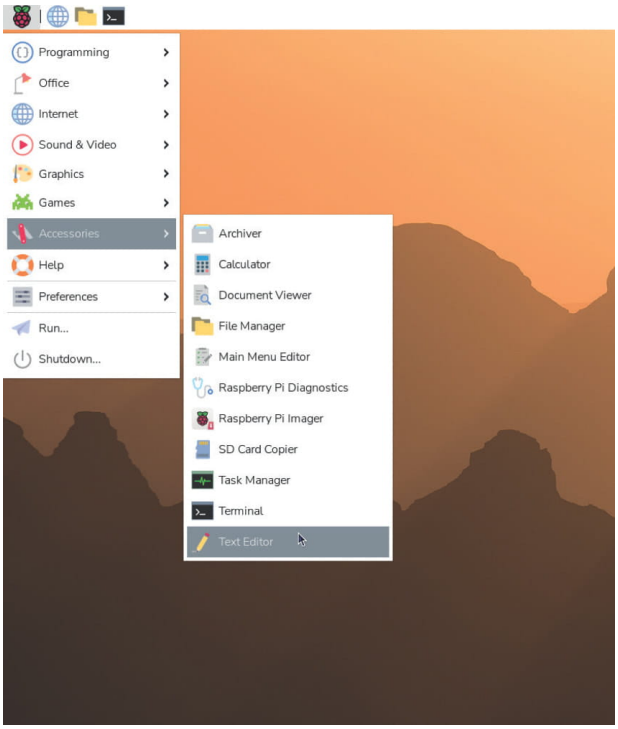

# Using Raspberry Pi OS

Navigate the official operating system and its applications

*Clicking the Raspberry Pi icon brings up an*

## *applications menu with various categories*

**W**hen you first power up your Raspberry Pi, you may (unless you set customisation options when writing the OS with Imager) be greeted by a Welcome Wizard that takes you through the initial setup process for the operating system, including locale settings, creating a user, and connecting to Wi-Fi. If you have already set these up in the Customisation section of Raspberry Pi Imager, however, you will instead be taken straight to the Raspberry Pi OS desktop.




Applications can be opened via the menu on the left

Along the top of the desktop is a taskbar. At its left end are some pinned applications – Chromium web browser, File Manager, and Terminal (command-line interface) – along with tabs for any other open applications or windows. At its right end are clickable icons including Raspberry Pi Connect, Bluetooth, Wi-Fi, and audio – right-click the last one to select an output. Other icons that may also appear here include a downward arrow if software updates are available, an eject icon if you have removable storage

connected (such as a USB drive), and a mic icon if you've ever connected a Bluetooth headset.




The Control Centre lets you change a range of settings

**YOU'LL NEED**

- Monitor
- Keyboard
- Mouse

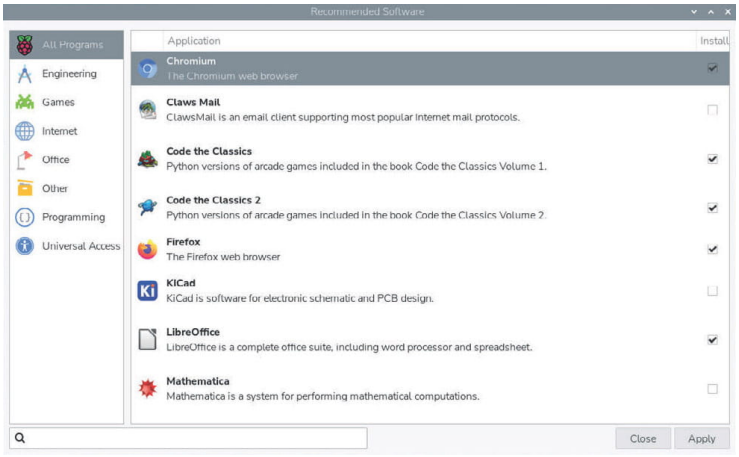
At the far right end of the taskbar is the current time (sourced from the internet). Hovering over it shows the date; clicking it brings up a calendar.

## Opening applications

On the far left of the taskbar is a Raspberry Pi icon. Clicking it brings up an applications menu with various categories. For instance, under Programming you'll find the Geany and Thonny editors; under Sound & Video you'll find the VLC Media Player. Just click on the name of an application to launch it.

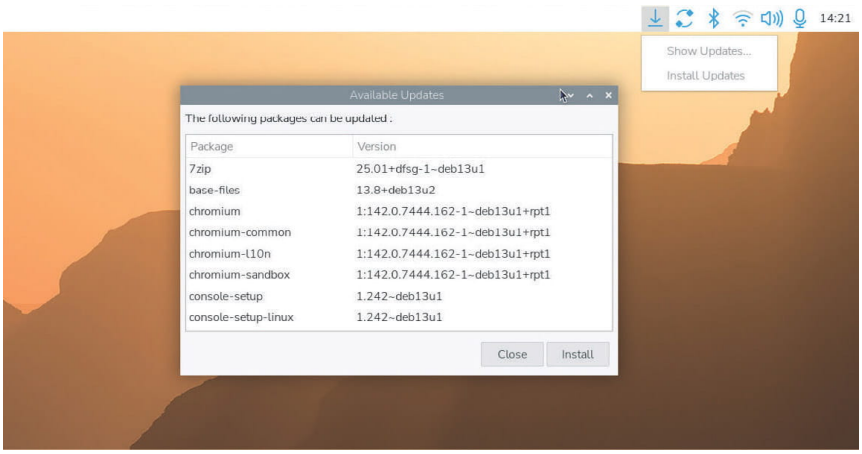
The number of applications that appear in the menu by default depends on which version of the OS you installed – the 'Full' one includes a larger

range of recommended applications. To install more applications, the easiest way is to use the Recommended Software tool, found under the Preferences category of the menu. Upon launching it, you can browse applications in various categories. Tick the checkbox for any app(s) you wish to install, then click Apply or OK. For instance, you may well want to install the LibreOffice productivity suite, which includes a word processor, spreadsheet, database, and other applications. Once installed, it will appear under the new Office category in the applications menu.




Install extra applications with the Recommended Software tool

If you want to uninstall an app, untick its checkbox in the Recommended Software tool. An even greater range of applications and software packages can be found in the Add/Remove Software tool, found under Preferences in the menu.




Click on the arrow icon in the taskbar to install updates

In addition, applications and tools can be installed and run from the command line by opening a Terminal window. Here, you can issue a host of different commands to become a power user of your Raspberry Pi. For more details, check out the *Conquer the Command Line* book: **rpimag.co/conquertcl.**

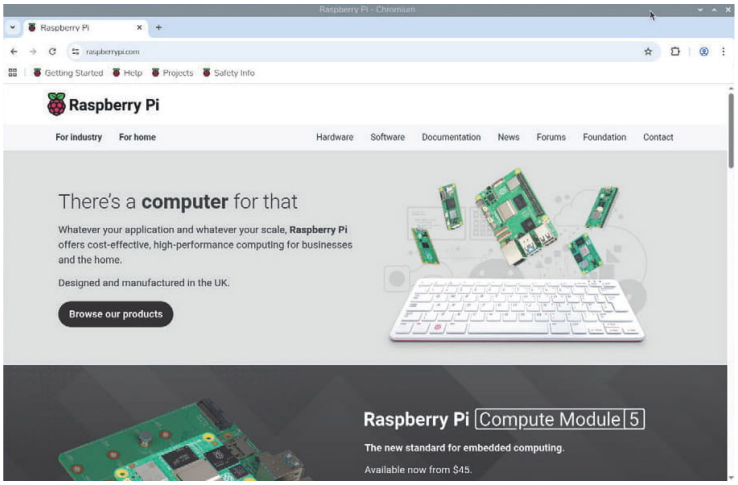
You can also find the book in the Bookshelf app, under Help in the applications menu. This lets you download digital PDFs of books and *Raspberry Pi Official Magazine* issues. Note that some titles are padlocked unless you are Raspberry Pi contributor (from £5 a month) and log in with your Raspberry Pi ID.

## System management

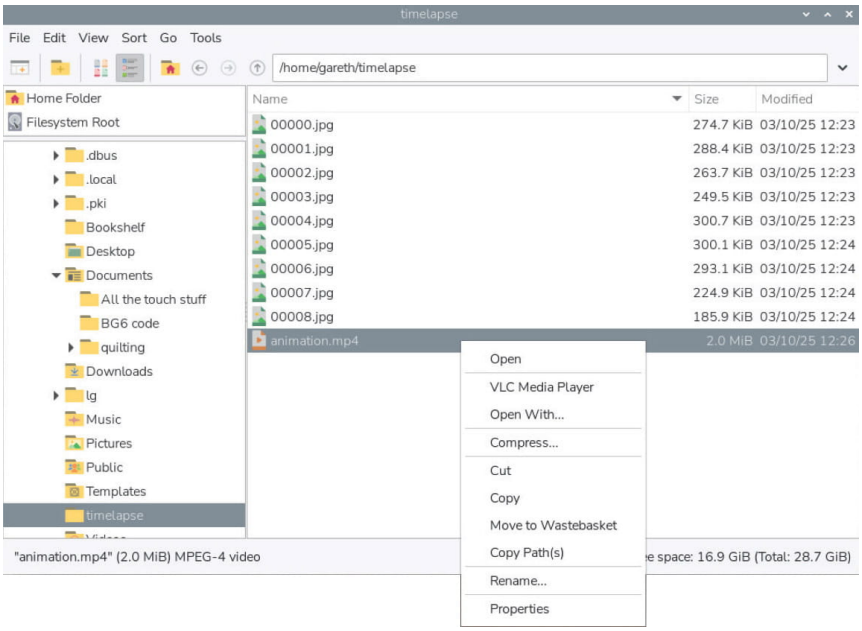
If you want to make changes to the settings for how your Raspberry Pi OS system works, the Control Centre is the place to go (replacing the Raspberry Pi Configuration tool in earlier versions of the OS). Found under menu > Preferences, it features numerous sections, selectable from the alphabetical list on the left. For instance, in Display you can enable screen blanking (to turn it black when the OS is not used for several minutes) and choose when/if an on-screen keyboard appears. Under Interfaces, you can enable or disable communication protocols like I2C and SPI, along with remote login methods like SSH and VNC. Localisation is pretty self-explanatory, enabling you to set the locale, time zone, keyboard layout, and Wi-Fi country.

As mentioned, if you see a downward arrow on the right-hand side of the taskbar, it means that software updates are available – for the system and/or applications. Clicking on the icon brings up two options: Install Updates to download and install them, and Show Updates to preview a list of them first.

Finally, when you've finished using Raspberry Pi OS for the time being, you will want to shut down your Raspberry Pi. It's important to do this properly rather than simply unplugging it, which could result in the operating system being corrupted. To do so, select Shutdown from the applications menu. This will bring up a dialog with four options: as well as Shutdown to shut the system down, you can Reboot it (shut down before restarting), select Lock Screen to return to a password-protected screen, or Logout of your user account (useful if your system has more than one user).




Visit websites with the built in Chromium web browser



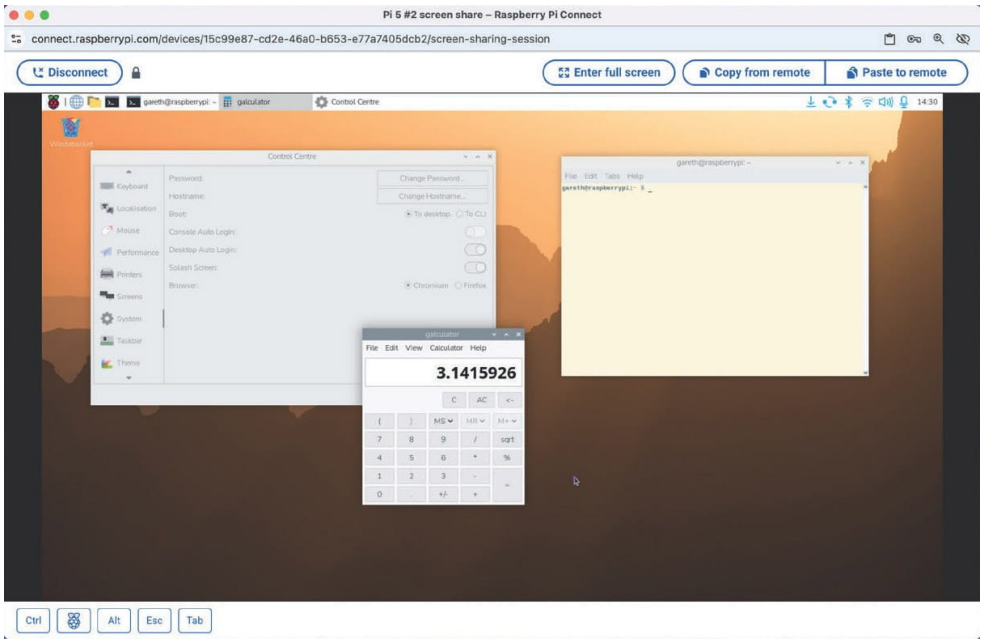

# File Manager

This is the file browser for Raspberry Pi OS and is similar to that found on other computers, enabling you to browse the folders (aka directories) and files on your system. The left pane shows the directories, while the right-hand pane shows the files and subdirectories for the selected directory.

Upon first opening the File Manager, it will show your user's 'home' folder, which contains subdirectories such as Documents, Downloads, Music, Pictures, Videos, and (once you've opened the Bookshelf app) Bookshelf containing downloaded book and magazine PDFs.

You can drag and drop files between folders. Double-clicking a file will open it in its default application; right-clicking lets you choose what to open it with, along with numerous other options.

If you insert a USB drive, you'll be prompted to open it in File Manager, where you can view its files and folders.




# Raspberry Pi Connect

While it's still possible to access the Raspberry Pi OS desktop remotely via VNC, and the command-line interface with SSH, Raspberry Pi Connect makes it easier to do both remotely – not only on your local network, but from anywhere in the world.

To set it up (if you haven't done so beforehand when writing the OS with Raspberry Pi Imager), click its circular icon on the right-hand side of the taskbar and select Turn On Raspberry Pi Connect. This will open the browser, prompting you to sign in with your Raspberry Pi ID – if you don't have one yet, click the link to create one. After authenticating, you're prompted to enter a nickname for your Raspberry Pi.

Once this is done, the Connect icon will turn blue to indicate it is working. You can now log in from any remote computer or device by visiting **connect.raspberrypi.com**. From the 'Connect via' drop-down next to your device, select 'Screen sharing' to view the desktop (or 'Remote shell' for its command-line interface). You can now navigate the desktop and use it as normal.

# Coding and digital making

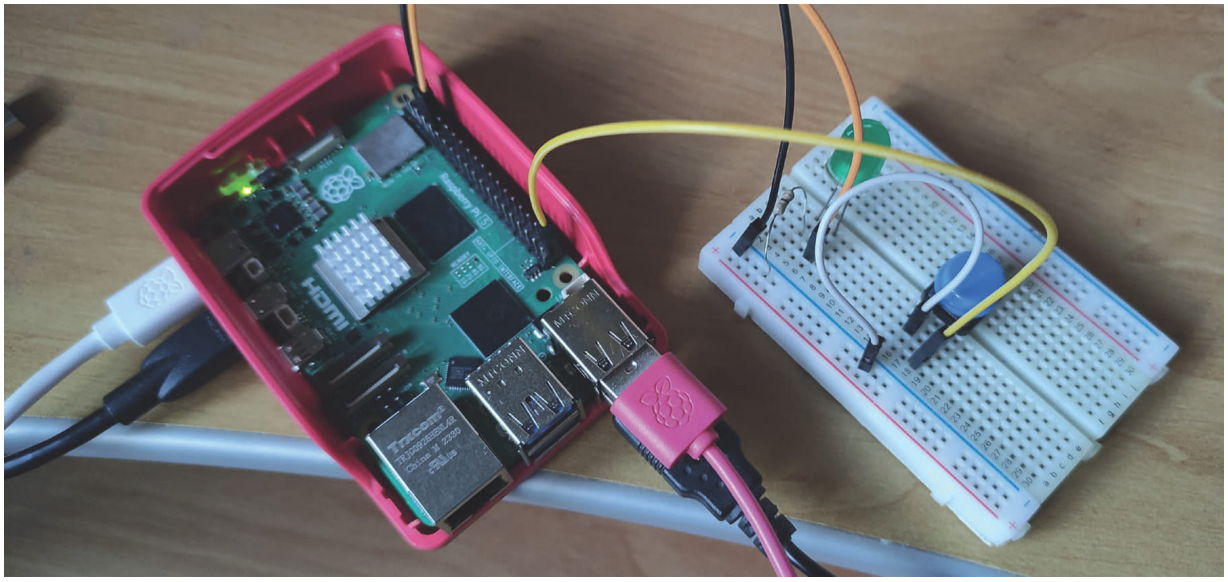
Program Raspberry Pi with Python and control an electronic circuit

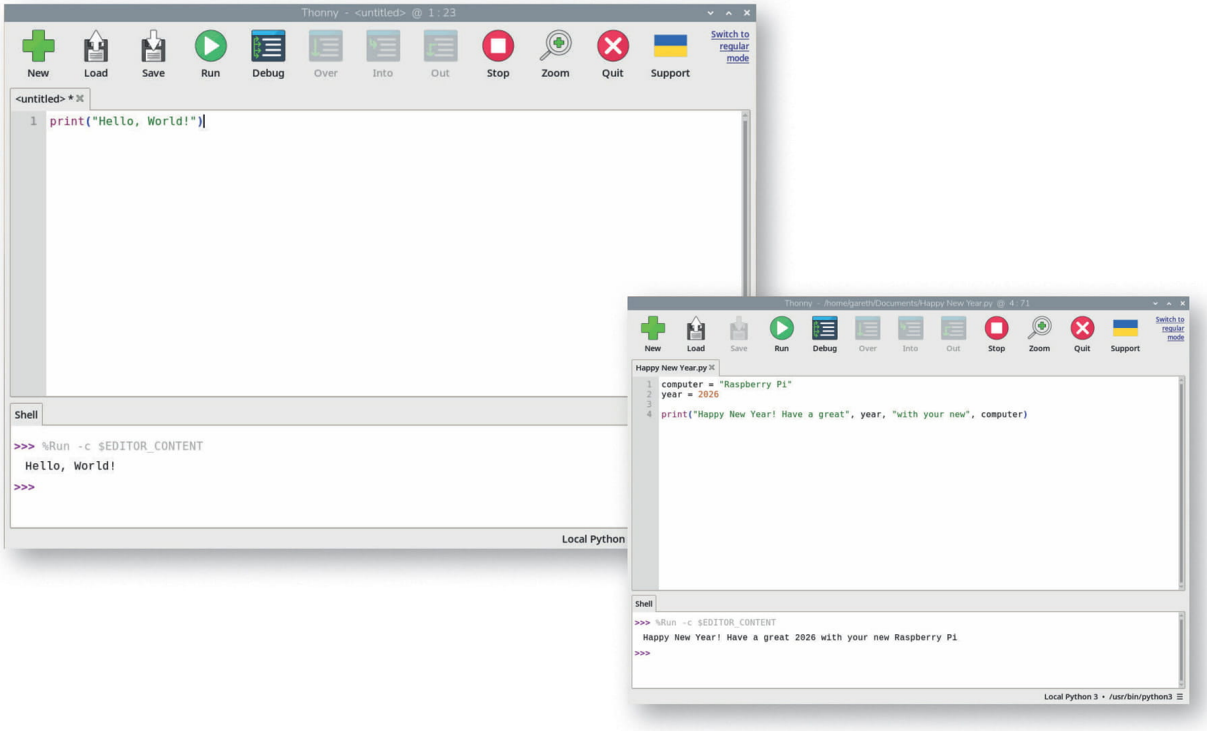
*One of the easiest languages to get started with is Python*

**O**ne of the best aspects of a Raspberry Pi computer (and the reason it was originally created) is the ease with which you can start programming on it and controlling electronic circuits connected to its GPIO header.

While you can code in a variety of languages, one of the easiest to get started with is Python. To get started, open the applications menu and select Programming > Thonny. This opens the Thonny IDE (integrated development environment) for coding in Python.

| YOU'LL NEED |
| --- |
| • Solderless breadboard |
| • LED |
| • 330Ω resistor |
| • Push-button |
| • Jumper wires |






## Hello, world

You can start by typing a simple one-line Python instruction in the Shell area at the bottom of the window:

```
print("Hello, world!")
```

Press the **ENTER** key and it will print the message ‘Hello, world!’ to the Shell area. As with other Python functions, brackets are used to enclose the output – in this case a text string within the quote marks.

Alternatively, you can enter the same line in the main coding area above the Shell area and click Run to run the program. The result will be the same. Try it.

Text strings and numbers can both be stored in variables, each of which has a user-defined name. For instance, you can store the text string ‘Raspberry Pi’ in a variable called `computer` , and a number in a variable called `year`, with:

```
computer = "Raspberry Pi"
year = 2026
```

Then print a celebratory message; for instance:

```
  print("Happy New Year! Have a great",
year, "with your new", computer)
```

This should output the message, ‘Happy New Year! Have a great 2026 with your new Raspberry Pi’.

## Python maths

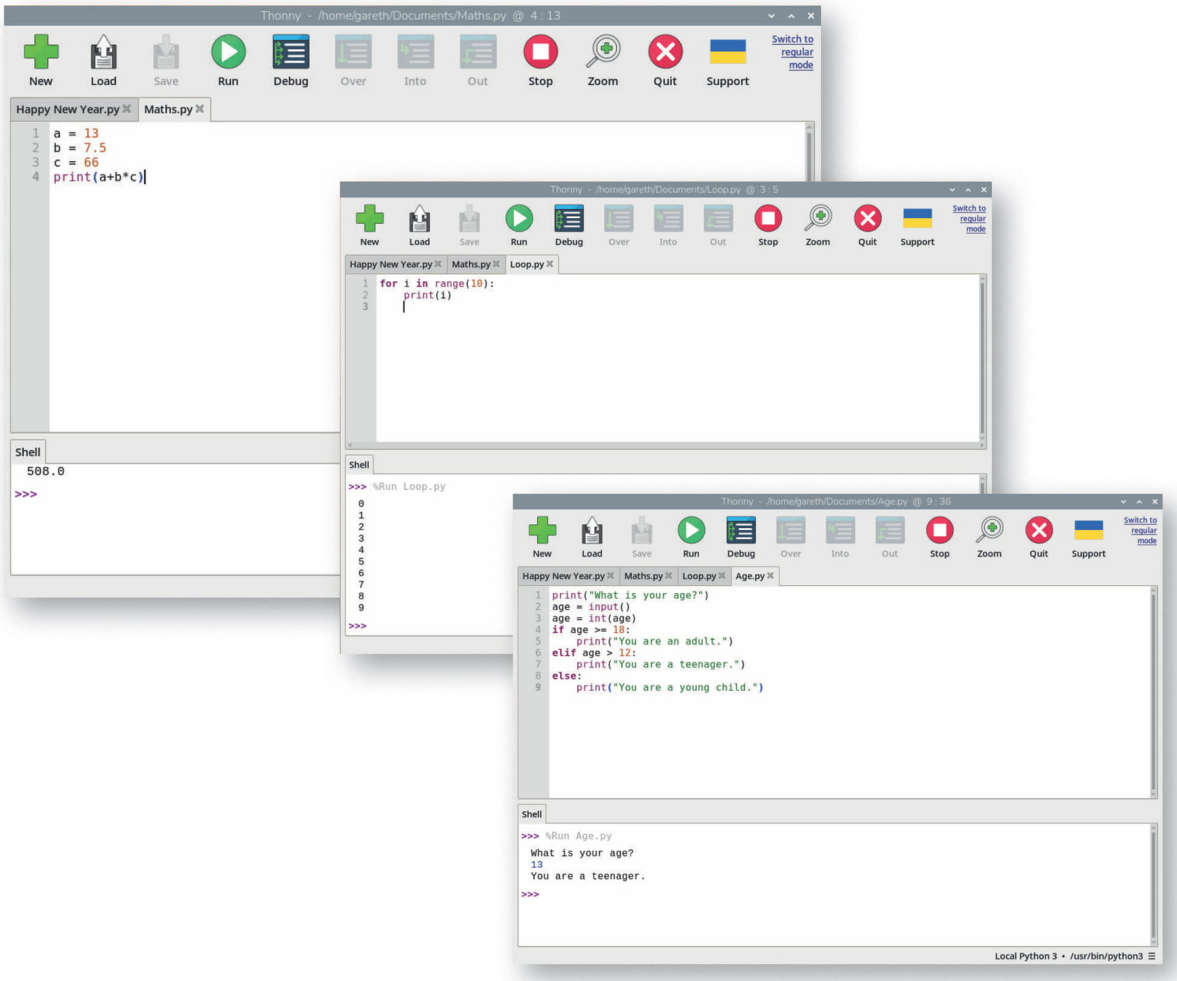
You can use maths operators to add (`+`), subtract (`-`), divide (`/`), and multiply (`*`) numbers. For instance, let’s use them on some variables:

```
a = 13
b = 7.5
c = 66
print(a+b*c)
```

The result given for this is 580.0 – since we have set a floating-point value for `b` (i.e. it's not an integer), the result is too and so has a decimal point. Note also that the BODMAS/PEMDAS rule applies, so multiplication is done before addition.

## Loop the loop

Another important aspect of coding is a loop. There are two main types in Python, using the keyword `while` or `for`. Both are followed by indented (typically by four spaces) lines that form the loop to be repeated. The difference is that `for` has `range` parameters to tell it how many times to execute the loop. For instance:



```
for i in range(10):
   print(i)
```

…will print the numbers 0 to 9 (it starts at 0 by default and finishes with the tenth loop).

The `while` keyword, on the other hand, performs the indented loop below it for as long as its variable has a certain value. If you just use `while True:`, it will execute the loop ad infinitum, which can be very useful for performing repetitive tasks.

## On condition

Another key concept of coding is the use of conditional statements which only execute the block of code below (again indented) when a specified condition is met. For instance:

```
print("What is your age?")
age = input()
age = int(age)
if age >= 18:
   print("You are an adult.")
```

Here, using Python's built-in `input()` function, we ask the user to input their age. We convert that string to an integer in the following line. We then use `if` to test a condition for it (here, whether it is greater than or equal to 18). If so, the indented line below is executed, printing the message.

We can add further conditions and outputs to this with `elif` and/or `else` statements:

```
print("What is your age?")
age = input()
age = int(age)
if age >= 18:
   print("You are an adult.")
elif age > 12:
```

```
    print("You are a teenager.")
else:
    print("You are a young child.")
```

## Blink an LED

Now we know a bit about Python, let's try blinking an LED with it. Connect the LED as shown in **Figure 1**, with its legs in two different columns of holes on the solderless breadboard (so they're not connected to each other). The longer leg is the anode (positive), which we will connect to one of Raspberry Pi's special programmable GPIO pins. The latter can be used as an input or output; here it's an output. We've wired ours to the GPIO 14 pin (**rpimag.co/gpio**), using a male-tofemale jumper wire.

The cathode (negative) leg of the LED is connected to a GND (ground, or 0V) pin via a 330Ω resistor. The resistor is vital to limit the maximum current flowing through the LED – too much current could damage it or even your Raspberry Pi. Make sure your Raspberry Pi is turned off when wiring up the LED.

With the LED connected, we can turn it on and off in Python – see the **blink.py** code listing. Using the GPIO Zero library makes things easier, so we import its `LED` function at the start (along with `sleep` from the time library). We then set GPIO 14 as an output pin for our LED. Finally, we use a `while True:` loop to repeatedly turn the LED on and off, with a one-second (`sleep`) delay each time.

**DOWNLOAD THE FULL CODE:**

**rpimag.co/annual27**

### blink.py

> Language: **Python**

```
001.  from gpiozero import LED
```

```
002.  from time import sleep
003.
004.  led = LED(14)
005.
006.  while True:
007.      led.on()
008.      sleep(1)
009.      led.off()
010.      sleep(1)
```

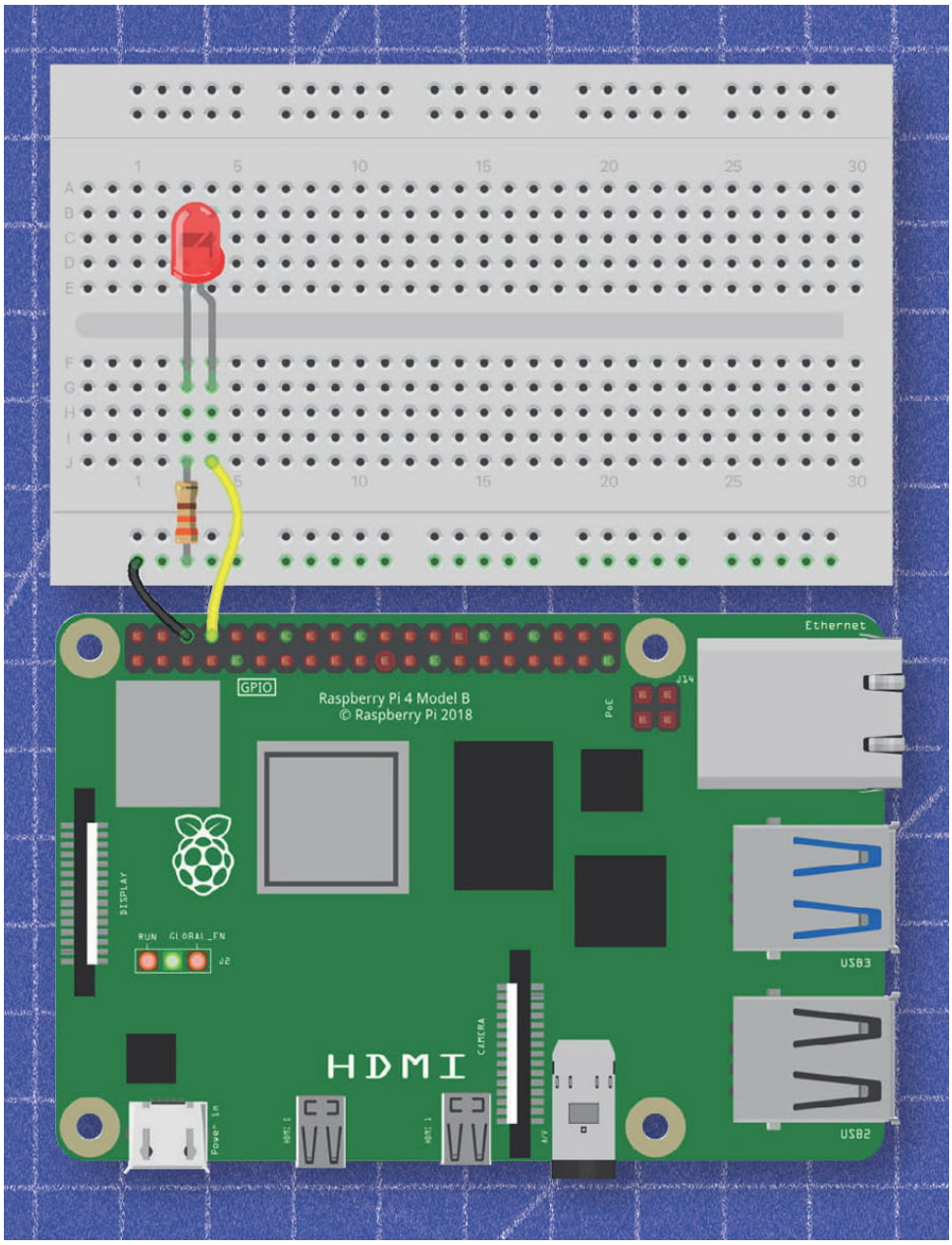



**Figure 1:** Connect an LED with a resistor

## Push the button

Now let's add an input device to our circuit, in the form of a push-button. Wire it up as shown in Figure 2. Note that here we're using a 'ground rail' (all its holes are connected) on the breadboard to connect both components' cathodes to the same GND pin on Raspberry Pi. You'll often see this in

circuits with multiple components. We've connected the other leg of our button (on the same side of the breadboard's central gap) to GPIO 21 (**rpimag.co/gpio**).

In our code (**led_button.py**), we also import the `Button` function from GPIO Zero and assign GPIO 21 to our button. In an infinite `while True:` loop, we then check whether the button has been pressed, using GPIO Zero's built-in `is_pressed` function. (Note that we could instead have used `if button.value == 1:`, where the `==` is Python's conditional form of equals.) If so, we light the LED, otherwise we turn it off.

Can you add more LEDs and/or buttons to your circuit? And perhaps also an active piezo buzzer to make a noise?

**DOWNLOAD THE FULL CODE:**

**rpimag.co/annual27**

## led_button.py

> Language: **Python**

```
001.  from gpiozero import LED, Button
002.
003.  led = LED(14)
004.  button = Button(21)
005.
006.  while True:
007.      if button.is_pressed:
008.          led.on()
009.      else:
010.          led.off()
```

*Let’s add an input device to our circuit, in the form of a push-button*

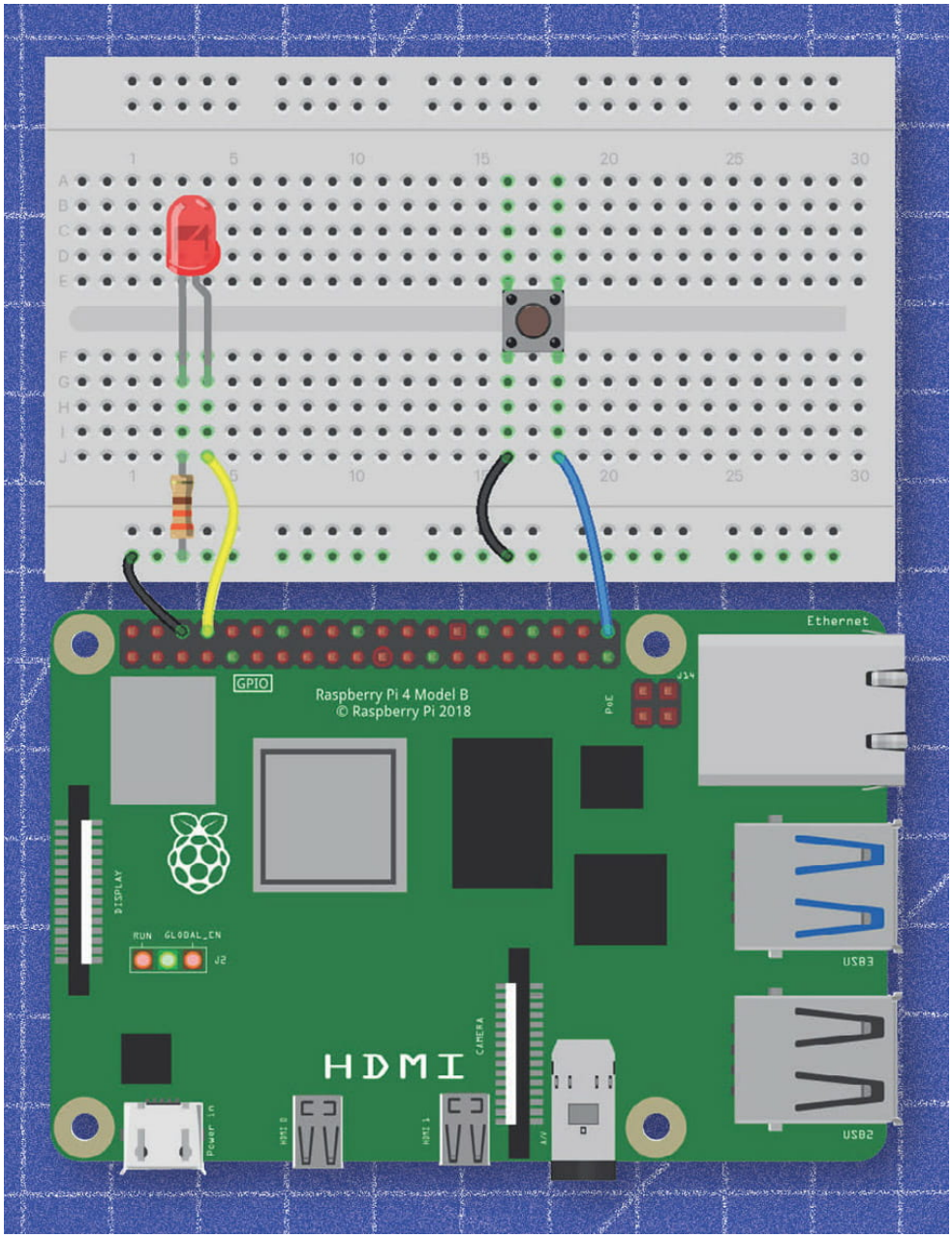



**Figure 2:** Adding a button to the circuit

# Project ideas

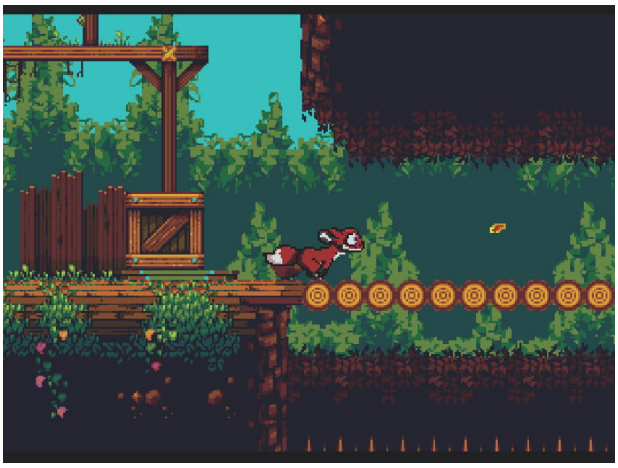
Raspberry Pi can be used in so many different and exciting ways

**W**ith its Linux-based operating system, processing power, and GPIO header, your Raspberry Pi can drive a wide variety of projects. We've highlighted a few popular software-based applications to get you started, but the potential uses go far beyond these. For further ideas, see this issue's 'Brilliant Beginner Builds' feature.

## Retro gaming

One of the most popular uses of a Raspberry Pi is as a retro games machine that can emulate classic computer and consoles. The main options available for this are Recalbox (**recalbox.com** and available in Raspberry Pi Imager), RetroPie (**retropie.org**), Lakka (**lakka.tv**), and Batocera (**batocera.org**). All of these feature a user-friendly front end and can emulate myriad systems such as NES, SNES, Mega Drive/Genesis, PlayStation, ZX Spectrum, C64, and Amiga.

Game ROMs can typically be added via a USB drive or over the network. Be careful downloading them from sites hosting copyrighted games illegally, however. There are lots of other legal ROMs available, including many modern 'homebrew' titles developed for classic hardware – see **rpimag.co/legalroms**.



Retro-style games include Tanglewood for Sega Mega Drive / Genesis

## Media centre

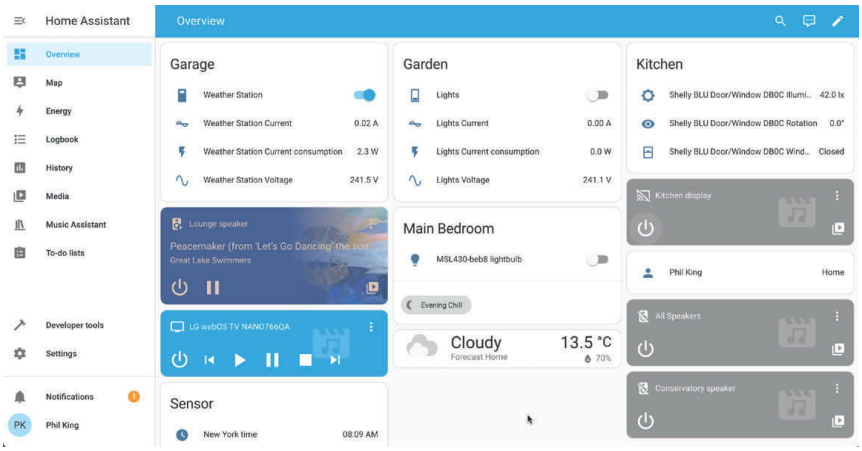
Your Raspberry Pi can be turned into a media centre to play video and audio on your TV. One of the simplest ways to do this is to install the specialist media player OS, LibreELEC (**libreelec.tv** and available in Imager, under the ‘Media player OS’ category). Based on the open-source Kodi software, it enables you to play files stored locally and on network-attached storage (NAS), as well as offering access via add-ons to popular streaming services such as YouTube, BBC iPlayer, and Spotify. An optional mini remote control makes navigating menus easier.

Alternatives to LibreELEC include Plex (**plex.tv**) and Recalbox (also a games emulator), along with audio-specific systems such as Volumio, moOde, and piCorePlayer. For more details on building the ultimate media centre, check out our detailed guide in issue 155 (**rpimag.co/155**).

## Home automation

As we explored in our issue 167 guide (**rpimag.co/167**), it’s easy to set up your own smart home dashboard on Raspberry Pi with Home Assistant – the free, open-source home automation platform. Rather than relying on a proprietary system, this enables you to build a fully customisable smart home setup tailored to your own preferences.

Once set up, you can view your Home Assistant dashboard with a web browser on any device. It’s easy to add all sorts of smart devices, including Google and Alexa ones. You can assign devices to areas/rooms and create automations – from scratch, or by adapting one of the many ‘blueprint’ templates shared by the Home Assistant community. In addition, ‘scenes’ enable you to set a state for a group of devices or entities.




The Home Assistant web dashboard

## Ad blocker

These days, many websites seem to be crammed with obtrusive adverts and pop-ups. While browser-based ad blockers are available, a smarter and more reliable method is to do it at a network level, to work on all connected devices. For this, you can install Pi-hole on a Raspberry Pi and then access its numerous options and statistics via a web dashboard.

Pi-hole works as a DNS sinkhole. When a device on your network requests a website, the request goes to Pi-hole first. If the domain is on the blacklist, Pi-hole blocks it, otherwise it works as usual. You can customise the blacklist, as well as adding sites to a whitelist. Since unwanted adverts are blocked before being downloaded, this has the added benefit of improving your network performance.

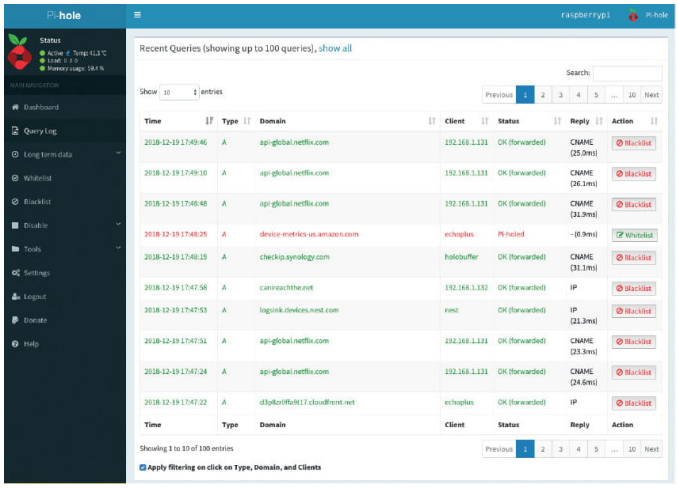
For details of how to set up a Pi-hole, check out this guide: **rpimag.co/pihole**.

## NAS server

With a suitable storage device connected, such as an SSD, hard drive, or USB flash drive, your Raspberry Pi can be turned into a network-attached storage (NAS) server. This means that the files held on it can be accessed wirelessly by other devices on your network.

The standard method for setting this up is to use Samba – not the Brazilian dance, but a free and open-source re-implementation of the SMB networking protocol. You can then share the drive as a folder over the network and grant access to authorised users with a password. There are a few steps to the process, as detailed in this guide: **rpimag.co/nasbox**.

If you want to access files remotely from outside your network, it may be better to set up your Raspberry Pi as a cloud server using a solution such as NextcloudPi (**nextcloudpi.com**) or ownCloud (**owncloud.com**).




Pi-hole lets you add domains to a blacklist to block adverts

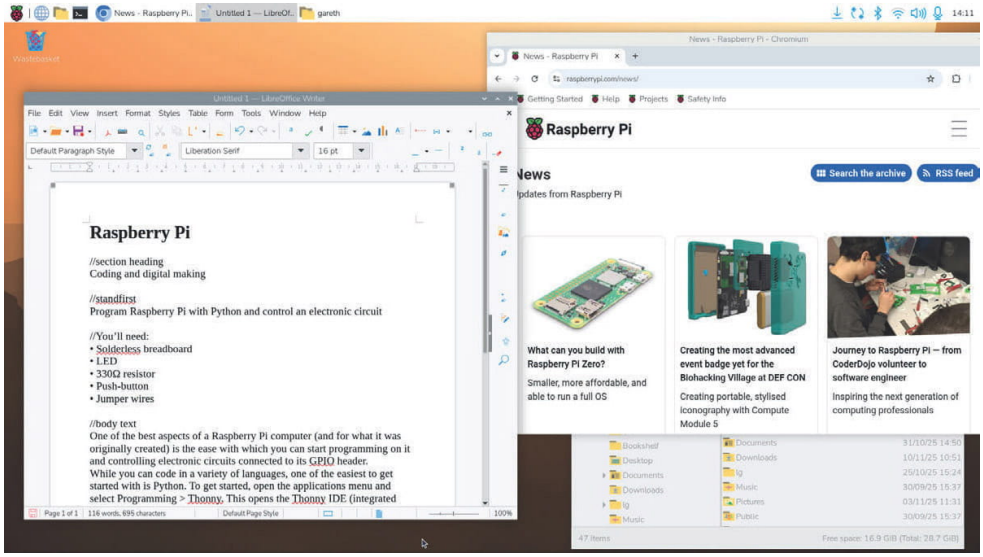
## Desktop PC

Your Raspberry Pi can also work really well as a standard Linux-based desktop PC replacement, giving you all the functionality you need for work and leisure. Along with the built-in Chromium and Firefox web browsers, there's VLC for media playback – it can handle video and audio streams (such as for radio stations) too.

For email, you can install the Claws Mail client, while the LibreOffice productivity suite includes word processing, spreadsheet, presentation,

database, maths, and drawing applications.

There are lots more tools and applications available to install in the Add/Remove Software tool and/or via the command line. These include the GNU Image Manipulation Program for editing photos and artwork, Inkscape for drawing vector graphics, and Audacity for editing audio files.




You can use Raspberry Pi as a desktop PC

OceanofPDF.com

# Raspberry Pi 500+



# The dream machine is here.

## Master physical computing with a mechanical keyboard, built-in SSD, 16GB RAM, 2.4GHz Arm processor, and GPIO. **Uncompromising performance, perfectly tactile.**

By **Lucy Hattersley**

**W**e are pleased to finally reveal the latest Raspberry Pi computer: 500+. This is the machine we have been waiting to share for over a year. It offers “uncompromising performance, perfectly tactile.”

In the words of its engineers: This is the “dream machine”. And this is the one that the editorial team want to use to code and write with.

Raspberry Pi 500+ puts the power of Raspberry Pi into an ergonomic and tactile mechanical keyboard with 16GB of RAM and a 256GB NVMe SSD. If you’ve ever wanted a modern-day Linux machine that you can plug in and start using for product development, coding, circuit building, and engineering, this is the perfect computer.

There are no compromises with Raspberry Pi 500+. The mechanical keyboard features Gateron KS-33 Blue keys with Cherry MX-style switches; the keys are low-profile with the top half of each key backlit. It is a 75% keyboard layout, which is our preferred layout, and offers the perfect balance between size and functionality. It’s a clickable joy to type on.

Inside is a quad-core 64-bit Arm processor paired with 16GB RAM and a 256GB NVMe SSD with Raspberry Pi OS pre-installed.

We’ve been using 500+ at work for a while now, slowly phasing out our laptops and migrating to it for writing, research, and coding. It’s a pure powerhouse and is a wonderful computer to use. Raspberry Pi OS is lightning fast and Linux is the best operating system around.

So, if you’re looking for a seriously fun computer that’s a coding and development control centre with a wonderful keyboard, fully upgradable

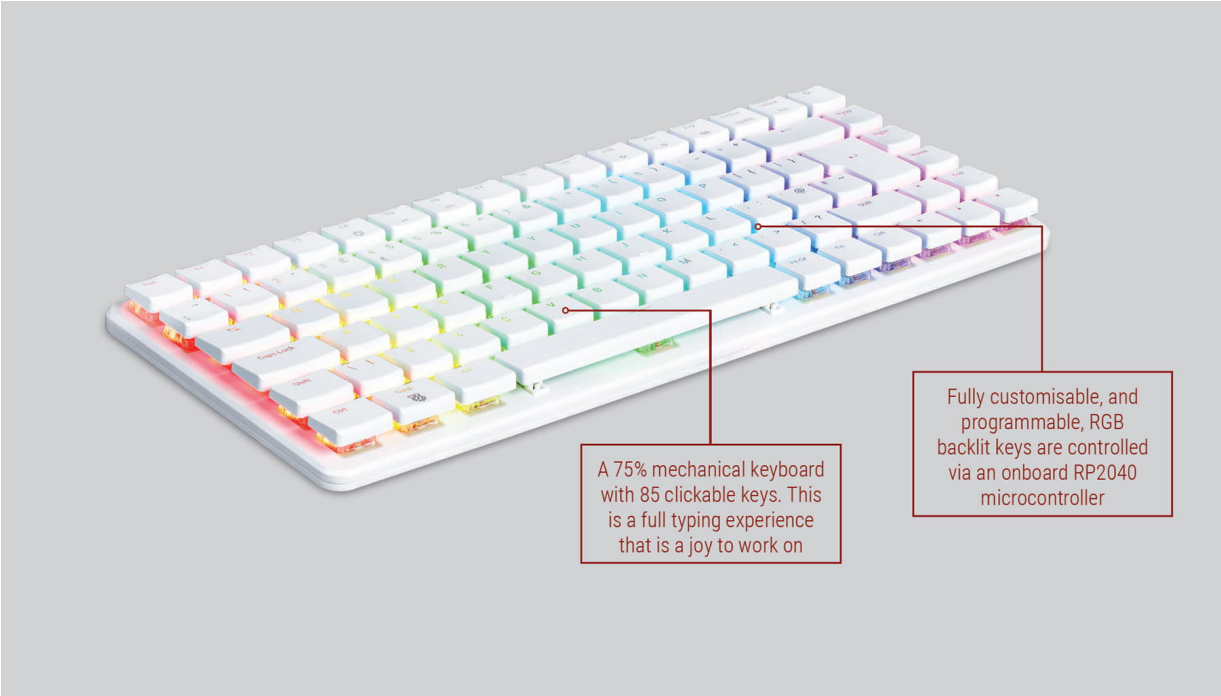
and repairable, with a host of features that you just don’t find on other computers, then Raspberry Pi 500+ is the one for you.

*There are no compromises with Raspberry Pi 500+*

**SPECIFICATIONS**

- Quad-core 64-bit Arm Cortex-A76 processor
- 16GB LPDDR4X SDRAM
- Integrated 256GB NVMe SSD with Raspberry Pi OS pre-installed
- Opening tool included for access to SSD (also requires screwdriver, not included)
- 2 × micro HDMI ports (supports up to 4Kp60)
- 2 × USB 3.0 ports
- 1 × USB 2.0 port
- Gigabit Ethernet port
- microSD card slot
- Horizontal 40-pin GPIO header
- 802.11b/g/n/ac Wi-Fi
- Bluetooth 5.0
- 5V DC via USB-C connector
- 85-key mechanical keyboard with clicky switches
- Replaceable keycaps (keycap puller included)
- Per-key programmable RGB LED backlighting

# This is the *dream* machine




## The clicky keyboard

Undoubtedly the star of Raspberry Pi 500+ is the mechanical keyboard. This provides a much more tactile experience than the chiclet-style keyboard found on a regular Raspberry Pi 500. Giving Raspberry Pi a mechanical keyboard makes it ideal for programmers and office workers who want the best keyboard around. It makes Raspberry Pi 500+ a unique experience that offers powerful desktop computing in a small package with a great keyboard.

A huge amount of time went into developing and determining the perfect keyboard for Raspberry Pi.

The keycaps are a custom in-house design by John Cowan Hughes. “We had a few different designs and took a poll,” says Simon Martin, Senior Principal Hardware Engineer at Raspberry Pi. The keyboard has a flat, and low, profile, with a tapered edge at the bottom of each key.

The keycaps can be replaced to a design you prefer (see ‘Change the keycaps’, overleaf).

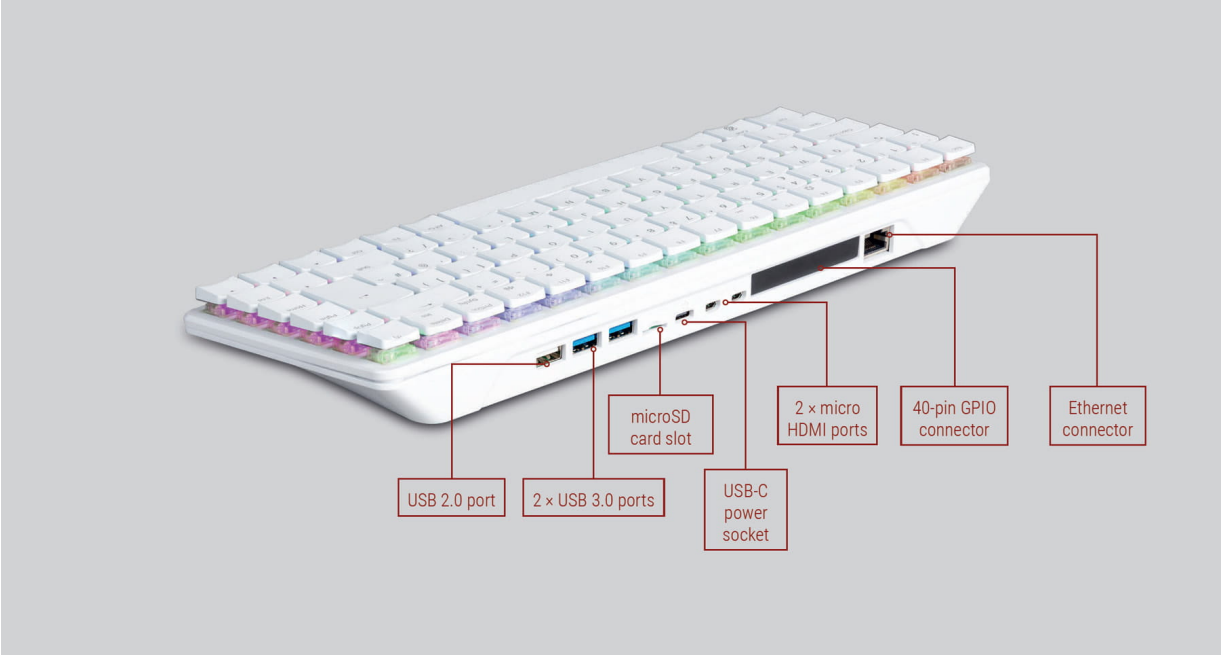
The switches below are Gateron Blue. This is a version of the classic Cherry MX Blue switches which are famous for a tactile bump and classic click noise. It has a sound like a typewriter key that is adored by writers, coders, and gamers.

These Blue-style switches are distinct from Brown and Red counterparts. Brown switches are tactile but have a gentle bump halfway down, which removes the click; Red switches have a linear switch with no tactile feedback or click at all – they are quiet but feel squishy.

Mechanical switches are usually a matter of taste, but almost everyone we know prefers Blue keyboards – although Blue switches can be quite conspicuous.



Around the back, we can see a range of useful ports




## Light it up

As well as being mechanically minded, the new keyboard has fancy RGB lighting. Underneath each key sits an RGB LED that can light up the keyboard in different patterns. Pressing **FN+F4** cycles through a range of keyboard patterns:

- Pure white
- Pure red
- Rainbow
- Cycle rainbow
- Heat map
- Single key press
- Off

Pressing **FN+F5/F6** adjusts the brightness of the keyboard backlighting.

**Get started: raspberrypi.com/500-plus**

| | |
|---|---|
| **Keyboard variants**<br>Raspberry Pi 500+ is available in the following keyboard variants:<br><br>• UK<br>• US | • Spanish<br>• Nordic<br>• German<br>• Japanese<br>• French<br>• Italian<br>• Portugese |

## Program the keyboard RGB lighting

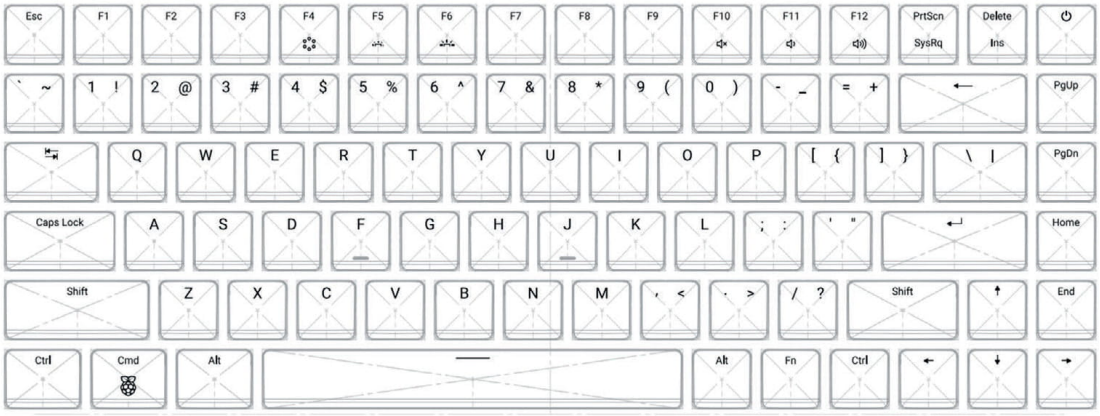
Inside Raspberry Pi 500+ sits an RP2040 microcontroller, used to control the keyboard and lighting. This makes it possible to reprogram the keyboard and lighting and create your own custom light displays.

As a demonstration, the team has developed a keyboard light version of Flappy Bird that you can play using the Raspberry Pi 500+ keyboard. This program, written in Python, can be uploaded to RP2040 inside Raspberry Pi 500+.

We will have an exploration of programming the keyboard and recreating Flappy Bird on it in a future edition.



Colour us impressed by the RGB lighting




The keycap design used in Raspberry Pi 500+

## Change the keycaps

Raspberry Pi 500+ comes with a key remover tool that is used to remove and replace the keycaps.

Raspberry Pi 500+ keycaps are a custom design and don't conform exactly to any of the standard profiles. Keycaps come in different profiles, such as:

- **OEM**. These are the default shape found on most factory-made keyboards.
- **Cherry**. Similar to OEM, but shorter.
- **Spherical All Row (SA)**. These are taller, with a 1970s retro vibe.
- **DSA (Double Shot ABS)**. Short, flat, and round. Most importantly, every row has the same shape.
- **XDA**. Similar to DSA but with a larger, flatter top surface.
- **MT3**. Very tall keycaps with a distinctly retro feel. As found on old terminal keyboards.

Note that most sets won’t have a power button, and the volume and RGB keys won’t be labelled. So you’ll have to remember which function keys to use. The quality of RGB lighting is also dependent on the keycaps.




Using the remover tool to change the keycaps

The keycap remover tool is easy to use. Simply place it under the edges of the keycaps, gently squeeze together to grasp the keycap, and pull it gently up. The keycap will pop off, revealing the mechanical switch underneath. To refit it, simply push the keycap back down and it will clip back into the switch.

## NVMe M.2 SSD

Moving deeper down from the keyboard, we find the other stars of the show. The integrated M.2 NVMe SSD is connected to the motherboard via a PCIe interface and is easily upgradable.

This provides the same kind of speed boost that using an M.2 HAT and SSD with Raspberry Pi 5 does, and is vastly faster than using a stock microSD card (although note that a microSD card slot is also provided – which is incredibly useful for backup and quickly prototyping alternative operating systems).

By default, Raspberry Pi OS uses PCIe Gen 2, and you can gain a further speed boost by switching to PCIe Gen 3. To do so, open Terminal and enter:

```
$ sudo raspi-config
```

Then select `6 Advanced Options` > `A9 PCI Speed`.

Choose `Yes` to ‘Would you like PCIe Gen 3 to be enabled?’

Choose `Finish` and `Yes` to the option to reboot your Raspberry Pi OS. When the operating system restarts, you will be running Raspberry Pi 500+ with PCIe Gen 3 enabled, which offers roughly twice the SSD speed. You can test the speed before and after using the Raspberry Pi Diagnostics app (found under menu > Accessories). The SD Card Speed Test option is designed to check if microSD cards are fast enough for Raspberry Pi, but will also speed-test the internal SSD. Click Run Tests.

Here are our results:

PCIe Gen 2

```
seq-write;0;0;420776;102
seq-read;442064;107;0;0
```

PCIe Gen 3

```
seq-write;0;0;799219;195
seq-read;853889;208;0;0
```

| | PCIe Gen 2 | PCIe Gen 3 |
|---|---|---|
| Write (MB/s) | 411 | 780 |
| Read (MB/s) | 432 | 834 |

## Raspberry 500+ vs Raspberry 500

Of course, Raspberry Pi 500+ isn’t the only option and Raspberry Pi 500 remains available. This previous model has some advantages. While some people might love the clacky nature of the mechanical keyboard, an office full of the chiclet-style keys found on Raspberry Pi 500 will be more amiable to the ears.

Likewise, the lack of backlit keyboards can be seen as a benefit in some low-light environments and a sealed unit may be better for some office environments (although the microSD card is easily removable). The 500 has a Kensington lock, which is useful for physically restraining the movement of the device.

Still. There's no getting away from the fact that Raspberry Pi 500+ is a substantial upgrade and – to our eyes and hands at least – the one we want to use. Here is a handy guide to some of the differences we've spotted:




The SSD inside Raspberry Pi 500+

| 500 | 500+ |
|---|---|
| 2.4GHz CPU | 2.4GHz CPU |
| 8GB RAM | 16GB RAM |
| microSD card slot | 256GB M.2 NVMe & microSD card slot |
| Tactile keyboard | Mechanical keyboard |
| No backlight | RGB backlight |
| 10 Function keys (11, and 12 as Fn alternatives) | 12 Function keys |

| | |
|---|---|
| No media keys | Separate media keys (Volume, brightness) |
| Power LED light | Power button with LED |
| Page keys with arrows | Dedicated PgUp, PgDn, Home, End keys |
| Fn on left | Fn on right |
| Sealed unit | Screw access with spudger included |
| Kensington lock | No Kensington lock |



Underside of a 500+, along with useful tools

## Getting inside

Staying true to the maker ethos, Raspberry Pi 500+ is easy to get into. There are five pan-head Pozi cross-head screws on the back of Raspberry Pi 500+ that can be easily removed.

Then a coin slot located on the plastic case underneath the space bar can be accessed, using the supplied spudger. Move the spudger around the line in the case to separate the keyboard from the lower case.

Gently lift up the keyboard and remove the cable connecting the two. Use the spudger to lift the black plastic retaining clip on the ZIF connector (Zero Insertion Force) on either the underside of the keyboard or the mainboard.

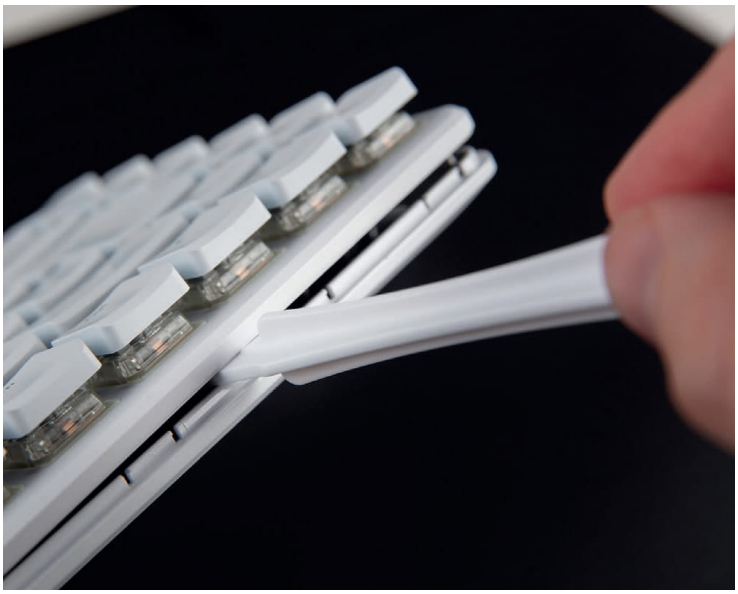
The SSD in our Raspberry Pi 500+ is a 2230. Your model may come with a larger 2280 size SSD. Raspberry Pi 500+ accepts the following sizes:

- 2230
- 2242
- 2260
- 2280

The default size is 256GB and you can easily upgrade this to a larger (or smaller) storage size. Remove the screw that holds the M.2 NVMe SSD in place and then gently ease it out. Now fit your replacement M.2 NVMe SSD in place and use the screw to secure it. Make sure the SSD is lined up with the groove in the screw so that it sits level.



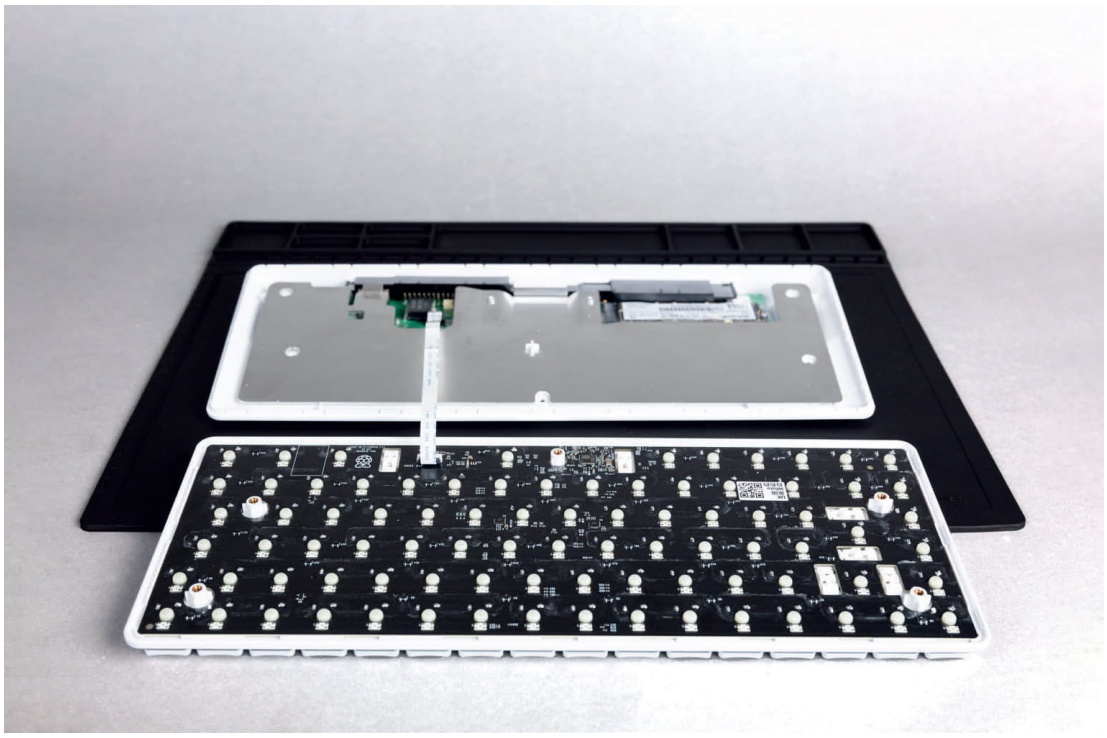
Raspberry Pi 500+ is easily accessible with five standard screws



A spudger is included to get inside the case

**CUSTOM SCREW**

The SSD screw is a custom design by Raspberry Pi. Normally these screws are found on the underside, but Raspberry Pi wanted the drive to be accessible without going underneath the board.



Flipping the keyboard up to take a look inside

*Gently lift up the keyboard and remove the cable*

## Going deeper

Once you have flipped the keyboard off the top of Raspberry Pi 500+, you will see the metal head spreader beneath. This can also be removed to provide access to the mainboard.

You will find inside the plastic light port blocker which, as the name suggests, blocks the light from the RGB keyboard from escaping out of the ports at the rear.

On the mainboard you will find the CPU, 16GB RAM (not upgradeable, sadly), and the connections. It is essentially a Raspberry Pi 5 transformed into a size suitable for the case of Raspberry Pi 500+.

Keen eyes will notice the PoE (Power over Ethernet) circuit that is unpopulated. While those with sharp soldering skills may be able to get PoE installed, this isn't planned as a standard feature and 500+ isn't qualified for PoE.

There is a J5 connector for a power cell battery if you want to add time permanence to your 500+. And there is a UART which can be used to obtain early boot stage information.

### ELECTROSTATIC PROTECTION

You will see a small pad fixed to the metal plate. Its primary function is to help with electrostatic discharge (ESD) protection, ensuring that static electricity is safely discharged and does not damage the keyboard's internal electronics.

### Mouse 2

The eagle-eyed might have spotted the new Mouse 2. This looks the same as the standard Raspberry Pi Mouse, but has redesigned innards and a new all-white design (instead of red and white). The sensor and performance remain as excellent as before. There is also a black model of Mouse 2 available.

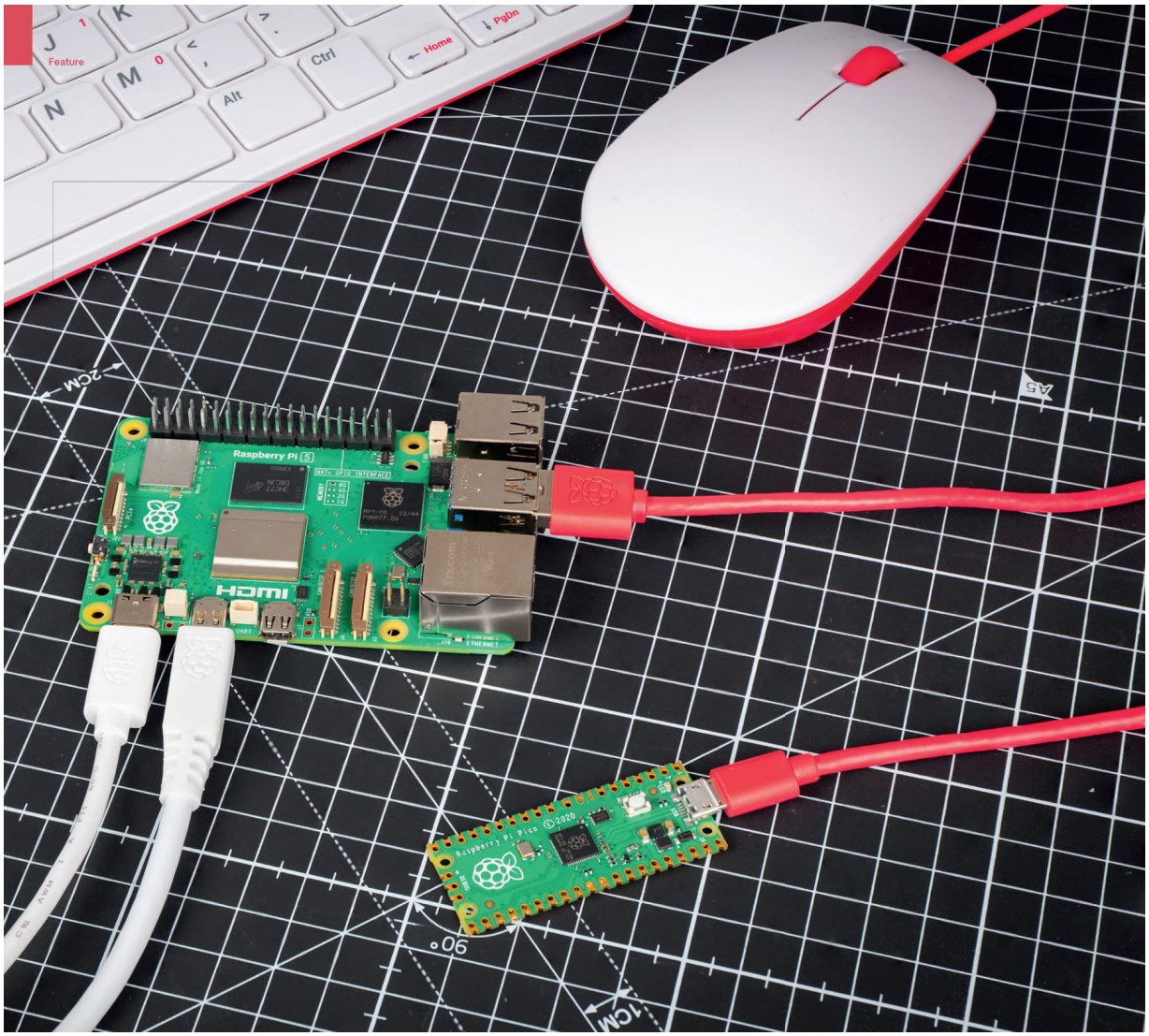
**rpimag.co/mouse2**



OceanofPDF.com

# PROBLEM SOLVING

## Get better at setting up and taking care of your Raspberry Pi devices. By Lucy Hattersley



## *You'll become a smarter tech troubleshooter*

**R**aspberry Pi is the computer for everybody. One group of people that it's especially good for is those of us who really do love computing.

The great thing about Raspberry Pi is you can take it apart, put it together, and start to understand what's going on inside. Unlike the glass and glue slabs that pass for computers, Raspberry Pi is a real computer that you can fix, and break, and fix again.

And that's what this feature is all about. We're going to look at setting up Raspberry Pi, along with some tips for picking the right accessories, and problem solving Raspberry Pi.

There is bound to be something here that you don't know. So read this feature and you'll become a smarter tech troubleshooter.

# Power up

The bare minimum you need to get started with a Raspberry Pi computer is a power supply and some form of boot media: typically a microSD card, but increasingly an SSD or USB thumb drive.

Newer Raspberry Pi models, from Raspberry Pi 4 / 400 onwards, use a USB-C connection to provide power; Raspberry Pi Zero and Raspberry Pi 3 (or earlier) models use a micro USB connection. So make sure you have the right cable to hand.

More important is keeping an eye on how much wattage your power supply is capable of. Many Raspberry Pi owners repurpose a power supply from other devices; using a standard phone charger is common.

There are lots of 5V/3A power supplies available that supply 15 W (V × A = W), which is enough for a Raspberry Pi Zero model, or Raspberry Pi 4/400 or earlier. Newer Raspberry Pi 5/500/500+ models operate best with a

27 W power supply and for these we recommend getting a Raspberry Pi Power Supply (**rpimag.co/powersupply**).

It is possible to run a Raspberry Pi Zero / Zero 2 or 4/400 or earlier with 15 W of power, you may run into problems with USB attachments that draw power from the device (such as mice, keyboards, thumb drives, or any HAT attached hardware).

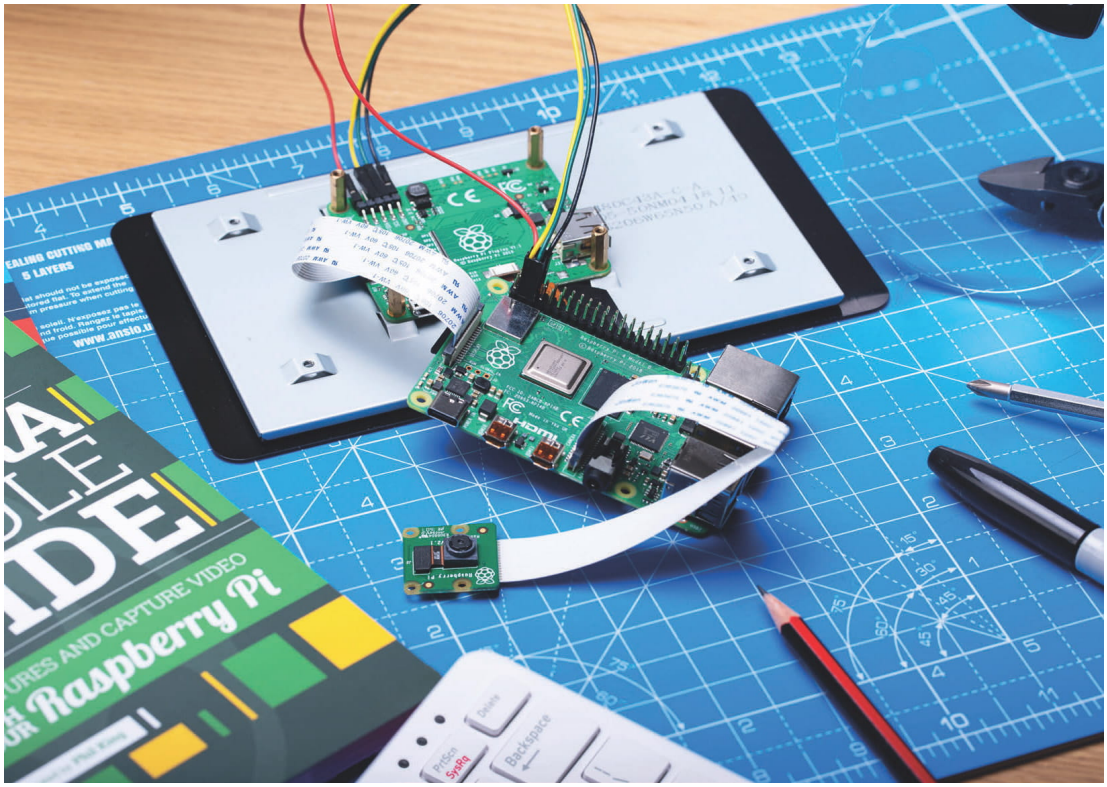
If your Raspberry Pi does not power up – or keeps resetting – then the first thing to look at is the power supply. Check it against the 'Recommended power supply' table and if it falls short, pick up an appropriate charger, such as a Raspberry Pi power supply (**rpimag.co/2wpowersupply**).




Raspberry Pi's Official USB Power Supply provides enough wattage to comfortably run the latest models

## FIND YOUR RASPBERRY PI

Take a close look at the board and it will tell you what device you are using, such as ‘Raspberry Pi 4 Model B’ or ‘Raspberry Pi 5’. Raspberry Pi Zero has it written on the reverse of the board, as do keyboard models such as Raspberry Pi 400 and Raspberry Pi 500. Head to **rpimag.co/products** to get a visual overview of all Raspberry Pi hardware.




Identify your Raspberry Pi by looking closely at the board

# Pick your boot media

Most Raspberry Pi models use a separate storage medium to boot the operating system. Typically, this is a microSD card, which slots into Raspberry Pi and contains an operating system image.

Raspberry Pi OS is the official image, but Raspberry Pi models can run a range of operating systems including Ubuntu and Recalbox.

You can also boot modern Raspberry Pi computers from USB storage, network storage, and via storage attached to the PCI Express 2.0 interface on a Raspberry Pi 5 (typically accessed via the Raspberry Pi M.2 HAT+, **rpimag.co/m2hatplus**).

You will want to use at least a Class A2 microSD card. The speed of the card is likely to affect the performance of the operating system. Raspberry Pi sells its own official SD Cards that are rigorously tested to ensure optimal performance (**rpimag.co/sdcards**).




Raspberry Pi's official microSD card has been rigorously tested

# Recommended power supply

| Model | Recommended power supply (voltage/current) | Raspberry Pi power supply |
|---|---|---|
| Raspberry Pi 5 | 5V/5A, 5V/3A limits peripherals to 600mA | 27W USB-C power supply |
| Raspberry Pi 4 Model B | 5V/3A | 15W USB-C power supply |
| Raspberry Pi 3 (all models) | 5V/2.5A | 12.5W micro USB power supply |

| | | |
|---|---|---|
| Raspberry Pi 2 (all models) | 5V/2.5A | 12.5W micro USB power supply |
| Raspberry Pi 1 (all models) | 5V/2.5A | 12.5W micro USB power supply |
| Raspberry Pi Zero (all models) | 5V/2.5A | 12.5W micro USB power supply |

> ## HOW BIG A MICROSD CARD?
>
> We define ‘ample space’ as at least 8GB to run Raspberry Pi OS with a graphical interface, or 4GB to run Raspberry Pi OS in ‘headless’ mode (accessible only via the command line or over a network). Only the first-generation Raspberry Pi uses a full-sized SD card; all models from Raspberry 2 onwards support microSD.

# Other cables

If you want to use Raspberry Pi directly, you’ll also need the following:

- a display
- a cable to connect your Raspberry Pi to your display
- a keyboard
- a mouse

Most of these items are easy to source and many Raspberry Pi owners repurpose old equipment. Raspberry Pi sells its own Keyboard, Mouse, Monitor, and Display Cable if you wish to use official kit (**rpimag.co/products**). You can use any of the USB ports on Raspberry Pi

to connect a USB wired keyboard or mouse. It is also possible to connect Bluetooth keyboards and mice from the Raspberry Pi OS interface.

Raspberry Pi models have a range of different display output ports and it is likely you will need to acquire a dedicated display cable. Raspberry Pi 4/400 and 5/500 models require a micro HDMI to HDMI cable (**rpimag.co/microhdmicable**). Raspberry Pi Zero and Zero 2 models need a mini HDMI cable (**rpimag.co/minihdmicable**).

# Installing your OS

Raspberry Pi computers check for an operating system on the SD card first, followed by the NVMe/USB boot drive and network drive (depending on the order set in raspi-config).

Raspberry Pi OS should be your first port of call for an operating system. Based upon Debian Linux, it is fast and efficient. Raspberry Pi OS features an effective collection of programming, internet, and desktop applications. And it has links to documentation, reference guides, and a Bookshelf application for reading our magazines and books.

Raspberry Pi has created an easy way to install Raspberry Pi OS onto your SD card using a program called Raspberry Pi Imager (available for Raspberry Pi, Mac, Windows, and Linux computers).

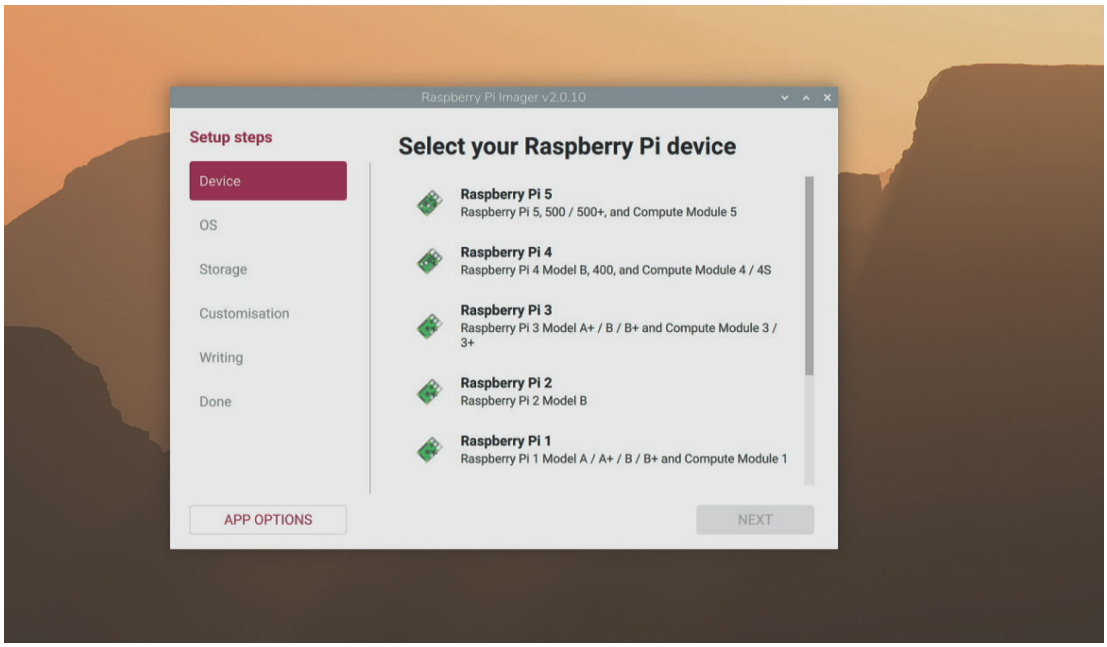
Download the Imager software from **raspberrypi.com/software** and follow the instructions at **rpimag.co/imager**. It is installed in Raspberry Pi OS if you have another Raspberry Pi up-andrunning, or you can install it in terminal:

```
sudo apt install rpi-imager
```

With your card in a USB microSD card reader (**rpimag.co/sdreader**), open the Imager software. Click on your Raspberry Pi device, click Next to select an operating system, and then click Next again to select storage.



Raspberry Pi’s official monitor




Use Raspberry Pi Imager to install an operating system

# Use OS customisation?

Before writing your OS to disk, you can choose customisation options. We can’t recommend highly enough that you click the EDIT SETTINGS

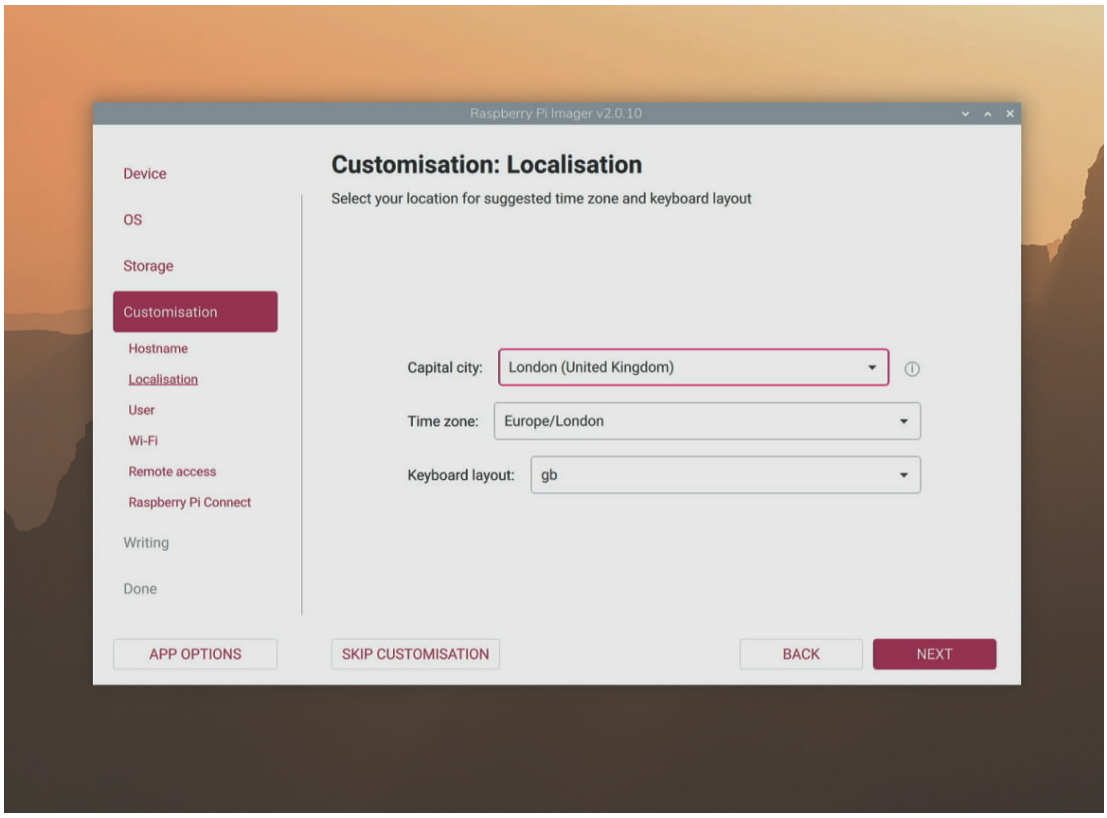
button.

Here are some of the options:

- **Localisation.** Set your local timezone and keyboard layout.
- **User.** It's important to choose a unique user name and password for your Raspberry Pi, otherwise there is a default user of pi and password of raspberry that everyone else can use on yours.
- **Wi-Fi.** Make set-up easier by adding your wireless network details before initial boot.
- **Remote access.** SSH is turned off by default, here you can turn it on for remote connection over a local network or beyond.
- **Raspberry Pi Connect.** Sign into your Raspberry Pi Connect account to connect your Raspberry Pi immediately to the official secure VNC/screen sharing system. You can learn more at **rpimag.co/connect**.

# Network install

It is now possible to install Raspberry Pi OS onto a storage device using a version of Raspberry Pi Imager downloaded over a network. This means that you no longer require a second computer to install Raspberry Pi OS. The only downside is that you need a reasonably recent model of Raspberry Pi (i.e. Raspberry Pi 4/400 or later).




Use OS Customisation to give Raspberry Pi OS custom boot settings

> *Raspberry Pi OS should be your first port of call*

You will also need a wired internet connection via an Ethernet cable, along with a monitor and keyboard.

To launch Network Install, power on your Raspberry Pi while pressing and holding the **SHIFT** key in the following configuration:

- No bootable storage device
- An attached keyboard
- An attached compatible storage device, such as an SD card or USB storage

A red and white screen will appear with ‘Install Raspberry Pi OS using Raspberry Pi Imager’. Follow the instructions to download Imager and

write an operating system to your storage device.

# Boot problems

A Raspberry Pi that is reluctant to boot is a great troubleshooting starter. You'll almost certainly run into this one at some point. Follow these steps:

- **Check the power supply.** First, check that your power supply is connected and that your power supply provides enough wattage for your model of Raspberry Pi computer (see 'Recommended power supply' table). If you have an alternative power supply, try using that one instead.
- **Check LED activity.** If a Raspberry Pi has power but fails to boot, it will – in many cases – flash the LED a specific number of times to indicate what is happening. Take a look at the 'LED Warning Flash Codes' box.
- **Check HDMI cable or network connection.** If you are connecting Raspberry Pi to a monitor, check the HDMI cable is the correct type, and it is correctly inserted. If you have another cable available, try swapping it out.
- **Check boot installation.** Remove the power supply, then carefully remove the storage media and reinsert it. If it still fails to boot, reflash the operating system with Raspberry Pi Imager and double-check you have the correct Raspberry Pi Device selected. If installing Raspberry Pi OS, try using the 'Recommended' version of Raspberry Pi OS. If you are using a different operating system, check with the OS provider that the version you are installing works with your model of Raspberry Pi.
- **Update the bootloader.** Raspberry Pi OS automatically updates the bootloader for important bug fixes. You can manually update the bootloader using a microSD card and Raspberry Pi Imager. Select Misc utility images > Bootloader and select a boot mode: SD Card Boot is recommended; NVMe/USB Boot and Network Boot are the other options. Please see 'Update the bootloader' on Raspberry Pi's

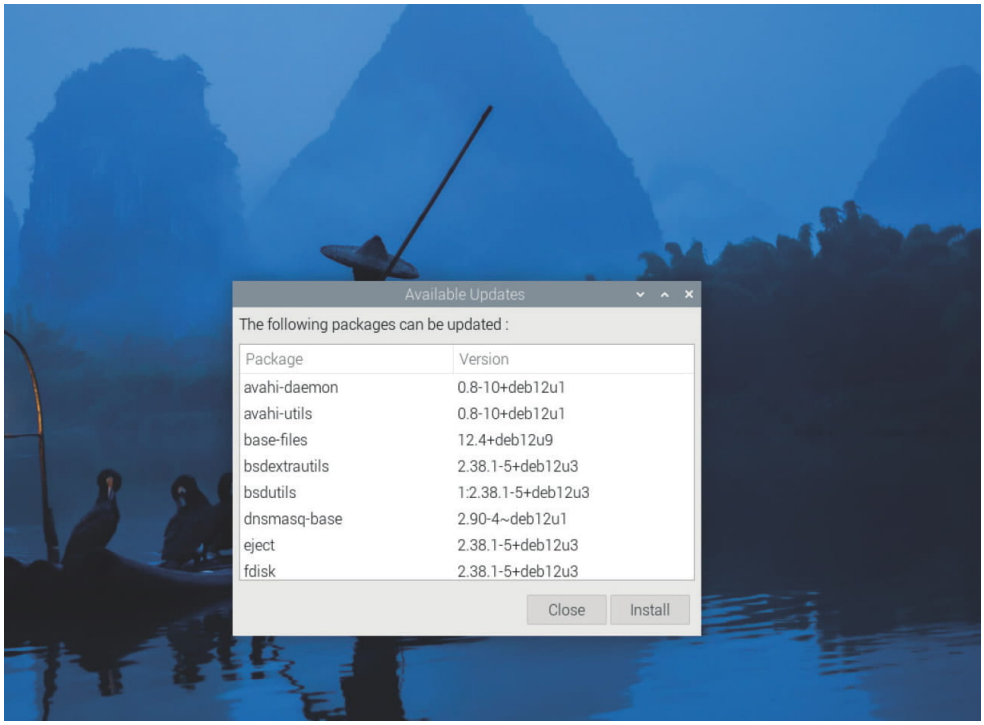
documentation for detailed instructions (**rpimag.co/updatebootloader**).

# LED WARNING FLASH CODES

If a Raspberry Pi fails to boot for some reason, or has to shut down, in many cases an LED will flash a specific number of times to indicate what happened. The LED will blink for a number of long flashes (0 or more), then produce short flashes, to indicate the exact status. In most cases, the pattern will repeat after a two-second gap. **rpimag.co/ledflashcodes**

| Long flashes | Short flashes | Status |
|---|---|---|
| 0 | 3 | Generic failure to boot |
| 0 | 4 | start*.elf not found |
| 0 | 7 | Kernel image not found |
| 0 | 8 | SDRAM failure |
| 0 | 9 | Insufficient SDRAM |
| 0 | 10 | In HALT state |
| 1 | 2 | SD card overcurrent detected |
| 2 | 1 | Partition not FAT |
| 2 | 2 | Failed to read from partition |
| 2 | 3 | Extended partition not FAT |

| | | |
|---|---|---|
| 2 | 4 | File signature/hash mismatch (Raspberr Pi 4 and 5) |
| 3 | 1 | SPI EEPROM error (Raspberr Pi 4 and 5) |
| 3 | 2 | SPI EEPROM is write protected (Raspberr Pi 4 and 5) |
| 3 | 3 | I2C error (Raspberr Pi 4 and 5) |
| 3 | 4 | Secure-boot configuration is not valid |
| 4 | 3 | RP1 not found |
| 4 | 4 | Unsupported board type |
| 4 | 5 | Fatal firmware error |
| 4 | 6 | Power failure type A |
| 4 | 7 | Power failure type B |




Check for updates using the Raspberry Pi OS GUI interface

> # DIG DEEPER!
>
> Click Show Updates to see a list of packages and versions. You can look these up online if you want to get an understanding for what different elements of Raspberry Pi OS and Linux are.

# OS update

If you did not use the configuration tool in Raspberry Pi Imager, the operating system will run a configuration wizard on first boot and guide you through the steps of connecting your devices, Wi-Fi or network cable, browser, and guide you through a software update.

Otherwise, the first thing you should do when booting up Raspberry Pi OS is check for updates. If updates are available, an ‘Updates are available’ icon will appear in the menu bar, click it and choose ‘Install Updates’.

If you are running Raspberry Pi OS headless (i.e. without a GUI interface), update using:

```
sudo apt update
sudo apt full-upgrade
```

# Remote access

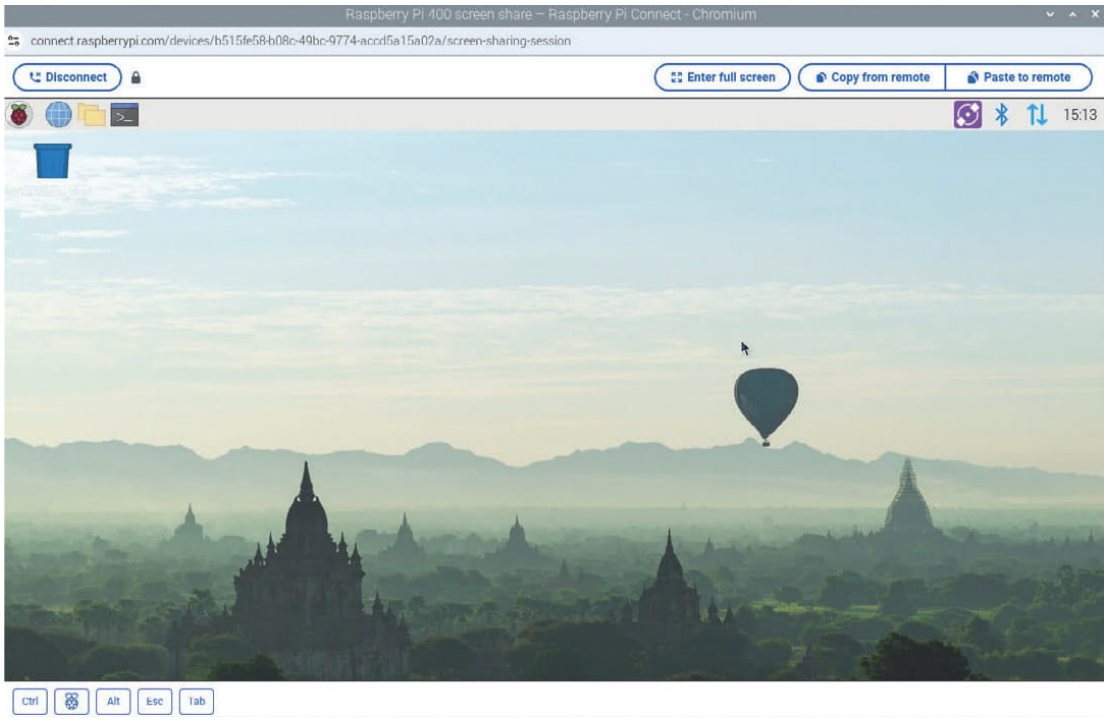
There are three ways to remotely access Raspberry Pi OS and they are all worth learning. Especially if you plan to run your Raspberry Pi without a monitor or keyboard:

- **SSH (Secure SHell).** You can access the text-based terminal of a Raspberry Pi remotely from another computer on the same network.

This is quick to set up and is fast to access, but accessing windows or GUI elements is an arduous process (**rpimag.co/ssh**).

- **VNC (Virtual Network Computing).** This process enables you to control the desktop of one device from another. VNC relies upon a server program (which will run on your Raspberry Pi) and a client program, which runs on the computer you are using to access Raspberry Pi (**rpimag.co/vnc**).
- **Raspberry Pi Connect.** Raspberry Pi Connect is a service provided by Raspberry Pi that provides free, out-of-the-box access to your Raspberry Pi from anywhere in the world. You activate the service in Raspberry Pi OS and can remote-access your computer from the Raspberry Pi Connect web service in any browser (**rpimag.co/connect**).

SSH and VNC options are both found in the Interfaces section of Raspberry Pi Control Centre. Raspberry Pi Connect is found to the right of the menu bar.




Raspberry Pi Connect enables you to access a Raspberry Pi from a web browser

# Update to a new major version

Occasionally Raspberry Pi will introduce a new major version of Raspberry Pi OS; it will be announced via the Raspberry Pi news channel (**raspberrypi.com/news**) and the article will have instructions on how to update to the new version.

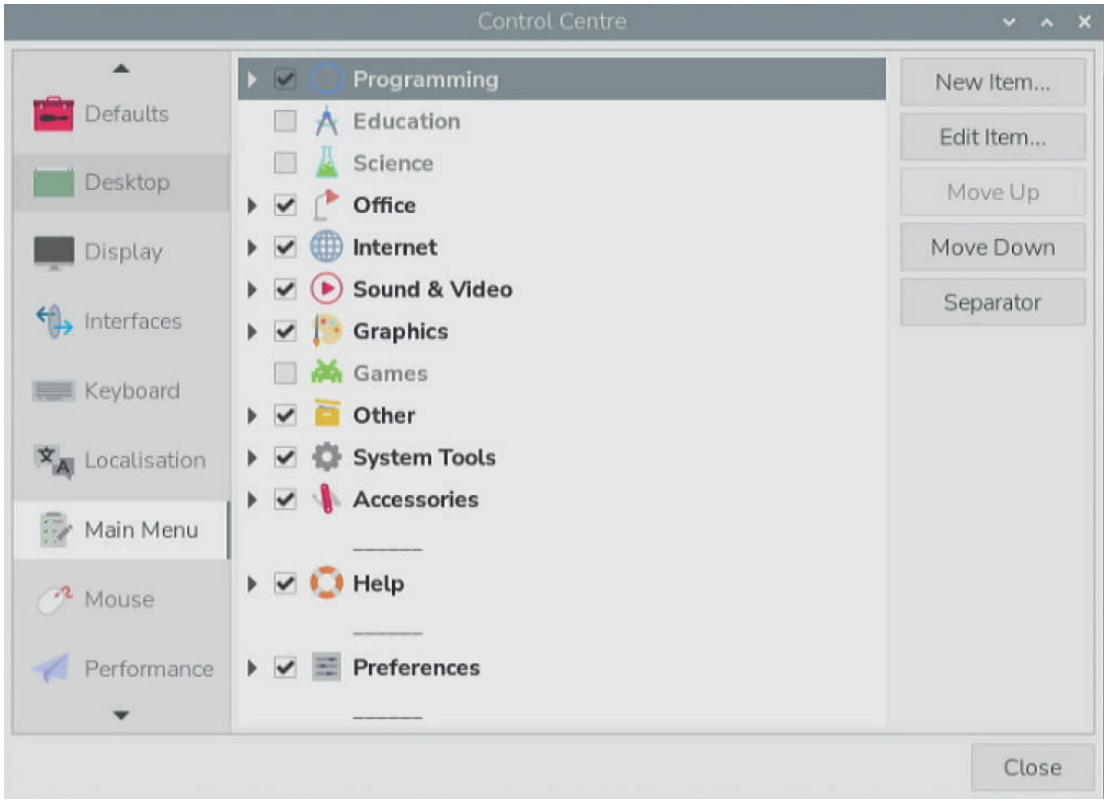
To update the operating system to a new major release on your Raspberry Pi, image a second, separate SD card with the new OS release. You can use a USB SD card reader or network storage to copy files and configuration from your current installation to the new SD card. Then, swap the new SD card into the slot on your Raspberry Pi, and boot.

# Raspberry Pi Control Centre

There are several tools to configure the settings on Raspberry Pi, but Control Centre, the graphical interface on the desktop, is the one you should become most acquainted with. Open Control Centre from Preferences in the main menu.

- **System.** Here you can change your Password, Hostname, Boot, and Login options as well as choose the default web browser (Chromium or Firefox).
- **Display.** Screen Blanking blanks the screen after ten minutes of inactivity (when enabled); the Onscreen Keyboard should appear and hide automatically when a touch-display is attached (and can be set to permanently appear here).
- **Interfaces.** Here you can turn on and off various Raspberry Pi OS system interfaces including SSH, VNC, and SPI.
- **Performance.** Click Configure on Overlay File System to enable a read-only filesystem. Here you can also turn Raspberry Pi 4s Case Fan on or off (and set the Fan GPIO pin or Temperature).
- **Localisation.** You can configure the UI language, Timezone, Keyboard, and Wireless LAN country code here.

These aren't the only configuration options you have in Control Centre. There's also appearance settings, a main menu editor, and mouse and keyboard settings which are used to customise the OS interface. In addition, the Screens tab can be used to identify, arrange, and adjust attached displays.




Adjust settings with Raspberry Pi OS Control Centre tool

# raspi-config

The raspi-config tool helps you configure Raspberry Pi OS from a command-line environment. Open a terminal and enter:

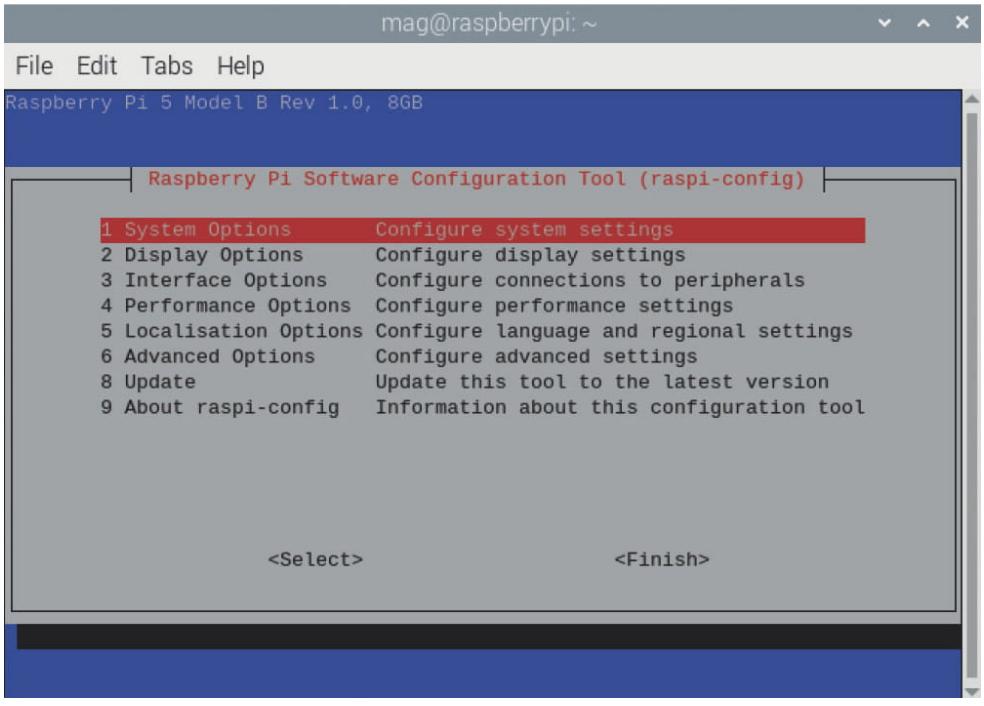
```
sudo raspi-config
```

Changes to raspi-config will **modify/boot/firmware/config.txt** and other configuration files. To navigate the configuration tool from the

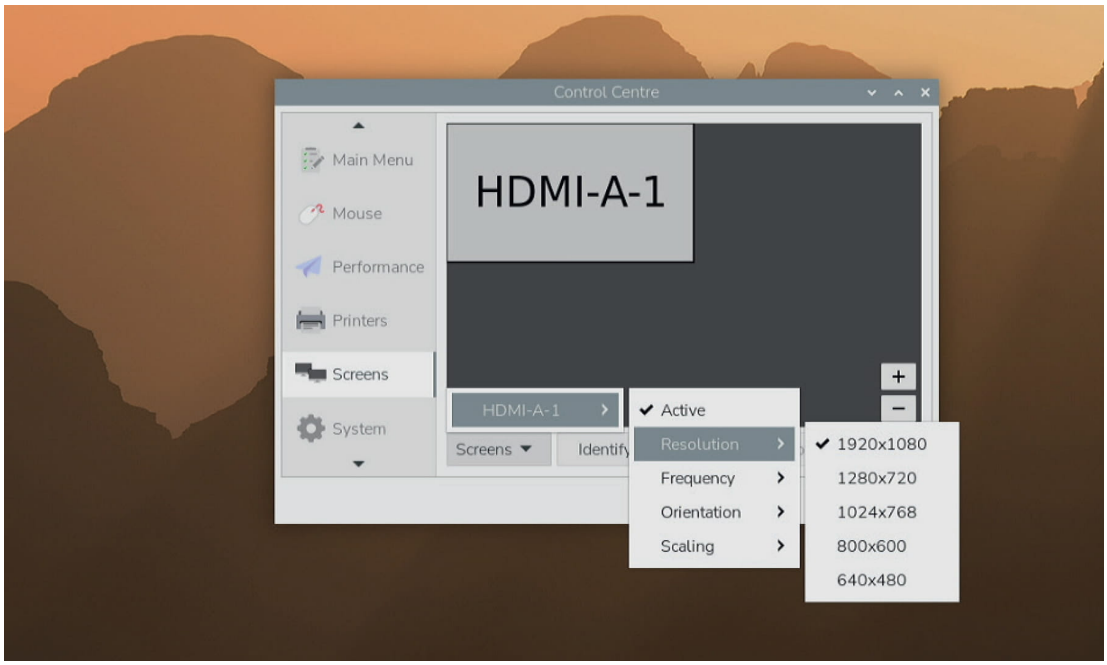
terminal:

- Use the up and down arrow keys to scroll through the settings list.
- Access the <Select> or <Finish> options using the right arrow or **TAB.**
- Return to the settings list using the left arrow or **TAB.**
- Type a letter to jump ahead alphabetically. For example, type E to jump ahead to ‘Europe’ in the time zone list.

> # TIP!
>
> Some advanced configuration is available in the raspi-config CLI, but not the Control Centre GUI.




The raspi-config tool enables you to adjust settings from the command -line interface




Set the screen resolution in the Screens tab of Control Centre

# config.txt

Raspberry Pi OS uses a configuration file called **config.txt**. Raspberry Pi looks for this special file in the /boot/firmware/ directory of the file system.

While it is still possible to edit the **config.txt** file directly from your Raspberry Pi OS installation, Raspberry Pi does not recommend this any more. Instead, we would direct you to the GUI configuration tools or raspi-config.

You can view the current active settings by opening a terminal window and using the following commands:

```
$ vcgencmd get_config <config<
```

… displays a specific config value, e.g. `vcgencmd get_config arm_freq`

```
$ vcgencmd get_config int
```

…lists all non-zero integer config options (non-zero)

```
$ vcgencmd get_config str
```

…lists all non-null string config options.

Take a look at the **config.txt** section of Raspberry Pi's documentation to become familiar with the commands you'll find in **config.txt**.

> # OLDER VERSION
>
> In Raspberry Pi OS Buster and earlier, Raspberry Pi OS stored the boot partition at **/boot/**.

# Adjust HDMI and display

Raspberry Pi uses the EDID of the HDMI-connected monitor to automatically pick the best resolution. You can manually choose a different resolution with the Screen Configuration tool found under Preferences in the menu. Click on Screens and select your connected display (HDMI-A-1 or HDMI-A-2).

> # SCREEN BLANKING
>
> If the screen periodically goes blank, it may be that screen blanking is enabled.
> Check the Display setting in Configurations and ensure Screen Blanking is set to Off.

# Command-Line Networking

It is possible to connect to a wireless network via the command line with `nmcli` (command-line tool for NetworkManager). Open a terminal and enter:

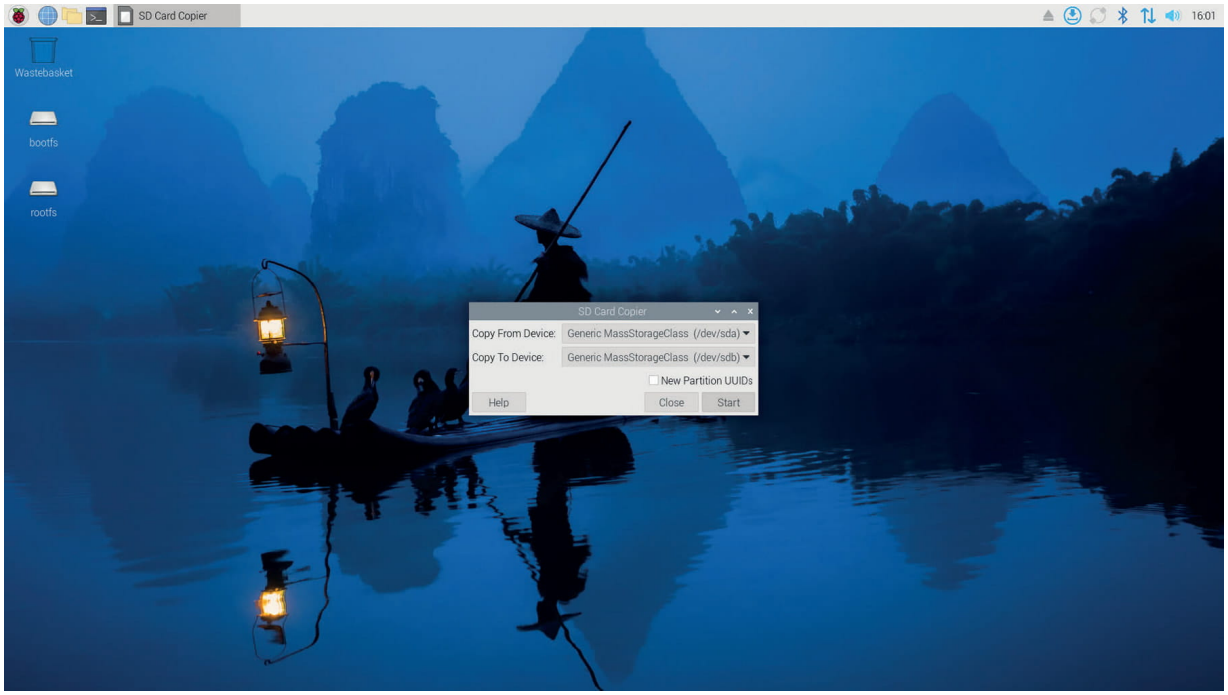
```
$ nmcli dev wifi list
```

…to scan for networks, and:

```
  $ sudo nmcli --ask dev
wifi connect <example_ssid>
```

…to connect. Enter your network password when prompted. For more commands and information, take a look at the Networking section of Raspberry Pi's documentation.

**rpimag.co/networkingdocs**




Use SD Card Copier to back up your Raspberry Pi

# Composite video mode

Composite video can be used to connect Raspberry Pi to CRT displays such as analogue televisions and monitors. In some countries these are widely used, and it's useful for classic retro projects.

To enable composite output, append this to the end of the 'dtoverlay=vc4-kmsv3d' line in **/boot/firmware/config.txt:**

```
dtoverlay=vc4-kms-v3d,composite
```

By default, this outputs NTSC mode. For more information and settings, see **rpimag.co/compositevideo**.

# Wayland vs X11

Since the 2023 version of Raspberry Pi OS (**rpimag.co/bookworm**) was released, all Raspberry Pi computers use a new display system called Wayland. This performs much better than X11, the system it replaced. You can switch between Wayland and X11 with:

```
$ sudo raspi-config
```

Now choose 6 Advanced Options > A7 Wayland > W1 X11.

> *Fortunately, Raspberry Pi OS has a built-in tool called SD Card Copier*

# Adjust the audio mode

Raspberry Pi has multiple output modes: HDMI, the headphone jack (if your device has one), and USB audio. By default, Raspberry Pi outputs audio to HDMI 1. If no HDMI output is available, Raspberry Pi outputs audio to the headphone jack or USB audio device.

- **Desktop volume control.** Right-click the volume icon on the taskbar bar to open the audio output selector. Available devices will be listed; pick the one you want to use.
- **raspi-config.** To change your audio output from the command line when using ALSA audio, open a terminal window and enter `sudo raspi config`. Select System Options > Audio and choose your required mode. Press ENTER to choose the mode and then select <Finish> to exit the configuration tool.

# Create a backup

Once you have Raspberry Pi OS running to specification, it's a good idea to create a backup. This enables you to restore all your files and system settings.

Fortunately, Raspberry Pi OS has a built-in tool called SD Card Copier that enables you to clone your boot drive. Attach a second storage device that is the same capacity (or larger). Open SD Card Copier under Accessories in the menu and choose your boot drive from **Copy From Device**. Select your target backup device from **Copy To Device** and click **Start.**

| Model | Composite output |
|---|---|
| Raspberry Pi 1 A and B | RCA jack |
| Raspberry Pi Zero | Unpopulated TV header |
| Raspberry Pi Zero 2 W | Test pads on underside of board |
| Raspberry Pi 5 | J7 pad next to HDMI socket |

| | |
|---|---|
| All other models | 3.5 mm AV jack |

The backup device will be a perfect copy of your boot drive and you can boot your Raspberry Pi from it.

# HAT+ and hardware

One of the brilliant things about Raspberry Pi is the ability to attach physical hardware to the GPIO pins. And that there are many, many third-party boards you can use to extend Raspberry Pi's functionality. Everything from motor controllers for robots, to AI accelerators, to audio DACs (and much more besides).

Raspberry Pi has created a HAT+ (Hardware Attached on Top) standard for third-party manufacturers. These HAT+ devices contain an EEPROM chip that holds the board manufacturer information, GPIO setup, and a device tree fragment (this allows Raspberry Pi OS to automatically load up the required drivers). If your hardware is HAT+, it should – in theory – work as soon as it is connected.

You can read more details about the new HAT+ standard in this blog post: **rpimag.co/doublestandards**.

The complete HAT+ specification can be found at **rpimag.co/hatplusspec**.

If you are having trouble with HAT+ hardware, power off your Raspberry Pi and try reseating the HAT+ hardware on the GPIO pins (carefully to ensure it is sitting on the pins correctly). If it still fails, try updating Raspberry Pi OS, updating the firmware, and check your power supply is enough to power both the device and HAT+ hardware.

Raspberry Pi has documentation on all its HATs, which you can find here: **rpimag.co/accessoriesdocs**.

Third-party makers provide their own documentation which you can consult to troubleshoot the device.

# Getting physical

Raspberry Pi is hardware and you should take steps to protect it physically as well as on the software layer.

- **Raspberry Pi Bumper.** We love the Bumper! It's a snap-on silicone cover that protects the bottom and edges of the board. It also prevents slippage and can help protect from vibration (**rpimag.co/bumper**).
- **Raspberry Pi Case.** Moving up from the Bumper is the Raspberry Pi Case. These combine high-quality construction with a minimalist design that enables you to quickly remove the lid and access the board. Raspberry Pi 5 models also have an integrated fan (**rpimag.co/case**).
- **Third-party cases.** There are many other cases available that offer a range of features above and beyond Raspberry Pi's official case. The HighPi Pro case offers a large internal volume area for HATs and VESA mount support (**rpimag.co/highpipro**), while ComfilePi offers touch panel cases with high-tolerant parts for industrial applications (**comfiletech.com**). Argon offers a popular range for home media that include infrared remote control, DAC audio, and large storage (argon40.com)
- **Cooling.** Raspberry Pi computers are designed to operate safely without cooling. If you want to put a Raspberry Pi under stress, then cooling will help it stay at a comfortable operating temperature, even under a heavy load. The official option is a Raspberry Pi Active Cooler for Raspberry Pi 5 that combines an aluminium heatsink with a temperature-controlled blower fan (**rpimag.co/activecooler**).

# COMPOSITE VIDEO PHYSICAL LOCATION

Composite video output can be found on some models of Raspberry Pi computer – see the table on the page opposite. Note: composite video mode is not found on keyboard models.

# NEW PARTITION UUIDS

By default, SD Card Copier will clone the partition UUIDs (Universally Unique Identifiers) which means you get an exact copy of the drive. This makes it difficult to mount both disks at the same time. It isn't a problem if you expect to store the backup, but if you want to keep it connected to Raspberry Pi and boot from either device, select the New Partition UUIDs checkbox.



Use a Raspberry Pi Bumper or a case to protect your computer

> *Raspberry Pi and digital computing is a vast subject*

# Getting help

Raspberry Pi and digital computing is a vast subject. You will routinely come across something you don't understand. Don't be afraid to look for help. Here are some places to get started.

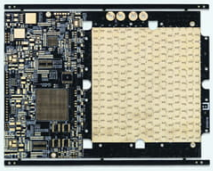
- **Help section**. Choose the Help section from the Raspberry Pi OS desktop menu. Here you'll find links to starter guides, Debian reference guides, and projects. Bookshelf is also here, which contains digital PDF editions of *Raspberry Pi Official Magazine*, its predecessor *The MagPi*, and various books.
- **Documentation**. Make sure you look at, and bookmark, Raspberry Pi's comprehensive Documentation portal (**rpimag.co/docs**). Here you will find comprehensive links and you can search by subject (**CTRL+K** accesses the search box).
- **Forums**. Raspberry Pi maintains a vibrant forum user base (**rpimag.co/forums**).

*Beyond the Standard* ——

# Explore PCBWay's Advanced PCB Solutions

## About PCBWay

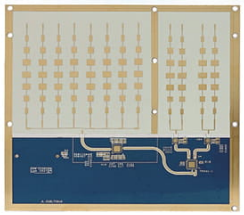
With specialized production facilities for advanced PCB manufacturing, PCBWay delivers high-precision, high-reliability solutions for demanding applications. From prototypes to large-scale production, we provide advanced PCBs that meet strict quality standards while maintaining competitive costs.



### HDI PCB

High Density Interconnect Technology

- 4–60 layer capability
- Advanced materials: FR-4, High-Tg, Halogen-Free, PTFE
- Finished copper thickness: 1/3oz–8oz
- Fine line/spacing: down to 2/2mil (inner layer / HOZ base copper)
- Supports micro BGA pitches from 0.35mm

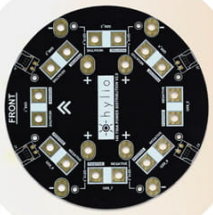


### High Frequency PCB

Engineered for High-Speed Signal Performance

- Compatible with premium RF materials: RO4003C, RO4350B, RO3003, RO3010, RT5880
- Minimum trace spacing: 2mil
- Minimum hole size: 0.15mm
- Finished copper thickness: 1–2oz
- Board thickness range: 0.2–3.2mm





### Advanced PCB Technologies

Specialized Processes for Complex Applications

- Embedded Copper PCB
- Backplane PCB
- Cavity PCB
- White / Partially Hybrid PCB
- Black Core PCB (After-dark Applications)

**Advanced Manufacturing. Reliable Quality. Global Support.**

www.pcbway.com

service@pcbway.com

# SUBSCRIBE TODAY
# FOR JUST £10

## Get **3 issues** + **FREE** Pico 2 W

### SUBSCRIBER BENEFITS

**Free delivery**
Get it fast and for free

**Exclusive offers**
Great gifts, offers, and discounts

**Great savings**
Save up to 37% compared to stores

### SUBSCRIBE FOR £10

**Free Pico 2 / Pico 2 W**

**3 issues of Raspberry Pi Official Magazine**

**£10** (UK only)

### SUBSCRIBE FOR 6 MONTHS

**Free Pico 2 / Pico 2 W**

**6 issues of Raspberry Pi Official Magazine**

**£30** (UK)

**$43** (USA)

**€43** (EU)

**£45** (Rest of World)




Subscribe by phone: +44 (0)1778 392472

Subscribe online: rpimag.co/subscribe

Email: **raspberrypi@warnersgroup.co.uk**

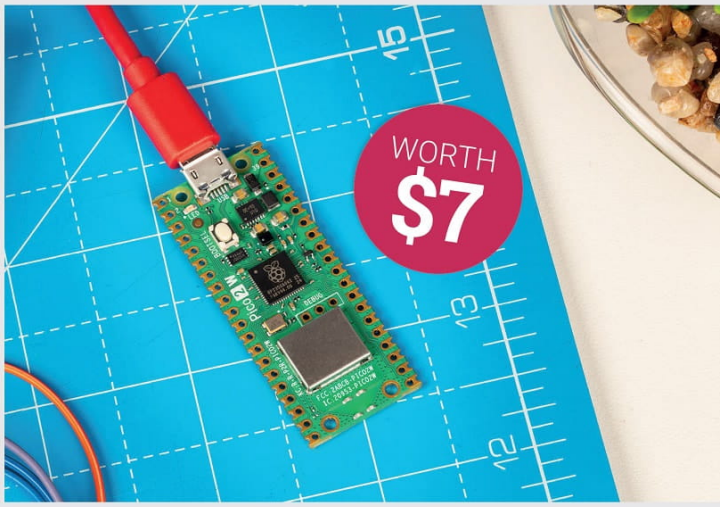
Subscribe for £10 is a UK-only offer. The subscription will renew at £15 every three months unless cancelled. A choice of free Raspberry Pi Pico 2 W or Pico 2 is included with all subscriptions.

SUBSCRIBE TODAY AND GET A

# FREE Raspberry Pi Pico 2 W (or Pico 2)

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

- A brand new RP2350-based Raspberry Pi Pico 2 W / Pico 2 development board
- Learn to code with electronics and build your own projects
- Make your own home automation projects, handheld consoles, tiny robots, and much, much more




Free Pico 2 or Pico 2 W. Accessories not included. This is a limited offer. Not included with renewals. Offer subject to change or withdrawal at any time.




Buy now: **rpimag.co/subscribe**

**SUBSCRIBE** on app stores

From **£2.29**

Available on the App Store

GET IT ON Google Play


[OceanofPDF.com](OceanofPDF.com)

# Project Showcases

**HexBoard**
A hexagonal MIDI controller

**Mini Air Hockey games table**
Heavily customised gaming

**Raspberry Pi Chessboard**
Test your cerebral might

**Raspberry Pi Fruitbox Jukebox**
Dial up the grooves




**Alpakka v1.0 controller**
A modular controller that you print yourself

**Whispering Wires**
Poetry by phone



**Bipedal companion robot**
Immobile desk assistant

**Operation Pico**
As in the classic boardgame

**Zero-G Lab Floating Platform**
Simulating space robotics

**Pepper Pi**
Victorian illusion, futuristic computing

**Avanade Intelligent Garden**
Interactive smart flowers



**Embedded MIDI Synthesizer on Raspberry Pi**
An electric accordion

**Reflecting telescope**
Turning a pipe into a telescope

**DOOM SNES 2025**
Pico-powered retro revival

**Vertical Runner**

A functional movie prop



**Bog body theatre prop**
Spooky performance art

**Yashica IR camera**
Surreal IR photography

**Sfera Baltic boat**
Upcycling a 60-year-old tug boat with Raspberry Pi

**Electronic Drum Business Card**
Playable contact details



**Flokk herd management**
Canadian cattle tracking

**Amstrad PPC 640 Cybderdeck**
The classic computer made much cooler



*OceanofPDF.com*

# HexBoard

## A unique way to make music. By **Andrew Gregory**



**Maker**

**Jared DeCook**

Stay-at-home dad of two wonderful (and very small) kids. He primarily work on projects during nap time, but loves finding parts they can work on together.

**shapingthesilence.com**

**Musical instruments are dependent on physics and maths for their sounds.** A guitar string changes pitch when its physical properties change, be that string thickness, tension or length. A note from a saxophone, or any wind instrument, changes according to the length of the instrument that the air is vibrating through. Any physical instrument needs to have some physical property that the player must interact with in order to play music, and this naturally limits who can play, and what can be played.

At least, that's how it always was. With MIDI and computers, this no longer has to be true. The instrument that you play is nothing more than an input device, like a keyboard and a mouse, and as such, it can be any shape or size that the maker wants. In this case, the maker Jared DeCook, maker

of the HexBoard, has chosen to make a grid of hexagonal keys and RGB LEDs, all controlled by a Raspberry Pi RP2040 chip.

## Custom controller

“The HexBoard is primarily a MIDI instrument, which means that it sends note information to software or hardware sound generators that play the sounds based on the notes you play,” Jared tells us. “It has a USB-C connector for connecting to MIDI software on a computer and a TRS MIDI jack for connecting to hardware MIDI devices. It has a very basic built-in sound generator that can play over headphones or built-in speaker, which is nice for when you want a quick and simple setup. All the unused pins from the RP2040 are exposed as headers on the board for ease of hackability for those interested in taking the hardware farther.”

On a piano, each key has two notes on either side of it; on the HexBoard, each key is touching six further keys, so you can physically reach more notes with one hand – which is useful if you don’t have massive mitts like Franz Liszt or Jimi Hendrix.




**01.** The HexBoard has 140 individual mechanical switches, each with a 3D-printed keycap lit by an addressable RGB LED

**02.** The unusual keyboard gives rise to unique chord shapes and melodic ideas

"The lit-up hexagonal keys allow for many different note layouts while indicating which note is which using colours," says Jared. "The different layouts included make it easy to play chords using simple finger shapes that don't change depending on the key, unlike an instrument like the piano. I designed the keycaps in TinkerCAD and designed the frame in Blender. I print everything on the Bambu Lab P1S, which just barely fits the frame into the build area."

## Why use an RP2040?

"The RP2040 chip is the brains of the whole project," he adds. "It reads the buttons, sets the lights, updates the screen, sends MIDI data, and generates

simple sound waves. No proprietary software is needed to run this as it's all handled by the RP2040.

"I appreciate how I can use PIO to handle updating the LEDs without blocking the CPU. I selected this chip for this project as it had all the I/O my project required with a good amount of headroom for future developments. That, plus it was very available during the big chip shortage that occurred in the early stages of this project."

> *I come up with different ideas on the HexBoard than I do on a standard musical keyboard*

While big companies may relentlessly focus group products to make sure they find an audience, Jared originally made the HexBoard for a pretty small audience: himself. We reckon argue this might have enabled him to focus on making the HexBoard awesome.

"Originally, I made it for me and my brother," he recalls. "I wanted a fun, hackable instrument that looked cool and was relatively easy to learn. I'm quite happy with where I am on that, so I'd say the maker/hacker/musician is my primary target.

"As the project has advanced, it's become useful to a variety of people. A beginner musician can use it to start learning basic chords and melodies. Any given melody or chord has the same shape/movement no matter what key you are in, so in many ways it has a simpler learning curve.

"A music producer can use the various layouts to spark inspiration as each different layout gives a unique way to consider the relationships between notes. I know for me, I come up with different ideas on the HexBoard than I do on a standard musical keyboard.

"With the latest update, it's now a useful tool for those experimenting in microtonal music. The compact size and wide range make it very useful for musicians who want access to a large number of keys while still being able to easily carry it around in a backpack. Beyond this, the colourful lights and animations make it plain fun for anyone to play around with."

## Quick FACTS

- Jared built a similar device in 2013, inspired by the Axis 49 instrument…
- …when this started to fail in 2021, he began work on the HexBoard
- It originally used a Teensy LC board as the source of the processing power…
- …but this was replaced with an RP2040-based module designed by Jared
- HexBoard is still under development, with new features being added all the time




Jared launched the HexBoard in May 2023, and has since overhauled the design to make it easier to assemble and repair, and to add new features



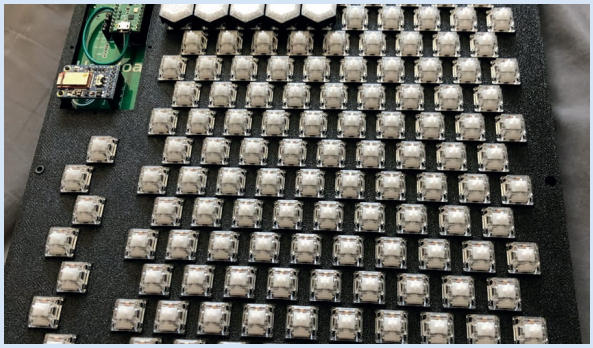



The screen enables users to navigate through the many layout options offered by the HexBoard

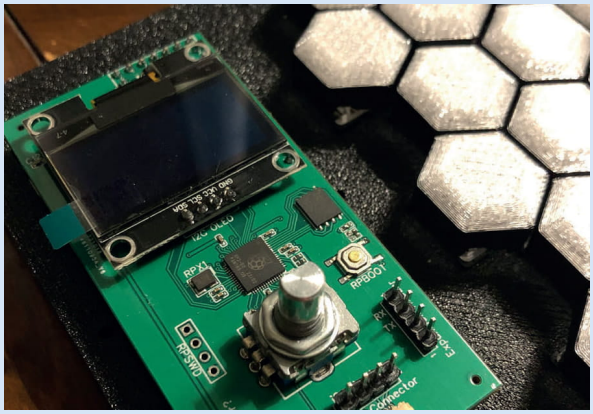
# How to build a HexBoard




1. Design a custom PCB with all the ports you need, breakouts to the RP2040 module, and room for 140 mechanical switches.



2. Solder the switches in place, then fire up the 3D printer and get to work creating a grid of tessellating hexagons.



3. Meanwhile, create your own custom RP2040-based board – add this to the PCB to handle all the ins, outs, and programming.

OceanofPDF.com

# Mini Air Hockey games table

## Offering a physical workout as well as a tactical one, this Raspberry Pi-controlled games table is a fun build. By **Rosie Hattersley**




**Maker**

**Chris Downing**

Chris is “a 42-year-old console modder from the backwoods of central New Hampshire” while CrashBash is a fellow modder and software engineer.

**downingsbasement.com**


**For the past 15 years, Chris Downing has been part of a community that takes old TV console games and turns them into portable versions using original hardware wherever possible, posting details on his site, Downing’s Basement (downingsbasement.com).** Raspberry Pi had “always been in the scene”, with many hobbyists using it to make portable emulation handhelds, “which in its own right was very cool”, says Chris, even though it was against the group’s “purist original hardware culture”.

Nonetheless, when it came time to create his own Doom-playing Mini Air Hockey Table and scoreboard, he instinctively turned to Raspberry Pi.

## Multicoloured swap shop

A change of company gave Chris access to bigger and more robust CNC routing tools and large flatbed printers, while a Yankee swap at the office party netted him a kids' mini air hockey table that went down a storm with his 5- and 7-year-old children. This spurred Chris on to create his own, using the Raspberry Pi 2B he'd bought for a portable project that had never materialised.

Given his interest in retro gaming, Chris didn't want a plain vanilla air hockey table, instead envisaging one with a DOOM theme and plenty of additional gameplay options. He's "a designer and fabricator at heart" with plenty of technical knowledge from his retro games builds. As well as the mini air hockey table, Chris already had a suitable ATX power supply, a fan to cushion the pucks, and scoreboard screens. He used CAD software to design and 3D-print parts for the screen housing and other structural elements. However, he needed an LED controller and an OS for the scoreboard interface and automatic score-detecting puck traps.




The Hyperion lights mimic the action when a games console is attached




**01.** Flashing lights in the legs and a celebratory sound are triggered when a point is scored on this Raspberry Pi Zero air hockey table

**02.** A video wall of LEDs underneath the 1/8-inch thick acrylic playing field are controlled using a Raspberry Pi build of Hyperion open-source software

## Quick FACTS

- Goal-scoring sound effects are played by using aplay
- Aplay plays a randomly selected audio file
- Delay-based ‘de-bouncing’ prevents players being wrongly awarded multiple goals
- Air Hockey Doom mode actually allows you to play a game of Doom
- Consoles and emulation mode are all supported

### Play to your strengths

Chris says coding is his biggest Achilles’ heel. “You play to your strengths the best you can, but when they fall short in an area, having someone who can bridge the gap is invaluable.” In this case, Chris’s modding buddy CrashBash was immensely more experienced with Raspberry Pi and able to offer professional software engineer input. “This project would not have gotten anywhere near my vision without his help,” says Chris.

There was one slight problem: the two men live more than 800 miles away from each other in different US time zones and would have a single weekend to work on the air hockey/DOOM project in person. As a result, recalls CrashBash, “the theme of the software-side was to, an extent, to make it able to be hacked together in such a way that it could be mostly completed in a weekend”.

Raspberry Pi was used for the puck trap sensors and scoreboard inputs. CrashBash quickly realised that the ‘Hyperbian’ Raspberry Pi build of the Hyperion Ambilight LED controller would be ideal and could be easily integrated using a Bash script that executes at boot. They would later add a

Raspberry Pi Pico and use leftover GPIO pins and a serial port to have lights flash on the scoreboard struts too.

> *Raspberry Pi was used for the puck trap sensors and scoreboard inputs*

Hockey rink and DOOM hockey modes use Python and the Ambilight Hyperbian effects API to generate static along with dynamic images such as DOOM's fire mode. Hyperbian even includes the option to create a video wall in which the LEDs recreate the images from a composite video capture card. "You get a rather low and somewhat lagged live representation of what is being played down on the main deck," says Chris, "but it still looks freaking cool!"

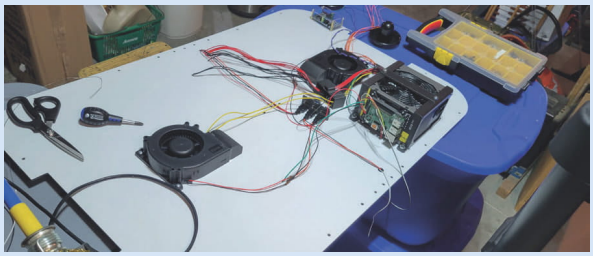


Chris demonstrates player modes and other control options in his YouTube video: **rpimag.co/airhockeyvid**

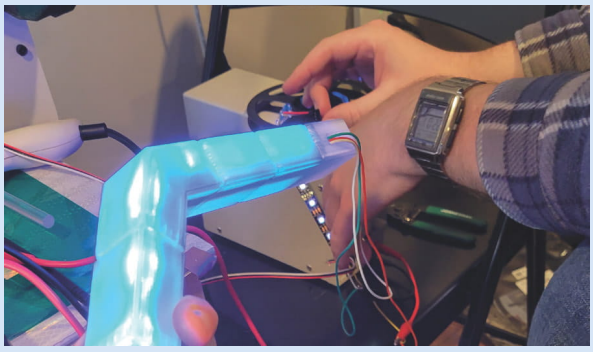



Chris designed and 3D-printed the mode-specific scoreboard display

## Ready player one



1. Adapt an existing gaming rig or build your own, adding Raspberry Pi, power supply, scoreboard display, fan to make the pucks ‘float’, and any lights and speakers you wish.



2. Downstall and install Hyperbian (**rpimag.co/hyperbian**), a ready-to-use OS including the Hyperion LED controller. A string of addressable LED lights attached to Raspberry Pi's GPIO pins can be made to respond to the air hockey or gaming action.



3. CrashBash wrote a Python script for Raspberry Pi Pico that recognises when a puck enters the trap, adjusting the score and triggering Hyperion to play celebratory sounds and lights.

*OceanofPDF.com*

# Raspberry Pi Chess Board

## Teenage strategist Tamerlan created a remote chess board in which a Raspberry Pi-controlled robot arm makes all the moves. By **Rosie Hattersley**




**Maker**

**Tamerlan Goglichidze**

High school student Tamerlan has a passion for engineering and innovation and initially taught himself using online videos.

**rpimag.co/piboard**


**California high school student Tamerlan Goglichidze has been coding and building projects since he was 11, teaching himself essential skills using online videos and tutorials before progressing from virtual to physical creations once his knowledge base gave him the confidence.** This early focus would be impressive enough in itself, but Tamerlan was also busy taking on all-comers as a chess player who travelled extensively to compete in International Chess Federation matches. His Raspberry Pi Chess Board combines the two interests.

## Winning strategy

He chose Raspberry Pi 4 as “the brain of the system, bridging the digital chess world with physical movements on the board”, noting its ability to handle advanced and complex systems as well as the versatile OS and accessible GPIO pins. Tamerlan felt it “was the best choice to handle both the chess engine calculations and stepper motor movements.”

His Raspberry Pi Chess Board project built on his experience of creating a Raspberry Pi-based MARS rover with a robot arm and a small JARVIS (from *Iron Man*) mechatronic simulation with animatronic servo eyes that move around and also integrated voice recognition and artificial intelligence.

### Quick FACTS

- Each chess piece contains a magnet, moved using a stepper motor
- To prevent the stepper motor returning home…
- …Tamerlan created an optimisation algorithm for his 8x8 grid…
- …with 2200 stepper motor steps per chess board side…
- …and a global position variable

## Pure magnetism

Seeing DIY but expensive automated chess boards online motivated Tamerlan to create a functional, budget-friendly version with similar capabilities that would capitalise on his growing engineering skills. He designed most aspects using Fusion 360 and used a standard chess board, cutting wooden boards as a base in his parents’ garage.




**01.** Raspberry Pi 4 remotely controls this Pi Chess Board in which a human player competes against a customisable chess engine

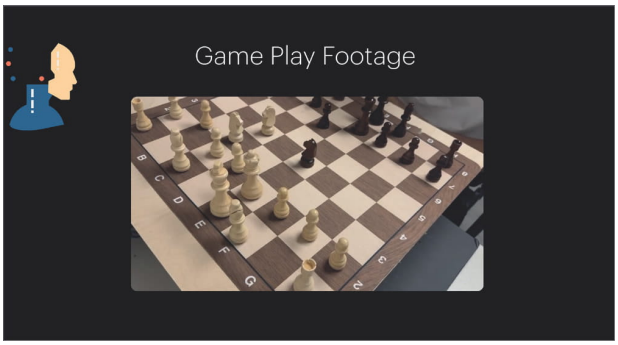
**02.** The stepper motor glides along a rail, moving chess pieces smoothly along the X and Y axes and deftly mimicking the knight’s move



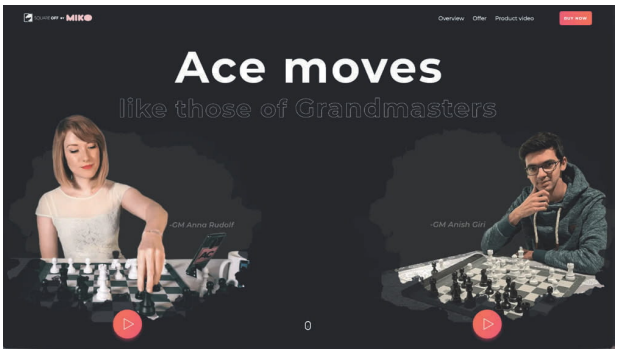
Tamerlan learned all about Raspberry Pi in his earlier Iron Man JARVIS project



Tamerlan pits his wits against his DIY Pi Chess Board




Tamerlan’s impressive website shows his Pi Chess Board in action




He was partly inspired by commercial chess games such as Square Off

Challenges for the automated chess system included accurately weighing each chess piece so the XY stepper motor mechanism and magnets could seamlessly move them across the board. Tamerlan tracked down an algorithm to minimise the stepper motor’s power consumption. He used a servo linear actuator with a magnet to move each chess piece, rather than an electromagnet, having calculated it would be more power-efficient. “To generate a strong enough magnetic field to lift or attract objects, high current is typically required, demanding a robust power supply and generating heat, which must be managed,” he explains. “In contrast, a servo linear actuator needs power primarily during movement, with significantly lower standby power consumption.”

Using magnets moved by a stepper motor rather than a robot arm allowed Tamerlan to make the chess board as compact as possible. He wrote Python code to control each piece, translating the coordinates for each of the board’s 64 squares into steps the stepper motor could execute. Raspberry Pi calculates the best moves according to player input and translates text such as ‘b3b5’ into moves. “Castling is straightforward and

can be handled by writing two different functions to cover the four possible cases and executing them accordingly.”

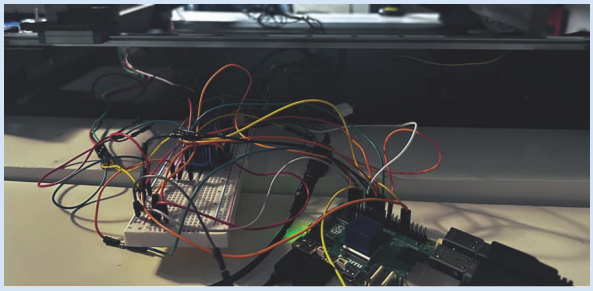
> *Seeing DIY but expensive automated chess boards online motivated Tamerlan*

He also needed a way to move pieces without pushing others out of the way, in particular the knight’s dog-leg movements. His solution has the knight do a half-step between squares, follow its move, and then reverse the half-step.

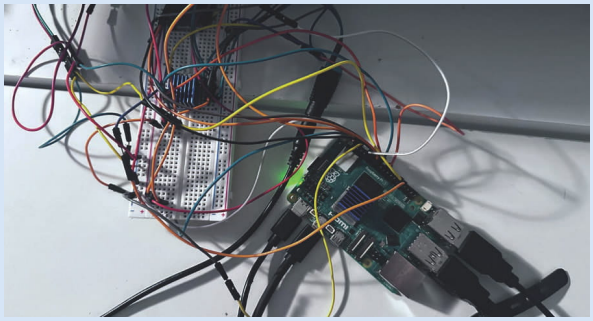
### Peer approval

Players can specify difficulty levels and whether to play as black or white pieces against the computer chess engine, and there is a web option for remote play against other chess fans. The project quickly garnered interest and plaudits online. Buoyed up by the success of his Raspberry Pi Chess Board which “gained remarkable attention” and, after being featured by a number of tech hobbyist sites, led to 40,000 visits to his own website (**rpimag.co/piboard**), Tamerlan has set his sights on a computing or engineering career.

## Calculated moves



1. The project can be made with a Raspberry Pi 4 or 5 and uses a standard chess board and pieces with a sturdy wooden base to which the electronics will be attached.



2. Tamerlan used two NEMA stepper motors, a servo, two linear rails, stepper motor belts, and plenty of jumper cables. He 3D-printed parts, designed in Fusion 360, for the mechanism.




3. Tamerlan's website provides details of how to overcome the issue of overheating electromagnets and inaccurate grabbing of chess pieces by mapping magnets fixed to individual pieces and moving them with the stepper motor.

OceanofPDF.com

# Raspberry Pi Fruitbox Jukebox

## Get into the groove and blast out those tunes with this retro jukebox. By **David Crookes**



**Maker**

**Jeremy Childress**

Jeremy has a degree in electrical engineering with a background in displays and communications, but he mostly works as a project manager now. He also loves teaching his children how things work.

**rpimag.co/piboard**

**It’s always good to come across a project that you can really make a song and dance about.** And, in the case of this awesome, authentic-looking jukebox based around a Raspberry Pi 4 computer, that’s exactly what we’ve been doing.

Maker Jeremy Childress has harked back to the 1950s for his first Raspberry Pi project, producing a machine that makes great use of a popular customisable MP3 player called Fruitbox. He’s done so at a fraction of the price it would cost to buy and fix up a retro jukebox while having lots of fun in the process.

“My mom has always wanted a vintage jukebox and several years ago, my parents bought an old Seeburg DS160 that never worked right,” Jeremy

says. “I tried messing with it a few times, but I could only get it to play a random record and only one at a time. So, I wanted to give my mom a working jukebox and, seeing the expense of a fully functioning vintage one, I thought I could do better for far cheaper.”

## Golden grooves

To begin, Jeremy researched jukebox software to see what was available. “Most of it was for Windows and didn’t fit what I was looking for,” he says. “I then happened to think about seeing if there were any Raspberry Pi jukebox projects I could work with and came across Fruitbox.” Excited, he went to his local Micro Center to pick up a Raspberry Pi 4 Model B 1GB computer. “I didn’t have plans to buy any other hardware until I was sure the software was going to work like I wanted it to,” he says.



The cabinet is completely modular. The marquee, monitor, button board and back panel bolt to the sides so it can be easily disassembled and moved if necessary


Johnny
CASH
DON'T
BE
CRUEL
ELVIS
PRESLEY
sentimentally yours
PATSY CLINE
HANK
WILLIAMS
01
WURLITZER
SELECTION
CREDITS
00
02
PAUSE
SKIP

**01.** The 32-inch monitor, running at 1080p, displays a retro Fruitbox skin modified to fit a larger display

**02.** The cabinet contains all of the components within the front section and ensures the amp is visible for easy control

Luckily, it fit his needs. “It had the vintage look and feel that I was looking for,” Jeremy explains. “Once I got it up and running, it was pretty easy to figure out, although I had some initial problems getting it up and running with the latest version of Raspberry Pi OS Bookworm. It was missing a library dependency that I had to install separately and I thought it was all running smoothly until I tried to get it to autostart at boot.

“Then I found the Fruitbox4dummies site which had a big notice saying there were issues between the latest version of Fruitbox and Bookworm. It suggested using a specific version from March 15, 2024 so I reinstalled everything and it all went smoothly from there. Just using Fruitbox with the keyboard, it appeared to function like I had hoped and the Fruitbox4dummies site was an amazing resource, explaining how to configure Fruitbox and customise the skin.”

## Quick FACTS

- This project uses Fruitbox with a custom skin
- It only took a month to plan and build
- The jukebox will accept quarter coins
- Labelled buttons allow songs to be selected
- There’s a swappable light-up marquee

## Cabinet craft

With the software in place, Jeremy was able to crack on with the rest of the build. “It was all about making it look as authentic as possible, but after considering buying and gutting an old jukebox, I decided to build my own cabinet,” he says.

He based the cabinet around a 32-inch monitor, making the sides, front and back panels out of wood. One of the panels was cut so it could fit an assortment of buttons that could be used to control the jukebox interface. These buttons were connected to Raspberry Pi using the joystick controller, “just because it would save me a lot of time making up wires since the kits already came with wires that connected to the buttons,” Jeremy says.

Another panel was made so it could include a coin slot. “I was just going to cut a slot to put coins in and rig up a switch to detect a coin,” he explains. He then discovered a programmable coin acceptor – one which only accepts quarters but acts like a button whenever a coin is inserted. Each quarter adds three credits. “The number of credits each coin provides is an option in Fruitbox,” Jeremy says. “It’s the only way to play songs with my machine besides a secret button on the back to add free credits.”

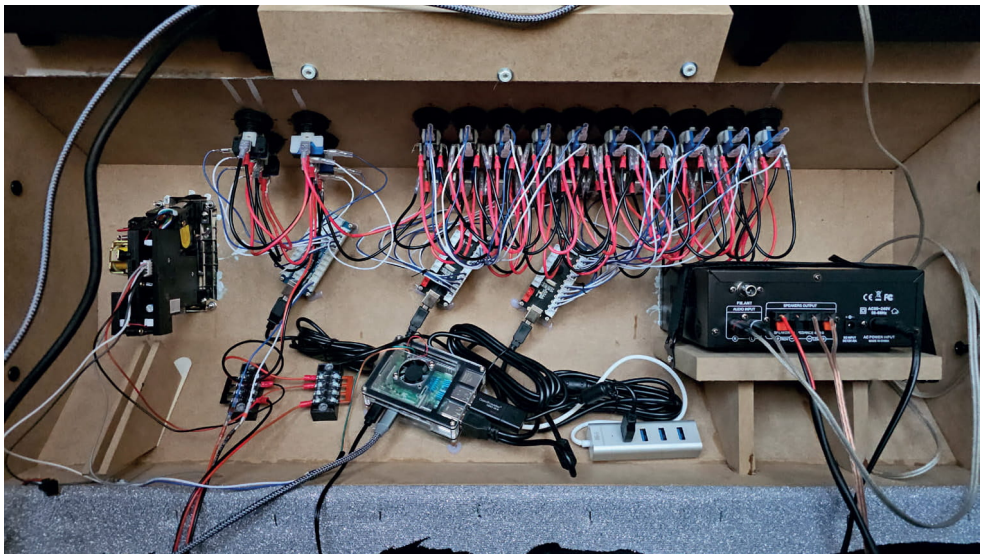
There were some issues connecting the coin acceptor to Raspberry Pi. “I initially wired it to one of the joystick inputs where it caused issues but when I wired it to a GPIO input, it worked as expected,” he recalls. The other major issue Jeremy faced was painting. “I tried using metallic paint to look like aluminium, but it was really streaky and it ran everywhere,” he says. “I had to sand it down and start over, which set me back a week.”

## Amp power

To speed things up and make the build a family affair, he enlisted his eight-year-old son, Luke, to help build the cabinet, sand, and paint. His nine-year-old daughter, Charlotte, also helped to modify the skin he used in Fruitbox. “She picked out some of the design elements and I had her program them in the skin,” Jeremy reveals.

As expected, aside from the screen, the Raspberry Pi is connected to an amplifier and speakers, enabling the jukebox to fill the air with sounds. “I debated whether or not to leave the amplifier exposed because I thought it took away from the vintage vibe somewhat,” he says. “But I ended up leaving it visible, because you can easily change the volume, it can be

controlled by a remote, it has microphone inputs for karaoke, and FM tuner, Bluetooth, and can play MP3s from a USB drive.”

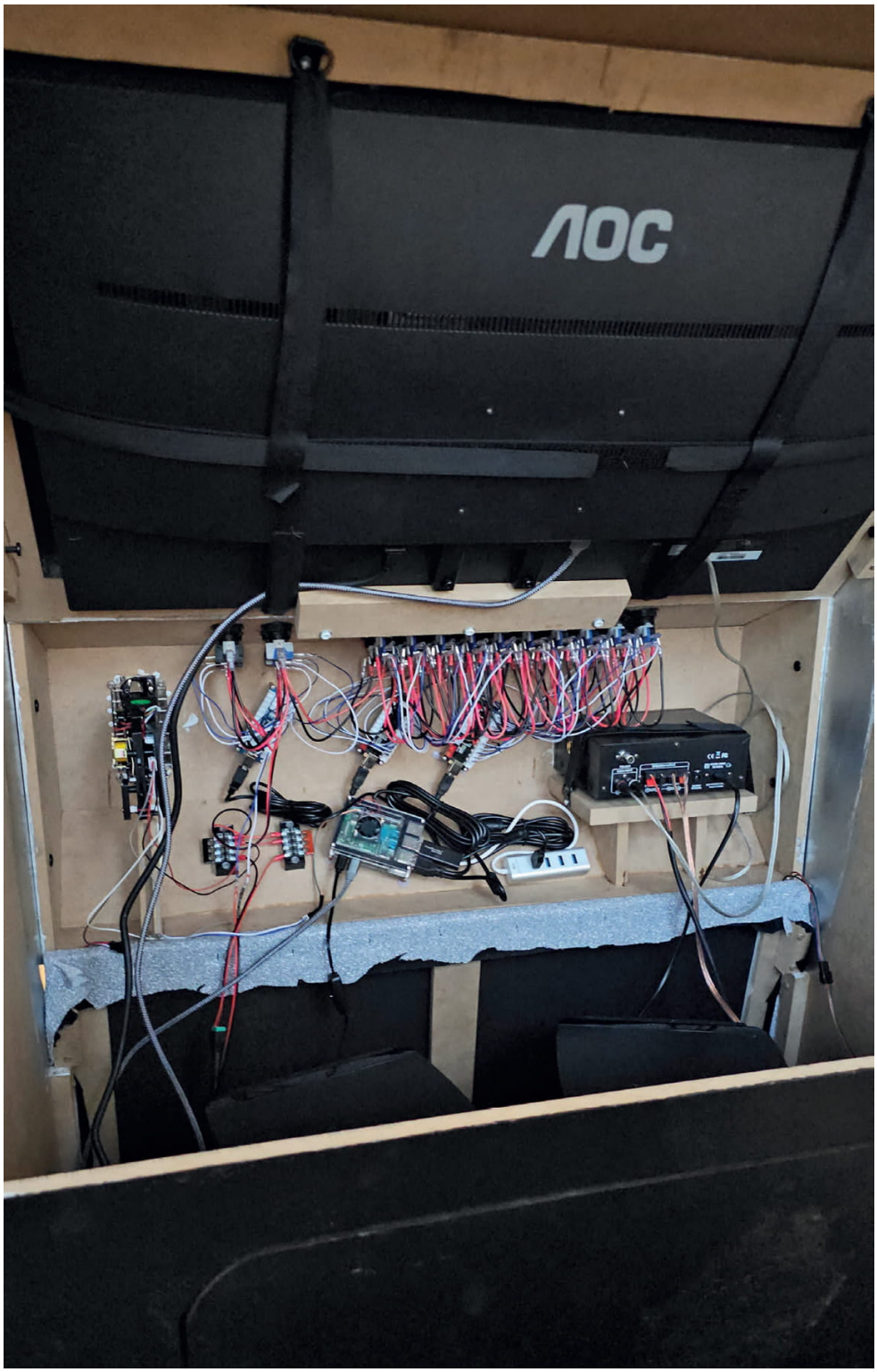


All of the hardware is housed in the button board. A couple of connections for power and the monitor are needed

> *It turned out better than I could have imagined*

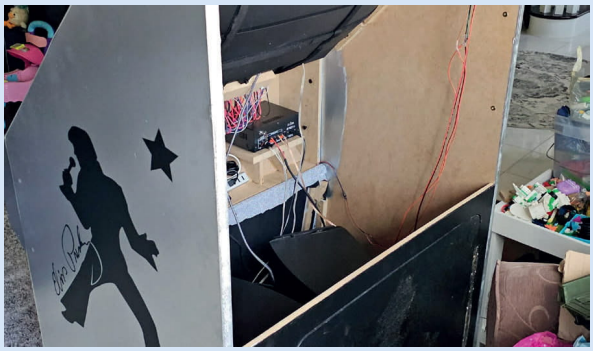
As a finishing touch, Jeremy also included a marquee at the top of the jukebox consisting of album covers back-printed on window clings stuck to some 1/8-inch plexi. These are lit using LEDs. More lights were fitted beneath the control panel, ensuring the overall build would look as good as it sounds. With a few extra design flourishes – some notes painted on to the front – it was ready to gift.

“It turned out better than I could have imagined,” Jeremy says. “The interface looks great. The buttons look and feel authentic and feel like using a real jukebox. My parents are very excited about it and the kids love to play around with it too.” It’s enough to make us want to dance with joy.




Jeremy says he would use a smaller monitor if he was replicating this build, so he could keep the cabinet under 24 inches wide

## How it works



1. Flick a single power switch and the jukebox springs into action. Everything is turned on, from the display to the amp and lights, and the Raspberry Pi 4 computer automatically boots to the MP3 player Fruitbox.



2. There are two ways of playing tunes. You can insert a quarter for three songs or hit the secret button on the back of the jukebox to get five ‘free’ credits. You’re then able to start picking some songs to enjoy.




3. There are 80 songs on each page and there are currently six pages. When a song is selected using the panel buttons, an animation shows the record being picked and played. Up to 20 songs can be queued.

OceanofPDF.com

# Alpakka v1.0 controller

## RP2040 is used for the core module of this accessible controller, now with wireless functionality. By **Rosie Hattersley**

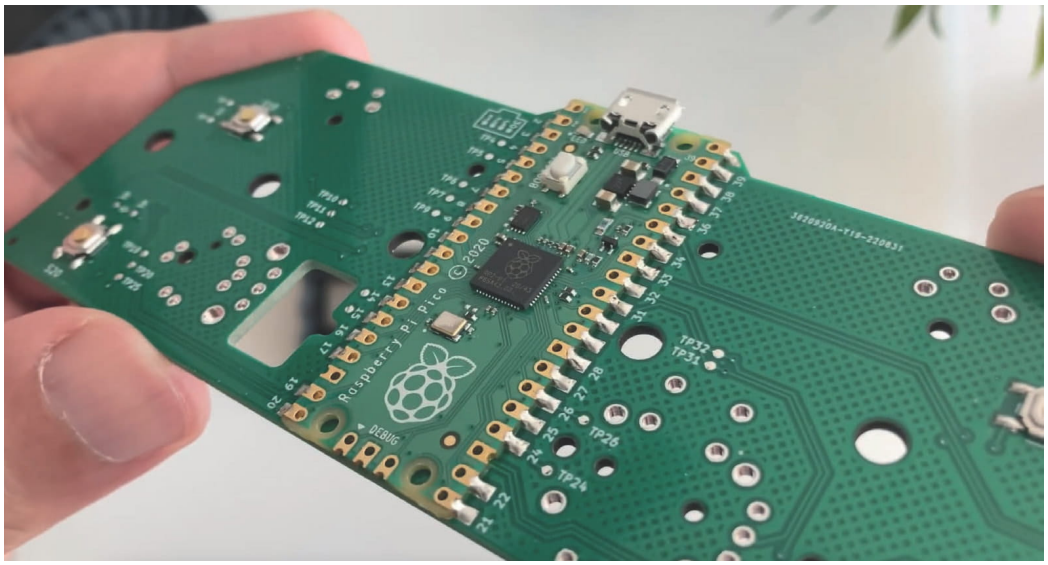



**Maker**

**Michael Przybilski and Marcos Diaz**

Finland-based software engineers Michael and Marcos regard firmware as their bread and butter.

**inputlabs.io**




Raspberry Pico freshly soldered to the custom Alpakka printed circuit board

**Software engineers Michael Przybilski and Marcos Diaz set up Input Labs in late 2022 with a mission of enabling “people with disabilities such as arthritis, who have difficulties holding and operating traditional pointing devices, such as computer mice, to comfortably work with computers”.** Now based at the renowned Maria 01 startup campus near Helsinki, Input Labs’ founders are passionate about open-source hardware and the mutually supportive communities that help drive innovation and improvements. A year into their journey, Marcos and Michael announced a 3D-printable version of a traditional gaming controller that “enables the user to aim comfortably, with two hands, while making the interaction with the computer very precise”. Alpakka v0, their two-handed reference design based around Raspberry Pi Pico, launched in late 2023 to an enthusiastic reception, and now the upgraded Alpakka v1 is available.




**01.** Alpakka is a 250MHz low-latency customisable input controller based around a Pico microcontroller

**02.** Two gyroscopes precisely track its location and compensate for input bias, making it a formidable games controller

## Complete user control

Alpakka struck a chord with gamers keen on the customisable case design – the 3D-printable low-polygon housing could be remixed in Blender for a smoother one – and the precision provided by two IMUs (inertial

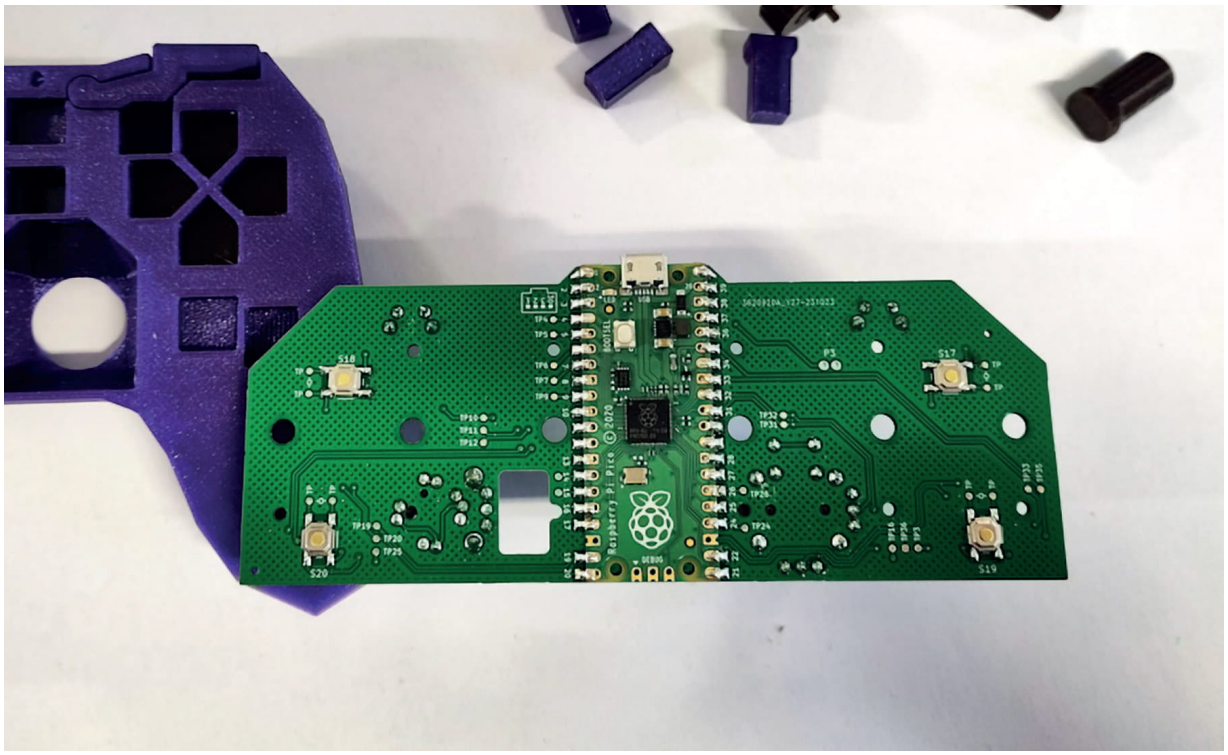
measurement units) featuring gyroscopes. Precisely detecting movement of the handheld device significantly reduces drift and also compensates for individual bias, both of which are manna for gamers, explains Michael. Alpakka's gyroscopes also make it ideal for flick-stick gameplay in first-person shooters.

The pair were pleased with how well Raspberry Pico addressed their needs, with plenty of addressable GPIO pins for the various triggers and buttons alongside a quality ADC (analogue to digital converter) and solid documentation and developer community. "While Alpakka itself was designed from scratch, we took of course advantage of all the other things that work around us," says Michael. The 3D parts have low tolerances, as the *HackSpace* review of the Alpakka v0 highlighted (**rpimag.co/alpakkareview**). The pair experimented with various hardware components, such as buttons, to find what level of required force would feel best for the triggers. To this end, they benefited from the invaluable input of a slew of eager testers, including a number with physical disabilities.

> *Alpakka's gyroscopes make it ideal for flick-stick gameplay in first-person shooters*



Input Labs sells Alpakka in kit form, with a choice of colours



*HackSpace* 3D-printed and tested Alpakka v0 and was suitably impressed

Michael says that it was a particular challenge to integrate the mechanical user interaction such as the triggers and buttons with the conductive touch area, but models from the likes of KiCad and Blender were helpful, as were Raspberry Pi's TinyUSB libraries. There is an accompanying 'Ctrl' app that allows users to configure aspects such as the gyroscope and button sensitivity. The 3D designs, PCB design, firmware, and Ctrl app can all be found in Input Labs' GitHub repositories (**github.com/inputlabs**).

## Considered approach

The attention to detail certainly paid off. *HackSpace* magazine tested the original Alpakka controller kit and reported that, once 3D-printed and built, "it feels great in your hand. It's sturdy and just weighty enough to feel solid" with a gyroscope that means tilting the controller dictates mouse movements.

Such elements reflect Input Labs' recognition that the Alpakka controller is useful for those with particular adaptive needs, but it's the gaming

community that will be the main customer base. Michael says their aim now is to make the best gaming controller around – “something that in gyro-gaming at least, we seem to be succeeding”.

December 2024 saw Input Labs unveil a second reference design. As well as a second thumbstick variant, Alpakka v1.0 provides a much requested (but less than straightforward) feature: wireless gameplay. Michael and Marcos spent many months on a low-latency communications solution. This is provided via a dongle and a Marmota module with Raspberry Pi RP20240 alongside an Espressif ESP8684-MINI-1. The Alpakka v1.0 can also be recharged via a USB-C connection on the module.

Marcos and Michael have since launched the new wireless-enabled Alpakka v1.0 components in the online store along with a configurator for customers who prefer a ready-to-assemble kit. Iterations and improvements are ongoing as the pair seek to make good on their intention to design and market the very pinnacle of gaming and user-controlled input. Keeping close to their existing and future customers is a must, notes Michael. “Luckily, we have the pleasure of a very active and knowledgeable community,” he adds. “Their expertise was, and still is, extremely valuable.”

## Quick FACTS

- Alpakka was initially designed for arthritis sufferers
- The original model brought in hundreds of customers
- It highlighted the need for quality adaptive input controllers
- Complete user configurability is a huge advantage
- Users can create and 3D-print custom models for their needs

WIZnet

# HARDWIRED ETHERNET

Powered by Raspberry Pi

Embedded Ethernet, designed for every integration

W55RP20 INSIDE

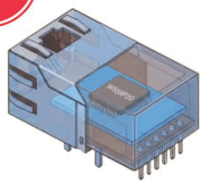



## W55RP20

RP2040 + W5500 in 8 x 8 mm SiP

- RP2040 Dual-Core Up to 133 MHz
- W5500 Hardwired TCP/IP Offload Engine
- 2MB Onboard Flash(XIP)

W55RP20 INSIDE

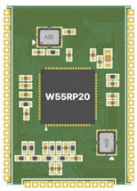


## WIZ-IP20

Serial-to-Ethernet Module in an RJ45 Form Factor

- UART/SPI, up to 921,600 bps
- TCP, UDP, MQTT/MQTTS, Modbus Support
- Unique MAC & Configuration Tool Available

W55RP20 INSIDE




## W55RP20 SoM

System on Module

- Compact Form Factor Easy to integrate
- Unique MAC Address Pre-programmed
- Custom Firmware Development Supported

**IDEAL FOR** | Industrial Automation | Smart Building | Network Security | Medical Devices | Remote Monitoring | Robotics

**Hardwired TCP/IP.**
**Reliable. Efficient.**
**Ready to Build.**

**Learn more at wiznet.io**

Follow WIZnet

OceanofPDF.com

# Whispering Wires

## This landline phone reads poems aloud. **David Crookes** has a listen



**Maker**

**Théo Z V Champion**

Théo, who goes by the internet names tzvc and Rootkid, has a master’s degree in computer science and says he has been tinkering with machines since the age of mythology.

**rpimag.co/whisperwires**

**When Théo Z V Champion visited a vintage market in London, he noticed that shoppers were rather intrigued by a rotary phone – the type that needed callers to choose a number by inserting their fingers into the relevant holes of a dial before turning the wheel in order to initiate a telephone call.** It was disconnected, but that didn’t prevent many people – acting out of curiosity or feeling pangs of nostalgia – from lifting the handset and placing it against one of their ears.

In each case, they were met with silence. They’d place the handset back on to the base, inevitably smile, and turn their attention to another retro device. But what if they had been able to listen to a voice? Would they have engaged with the phone for longer and potentially gained something more

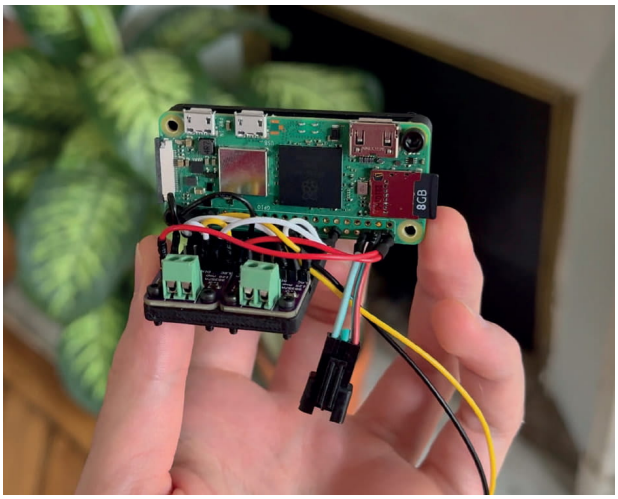
from the experience? With that thought in mind, Théo acquired an old landline phone of his own – albeit a push-button type – and he got to work on retrofitting it with a speaker.

The concept was that, upon lifting the handset, the user would hear a voice reading out a poem and this, he explains, would indulge a burning passion to mix technology with art. “I believe technology and engineering are underrepresented in art, so I create pieces that use technology as both the medium and the message to reveal the invisible world of engineering around us – be they algorithms, communications/ surveillance tech, artificial intelligence, the internet and so on.”

## Listen up

Théo selected a vintage French Socotel S63 phone, opting for a model that was designed by the Centre national d’études des télécommunications in the 1980s as a standard-issue device for homes the length and breadth of France.

He then opened the housing and decided to make a Raspberry Pi Zero 2 W single-board computer the heart of his modernised device. Since Raspberry Pi Zero 2 W doesn’t have an on-board audio amp, he connected it to a pair of 3 W amplifier chips (MAX98357A). “This would drive audio to the phone earpieces via I2S,” says Théo, who attached all of the components to a 3D-printed support and inserted it into the housing.



Raspberry Pi Zero 2 W is connected to the two small 3 W amps and these components are housed in their own 3D-printed case which is then secured inside the phone




**01.** The project uses an old French push-button Socotel S63 phone which once came in a variety of colours. There were rotary dial versions too

**02.** No modifications were made to the housing, so it looks, to all intents and purposes, like an original phone albeit with a modern twist

> *I believe technology and engineering are underrepresented in art*

Raspberry Pi Zero 2 felt like a perfect fit and not simply because it was small enough to squeeze into the relatively cramped space within the phone. “I needed the phone to have internet access over Wi-Fi and to be able to play audio,” he explains. “Being a software engineer, having a full Linux environment is also a blessing for me as opposed to more constrained environments such as ESP32, MicroPython, and so on. All the tools and packages I could think of are compatible with this single-board computer.”

## Finding inspiration

With the components sorted out and a switch hook connected so that, upon lifting the phone’s handset, the Raspberry Pi would spring into action, Théo began to work on the software. He’d discovered the Poetry Foundation website (**poetryfoundation.org**), which contains large collection of poems, and was delighted to find that a good number of them included audio narrated by the author.

“I poked around the website code and was quickly able to find an open access to their database,” he recalls. “I was able to download all the audio poems, but there was one issue: the poems contained an intro/outro by the author.”

Théo wanted the poem to sound like the poet was actually on the other side of the phone, not just a recording. “One of the main challenges was to clean up all the audio-narrated poems from their intro/outro to only have the actual poem play when the phone is picked up,” he says. “The solution involved transcribing all the poems and then using a large language model (LLM) to cut out irrelevant portions.”

## AI for rhyme

To transcribe the audio poems, Théo used WhisperX (**rpimag.co/whisperx**), an advanced, fast automatic speech recognition model which is capable of transcribing audio with word-level timing. Building upon OpenAI's Whisper model that has been trained on about 680,000 hours of diverse audio, it allowed Théo to get a handle on exactly when each word was spoken in an audio clip. He could then turn to Open AI's LLM, GPT-4o-mini, to complete the next stage of the task.

"I passed the transcription into GPT-4o-mini, asking it to remove any intro/ outro [material]," he explains. "I was left with a database of more than 3000 perfectly cut audio poems." This method also meant that the device is 'future-proof', so to speak. "The database updates every day as new poems are uploaded to the Poetry Foundation website," Théo reveals, and Whispering Wires is capable of playing them all.

Of course, there are some ethical questions surrounding this move: is it right to remove the credit from the beginning of the audio simply to gain a seamless experience? Théo has already considered this and he says all of the information about the poets and the names of the poems is being saved and organised in a database. He plans to create some way to show the credits when the phone plays the poem and he's currently looking for a small enough display.

In the meantime, Théo has been working on making his project as authentic an experience as possible. Although sound comes out of both the handset speaker and an extra earpiece that allows two people to listen to the poems at the same time, Théo has ensured that the audio isn't played with crystal-clear clarity but instead sounds just like the phones did when they were being used a few decades ago.

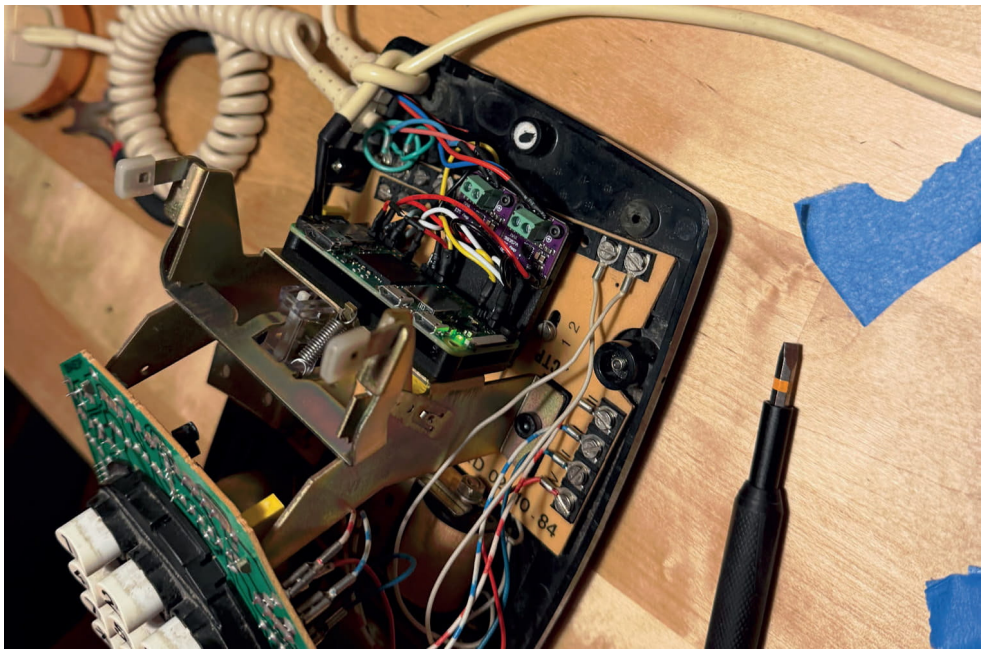
"To achieve the vintage phone audio quality, I researched what frequencies phones operated at back then and applied the same filter with code (300Hz to 3.4kHz band range at 8kHz sampling frequency) to get the authentic 'phone grain'." He's also added a hang-up sound followed by a line-disconnect tone when a poem ends so that the listener has the feeling the poet has hung up. To that end, it's shaped up well as a head-turning art piece and if the response from Théo's YouTube video is anything to go by, it is certainly getting people talking.

## Quick FACTS

- This project uses an old landline phone
- State-issued, it’s technically still the French government’s property
- It makes use of 68 hours’ worth of poems
- AI is used to edit the poem audio
- The project cost $60

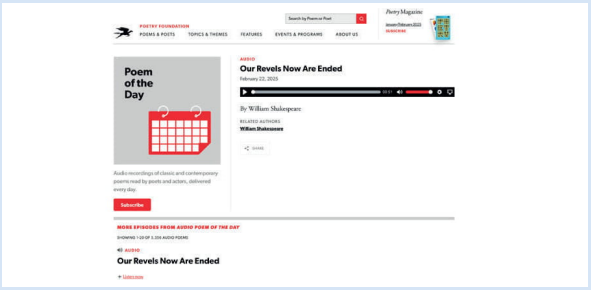


The phone has a cache of a few gigabytes of poems for use offline, while audio is streamed when online with the phone directly connecting to the internet using Raspberry Pi’s wireless LAN capabilities



Rather than strip the phone internals, Théo has simply added the wires and modern electronics, which happen to fit neatly into the space towards the rear

## Making poetry




1. A database created using artistic works pulls narrated poems from the Poetry Foundation website. An AI model called WhisperX creates word-level accurate transcriptions of those poems and a large language model strips them of any intros and outros.



2. When someone picks up the handset, the phone will automatically and immediately pull a random poem from the database. It will begin to play the audio through the earpiece, complete with an authentic phone audio grain for added nostalgia.



3. When the poem is finished, the poet ‘hangs up’ and this leaves the listener with a disconnect tone. All the listener has to do is put the handset back on the phone’s base and pick it up again. The phone will then pull another random poem from the database.

OceanofPDF.com

# Bipedal companion robot

## We all need a friend, whether they be human, furry, or long-term electronics project. **Rob Zwetsloot** befriends this bipedal pal

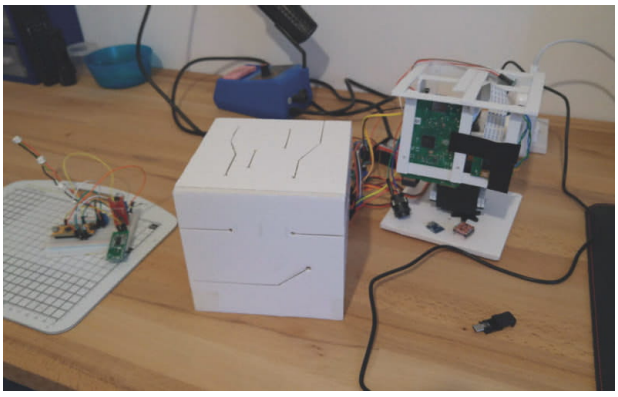


**Maker**

**Dan Nicholson**

A software engineer by day who has been building open-source robots for five years, logging the build process as he goes.

**makerforge.tech**



The robot started off life as a cube

**A robot companion is a bit like a pet, but without the muss or fuss as a Tom & Jerry cartoon once put it.** While this one stands on two legs, it currently cannot walk; for maker Dan Nicholson, however, that wasn't the point of his long ongoing project.

"It includes an easy-to-learn, custom software framework that supports functionality like object and face detection, animation, voice recognition, text-to-speech, AI integration, and much more," Dan explains. "I've focused heavily on documentation, and made design choices to keep the project accessible to as many people as possible. The goal is to make an autonomous desktop buddy that others can build, to inspire people to get involved in robotics and making."

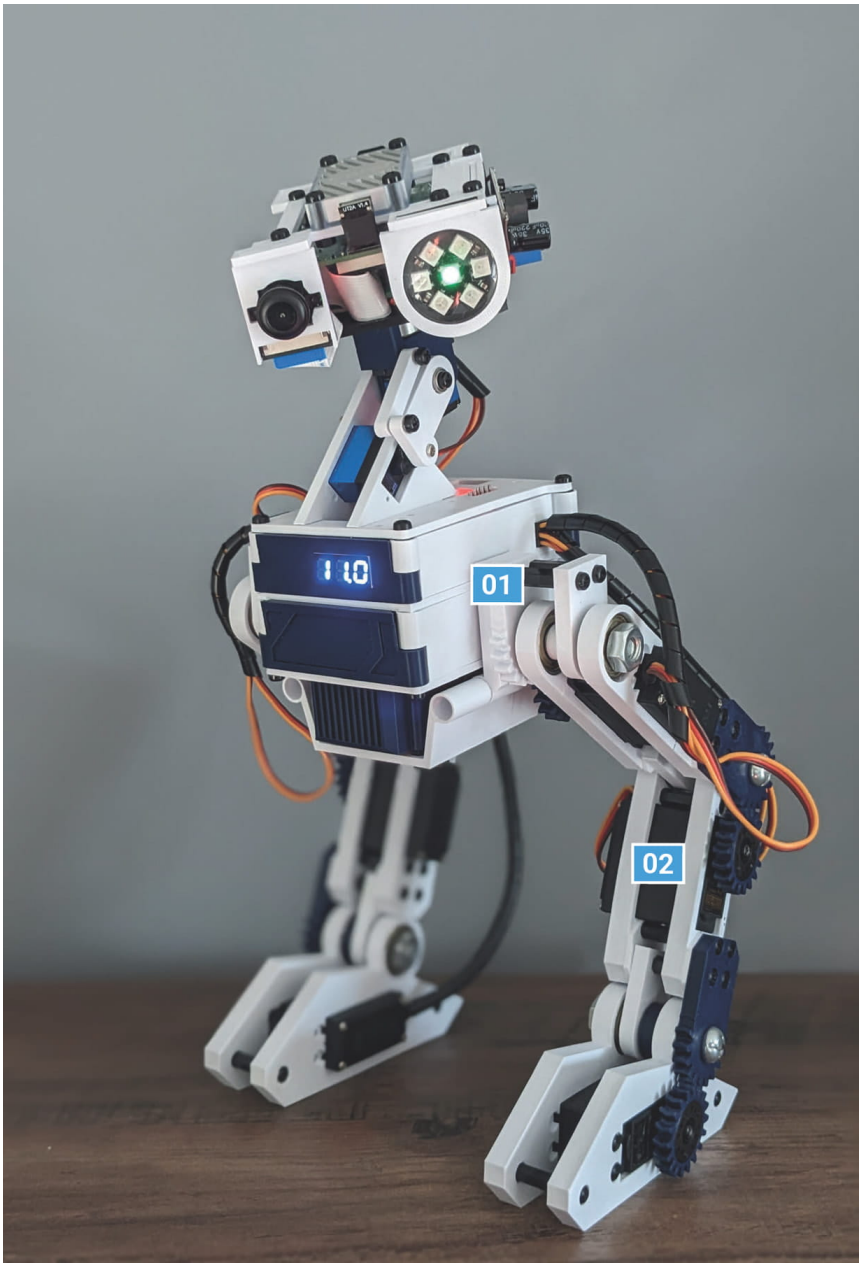
He's been working on it since 2019, and has been featured on Raspberry Pi's blog with an earlier iteration of the robot (**rpimag.co/bipedalrobot**), although it started life as an animated cube before evolving via wood builds and 3D prints to the version today.

"During Covid lockdown, I was given a Raspberry Pi 3B+ and wanted to try building something with it," Dan tells us. "I had no idea what I could or couldn't do, and I only had a basic understanding of electronics, robotics, and 3D printing, so I thought a desktop companion would be a challenge… [And] it certainly was! Five years later, I'm still learning, building, and adding to the project."

## Quick FACTS

- All the components required are easy to obtain or make
- There have been three entirely separate 3D-printed chassis
- All the custom PCBs are open source, like the rest of the robot
- Other folks are adding wheeled legs so it can move

- Despite all the videos, Dan finds them tricky to make.




**01.** Different components allow for various combinations of functions

**02.** The legs are full of servos, but it cannot walk… yet

## Tumultuous process

We often bang on about how taking part in a hobby is part of a process of learning and experimenting, and Dan has been learning all about it over the last few years, describing the experience of building the robot as “awful, and also loads of fun”.

“I’ve come to realise that the value of the project for me isn’t just in having a robot, it’s also in the learnings that I’ve taken away from the years of trial and error,” Dan says. “I haven’t had any major disasters (yet!), but I have had a lot of things that haven’t worked the way I wanted. But because I don’t have a deadline and can take my time and try things out, my knowledge has become much stronger because of every experience, both good and bad.”

As the project is ongoing, Dan reckons the functionality of the robot varies day-to-day, with a long list of features that can create “really interesting behaviours” when enabled in various combinations.

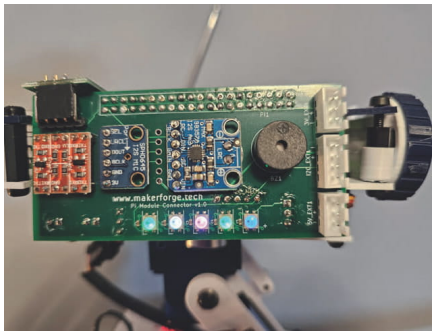
## Always improving

As this is one of those projects that will never be completed, Dan has a long list of things he’s contemplating adding or upgrading.

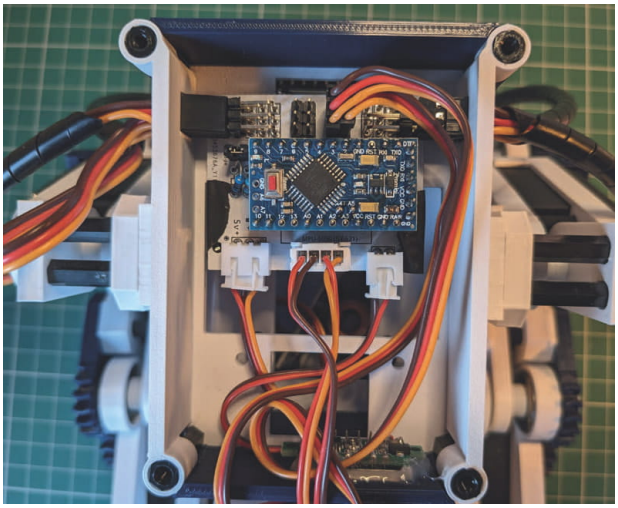
“I wish I had the time to do them all,” he admits. “Right now I’m working on swapping Raspberry Pi 5 for Compute Module 5 and a custom carrier board. This will not only shrink the build and expand the existing functionality, but also allow other modules to be added easily for community variations. The great thing about this build is that, as new technology becomes available, the entire project direction can shift. A recent example is the Raspberry Pi AI Camera, which now handles object detection directly on the module and leaves the Raspberry Pi free to do other things. I’m excited to see what comes next.”

He has managed to build a small community of folks around it too, who are doing their own changes and upgrades – including the ability for the robot to walk. Maybe we’ll get it back in the magazine when it’s mobile.

> *I've come to realise that the value of the project for me isn't just in having a robot, it's also in the learnings that I've taken away*




There are many custom PCBs used in the build – all of which are open source



An Arduino Pro Mini microcontroller also helps Raspberry Pi to control the robot

OceanofPDF.com

# Operation Pico

## Prolific maker Kevin wanted to recreate the classic board game Operation using Raspberry Pi Pico. **Rosie Hattersley** is buzzed



**Maker**

**Kevin McAleer**

Kevin makes robots and shares his builds and expertise via weekly online broadcasts.

**kevsrobots.com**

**Kevin McAleer is a master at making coding fun, showcasing the creative possibilities via a tranche of simple but clearly explained videos and tutorials that are both memorable and off the wall: rpimag.co/kevsyt.** One of the very first projects of Kevin's that we featured involved a Raspberry Pi and Billy Bass, the singing animatronic catch of the day. His latest project is a build-ityourself version of Operation, a hugely popular game beloved of those of us who grew up in the late 1970s or 1980s. Pico Operation showcases how easy it is to use a Raspberry Pi

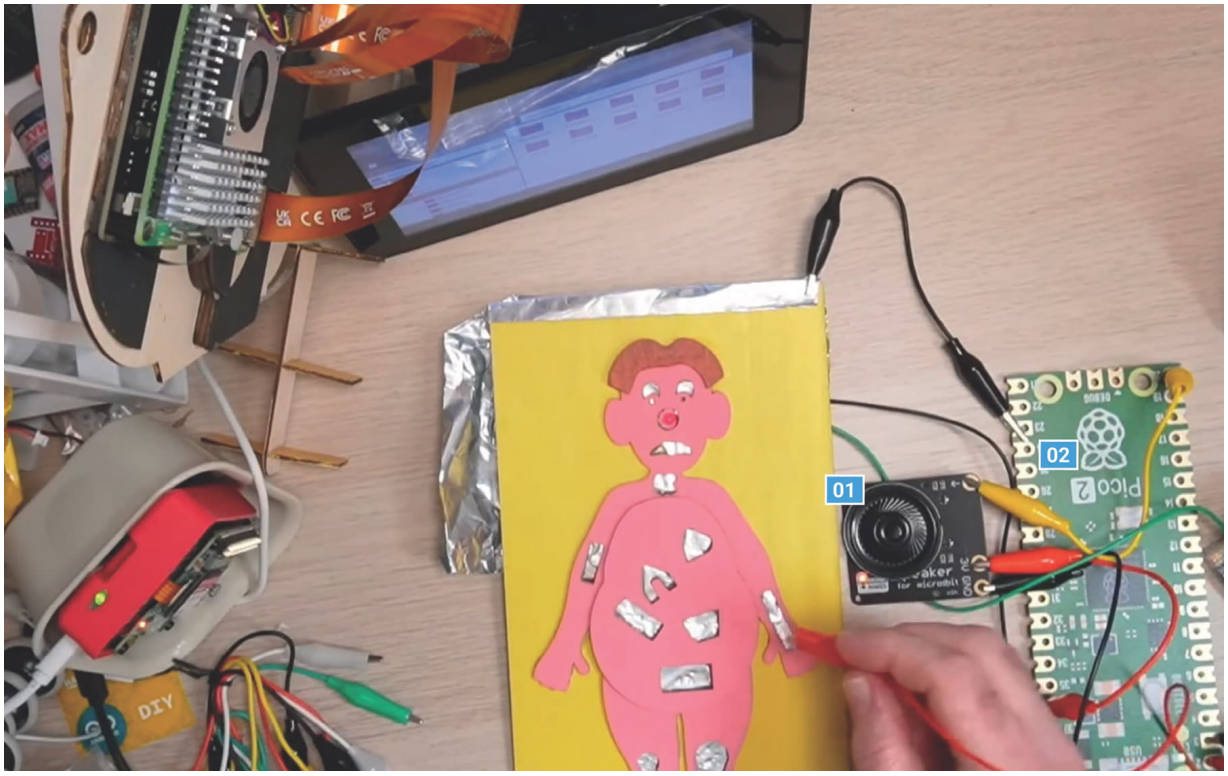
Pico microcontroller and MicroPython code to recreate the Hasbro board game.

## Break it down

Kevin has a knack for making coding accessible and has been showing off his inventive spirit ever since 3D-printing his own Adafruit Mini Mac case with a first-generation Raspberry Pi inside. He credits Raspberry Pi with helping him learn and master Linux and Python, his go-to choice of coding language.

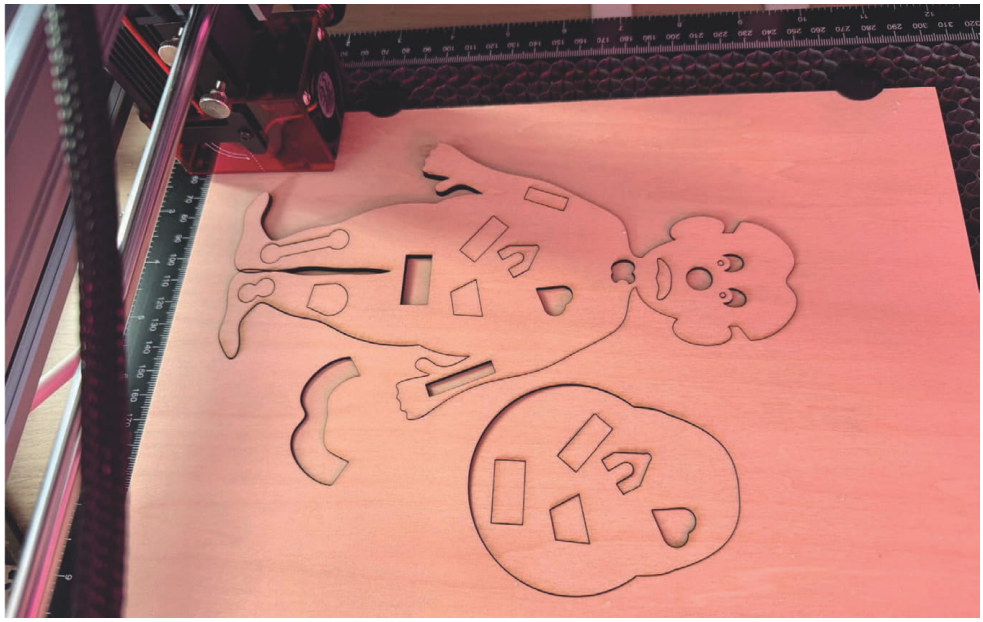
Here, he uses crocodile clips attached to Pimoroni's Pico Jumbo – a "hilariously oversized" version of Raspberry Pi Pico 2 on a matching PCB – connected via GPIO pins to a laser-cut patient from the game Operation (**rpimag.co/operation**). The Jumbo is "perfectly suited for use with crocodile clips and therefore requires zero soldering – you can simply unclip everything after playing and put it on the shelf!" He was also impressed with the MonkMakes Speaker which, although labelled as being designed for micro:bit, works perfectly well with Raspberry Pi.

> *Operation Pico has a range of tuneful responses when a shaky hand causes the crocodile clip tweezers to touch the sides of the cavity and trip the buzzer mid-'operation'*




**01.** Pico Operation updates the classic buzzer-wire game with catchy tunes played via a MonkMakes speaker

**02.** A Pimoroni Pico Jumbo controls everything, although you could just use a standard Pico



Laser-cut plywood pieces recreate the classic Operation patient

The setup also includes a large 10 mm LED nose and a crocodile clip used as the surgeon's removal tool, which "will complete a circuit if it connects with the aluminium foil on the back of the Operation model." The closed circuit is detected in the same way as a button press and causes the speaker to buzz and the patient's nose to light up.

## Listen up

Operation Pico has a range of tuneful responses when a shaky hand causes the crocodile clip tweezers to touch the sides of the cavity and trip the buzzer mid-'operation'. "The original Operation game simply had a buzzer that was connected to the battery and made a circuit whenever the tweezers touched the sides of the cavity," says Kevin. "For this project I wanted to include a library for playing music through a simple speaker. I created a music library and included a little tune at the start of the game." He points out that the MicroPython code for the game is only 45 lines long, while adding music extended it to 107 lines, "so it is quite a simple project for people to easily recreate themselves".

It was the first time Kevin had used a MonkMakes mini amplified speaker. He says "it is surprisingly loud and gives the game an extra dimension, and was lots of fun when creating the music player library". Kevin explains the music creation process in his excellent accompanying video: **rpimag.co/operationyt**.



Kevin used Fusion 360 to faithfully recreate the original game’s look and feel

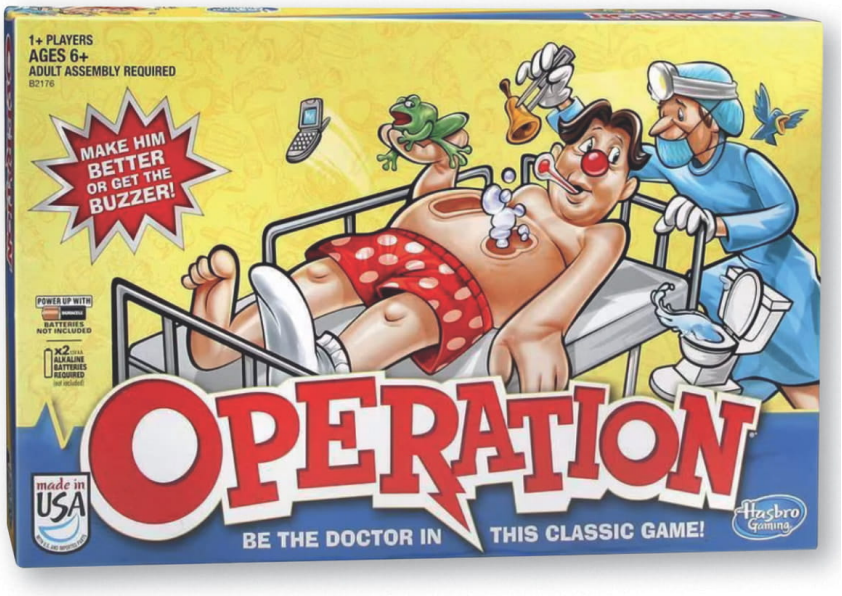
## He’s got the look

The hardest aspect of the Operation Pico project was getting the iconic patient character design just right. After all, game-playing families from over five decades know exactly what he’s supposed to look like. Kevin painstakingly recreated the man in Fusion 360, using circles and splines to mimic the original design. He then glued together multiple layers of plywood to give the character some depth.

As someone commented on the accompanying YouTube video, “Genius! Love how you’re combining classic gameplay with modern tech.” We couldn’t agree more

## Quick FACTS

- Operation has made Hasbro $40 million to date
- Creator John Spinello was paid $500 for it in 1964
- The original patient is called Cavity Sam
- Kevin has more than 50 active Raspberry Pi projects
- He’s keen to hear from anyone currently running more!




The original game’s patient is called Cavity Sam

## Body builder



1. Using Kevin's body part templates (**rpimag.co/operationpico**), cut out or laser-cut individual body pieces and spray-paint them. Attach aluminium foil as a base for each body part.




2. Use the GPIO pins on a Raspberry Pi Pico 2 or Pimoroni Jumbo to connect the board electronics, MonkMakes speaker and crocodile clips, along with a power source. Operation Pico code is on Kevin's GitHub page: **rpimag.co/operationgit**.



3. Kevin tracked down a chunky 10 mm LED to serve as the patient's red nose, triggered whenever the player touches the aluminium foil while removing a body part with a crocodile clip.

OceanofPDF.com

# Zero-G Laboratory floating platform

## Helping explore space by building earthbound robots. **Rob Zwetsloot** investigates the final frontier

**Maker**

**Space Robotics Research Group (SpaceR)**

A research group from the University of Luxembourg that specialises in space-bound technology.

**spacer.lu**

**Zero gravity is weird.** Most of us can’t really conceptualise a weightless environment like you experience on space missions, and there’s even a difference between pressurised atmospheres inside a spacecraft and vacuum as well, and even the tiniest variable can affect things. To better understand the various effects, special equipment has to be created.

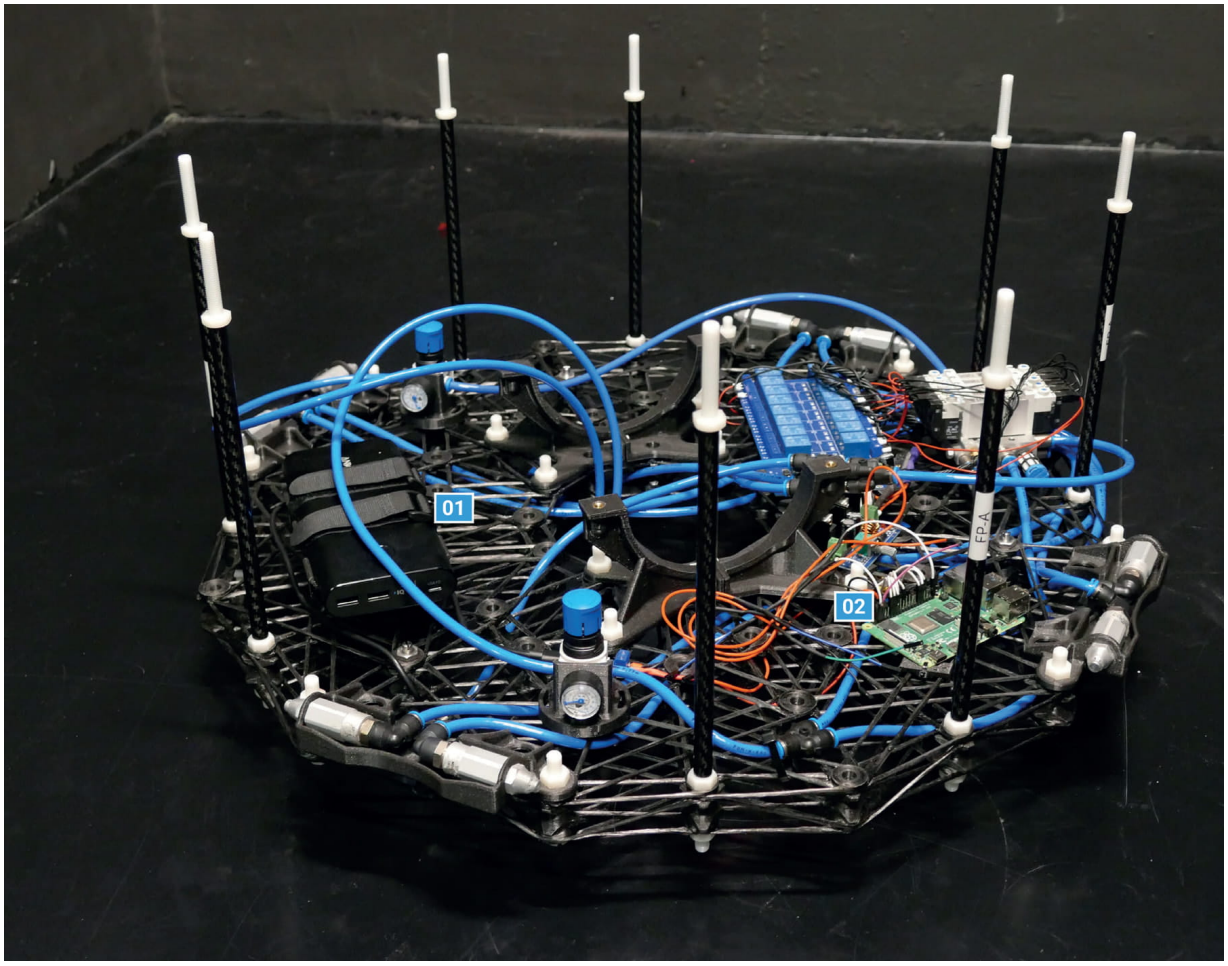
“The Zero-G Laboratory is specifically designed to emulate scenarios like spacecraft rendezvous, docking, capture, and other interactions between spacecraft,” Barış Can Yalçin of the team tells us. “It is equipped with advanced infrastructure, including space-like lighting conditions, a motion capture system, an epoxy floor, mounted robotic rails, and the capability to

integrate onboard computers and large mock-ups. These features enable researchers to conduct a wide range of experiments for unique orbital scenarios, allowing for hybrid emulations with robots that integrate hardware and pre-modelled software components. The facility can be operated in real-time and accurately emulate orbital robotics scenarios.

## Filling a need

Space is still a huge industry, and labs like this are created to make sure missions are safe and reliable.

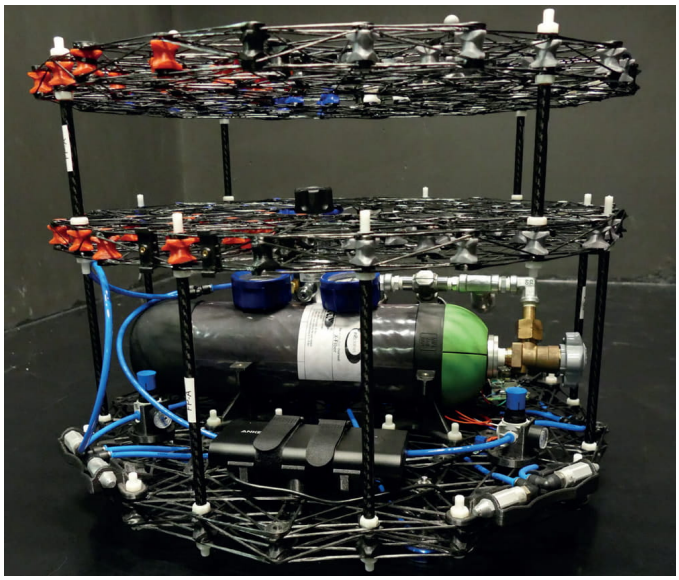
“Orbital Servicing, Assembly, and Manufacturing (OSAM), Active Space Debris Removal (ADR), and Asteroid Mining (AM) are becoming increasingly significant in both research and commercial sectors,” Barış says. “Earth’s orbits are filling up with outdated space assets, while the number of planned missions is set to rise sharply in the coming years. Additionally, there are plans to establish multiple space stations and large structures in Earth’s orbits over the next decade, which will be partially assembled and/or manufactured in space. These activities require higher levels of autonomy and close interaction. To ensure safe, secure, and reliable in-orbit operations, it is essential to validate and verify Guidance, Navigation, and Control (GNC) algorithms on the ground before launching missions. Consequently, there is a growing need to develop effective experimental setups for testing these algorithms. Therefore, in the academia and industry, floating platforms have been developed and frequently used by many institutions to emulate orbital robotics scenarios.”



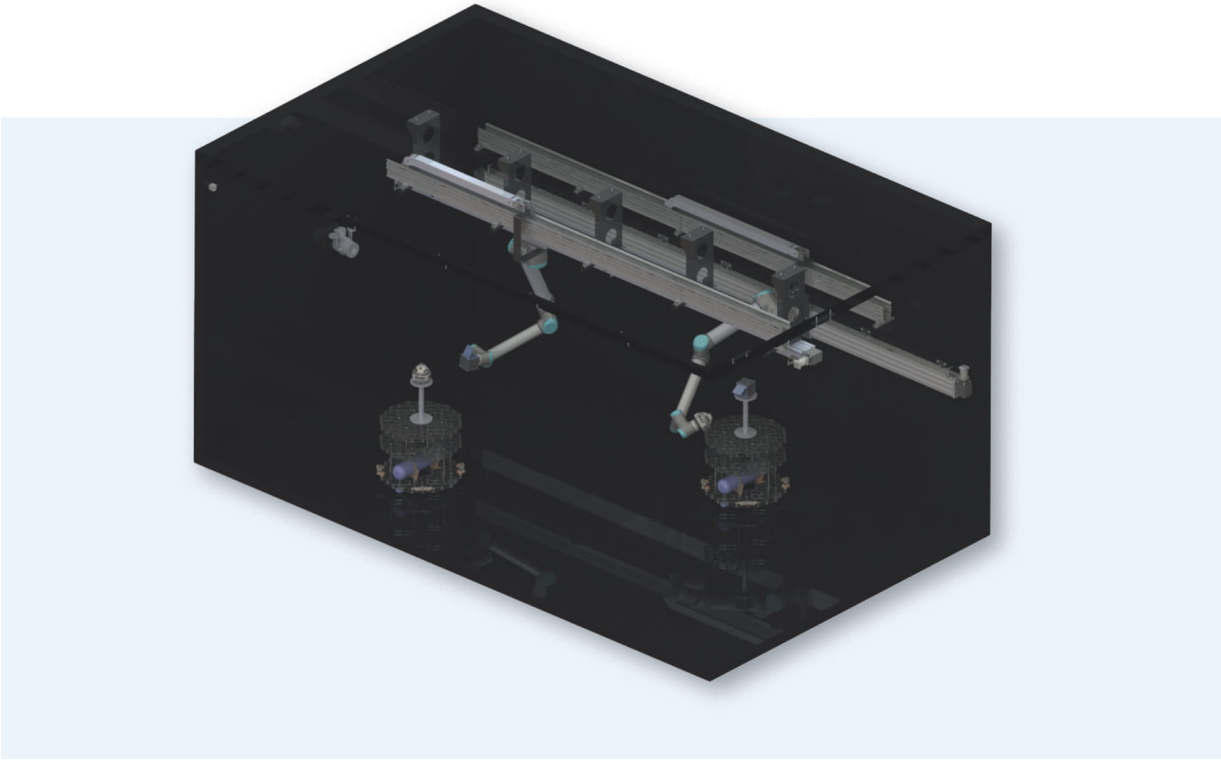

**01.** It’s all powered by a 5 V mobile battery

**02.** Raspberry Pi both helps control the platform and communicate with other robots

> *The floating platform of Zero-G Lab is successfully performing SIL (software-in-the-loop) and HIL (hardware-in-theloop) capabilities*



The fully constructed platform looks like a strange K’Nex set



A visualisation of the robots working together in experiments

Where does Raspberry Pi come in with this scenario? Its small size, low cost, and huge support and compatibility, along with ROS (Robotics Operating System) being available for Raspberry Pi, allow it to perform a

great many functions for a project like this. Image processing, data analysis, control algorithms, wireless connectivity, and more make Raspberry Pi ideal for prototyping, then embedding in systems like this.

## Some light construction

The system works, to be reductive, like an air hockey table. The floating platforms have an ‘air-bearing’ that levitates them off the floor a little like a hovercraft, and they have a series of nozzles that grant 3 DoF (degrees of freedom) – they can move along X and Y axes, and rotate around the Z axis. For this to work, the platform needs to be light.

“The floating platform is constructed using additive manufacturing with lightweight carbon-fibre material, which helps extend experiment duration, allowing for the emulation of complex scenarios in the Zero-G Lab,” Barış explains. “The lighter the floating platform, the less compressed air it consumes. Moreover, the floating platform features a modular design, allowing easy disassembly of the middle and upper plates, which can also serve to carry equipment for various emulation scenarios. Its string-like topology provides ample space for mounting multiple components. 3D-printed supports can be easily integrated into the string structure, enabling the assembly of additional equipment such as debris removal systems, debris mock-ups, refuelling or docking mock-ups, sensors, and more. Each plate has a 60 cm diameter, and the distance between them is adjustable, offering flexibility for accommodating different types of equipment, making the platform highly versatile for various applications.”

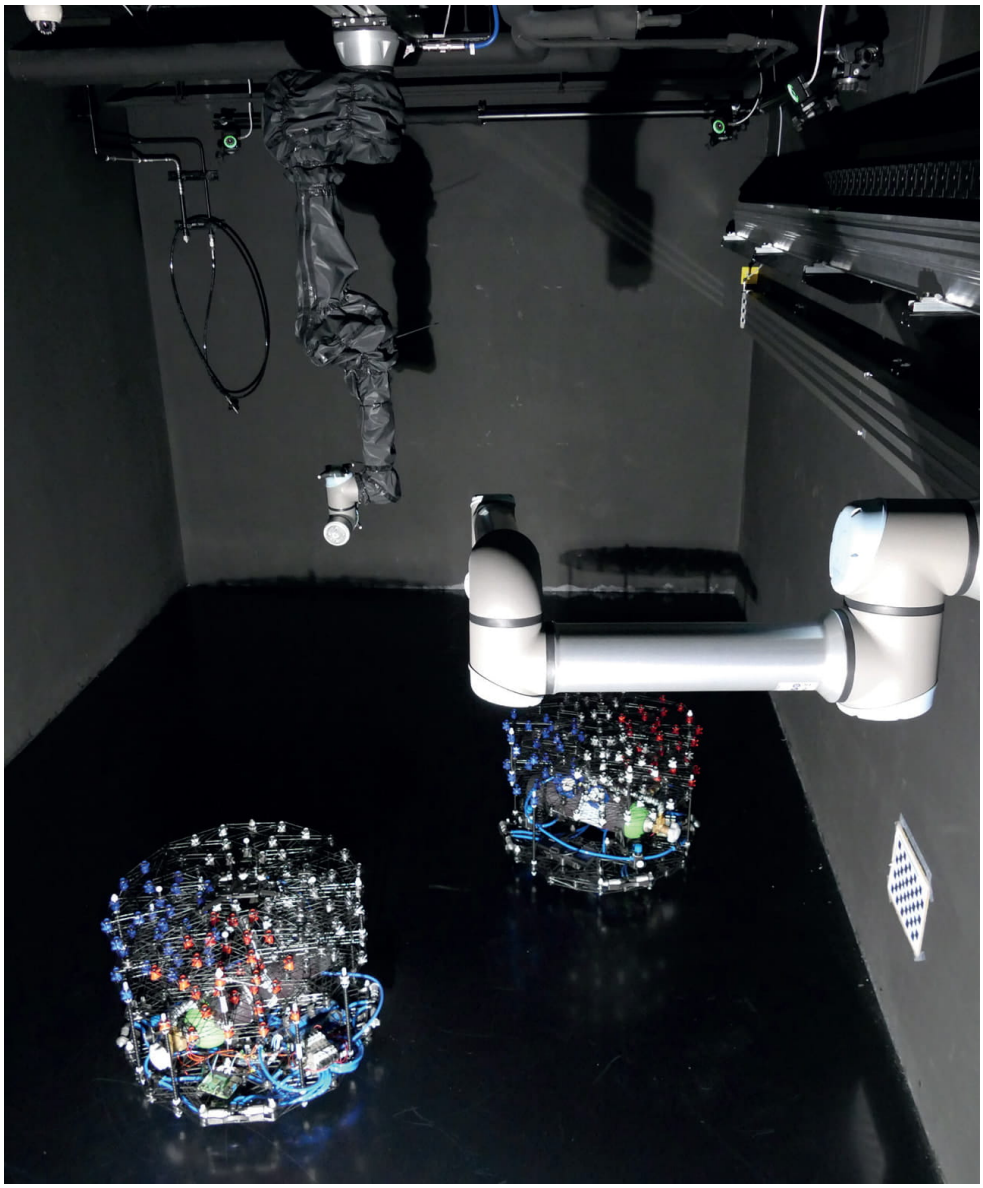
## Lift-off

The good news is that the floating platform works as expected. “From the numerical results gathered during the experiments, we confirmed that the proposed floating platform used with Raspberry Pi is suitable for emulating on-orbit scenarios” Barış reveals. “The floating platform of Zero-G Lab is successfully performing SIL (software-in-the-loop) and HIL (hardware-in-theloop) capabilities. Several mission-specific proof-of-concept tests such as rendezvous and docking, on-orbit interaction, landing, etc. that leverage the floating platforms in the Zero-G Lab have been realised.”

Maybe a Raspberry Pi-powered robot will be tested on here for a future mission? Only time will tell.

## Quick FACTS

- Parts of the design are patented
- The platforms communicate with the rest of the lab's robotics
- This communication is handled via ROS
- Solenoid valves manage the pressure, to a max of 10 bar
- A pressure regulator helps control the external compressed air source

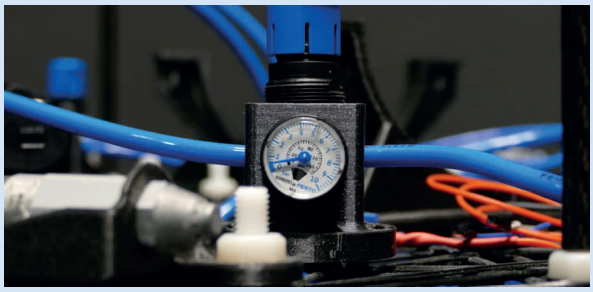


The laboratory where the experiments and tests are carried out

## You will believe a platform can float



1. “A 3 litre carbon-fibre compact air bottle is mounted on the floating platform as the compressed air source,” says Barış. “The selection of carbon-fibre material for the air bottle aims to reduce the overall weight carried by the floating platform.”



2. “The criteria for selecting an air-bearing are directly linked to the weight of the floating platform and the load it will support. In this floating platform setup, a 40 mm diameter flat round air-bearing is employed, capable of lifting a maximum of 22.5 kg when actuated at 5 bar. When combined, three air-bearings can achieve a maximum lifting capacity of 67.5 kg.”



3. For longer operations, two 30 litre pressure bottles with a maximum capacity of 300 bar can be connected to the platform for a direct source. This extends use from 40 minutes to up to eight hours of operation.

OceanofPDF.com

# Pepper Pi

## Raspberry Pi fans with multiple models will relate to the rationale behind the origins of the transparent computer, says **Rosie Hattersley**



**Maker**

**Martin Spendiff and Vanessa Bradley**

Serial inventors Martin and Vanessa enjoy videoing their Raspberry Pi builds and have inspired many makers to create their own fun projects.

**veeb.ch**

**Documenting quirky makes comes naturally to creative couple Martin and Vanessa.** Collectively known as Veeb, they've been making madcap designs accompanied by fun videos of each build for the past five years. Their latest project, Pepper Pi, makes use of a headless Raspberry Pi 5 they had available to recreate a Victorian illusion known as Pepper's Ghost in which an object appears to float in a glass dome. The project also has its roots in early 1900s stock tickers. Impressively, Pepper Pi functions as a perfectly usable transparent PC, too.

## Seeing is believing

Originally conceived during lockdown, Veeb's videos soon caught on (and meant they didn't need to do quite so much worthy documentation!). Imitation is flattering, and when Martin and Vanessa realised other people enjoyed some of the videos they'd made combining old tech with new sufficiently to create their own versions, they embarked on an ongoing series. "We started to hear about people reproducing what we'd done. That felt good and was enough to make us keen to keep sharing projects whenever we thought we'd done something vaguely interesting." Veeb now have quite a following – 11,500 according to their YouTube follower clock: **rpimag.co/ytsubscounter**.

The day-to-day business involves custom builds and "reimagines vintage design for contemporary living". Martin tells us. "The Pepper Pi project was prompted by the fact that there are headless Raspberry Pi computers dotted about our office taking care of various tasks."

Although "ever so useful", a scattering of hardware components doesn't produce quite the contemporary finish to their aspirational design studio that, say, scatter cushions or artfully placed house-plants might. Instead, the aim was to spruce things up and to make a selection of "Raspberry Pi computers into interesting things to look at".



01. Pepper Pi recreates a Victorian mirror illusion in which an image appears to float

02. Any Raspberry Pi can be used, but Veeb chose Raspberry Pi 5 with 16GB RAM so it would also serve as a PC

03. Mirror angles and a privacy shield plus some **mpv.io** footage create the compelling visual trickery

> *Veeb now have quite a following – 11,500 according to their YouTube follower clock*




1920s ticker tape machines and Victorian illusions inspired this build, as Veeb's video explains

## Old hands

Martin and Vanessa have many Raspberry Pi builds under their belts. Here, they used Raspberry Pi 5 to run the statistical package R (**r-project.org**), "so we wanted to use something meaty for number-crunching". However, the illusory effect with the screen could be done using "more modest hardware" – they have since created one using Pico. A previous project involving an old Super 8 video viewer (**rpimag.co/boostbox**) showed them the potential of lower-resolution screens and how to make the most of them.

They still use the Boostbox cyberdeck daily. As with the Pepper Pi project, it combined “a redundant bit of technology and something box-fresh”.

Their take on the 150-year-old Pepper’s Ghost illusion involves a pair of mirrors bouncing a hidden image to make it appear within a transparent dome they bought online. They had seen versions hidden in arcade cabinets, but worked out the trickery for themselves. It took a while before they hit on the idea of using a privacy screen to shield the image so it was not on obvious display. In Vanessa and Martin’s version, the source image is hidden behind an acrylic disc which they commissioned “a nice man in Germany (**formulor.de**)” to laser-cut.

Originally, they set the viewing angle for the mirror to 45 degrees, but soon realised this didn’t work well for viewing the illusion when the Pepper Pi was sat on a desk and in use as a PC. Since the angle of the reflector is pretty important, Veeb created a shape calculator to assist anyone keen to recreate the project: **rpimag.co/pepperpi**.

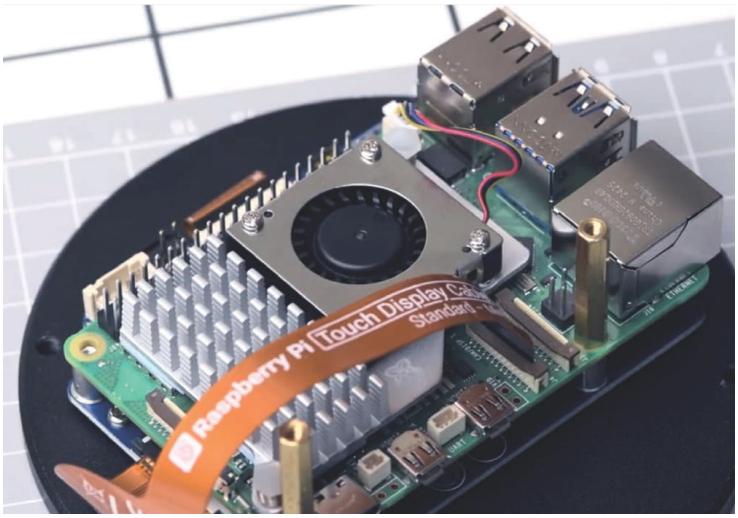
With the trigonometry taken care of, (and an optional screen attached), they ended up with “a functional computer that has a nice steampunk vibe”. You can even run it as an ornament that can act as a server while intermittently playing that hologram message that Princess Leia sent via R2-D2. Martin fervently hopes someone will actually do this!



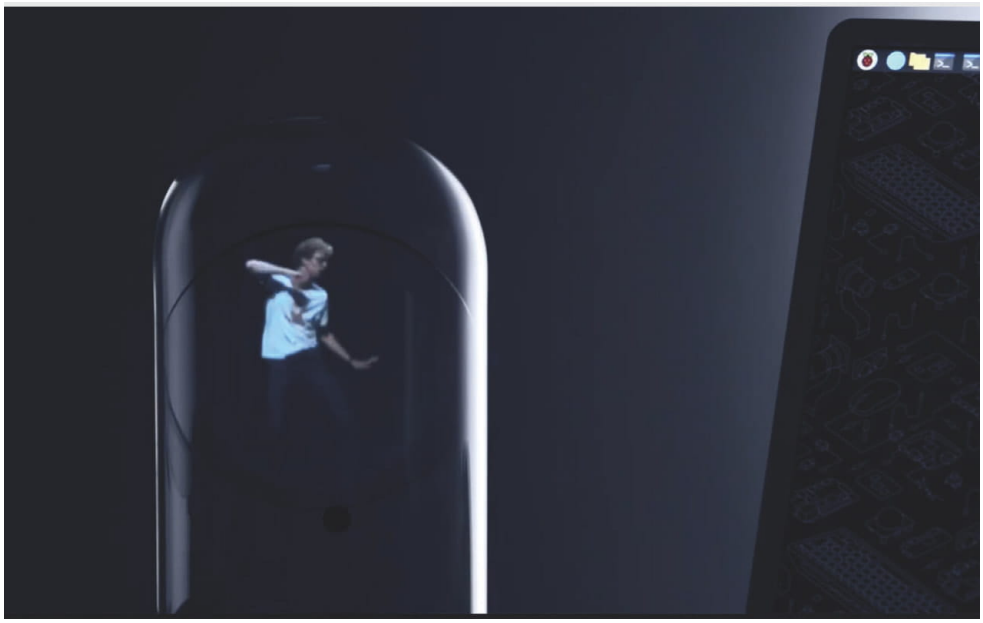
To the uninitiated, the dome simply looks like quirky desk furniture rather than a transparent PC

## Quick FACTS

- They used R Server to write a program to create images from data
- They also used it to make a program to count YouTube followers
- Veeb ran a contest to win their Pepper Pi
- Entry involved submitting a poem about waffles
- More than a hundred poems about waffles ensued

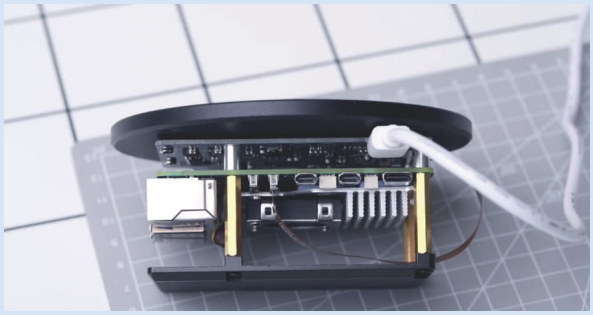



Mounted on the underside of the disc-shaped base for the glass dome, Raspberry Pi 5 plays the images or videos

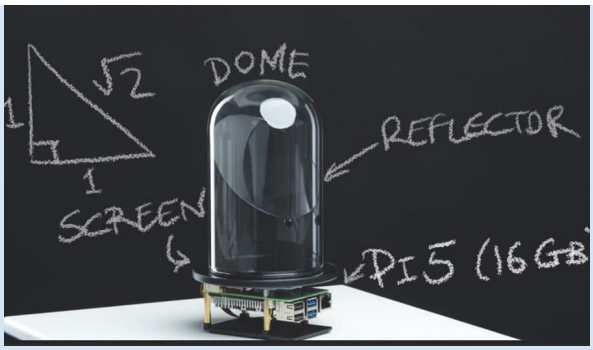


The duo had fun seeing what they could make appear inside the glass dome

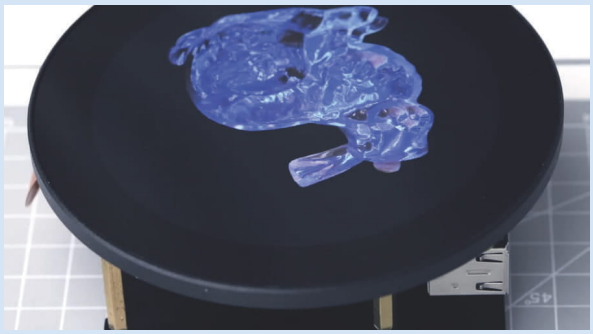
## Use your illusion



1. You will need a Raspberry Pi 4 or 5 computer, ideally with 16GB RAM, plus a suitable dome for the illusion, a base, and two small circular mirrors.




2. Set up the source image and mirrors, using the handy reflector shape calculator on Veeb's website: **rpimag.co/pepperpi**. Veeb used MPV (**mpv.io**) and some tweaks to Raspberry Pi OS to include video effects.



3. Further construction details plus the all-important maths (for calculating mirror angles) are shared in this fantastic YouTube video: **rpimag.co/transparentcomputer**.

*OceanofPDF.com*

# Avanade Intelligent Garden

## Inviting visitors to interact with the plants at the Chelsea Flower Show proved a positive showcase for AI and Raspberry Pi 5, reports **Rosie Hattersley**




**Maker**

**Pete Gallagher**

Electronic and electrical engineer Pete works for Avanade, and has been an enthusiastic user of Raspberry Pi since its 2012 launch.

**rpimag.co/avanade**


**Keen gardener King Charles III famously promoted the idea of talking to plants in a 1986 episode of children's TV show *Blue Peter*.** This same idea could be found at this year's Chelsea Flower Show, where technology company Avanade embedded sensors in trees and used an app to prompt conversations about each tree's health.

Alongside this, Avanade's Pete Gallagher created a bespoke Pollinator Camera, based on Raspberry Pi 5 and the Hailo AI accelerator, to photograph and detect the number of insects visiting the show garden each

day. The Intelligent Garden won a Gold award for Best Construction at Chelsea Flower Show.

## Trees, please

School children and conservation charities often use a form of AI to record wild visitors using hidden cameras and Raspberry Pi to identify them, but such uses could be better known. Avanade decided to bring a similar form of citizen science to people who may be unacquainted with how AI – or, more properly, machine learning – could be used to enhance their very own gardens.

To this end, Avanade approached renowned gardeners Tom Massey and Je Ahn to design an appealing garden with plenty of different species of trees, as well as plants that would attract insects and butterflies. The Times journalist Lucy Bannerman described the Intelligent Garden as a “garden that tells people how it is feeling”. Given the blazing heat in London one late May, it mostly reported (via an auto-generated text message) being pretty thirsty!

Avanade’s Helen Woodfield explains that plenty of trees are being planted in the UK. Leaf cover helps to cool streets, remove pollutants, and provide much-needed shade. However, all the stats say that street trees in cities are really suffering; 50% die within ten years, wasting money and the goodwill of those who fundraised or paid for each tree. Having people adopt individual trees, regularly watering them with the necessary 30 litres per week, completely changes this narrative.




**01.** A Pollinator Camera installed on the pavilion roof at Chelsea Flower Show recorded insect visitors

**02.** Raspberry Pi 5 and Hailo AI accelerator automatically recognised and recorded photos of bees and butterflies

> *All the stats say that street trees in cities are really suffering; 50% die within the first ten years*

## Nurturing green shoots

Alongside the chatty trees, the Intelligent Garden featured a Raspberry Pi 5-based AI Pollinator Camera Trap on the roof of the main Chelsea Flower Show pavilion which was covered in a meadow of wild grass. The bespoke

bird box by Sebastian Cox is a thing of beauty in its own right. “This camera trap aims to replicate a Royal Horticultural Society initiative called FIT (Flower Insect Timed) counting, which aims to monitor the number and type of pollinating insects visiting a quadrant of garden space and landing on flowers so the garden owner can assess whether they are attracting enough butterflies and bees,” says Intelligent Garden designer Pete.

“By supplementing good horticulture and science with the power of AI, we hope to improve the chances of urban tree survival and help to maintain the existence of our much-needed urban green spaces.”

The Pollinator Camera recognises pollinators and keeps a running total of the number of insect detections in the past 30 minutes over a 24-hour period along with thumbnail images of each insect or butterfly identified. “It can detect flowers, bees, and butterflies currently and, when insects appear in frame, we take still images as well as videos, both with bounding boxes to show the objects for use in the application,” Pete explains. The YOLO object detection setup currently records each instance, with tags added using Label Studio. He tried using AI to label them, but it was hit and miss so he ended up doing this part manually.




A custom-built bird box houses the Pollinator Camera



Pete in the Intelligent Garden with the Pollinator Camera hardware

The next version of the Pollinator Camera software will identify individual bee species and insect types and record how many of each visits the garden. The live data means the thumbnails already being logged can be added to the dataset (or zoo) to train the next iteration of the software to improve its accuracy. “Of course, if you start training on an image that’s got a box around it, it’s going to be very quick to identify boxes, not bees. It’s the same for the video as well.” To get round this, Pete created “a little

piping script” that captures one-second interval screenshots from the video. Avanade could use each of those “because of course, the position of the bee would change, and it would rotate, and it seemed important to train it on all of those”. There are detailed build instructions at **rpimag.co/intelligentgarden**. Meanwhile, Avanade urges makers and nature fans to try and build their own versions and share their findings. There’s even talk of a competition for the best young citizen science projects based on the one at Chelsea Flower Show!




The Pollinator Camera detects and logs numbers of bees arriving in the flower garden transect

## Quick FACTS

- Pete previously built stamp vending machines with modems for the Royal Mail
- He also worked on a project to save elephants near the Zambezi river
- Avanade’s aim was to introduce AI to people who have only heard about its downsides

- Citizen science and engagement with nature caught the imagination of visitors
- The Intelligent Garden won the Chelsea Flower Show's Best Construction award

## Set up a smart garden



1. A pre-built or 3D-printed bird box can be used to hide the camera-based setup. Using an 8GB Raspberry Pi 5 with an HQ Camera, 40W power supply, and the Raspberry Pi AI HAT+ 26 TOPS made integration very simple, says Pete.



2. Installed on the pavilion roof, the camera automatically logs insect visitors and sends data back to Avanade's server via a fixed Power over Ethernet connection.




3. Insects identified are shown in a bounding box. The thumbnail imager can be dragged into your dataset for ML training purposes. Build instructions for the Pollinator Camera can be found at **rpimag.co/intelligentgarden**.

*OceanofPDF.com*

# Embedded MIDI Synthesizer on Raspberry Pi

## A synthesiser squeezed into an accordion sounds great. By **David Crookes**



**Maker**

**Sergey Antonovich**

Sergey is an embedded systems engineer specialising in real-time system software and sensor integration for self-driving cars and delivery robots.

**rpimag.co/rpimidisynth**

**Although it's often considered to be a quaint, old-fashioned instrument in large parts of Europe and North America, the accordion is more popular than you may think.** As well as playing a significant role in traditional, folk and ethnic music, it's used to produce jazz, pop and classical tunes. It's also of great cultural importance and a great symbol of musical expression.

Sergey Antonovich is certainly a fan and he loves to play the button accordion. He also spends a good chunk of his free time studying music theory and creating digital music instruments and accessories. To that end,

he’s been working on bringing digital synthesiser capabilities to an acoustic accordion while preserving the instrument’s authentic feel and sound.

> *Without a reliable and flexible sound engine, the instrument cannot speak*

To achieve this, Sergey has been using the Raspberry Pi Compute Module 3+, which he praises for its balance of performance and thermal efficiency. He’s also produced a version that uses the Raspberry Pi Zero 2 W computer and created it with simplicity in mind so that anyone can try it out and see how it works.

The result is a system that is more flexible and less expensive than off-theshelf MIDI synthesiser alternatives, that’s still capable of playing beautiful, digital sounds. What’s more, Sergey has fitted all of the electronics inside the accordion rather than have it trailing externally, which has allowed him to create a self-contained musical experience offering the very best sound quality.

## Melodic move

The project has been challenging. “When designing a digital musical instrument, the sound engine – the synthesiser – is not just a component. It is the instrument, in the ears of the audience,” Sergey explains. “No matter how sophisticated the keyboard, pressure sensors, or expressive interface may be, without a reliable and flexible sound engine, the instrument cannot speak.

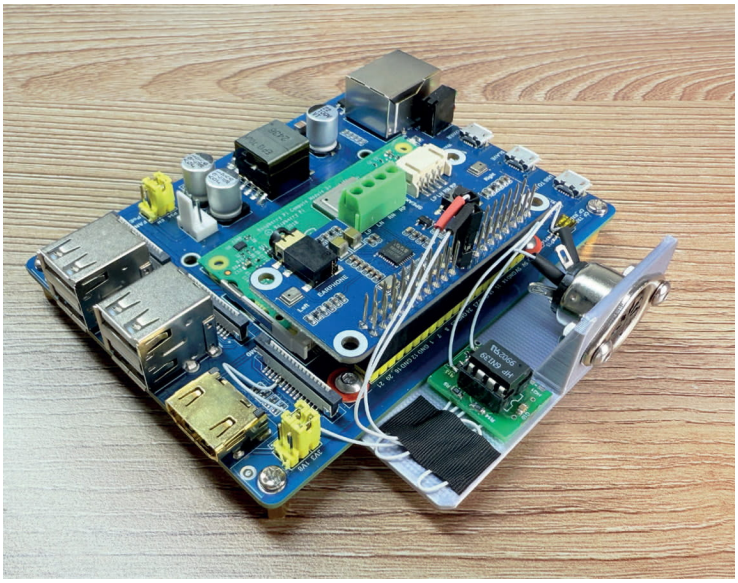



**01.** It still looks like an ordinary chromatic accordion, albeit with new control knobs

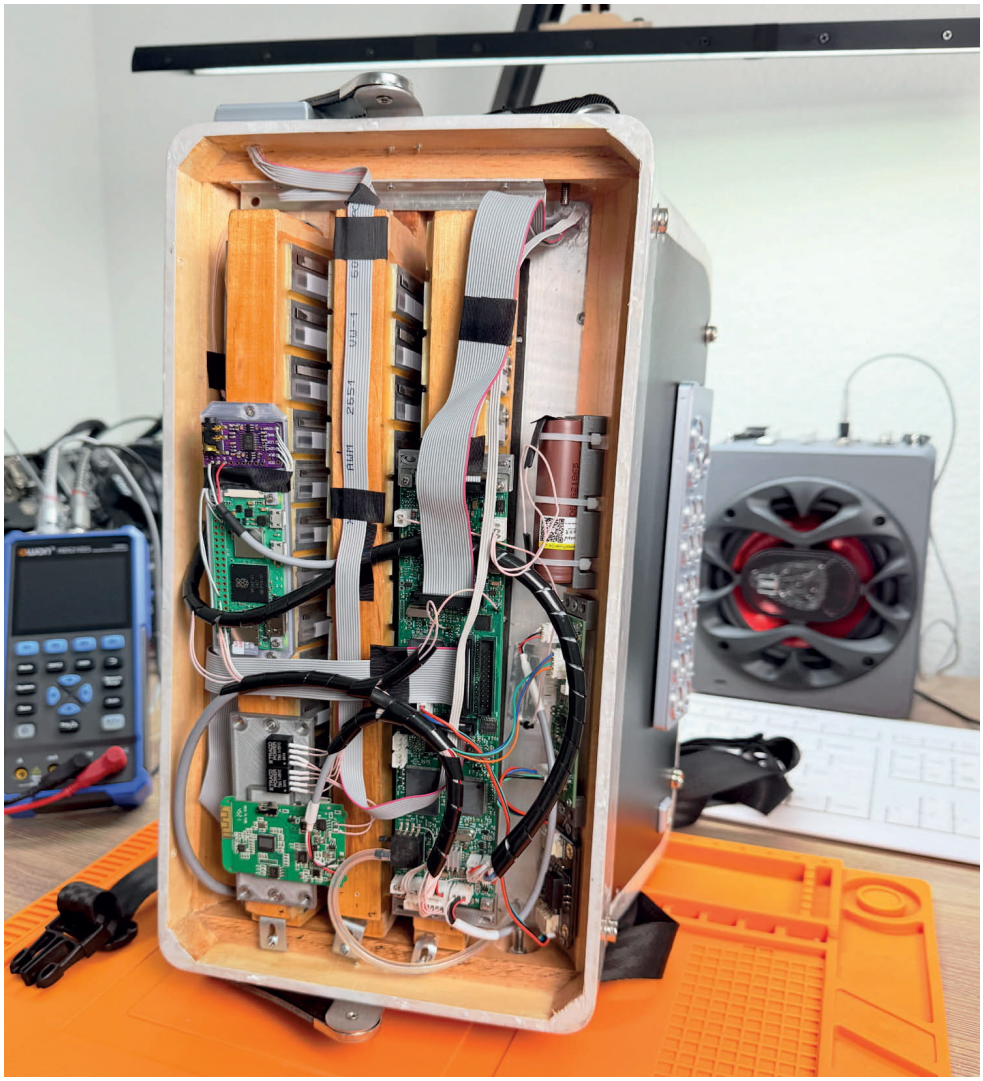
**02.** This white box on top is a digital screen that displays information such as sounds and battery life

"In my work developing digital accordions, this challenge became central," he adds. "I had already developed a working logic board based on a microcontroller – responsible for scanning buttons, detecting bellows pressure, and generating MIDI messages. But I needed a standalone, embedded MIDI synthesiser: a system that receives MIDI input (via

hardware DIN or USB), and outputs audio in real time. Crucially, it had to be small, reliable, low-latency, and flexible enough to support wavetable synthesis with user-replaceable sound banks.”



This is the first headless prototype on Raspberry Pi CM3+. It boots straight into performance mode. With 3ms latency, it is in the same class as hardware synths



All of the components fit inside the right-hand half of the accordion and it doesn’t have the limitations of closed systems or hassle of external gear

Although Sergey did consider external MIDI synthesiser modules such as the MIDIPLUS miniEngine Pro, Ketron SD2 or Ketron SD1000, he didn’t want separate cables, power, and clunky mounting. He also thought about using integrated hardware solutions offered by a small number of companies whose integrated circuits provide sample-based wavetable synthesis in compact form. But, again, he saw problems.

“They present several challenges for independent developers,” he says. “There is no public access to development kits or documentation. They use proprietary sound bank formats, editable only with internal or vendor-provided tools. They have locked feature sets, meaning custom user banks

are difficult to implement without proprietary tools, licences or vendor support. And cost-effective ICs like the SAM2695 or VS1053b rely on compact, generalised samples that may not fully capture the nuance of acoustic instruments.”

## Bellows power

As a result, Sergey began exploring a fully open approach. “I decided to use FluidSynth, a software synthesiser that supports SoundFont 2 wavetable banks that can be tuned for real-time operation on Linux-based single-board computers,” he says.

FluidSynth is a software synthesiser that responds to real-time MIDI input by generating audio and it has a shared library that can be used in other programs. It also has a built-in command-line shell. Running it on Raspberry Pi Compute Module 3+ appeared the most logical solution to Sergey’s problems and it means he could take advantage of the four ARM cores and 1GB of RAM, which is enough to load most SoundFont 2 banks.

“Using Raspberry Pi for audio synthesis offers several key advantages,” he says. It has an open software stack with no vendor lock-in and full control over tuning, latency, and effects. When combined with FluidSynth, it also enables full support for the SoundFont 2 support, allowing users to replace or customise their sound banks, which is a highly requested feature among musicians.

## Quick FACTS

- Sergey creates interactive digital musical instruments
- They can mix acoustic and electronic sounds in real time
- This project creates the sound engine – the synthesizer
- It’s based on Raspberry Pi hardware (CM3+ or Zero 2 W)

- It runs FluidSynth and supports SoundFont 2

“It also offers a good performance-topower ratio which is critical for embedded use in sealed housings,” adds Sergey. “And there’s a strong developer ecosystem with up-to-date kernels, documentation, overlays, and toolchains.”

Raspberry Pi Compute Module 3+ can be integrated in a compact board design as well. “For benchtop prototyping, I used the Waveshare Compute Module PoE Board, Waveshare WM8960 Audio HAT, and a simple DIN MIDI input schematic on a 6N139 optocoupler,” Sergey says.

## Sound journey

For the instrument to work effectively, the time between pressing a key and the sound playing needed to be short. “The most critical metric for live performance is trigger-to-sound latency and, while hardware synthesisers typically achieve 1.5 to 3 milliseconds (ms), software solutions must carefully balance CPU load and audio buffering to stay within acceptable range – ideally below 10 ms to remain imperceptible to the performer,” Sergey explains.



The system is also designed for safe power removal: thanks to a read-only root file system, the module can be switched off simply by cutting power

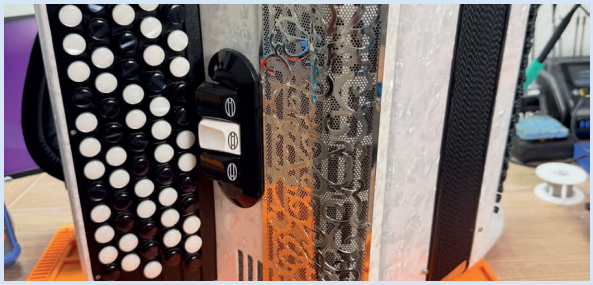
Using a minimal Buildroot-based Linux system and carefully tuned audio parameters, he achieved a consistent average latency of 3ms (from receiving a MIDI event to the sound being generated). Including MIDI transmission of around a millisecond, the total response time has come in at 4ms. ‘That’s fully competitive with dedicated hardware synths,” Sergey notes.

The system also handles playing 64 different notes simultaneously, which Sergey says is sufficient for a digital accordion with layered auto-accompaniment. “The audio output is clean, and no artifacts or dropouts occur under load.”

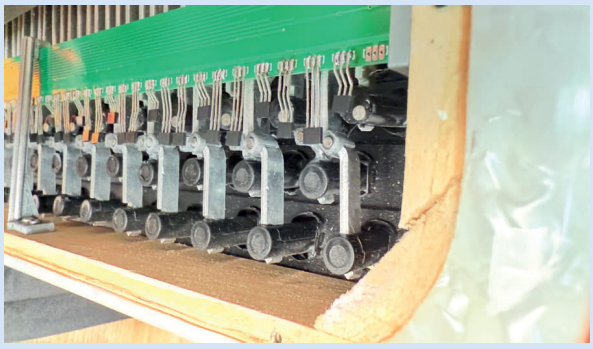
The result is excellent latency, full musical responsiveness, and great sound quality in an instrument that is fully self-contained and can be played without relying on any external gear – enabling untethered movement on stage when paired with a basic wireless audio transmitter.

And while the instrument can be played electronically, the acoustic sounds still respond naturally to bellows movement while digital voices can be configured to ignore or follow the same motion. “It confirms the viability of this setup in a real, stage-ready instrument,” he says.

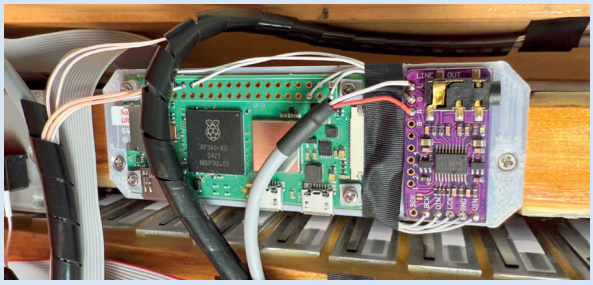
# Embedded MIDI synthesizer on Raspberry Pi



1. Accordions work by using the instrument's bellows to force air through reeds. This produces a sound. In Sergey's system, there is a pressure sensor that can work out how the bellows are behaving.



2. The keys of an accordion control the sound's pitch. By having sensors on levers, Sergey's program can work out whether they're open or closed and interpret which keys are being pressed. A sound is then created.



3. All of the processing is carried out on the synthesizer developed using Raspberry Pi Zero 2 W. It was important to avoid sound delays greater than 20ms since they're noticeable and make live performances difficult.

OceanofPDF.com

# Track the stars with a reflecting telescope

## Gaze deep into the night sky with drainage piping and an affordable microcontroller. By **Andrew Gregory**



**Maker**

**Tim Ritson**

When he’s not working as a power systems engineer, Tim’s cramming loads of complicated astrophysics into home-built projects like this one.

**rpimag.co/esp32telescope**

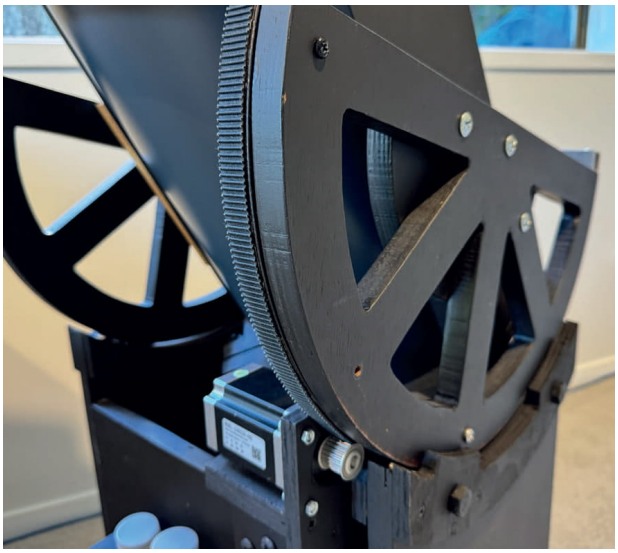
**Telescopes come in many shapes and sizes.** There are refracting telescopes, which use a series of lenses to make faraway objects seem larger – this is the type of device first used by Galileo to observe the planets. The telescope was improved by Sir Isaac Newton with the invention of the reflecting telescope, which used mirrors to reflect the light to an eyepiece. A larger aperture, and larger mirrors, means more light, and more magnification.

Now Tim Ritson has taken Newton’s basic design and refined it with the addition of computerised motor control, Wi-Fi, and HTML.

“To start with, I really just wanted to see if I could build a basic telescope,” he says. “Turned out that part was relatively easy. The mirror and a couple other components are a little expensive, but the rest was all just obtained locally. It was hard finding the right-sized tube to fit my mirror. I had to drive to a farm supplies store in the country and strap a 2m-long section of culvert pipe to the roof of my station wagon! But after that the scope came together with few hiccups, and it worked pretty much first time.”

So what do you do with a reflecting telescope with a 1200mm focal length and 200mm diameter mirror? We suspect that most of us reading this would think about computerising it. And that’s exactly what Tim did.

“I built a stand for it. Then I wanted to see if I could motorise it. If that worked, I would program it to track stars and find objects just by entering coordinates. One step at a time, just seeing how far I could get.”



The user can elevate the telescope from 0 degrees (horizontal) to 90 degrees (straight up)


01
02

**01.** You can see loads of celestial bodies though this telescope – the Orion Nebula is visible as a wispy grey cloud

**02.** One full circle of rotation can be divided into 160,000 steps — that's a lot of accuracy for a telescope made out of a drainage pipe!

> *I had to drive to a farm supplies store in the country and strap a 2m-long section of culvert pipe to the roof of my station wagon*



**Warning!**

**Electrical Safety**

Please be careful when working with electrical projects around the home. Especially if they involve mains electricity. **rpimag.co/electricalsafety**



If anything, the telescope is too powerful to look at the moon comfortably, as it appears too bright

## Search the stars

When you're looking at something a long way away, even a tiny range of motion makes a huge difference. "I used two 200-step NEMA23 motors from AliExpress, with TB6600 drivers that convert a logic 5V pulse from the ESP32, into the electrical signal the motor needs to move exactly one step," Tim explains. "They also have a 'micro stepping' feature which divides each step into up to 32 steps, so the motors have 6400 steps in one revolution.

"When combined with the gear ratio between the small 20-tooth cog on the motor to the timing belt with around 500 teeth per full circle… each step corresponds to moving about 8 arcseconds, or 0.002 degrees. To put it another way, if you did one step per second, it would take almost two full days to make a full revolution!

"In terms of altitude (up/down), it can move from horizontal to straight up (0–90°). And in terms of azimuth (think compass heading), it can rotate infinitely (I built in a slip ring so the power cable doesn't get twisted as it

rotates). With these two movements, a full hemisphere of sky is within sight. If coordinates for an object that is below the local horizon are entered, the altitude calculation will give a negative number, so I have programmed it to not try and rotate to it. I don’t want to find out which bit breaks first!”

Users are able to control the telescope manually using a joystick, but there’s more to this build than that: “For automated control, the ESP32 hosts an HTML page that can be accessed from my phone when on the same Wi-Fi network (or it will create its own network if my home network is unavailable). This page has inputs for selecting a target and turning ‘tracking’ on and off, and gives me output data, such as the current position of the scope, and coordinates of the target.”

## Automatic astronomy

“I have programmed a list of the 50 brightest or interesting stars, which can be selected via a drop-down,” adds Tim. “You can also enter coordinates (right ascension and declination) for any target which you can get online. Incidentally, it cannot currently track planets, as they move differently compared to background stars. Most stars essentially have fixed coordinates and don’t move relative to each other (this is a slight simplification) so the altitude and azimuth of a star can be easily calculated, based on a given location and time. Planets trace more complex paths so will require a bit more programming to track accurately. This is on my to-do list!”

The maximum magnification of this telescope is 133×, which is good enough to make out the number plate on a car parked 1km away. But that’s not the most important feature according to Tim: “a wider aperture is also important as it collects more light. It sees objects in the sky that you wouldn’t even know are there because they are too faint for the naked eye. It’s like revealing an image drawn with invisible ink.



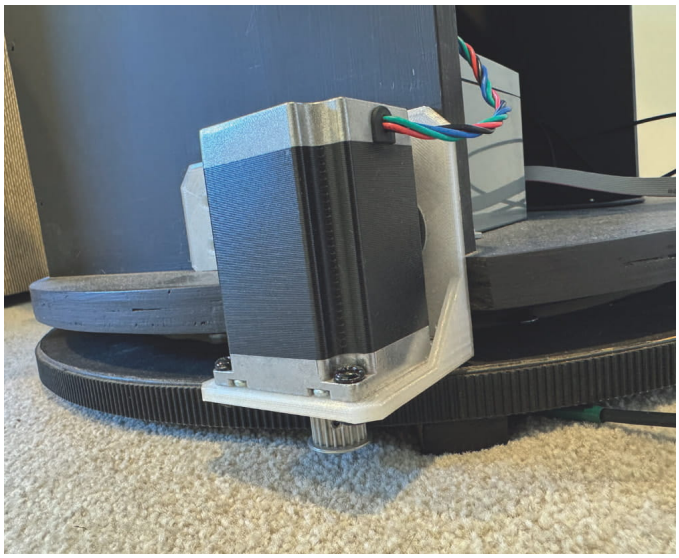
No lenses here: the power of a reflecting telescope comes from mirrors

## Quick FACTS

- The world's largest telescope is in La Palma in the Canary Islands, with a primary mirror 10.4 metres in diameter
- Tim used an ESP32 for Wi-Fi and Bluetooth capabilities; if you're happier programming in Python than in C, you could use a Raspberry Pi Pico W instead
- Tim had always wanted a telescope, ever since seeing the moon through a friend's telescope as a kid
- An earlier attempt at controlling the telescope used a 3D-printed geared wheel, which took 24 hours to print…
- …This didn't work, so Tim used a timing belt instead

"My first target was the moon – but when full, the large aperture reflects so much light into the eye piece it is slightly uncomfortable. I quickly turned to planets, Saturn and Jupiter. I could see the rings and moons of Saturn. Unfortunately at the moment, Saturn is in a part of its 30-year orbit

around the sun where the rings are almost side on to us, so they just look like a line. The stripes of Jupiter were clearly visible, as were its moons. They pretty much look like stars, except they form almost a perfect line through the planet. I have also looked at Mars and Venus, but they are less interesting. Once I had done the planets, I turned to star clusters and deep space objects. I have ticked a few off, including the Jewel Box (a tight group of stars near the Southern Cross), Omega Centauri (a globular cluster, which looks like countless tiny points of light which form a brighter centre and fade out towards the edges), and the Orion Nebula (not quite as colourful as all the Hubble pictures you see, more like a wispy grey cloud). I also managed to catch a comet, C/2024 G3 (ATLAS). I haven't seen a galaxy yet – I absolutely can't wait for this."

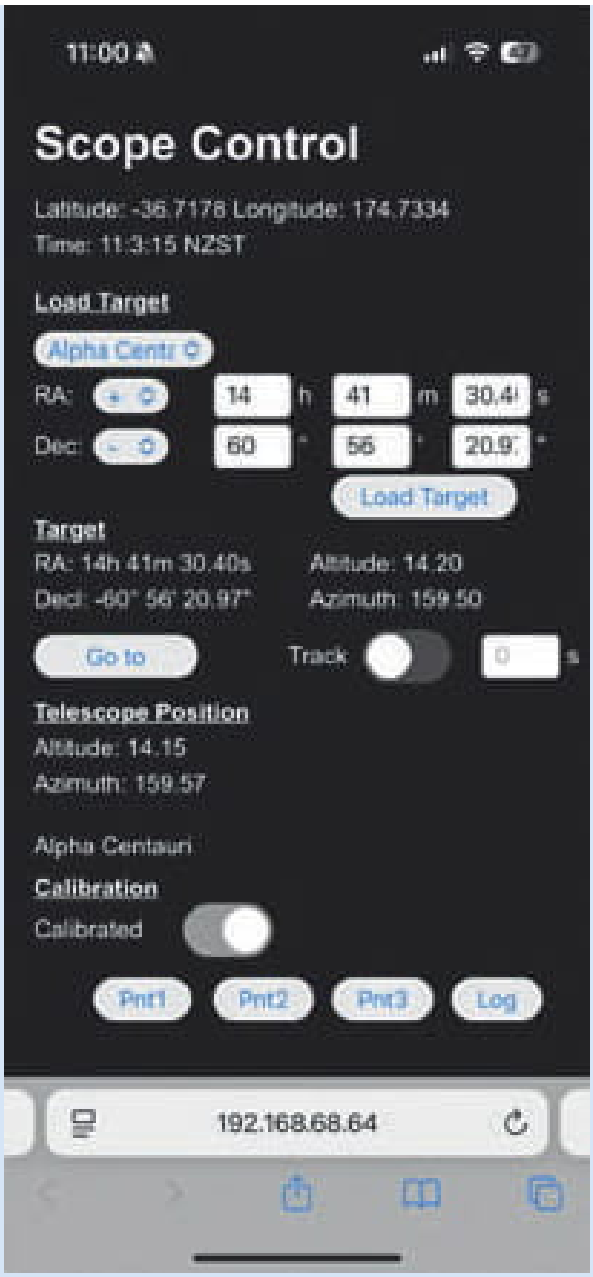


The stepper motors need driver boards to convert the signal from the ESP32 to the correct voltage

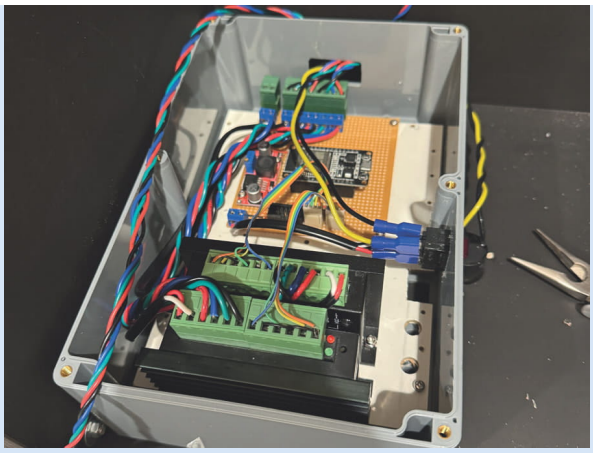
## Reach for the stars



1. Users can control the telescope manually using a joystick wired to the controller…




2. … or via a drop-down menu of 50 stars that Tim has coded into the software.



3. There’s an ESP32 board controlling stepper motors, and taking input via Wi-Fi and the joystick.

OceanofPDF.com

# DOOM SNES 2025

## Emulating a special co-processor for a bygone console is no problem for RP2350. **Rob Zwetsloot** plays with power while knee-deep in the dead




**Maker**

**Randal Linden**

Randal has a long history porting and emulating games in the video game industry, and worked with Matt Cope and Gavin Bell of Bitmap Bureau on this project.

**limitedrungames.com**

**bitmapbureau.com**


**Did you know that you can still technically make games for old consoles?** It's been a popular thing to do for special editions and anniversaries (promo material for the film *Sonic the Hedgehog 3* was even distributed on working Mega Drive / Genesis cartridges). Now is the turn of DOOM on SNES, whch celebrated its 30th anniversary with an updated re-release.

"We've updated and expanded the game with tons of new features including 14 new levels, improved frame rate and performance, added

circle-strafing, and even vibration effects with our new rumble game controller.” Randal Linden of Limited Run Games (LRG) tells us. “The final project uses custom software, tools, firmware, and hardware to create a unique cartridge that works in any Super Nintendo!”

That custom hardware? A Raspberry Pi RP2350 microcontroller chip, of course.

> *The RP2350 offered advantages that were impossible to resist*

## Super effects

DOOM on SNES was one of the few games to make use of the Super FX chip created for Star Fox (named Starwing in the UK, for weird legal reasons). This resulted in an extra hurdle in creating new, compatible carts for 30+-year-old console.

“[The Super FX chip] is a high-speed RISC co-processor that can be used to accelerate and optimise game logic, math, and graphics rendering,” Randal tells us. He would know: he was basically a one-man team behind the original port to SNES. “When we started working on DOOM SNES 2025, our initial focus was on FPGAs to simulate the Super FX chip, but the RP2350 offered advantages that were impossible to resist: first and foremost was the lower cost of the device, both in terms of production but also for development. Next, the numerous GPIOs and efficient PIO programs enabled us to interface the various memory devices used by the cartridge hardware with the Super NES easily and effectively. Finally, the Raspberry Pi SDK is open source and fully documented, which made our bring-up process nice and easy.”




**01.** The RP2350 chip replaces – and upgrades – a Super FX chip

**02.** It slots in to original SNES hardware and runs without any modding to the console

To prove the concept would work, LRG put together a mini console they referred to as Imp with off-the-shelf parts and a Raspberry Pi Pico 2.

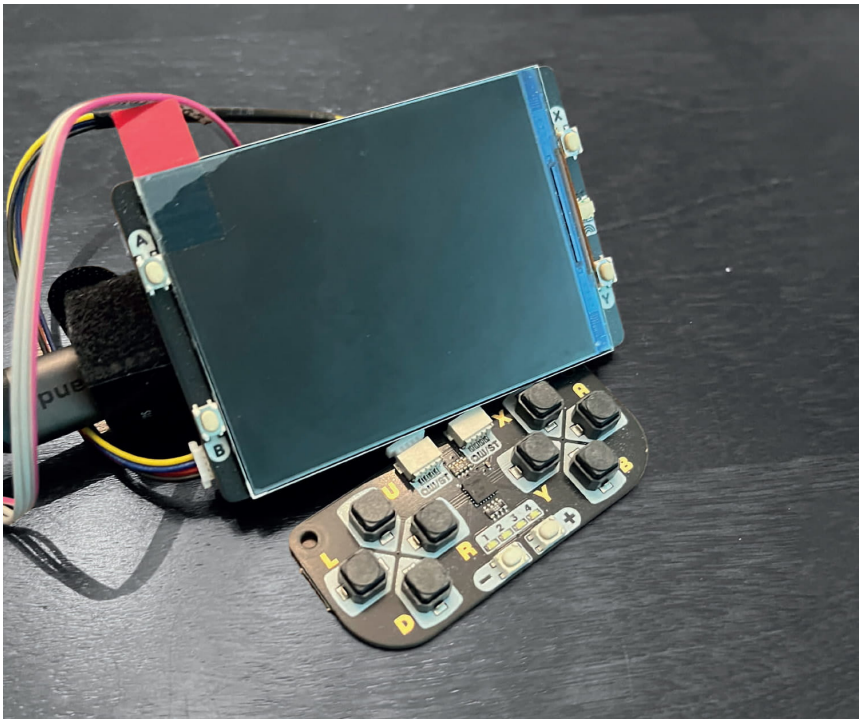
“We were able to prove our new ‘FX 3’ system was rendering and playing the game properly, even though at this point we weren’t connected to a SNES. We could also run lots of metrics and debugging quickly to get the FX 3 core stable,” Randal mentions. The original SNES port used a slightly upgraded version of the Super FX chip created after Star Fox,

colloquially referred to as Super FX 2, hence the FX 3 moniker. “We then designed our initial SNES PCB designs around socketing a Pimoroni PGA2350; it was a great way to prove the concept was then able to run on a SNES itself.”

## Rip and tear

In the final product, RP2350 performs three main functions according to Randal: a high-speed interface to multiple memory devices, simulation of Super FX, and ‘graphics format conversion operations’.

“We have a pre-production tool which takes the Super NES program executable code, processes it, and generates C source code which is then compiled using the Raspberry Pi Pico SDK,” Randal explains. “The resulting program includes logic which sets up PIO and DMA chains to respond to memory accesses in parallel with the Super FX simulation. After the frame buffer is generated by the Super FX simulation, the graphics conversion logic processes the image from its chunky graphics format into the required Super NES planar format.”




The proof of concept was a little system nicknamed Imp, one of the enemies in DOOM




The packaging apes the Super Nintendo game boxes of the time

## Quick FACTS

- Randal started the original port in his spare time
- id Software didn't know about it until it was ready
- The Super FX used in DOOM was clocked at 21MHz…
- … and had 512kbits (64kB) of RAM
- The Imp prototype used a tiny Qw/ST Pad

Creating the game like this also allows them to add more content that wouldn't have fit on the original cart – supposedly only 16 bytes were free when it was released.

## Playing with super power

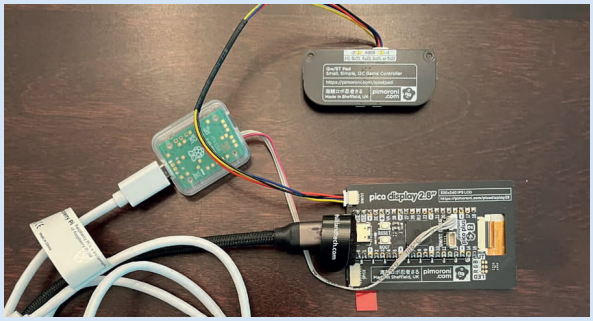
“DOOM is widely recognised as one of the greatest games of all time, but when it first came out, the best (and only) way to play the game was using an expensive PC,” Randal says. “At the time, many people didn’t have a high-end computer, but they did have a Super Nintendo, so it was for them that I developed DOOM for SNES.”

The game itself is out and ready to slam into your slightly yellowed Super Nintendo as of 2025. The moons of Mars are waiting your return.

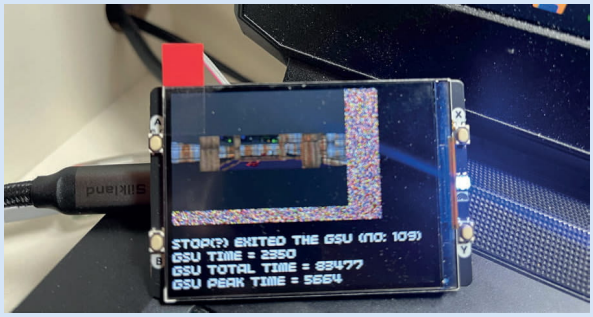



It is much more powerful than any game released during the original run of the SNES

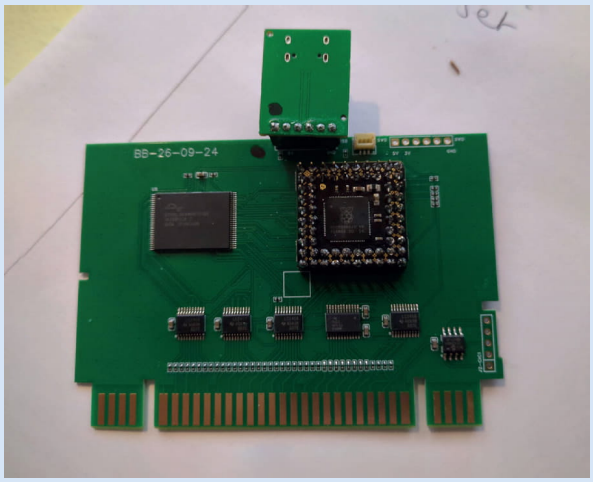
## Enhancing a port



1. Like a lot of electronics projects, you can create a prototype of your final circuit with parts you have lying around.




2. Debugging is important, so once the circuit is created, it’s time to test, test, test.




3. Finally, you can start creating custom PCBs and add an RP2350 into it, even if the SNES didn’t originally use one.

OceanofPDF.com

# Vertical Runner

## A cyberdeck that looks as good in real life as it does in a film script. By **David Crookes**




**Maker**

**James Reeves**

James is an American filmmaker known for his futuristic-hesh style. He is currently pursuing becoming a more versed ethical hacker.

**rpimag.co/verticalrunner**




The aluminium grooved grip with threaded holes was primarily chosen to look cool on screen

**Many Raspberry Pi projects are created for use in the real-world.** Not Vertical Runner. This one was designed as a film prop for a short science-fiction thriller that James Reeves was writing with his wife. It was set to star a lonely, introverted, skilled hacker looking to access rare cars and it was going to be shot at night.

“I wanted a really strong title sequence showing our main character coding, printing, and assembling and I wanted something unique as her main hacking tool,” James says. “It needed to be an extension of the main character and look great on film at night, perhaps in a phone booth, around a corner, or adjacent to a neon sign.”

James also wanted his device to double as a principal light source for the character, exposing the main character’s face. “It kind of just spiralled from there,” he reveals. As time went on, James learned about ethical hacking, cybersecurity, and Linux. The end result was a handheld computer that actually worked.

## Hacker training

Initially, James had little understanding of what a hacker would do. To learn, he built a hacking lab using two Raspberry Pi 4 computers – one acting as an “attack” machine and the other as a “victim”. He played around, getting comfortable with Raspberry Pi OS before hitting upon a configuration.

The Vertical Runner, he surmised, should comprise a Raspberry Pi 4 8GB, a Waveshare 3.5-inch touchscreen LCD, a Fly Way Bluetooth 3.0 keyboard, and a BrosTrend Dual Band Wi-Fi adapter, all powered using a Vilros battery pack. “It also needed to be an extension of the main character – someone agile, flexible, determined, resourceful and punk,” James says.

For this reason, he mixed practicality with aesthetics. “The screen had to work as a key light,” James explains, figuring a vertical screen would work best. “This orientation lent itself to the lighting plane shape I wanted for the main character’s jaw line,” he adds. But while it also looked cool and futuristic, it wasn’t without its difficulties.




“The main challenge was getting the entire screen to correctly display vertically,” he explains. “I could get the display to work, but it was only displaying a 4:3 aspect ratio regardless of the orientation and it would not fill in the bottom 200 lines of pixels. Finally, I just decided to try my own lines of code for the **config.txt** file and after about two hours of doing that, I got it to work.”

## Quick FACTS

- The device was designed as a film prop
- James wanted it to be agile, flexible, and punk
- It evolved into a fully fledged cyberdeck
- It doubles as a key light for the main character

- A strap enables the device to be easily worn



James Reeves working on a night shoot for one of his films

> *The main challenge was getting the entire screen to correctly display vertically*

## Punk aesthetic

James then sought to perfect the look. “My friend, Michael, owns and runs a 3D printing business called Precision Additive and we started putting concepts together,” James says. “We went through three iterations before the final one and added an unusual grip which I salvaged from a rotational handheld camera rig.”

The grip was included to allow an actress to use the device in different positions. “It could be rotated for use as a self-defence weapon too,” James says. “The handle also allowed me to use small lighting devices or car mount systems.” By producing a real cyberdeck, James was able to create something believable.

“I still intend to use it a film prop, but I’ve created a fully-functional penetration-testing mobile computer which works fantastically well,” James says. “It could have a better Wi-Fi monitoring/injection adapter, but I’m really happy with it. I’m now looking forward to starting the build of another cyberdeck.”

**01.** A dual-band Wi-Fi dongle (with antenna) allows the cyberdeck to run in monitoring or injection mode

**02.** The screen is mounted vertically and connects to a Raspberry Pi 4 8GB computer



A honeycomb backing is used to help prevent the device from becoming too warm

OceanofPDF.com

# Bog Body theatre prop

## It’s not every day a maker emails **Rosie Hattersley** with a tall tale about Raspberry Pi bringing a mythical creature back to life




**Maker**

**Sean M Tracey**

Sean runs a company that claims to be able to build anything – often with the help of Raspberry Pi.

**mitchelltechnologies.co.uk**


**rpimag.co/bogbody**

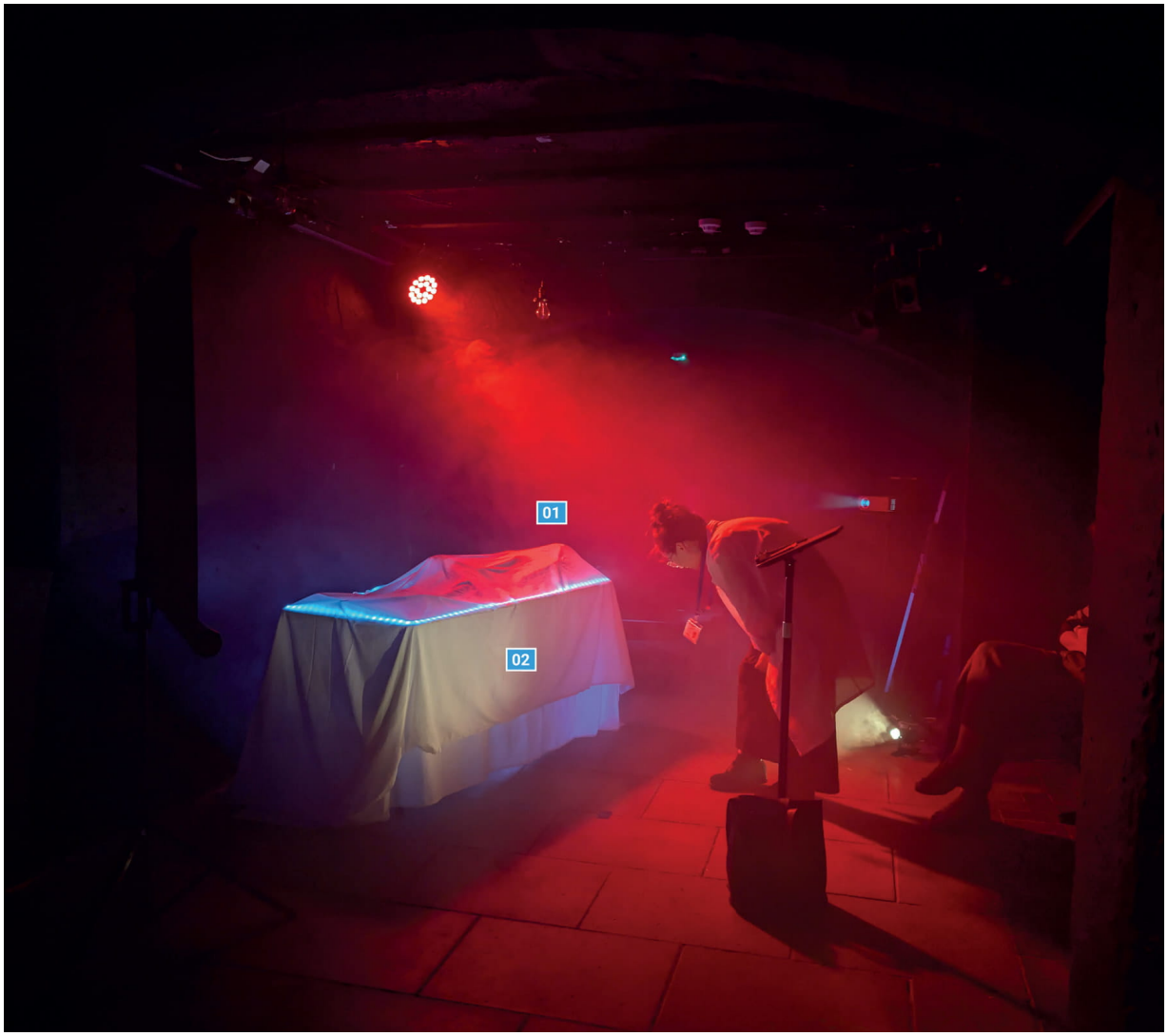
**We last encountered Sean Tracey engaged on an international trade mission somewhere between London and San Francisco, stealthily trying to set up a Raspberry Pi cluster computer in an anonymous-looking briefcase with the aid of some rather whiffy spray paints without attracting the attention of the conference hotel staff.** This issue, he’s left the latex special effects to the theatrical pros while he takes on the task of animating a fake 1000-year-old corpse with a little help from Raspberry Pi.

## Deadly serious

Sean helped attract attention to a previous company he worked for by animating a cutout of Elvis Presley and placing it in the office window, where it would follow people as they walked by and block their view of what was going on inside. Whether this had any bearing on his recent project is a moot point, but earlier this year Sean was approached by a theatre company about building a life-sized animatronic mummy that could rise up from a table unassisted. It would need to reach out for an actress on-stage in what, quite evidently, was going to be a horror-themed play. The titular bog body was to be a life-sized animatronic mummy that could rise up and then reach out for an actress on-stage.

> *Sean was approached by a theatre company about building a life-sized animatronic mummy that could rise up from a table unassisted*

The project wasn't as easy as it might sound, says Sean, "especially when there are so many unknowns, which is how things tend to be when working in theatre productions". Practical issues included how much load certain types of motors could take and the materials that would make up and support lifting a full-sized latex-covered plastic skeleton weighing around 8kg from resting to bolt upright in less than 30 seconds. The whole thing also needed to be collapsible (for ease of transportation and storage) and able to be up and running on stage in less than ten minutes.




**01.** Bog Body showcases the folklore behind corpses preserved for centuries, with the twist that this one appears to be alive

**02.** Two stepper motors are controlled wirelessly by 8GB Raspberry Pi 5. They enable the latex-clad skeleton to silently sit up and reach towards the actress

Having used Raspberry Pi in approximately 80% of his previous creations, Sean immediately knew his best bet would be using one to run some custom Python code that could hook into the theatre's systems sat at the heart of the project. "Honestly, I don't think it would have been possible to build this kind of project without it."

## Use your illusion

Being “a big believer in having more capacity than you think you might need”, Sean set up Raspberry Pi 5 (8GB) knowing it would provide reliable networking capabilities, could operate silently – critical for maintaining the eerie atmosphere of the play – and could easily drive two motors via a stepper motor HAT, With reliable networking capabilities, and the ability to drive at least two motors simultaneously. “It’s always better to have more power and not need it than to need it and not have it!”

Importantly, Sean remembered the Elvis project and corrected an error. The cutout was supposed to move in lock-step with the curious passer-by, but precision of movement that stepper motors offer came at the cost of speed. “You could set the motors to move an exact amount, but at no more than a few revolutions per second.” Sean resolved this with some clever gearing and C++ coding. “For the mummy, precision over speed was exactly what I needed, so having almost the exact same motors I used a decade ago powering the raising of the mummy was perfect!

Libby Morris, who Sean describes as “an incredible London-based horror artist who’s built more terrifying things than I could ever imagine!” – built up the body’s latex skin in layers to look as though it had deteriorated over a long period of time. The two of them had less than a month to bring the bog body to life, complete with 14 3D-printed parts and counterweights carefully geared to control the speed at which it rises up and extends its finger.

As well as writing code to control the bog body, Sean had to work out how to connect everything and make it work with the Theatre Control Panel (TCP), a popular piece of software used in theatres for controlling sound, video, and lighting cues. “Thanks to the huge ecosystem of Raspberry Pi HATs and software packages, bringing all of the parts together that we would need to bring the mummy to life, it took less than two days after we had the work package signed-off,” he says.

Bog Body opened to much acclaim at Camden Fringe Festival in August and has since been on tour in the UK.



Actor Jen Tucker encounters the bog body in the dress rehearsal




Bog Body cast and prop onstage at Camden Fringe Festival



Sean shows off the cadaver before it’s hidden under a sheet until the very end of the play

## Quick FACTS

- Bog bodies are naturally mummified and preserved, often in peat bogs
- Several dozen have been unearthed in the UK and Ireland
- The pH levels of bogs effectively pickle the bodies, preserving them
- Sean’s bog body is animated using counterweights and stepper motors
- Raspberry Pi 5 controls them in a similar fashion to a grandfather clock

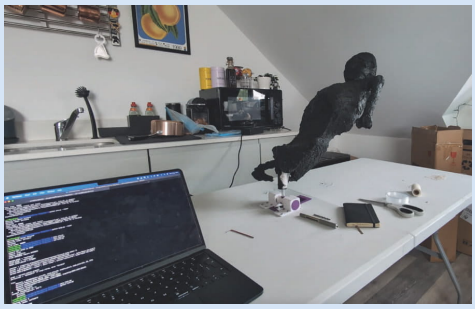
## Looking lively



1. Theatrical horror artist Libby Morris covered a life-size plastic skeleton in latex skin. Sean then drilled holes where it would need to pivot so it could be raised.



2. A steel bar hidden beneath the body and attached to geared counterweights causes the bog body to smoothly and silently sit upright and fall backwards.



3. Raspberry Pi 5 controls the NEMA stepper motors remotely using an API Sean wrote for the project. Once the play has finished touring, he hopes to post the code for animating mummies on his website.

OceanofPDF.com

# Yashica IR camera

## This brave build uses technology to take features away. By **Andrew Gregory**



**Maker**

**Malcolm Wilson**

A tinkerer since he was a teenager, Malcolm now works in the automotive industry.

**rpimag.co/camerahacksmj**

**In the olden days, we'd point our cameras at scenes, press a button, then have to wait to get the film developed so we could see if the photos we'd taken were any good.** The process was slow, and it forced the user to be selective and deliberate with what they were doing. Nowadays our cameras are our smartphones, and we hardly even look at the photos we take, still less print them out and have them around the house.



The changing colours of autumn leaves look great under IR light

One maker who wanted to go back to the way things were is Malcolm Wilson. Malcolm's a prolific camera hacker, adding 3D-printed bits, Raspberry Pi Camera Modules, and copious amounts of Python code to get things to work just so. This is his latest build: it's based on a Raspberry Pi Camera Module 3 NoIR, and it's just the thing if you like taking weird infrared photos that make it look like you're wandering in a post-apocalyptic landscape. "[Photography] was a hobby that I picked up during the pandemic to stay creative", says Malcolm. "I started out shooting mainly film and then branched out into modifying cameras. I've always loved the look of Aerochrome film and wanted to experiment with different ways of achieving the same look without spending thousands on a converted speciality camera."

## Quick FACTS

- Malcolm took up photography as a hobby during the pandemic
- No cameras were harmed in the making of this project…

- … as the donor camera was already broken when Malcolm acquired it
- This build uses a Raspberry Pi Zero 2 W and a Raspberry Pi Camera 3 NoIR



It’s perfect for making moody, melancholy montages







This build requires a variety of skills to get right

**01.** This camera is built into the body of a Yashica Electro 35 film camera – as used by Peter Parker in the 2012 film, *The Amazing Spider-Man*

**02.** The screen on the back tells you how many photos you’ve taken… and not much else

> *It was my first time using the Camera Module 3 and I was pleasantly surprised by the image quality and the speed of the autofocus*

“It was my first time using the Camera Module 3 and I was pleasantly surprised by the image quality and the speed of the autofocus. I wasn’t expecting it to work so effectively as a point-and-shoot with minimal tweaks,” he says.

Keen-eyed readers will notice that there’s something missing compared with most camera builds: a screen. There’s a viewfinder, and a little OLED that tells you how many shots you’ve taken, but most cameras show you what the photo is going to look like before you take it. Not this one: instead, you capture the moment as the muse directs you, turn the camera off, then turn it back on when you get home. Its Raspberry Pi Zero 2 W then connects to your home Wi-Fi, enabling you to SSH in to retrieve your images.

The electronics of this project comprise the aforementioned Raspberry Pi Zero 2 W, Camera Module 3 NoIR, 720nm IR lens filter, a rocker switch for turning the camera on and off, a momentary push-button, LiPo battery with 5V buck converter, and an I2C OLED screen. Assembly takes some doing: Malcolm says that anyone willing to try this project needs to be comfortable with disassembling a film camera, cutting metal, soldering, writing and editing Python code, and 3D printing – there’s a 3D-printed mount for the Raspberry Pi Zero 2 W within the camera because, as Malcolm says, “It took me awhile to get the right orientation to fit all of the components in the camera body.”

There’s enough battery life to handle a long walk in the woods, and plenty of storage – “I’ve been able to get 2–3 hours of use; I haven’t done a formal test,” says Malcolm. “I didn’t put any restrictions in the code for the max number of images. The only limit is the size of the internal memory card.”

So, make your own (or something similar) and get out into your nearest forest to take pictures of the autumn leaves – they look amazing in IR.

OceanofPDF.com

# Sfera Baltic boat

## A 60-year-old tug boat is flawlessly piloted by the latest Raspberry Pi and Compute Modules, finds **Rosie Hattersley**



**Maker**

**Ulderico Arcidiaco**

Ulde is the co-founder, CEO, and chief hardware engineer at Italy's Sfera Labs.

**sferalabs.cc**

**In 1962, The Baltic, a 26 metre-long Norwegian tug, took to the oceans.** Unlike today, boats of the 1960s were largely manually controlled. Little wonder because the most widely used computer was IBM's 1401: it had a 16,000-character core memory and weighed several hundred kilos. The 1401 was so expensive that most companies would generally rent one.

The 1401 computer took 17 seconds just to boot up and lacked both software and interfaces needed to work in industrial, let alone marine, scenarios. Other early computers such as the renowned DEC PDP-1 would have been no more suitable or affordable to control a commercial craft. Yet by the time The Baltic tug had been in operation for 20 years, computing was beginning to make real inroads, and computer-controlled ships came on

stream. Around this time, The Baltic was taken out of service and sold on. Eventually, time and computer capabilities caught up with the venerable craft, and in 2021 The Baltic embarked on a new life, with a brand-new control system built around Sfera Lab's Raspberry Pi and Compute Module Strato and Iono hardware: **rpimag.co/piboatrestore**.

## Heritage value

By the early 1980s when The Baltic tug was decommissioned, huge strides had been made in computing capabilities and costs had plummeted. After languishing on the south coast of England and then voyaging to the Mediterranean, The Baltic's fortunes changed. "The new owner had the ambitious goal to fully refit the boat and transform it into a yacht, preserving the original design and heritage of the unit," explains Ulderico Arcidiaco, the man tasked with modernising and fully automating the revitalised craft.

Standard PLCs (programmable logic controllers, often described as industrial computers) are used for the power management and some automation aspects of most ships, while everything else is done separately. Ulde was approached to add automation and decided on an audacious plan for an IPMS (Integrated Platform Management System) such as those found on modern military units or very large commercial ships. The fully integrated approach controls of all of the ship's subsystems. The setup would cost a fortune to implement the traditional way, but made sense using Sfera's Raspberry Pibased PLCs and the firm's own supervision software. He installed Strato Pi units to provide full, integrated control of The Baltic's core and supplementary functions, from power distribution to tanks and pumps control, navigation, alarms, fire, lighting, stabilisers, chargers, inverters, battery banks, and video feeds.




**01.** Former tug The Baltic had a Raspberry Pibased makeover in 2021 to become a fully automated pleasure craft

**02.** Sfera Labs' Ulde Arcidiaco has now upgraded the 63-year-old Norwegian tug to take advantage of both Raspberry Pi 5 and Compute Module 5

> *IPMS-class architecture would cost a fortune to implement the traditional way, but made sense using Sfera's Raspberry Pi-based PLCs and the firm's own supervision software*

The first phase of the refit saw Ulde install Strato Pi Compute Module Duo and Raspberry Pi Compute Module 3+ on The Baltic. These were replaced in 2022 with two Strato Pi UPS (uninterruptible power supply) modules with 4GB Raspberry Pi 4B. "It was a plug and play replacement," he notes. "The combination of a more powerful CPU and more memory resulted in a 2× to 4× increase in performance, buying us ample margin to address the new needs."



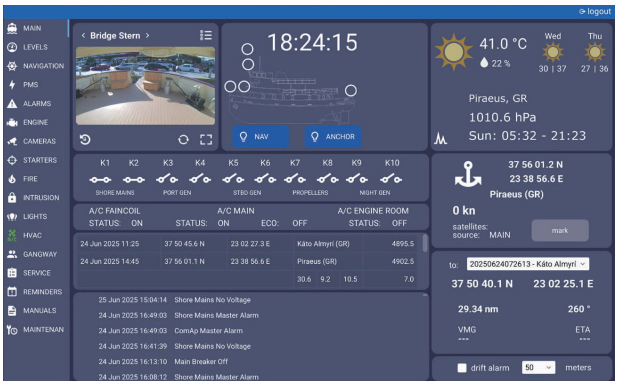
The bridge is very high-tech now, thanks to the Raspberry Pi-powered upgrades

Nonetheless, by this year, Ulde saw the sense in upgrading. “When sailing, the amount of data that has to be processed and stored in real-time is significant,” he observes, and he knew from the outset that performance updates would be needed over the years, especially with new onboard systems and services to support. Two fully redundant Strato Pi Max XL units with Compute Module 5 and 8GB RAM plus a 480GB SSD now function as the core IPMS processors. Performance-wise, The Baltic’s upgraded setup is now around 15× more powerful than the original CM Duo setup Ulde first installed.

## Time and tide

Having been born just a couple of years after The Baltic first set sail, Ulde has been an electronics and computer enthusiast since his teens and says he has seen “a meaningful chunk of the computer evolution”. The founder and co-CEO of industrial computer specialist Sfera Labs, Ulde says Raspberry Pi is one of the most important developments that happened throughout this journey. “Raspberry Pi opened a whole range of new opportunities, similar to what Linux did on the software side.

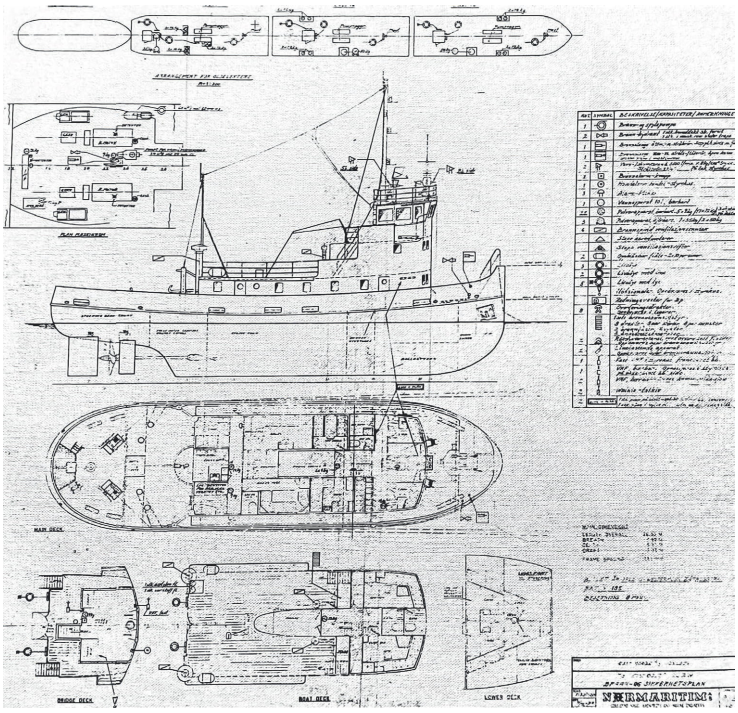
“Just as it was inconceivable to install a computer on a yacht in 1962, so it was unthinkable until a few years ago for a relatively small company such as Sfera Labs to be creating products and software capable of implementing a full IPMS on a complex vessel.”




Sfera’s bespoke IPMS interface controls almost every aspect of the ship

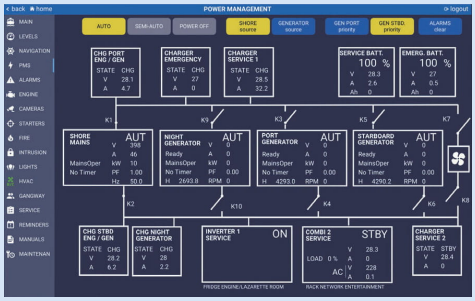
## Quick FACTS

- Navigation involves significant amounts of data processing and storage
- The Baltic has more than 4500 quickly changing data points
- Sfera’s hardware also has to cope with video from ten onboard cameras
- Upgrading to Strato Pi Max XL provided a 5× performance jump
- The upgraded CM5 system now shows a CPU load of just 20%



Original drawings showing The Baltic back when it was a lowly tug-boat

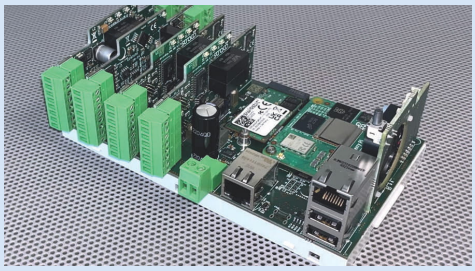
## Captain’s log




1. Sfera Labs’ take on IPMS (Integrated Platform Management System) provides detailed insights into the power being generated and distributed on board.



2. The Baltic's 2025 upgrade saw the Strato Max and Iono hardware that governs the propulsion, power generation, temperature, and automation updated to Raspberry Pi 5 and Compute Module 4: **rpimag.co/baltictug**.



3. Upgrading the IPMS to more powerful Compute Module 5-based Strato Pi Max XL units significantly improved its efficiency and failover support.

*OceanofPDF.com*

# Electronic Drum Business Card

## This is a project that is certainly worth making a noise about. By **David Crookes**



**Maker**

**Sergey Antonovich**

Sergey is an embedded systems engineer specialising in real-time system software and sensor integration for self-driving cars and delivery robots.

**rpimag.co/drumcard**

**Want to drum your name into someone's memory at a network event?** Then Sergey Antonovich has you covered. The embedded systems engineer has reinvented the age-old business card by turning it into a playable electronic drum kit – and we'd hazard a guess that this one won't end up languishing forgotten in someone's pocket.

Sergey got the idea for the project some months ago. He enjoys building digital musical instruments including ultra-portable all-in-one digital accordions and he was inspired by business cards which fit an entire Linux system on a tiny PCB. Producing a business card which could be played felt like a natural fit. "It does three jobs at once," he says. "It instantly grabs attention when you hand it over, it communicates exactly what I do, and it invites people to learn by reproducing the design."



The cards are being prepared for hand-soldering. They are normally created in batches of ten

> *It instantly grabs attention when you hand it over*

This led him to create a card that incorporated a F1C100s system-onchip running Linux as well as a TTP229 capacitive touch sensor. Recipients simply needed to power it up and listen via headphones connected to a 3.5mm TRS audio jack while tapping on an image of a drum kit printed on the card. “Touchpads were hidden under the silkscreen art so you could tap drums right on the print,” Sergey says. “It worked, looked great, and felt like a real instrument.”

## Digital Drumming

Yet Sergey wasn’t entirely happy because the project proved costly and complex. “A Linux image/toolchain is doable, but heavy for beginners or classrooms,” he notes. “And even with trimming, a boot time of a few seconds was noticeable. I wanted instant-on.”




**01.** It’s a fully functional instrument with touchpads hidden under the colour UV silkscreen

**02.** Sergey designed it to be fun and intuitive: tap the printed cymbal to hear a crash




The RP2040 was soldered manually onto the PCB with the use of a hot plate and hot air

To resolve these issues, he turned to the Raspberry Pi RP2040 microcontroller. “It’s inexpensive and it uses external QSPI flash which is large enough for multiple stereo drum samples,” he says. “It can also be programmed using CircuitPython and this takes you from idea-to-music in an evening. There’s no heavy toolchain.”

Indeed, as Sergey points out, he really did have a build ready within hours. “I started from a CircuitPython build for a 16MB RP2040 board, prepared a tiny drum sample set, and wrote a short script that scans pads and plays stereo WAVs via ‘audiomixer’ and ‘audiopwmio’.

## Feel the beat

The result has been a fresh RP2040 + CircuitPython version optimised for instant power-on via USB-C that is ideal for beginners while also being cost-effective and quick to create. The touchpads are read directly by RP2040, so there is no need for an external touch integrated circuit, and 16MB of external flash memory is more than sufficient for the code and sound samples.

Thanks to CircuitPython's touchio module, RP2040 senses the pad capacitance directly. "A finger increases the capacitance, which leads to changes in the charge/discharge time that results in a reliable 'touched' flag," Sergey explains. An LED gives instant visual feedback and the sound is output via two of RP2040's PWM channels.

Sergey now wants other people to make the project their own, which is why he has made it open source. But even though he has printed his own details on the reverse, including a QR code pointing to his LinkedIn profile, he says its use goes beyond sharing contacts.

"It's a memorable handout that grabs attention the moment you tap it," he says. "But the goal isn't just to hand someone a cool card – it's to give you a tiny, hackable instrument you can learn from and extend."

## Quick FACTS

- Each card costs around $6 to make with DIY assembly
- Tap the printed drums on the card to hear the sounds
- Sound samples are stored in the memory
- It feels instant – latency isn't perceptible in play
- The audio is playable via plug-in headphones

OceanofPDF.com

# Flokk herd management

## Trade tariffs and traceability are two good spurs to improving Canadian cattle tracking using Raspberry Pi handsets, learns **Rosie Hattersley**



**Maker**

**Mark Olson**

Mark is an enterprise IT veteran and founded Flokk to bring digital record-keeping to the Canadian ranches where he grew up.

**flokk.ca**

**Keeping tabs on cattle ranging over enormous ranches in rural Canada is a serious challenge.** “Ranching operates outdoors, in all weather, in remote locations where there is no power or data connectivity,” explains Flokk founder and CEO Mark Olson. Record-keeping has traditionally been done using pocket notebooks, which of course are easily lost, damaged or trampled underfoot while the rancher struggles to check on each animal’s health and well-being. It’s the sort of task that hapless TV vets might have nightmares about. Nonetheless, detailed and fully up-to-date information

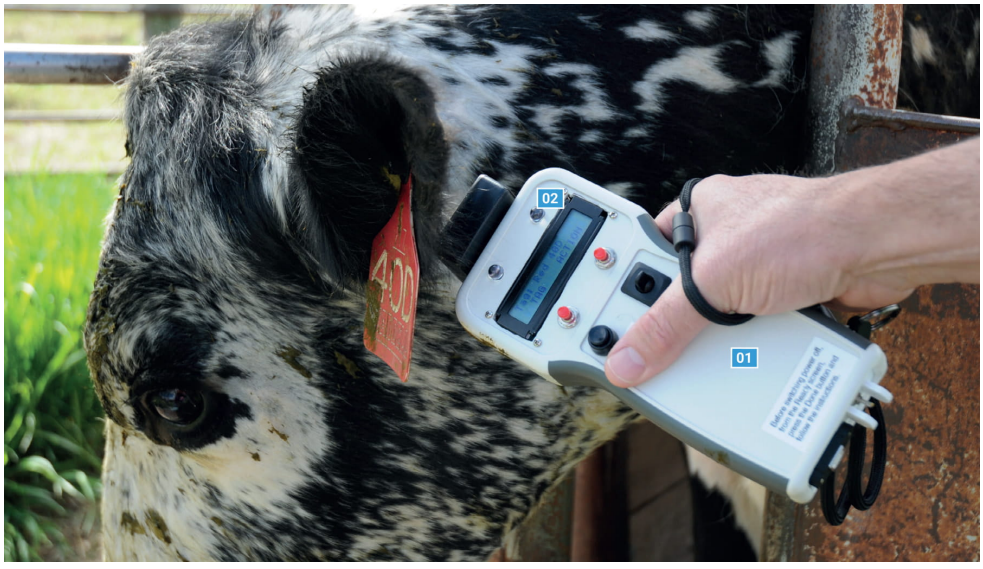
about each cow is mandatory in the highly regulated world of animal husbandry.

Computer scientist and farmer Mark Olson applied his knowledge of data collection to address the issue, coming up with a ruggedised scanner that has Raspberry Pi Zero W at its heart and which automates much of the record collection process.

## Rural reach

Mark grew up on a farm in Alberta, Canada and, after briefly exploring the computerisation of agriculture when he graduated in the mid-1980s, went on to work in enterprise IT and management. Open-source software has always interested Mark, not least because his early forays into building home servers were of the DIY, fish-the-hardware-outof-the-dumpster variety. Linux was the only platform he could afford and that would run on the hardware he'd sourced. An interest in Raspberry Pi is therefore no surprise. It's "a logical extension of my interest and aptitudes in open source," says Mark, who uses Debian Linux for both Flokk's website and online services.

This contrasts with "digital solutions for ranching [that] currently attempt to use hardware built for offices, not ranches. Consumer hardware is expensive and delicate, and unsuitable for the harsh environment of ranches."



01. Flokk's ruggedised handheld scanner records crucial animal data

02. Details are saved to a Raspberry Pi Zero W inside and sent to a central server via Starlink satellite broadband

Traceability is not only desirable; it's legally mandated. All Canadian cattle are fitted with a tag with an RFID code so that if an issue arises with the animal, it can be traced back and any other animals that might have been impacted by the issue located and investigated.

Flokk digitises data collection and record keeping for cow/calf ranching, a $5 billion industry in Canada and a trillion dollar industry globally, moving it from the small paper pocket notebooks ranchers use today to a digital solution.

Mark's LinkedIn article about the importance of such record-keeping to the Canadian agricultural economy (**rpimag.co/canadafood**) offers some important insights. He talks of the clear need for beef traceability and for Canada to secure its future food security via agricultural exports, keeping data (saleability, individual animal traceability, pregnancy, and other health stats) ondevice and freeing up the rancher's time and providing autonomy – not just laboriously filling out paperwork but capturing it via in-field scans and having it immediately be digitised.

> *His early forays into building home servers were of the DIY, fishedthehardware-out-of-the-dumpster variety*

He points out that Canada spends around 40% of its agricultural services budget on inspection and control and says this could be massively reduced using technology to tag and monitor agricultural assets. Starlink provides "a ready answer for rural broadband and wireless connectivity" and he rates it above 5G connectivity, "which is both far more expensive to install and requires time to roll out along with ongoing maintenance".




RFID tags are read by the Flokk handheld scanner and details uploaded to a central database

No platform other than Debian, the flavour of Linux underlying Raspberry Pi OS, lets you take the same software, and skill set, and apply it all the way from battery-powered handheld, through SBC desktop, all the way through to multinode server.

## Hard-headed approach

Flokk chose Raspberry Pi for its affordability, high capability, and compatibility with other open-source products. “Raspberry Pi Zero W uses a full-function Broadcom system on a chip, not a microcontroller, and Linux as the OS, [so] Flokk can leverage standard open-source tools. We can add functionality in days and be confident our platform will support whatever software or functionality we will need to deliver in future.”

It was important that the cost of the handheld scanners wasn’t a barrier to ranchers adopting them. Flokk will make its money from subscription, rather than the hardware. The end of life for a Flokk handheld will most likely not be the result of wear and tear or obsolescence –it will get dropped into a pen of cattle and stomped on. Flokk has to be able to rapidly, and affordably, recover a customer’s data from offsite backup and ship them a new Flokk ready to use.

Mark wrote all the code himself, devoting more than 1000 hours to both code and platform development. When they began the process, Raspbian/ Raspberry Pi OS would not run on the handheld scanner, so they used Tiny

Core Linux instead. Future iterations may use this alongside Raspberry Pi OS.



The ruggedised handset is ideal for use by ranchers on the move

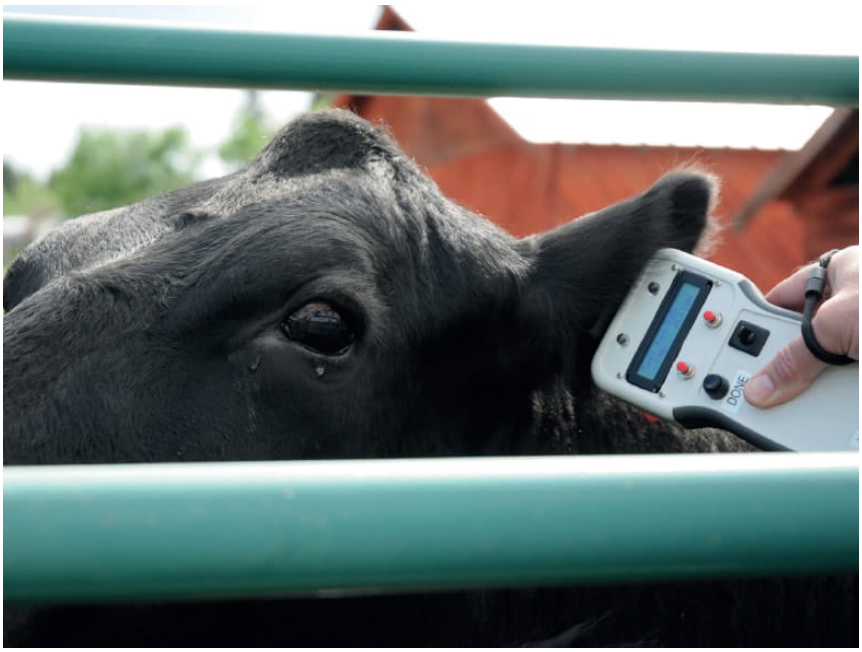
Mark says Raspberry Pi Zero's capabilities were "a perfect match; right price, power efficient, and exactly the I/O we needed". Flokk is currently preparing its next iteration of hardware, migrating to Raspberry Pi Zero 2 W, upgrading the handheld scanner's display, adding a GPS receiver and, perhaps, a camera.

For now, Mark is busy garnering investor interest in Flokk so it can be rolled out at scale across Canada's ranches. Demonstrating its real-world use is imperative. "The Flokk I proudly use for investor presentations has brown stains on it; and the source of those brown stains is exactly what you think it is."

## Quick FACTS

- Every livestock animal in Canada should have an in-ear RFID tag
- Most ranchers have no means of reading the CCIA tags, so keep paper records
- Missing and inaccurate record-keeping impacts traceability efforts
- Dedicated digital readers are expensive and have no other functions

- Flokk works with record management software, making compliance easier




Blockchain protects records from interception and exploitation once each animal’s details have been logged

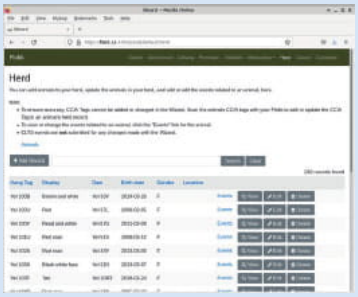
## Stand by to scan



1. Flokk runs Tiny Core Linux code on Raspberry Pi Zero W alongside an RFID reader inside a Hammond Hand-Held T ruggedised case.




2. Flokk's record management platform makes keeping detailed logs about each animal straightforward, helping ranchers comply with husbandry regulations and log medications, health, and welfare details.




3. Data is stored locally on an SD card and records can be automatically uploaded to the online management dashboard over the Starlink broadband network or when the rancher is back at base.

OceanofPDF.com

# Amstrad PPC 640 Cyberdeck

## A broken retro portable PC has been given a bright future. By **David Crookes**




**Maker**

**Mikey Damager**

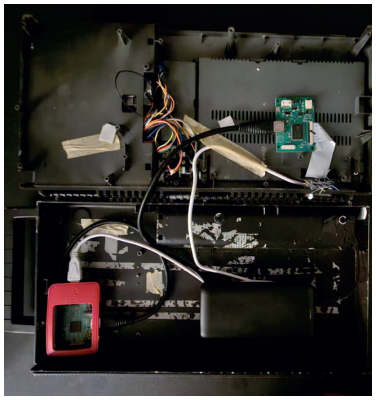
Mikey is a UK-based artist, technologist, and technowomble. He builds interactive narratives, musical interfaces and AV installations from code and old bits of hardware he finds lying around.

**rpimag.co/ppc640deck**


**When faced with a broken Amstrad PPC 640, Mikey Damager had two choices: to return the machine to its former glory or tear it apart and rebuild it using modern parts.** He decided to do the latter, turning what was Amstrad's first portable IBM PC compatible computer, released in 1987, into a cool-looking cyberdeck powered by Raspberry Pi 4. It produced a machine capable of running an interactive fiction project for Mikey's master's degree.

"I wanted to explore AI and machine learning to see if I could incorporate some of the tools into a creative workflow in a way that felt somewhat critical and not too detached," he says. "I ended up making

something that uses LLMs to explore what it's like to be existentially brutalised by an evil corporation which has hidden access to pseudo-sentient technology. It's supposed to be somewhat tongue-in-cheek and satirical."



The original innards have been replaced, but Mikey has retained the motherboard in case someone wants it

## Key to success

Mikey had considered repairing the original machine, yet doing so would have entailed a huge amount of work. "The screen was completely smashed and the case was pretty dinged-up. A few bits of plastic had also snapped off."

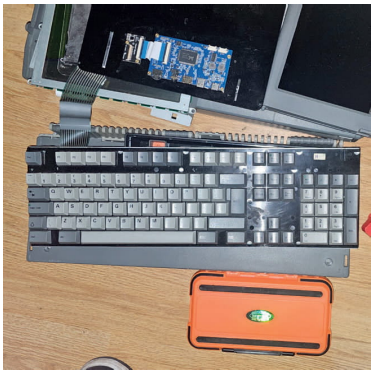
Deciding an upgrade was preferable, he opened the case, and detached the screen and keyboard. "The chassis is basically a plastic suitcase with a screen and a keyboard attached," Mikey says. "Once inside with a screwdriver, the motherboard practically leapt into my arms. I was left with a big empty box that I could fill up with new gadgets."

The screen was replaced with an eight-inch 4:3 LCD panel connected to an HDMI driver board. Mikey used screen repair tape to fix this panel to a sheet of 2mm acrylic for protection and he connected the display to Raspberry Pi before wiring the PPC 640's original LEDs and switches to the new hardware, allowing the system to be easily powered up.




**01.** The project is faithful to Amstrad’s original cool-looking case design

**02.** All of the components including the battery fit neatly inside



Mikey says the PPC 640’s plastic had become brittle over time, so care was needed when working on the case

## Quick FACTS

- The project uses a case from an Amstrad PPC 640
- There’s plenty of space, offering lots of design choices
- Finding suitably sized screens and keyboards is easy

- Raspberry Pi is being used in kiosk mode
- The computer has been coloured using car bumper spray paint

> *The screen was completely smashed and the case was pretty dinged-up*

Replacing the keyboard proved easy too. “One of the best things about the PPC is that it has a full-size keyboard, which means that if you just remove the little plastic tangs where the Fn key should be, you can grab pretty much any random full-size mechanical keyboard, de-glove it, and there’s a good chance that it will fit almost perfectly.”

## Black is the new grey

The floppy disk drive was also retained even though it was disconnected. “I wired the write-protect switch from the floppy drive to an Arduino to turn on a small screen when a disk is inserted, but it doesn’t read any data,” Mikey says.

The biggest challenge was the development of the front end. “I needed what’s running on screen to look and feel suitably cyberdecky,” he explains. “The piece is built around a React app that’s styled to look like an OS. Raspberry Pi is running FullPageOS, so it’s just a kiosk that boots straight in to a web page over Wi-Fi, with the back end running either from my laptop or in the cloud.”

It means Raspberry Pi is just handling the display and user input, but it ensures the cyberdeck does the intended job. As a finishing touch, Mikey sprayed the grey computer black and added colour to some of the keys, but he’d like to go further. “There are a few cosmetic improvements I’d like to make, such as new badges and branding,” he says. “I’m also planning a better power solution because currently it’s running from power banks that I’ve hidden inside and I’d like something a bit more elegant.”

OceanofPDF.com

# Maker Guides

**Build the ultimate media centre**
The perfect box for watching your faves

**Play Steam games using x86-64 emulation**
Modern games on Raspberry Pi

**The A to Z of Raspberry Pi**
26 essential things to know

**Create a Sense HAT interactive pixel pet**
Program your own pocket-sized monster

**Pico drum machine**
Percussion at your fingertips

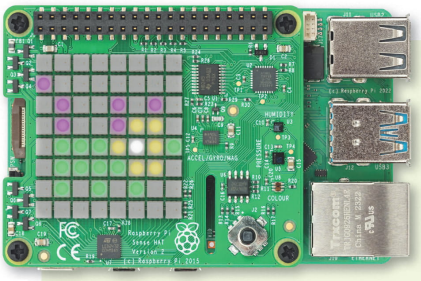
**Home automation made easy**
Make a smarter home with Raspberry Pi

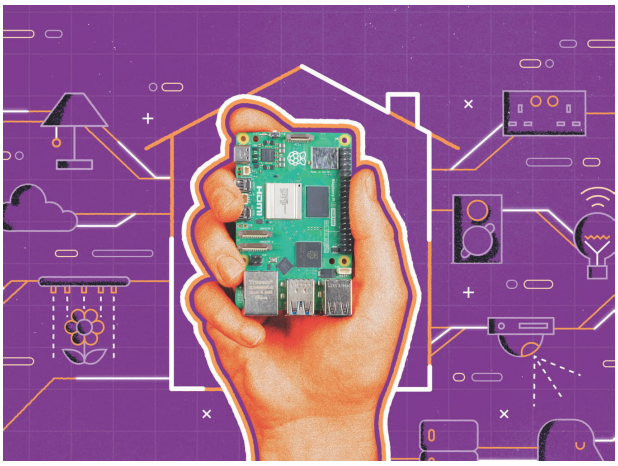
**Hack a robot arm**
Powering up a kid’s toy

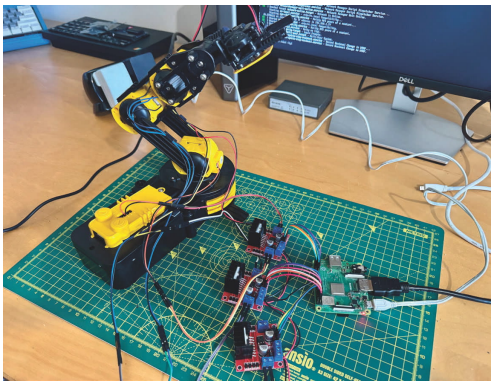
**Extreme Raspberry Pi**
At the very limits of computing




Raspberry Pi Documentation
PiCADE







[OceanofPDF.com](OceanofPDF.com)

# Build the ultimate Media Centre

## Whether it’s rugged server cases or beautiful set-top boxes, we’ve got you covered. By **PJ Evans**

**Building a media centre is a wonderful home project.** You can take full control of your music, video, and television experience.

Like all computer projects, there are two elements: hardware and software. Building your perfect media centre is no different. There’s lots of choice and what’s best will depend on your use case. So, we’ve gathered up the very best in hardware and software to help inspire your media centre project.

## Argon ONE cases

Unless you’re a bit of a minimalist and laugh in the face of static electricity, you’re going to need a case with some active cooling included. Media centres can be resource-hungry things and will use techniques like hardware graphics acceleration to process video. It also matters whether you want a box tucked away that can be accessible over the web or via dedicated client, or if you’re after a case on display for use in your home theatre. Either way, Argon has you covered with the excellent ONE range (**argon40.com**). Choose from ONE V3 for a good-

looking unit with IR (Infrared) control built-in or the more industrial ONE V5 for dual NVMe storage and built-in DAC (but no IR). See also the excellent KKSB Case which can contain a DAC Pro HAT (**rpimag.co/kksb**).



The Argon ONE V5 is the case of choice for a display-less media server

# Argon IR Remote

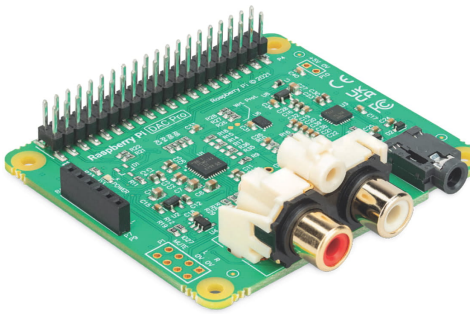
If you're keen on the ONE V3 case with its sleek lines and small footprint, you might want to take advantage of its onboard infrared receiver. Argon sells an optional remote control (**rpimag.co/argonremote**) with the drivers to help integrate with your choice of media centre software. It's inexpensive and very Apple-like in design. Pair it with the ONE V3 and you can design and build your own version of Apple TV or Fire TV Stick and enjoy it from the comfort of your sofa.



Add this sleek remote to the IR-capable Argon ONE V3

# Digital-to-analogue converters

An essential investment if you're looking at an audio-first project and are not using HDMI for audio, a digital-toanalogue converter (DAC) provides a much better audio experience than the capabilities of any Raspberry Pi. They come as HATs and provide a variety of outputs, from a headphone jack to RCA line-out connectors, and some even have amplification onboard to connect directly to speakers. Combine one of them with a touchscreen and you have a standalone media centre. Costs vary by capability, but the official DACs are a great place to start (**rpimag.co/dacpro**).



The official DAC Pro HAT adds superb audio to your project

> Media centres can be resource-hungry things and will use techniques like hardware graphics acceleration to process video

## SmartiPi Touch 2 and Touch Display

If you miss jukeboxes or fancy a more interactive experience, how about creating a touchscreen-controlled media centre? We love the SmartiPi Touch 2 enclosure for the official Raspberry Pi Touch Display (v1). It can be wall-mounted or stands freely and the larger rear enclosure means you can fit a decent DAC inside for improved sound. When paired with a dedicated music OS such as moOde, you can control everything without a dedicated keyboard. Perfect for communal spaces (**rpimag.co/smartipi2**).



Add this SmartiPi case to a Raspberry Pi and touchscreen for your own jukebox

# Raspberry Pi Zero

What's that? A Raspberry Pi Zero as a media centre? Well, not exactly. If whole-house audio is your thing, the humble Zero is a great solution for creating casting points around the house. Simply add a DAC HAT, a case (dedicated DAC cases are available), and then install Shairport Sync (**rpimag.co/shairport**). Connect to any active speaker system and you will then see that system advertised as an AirPlay destination. Small, cheap, and energy-efficient.



A Raspberry Pi Zero W or 2 W with a DAC makes whole-house audio cheap and easy

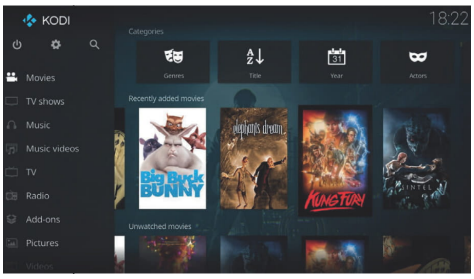
# Essential media centre software

# You've got all the physical goodies, now what about the software? PJ Evans covers your options

**It's all very well having a fancy case, remote control, and AirPlay setups all around the house, but how are you going to make it all work together?** There are many options for accessing and playing back your media, so it's worth taking time to look at the capabilities of the main contenders. There are three main use cases: local media playback (like an Apple TV), client playback (like Spotify), and audiophile-grade options. We'll look at our favourites from these categories.

## LibreELEC & Kodi

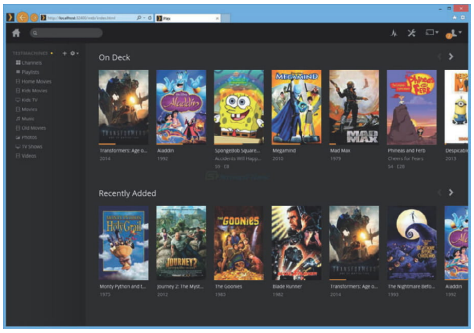
Kodi has been a long-term player in our first category, for local media. The software is mature and its plug-in architecture means that literally hundreds of add-ons can be installed to increase its functionality. It's a perfect solution for building your own Fire TV Stick or Apple TV rival, with a highly customisable interface and support for HD or UHD (4K) films, lossless music, and photo libraries. For Raspberry Pi, LibreELEC (**libreelec.tv**) is a dedicated operating system for Kodi which ensures smooth installation and no wasted resources. You can create a LibreELEC image directly from Raspberry Pi Imager (**rpimag.co/software**).




Kodi with LibreELEC is your fastest route to a full-featured living room setup

## Plex

Want to get to your media anywhere in the world with offline access? How about a dedicated music app or onthe-fly video transcoding? Best of all, would you like to do this without having to fiddle about with routers and DNS (Domain Name System)? Plex is a fantastic choice for a media server (**plex.tv**). It's not designed to be accessed directly, but supports web browsers, mobile apps, and there are clients for Apple TV and similar. If you want access outside the home and to download media, you'll need a paid-for subscription, but for hassle-free media management it's hard to beat. Don't mind routers and DNS? Have a look at our Jellyfin tutorial overleaf.




Plex makes accessing your content from anywhere simple

## Shairport Sync

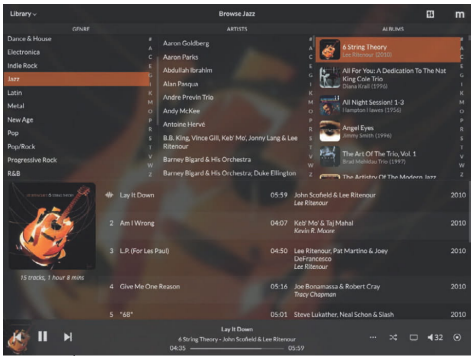
This 'headerless' server (it has no user interface) is squarely aimed at whole-house audio projects. It's a small service that, once installed, converts the Raspberry Pi into an AirPlay 2 target, supporting a wide range of casting devices, including iPhones. Better yet, it supports the multiroom sync protocol. Perfect for creating cheap audio 'nodes' using a Raspberry Pi Zero, DAC HAT, and speakers. Best of all, it's open source so completely free to use (**rpimag.co/shairport**).



Build cheap AirPlay receivers with Shairport Sync

## moOde

If you want to take your music listening seriously, Raspberry Pi has become a serious contender in the audiophile world. Companies such as Bang & Olufsen have produced DAC (digital-toanalogue converter) HATs using the finest components, turning the Raspberry Pi into a music-lover's delight. Even the lower end of the market produces some very good audio output. To get the best control, there are a few dedicated operating systems available such as moOde (**moodeaudio.org**), aimed at producing the best playback experience. It works great with a touchscreen or you can control it from a browser. See also the popular Volumio (**volumio.com**).




moOde is one of the best choices for serious audio folk

> Raspberry Pi has become a serious contender in the

# Project Recipes

## Set-top Box

- Raspberry Pi 5
- Argon ONE V3
- Recalbox & Kodi

## Audiophile’s Dream

- Raspberry Pi 5
- KKSB Case
- Raspberry Pi DAC Pro HAT
- moOde

## Whole-House Audio

- Raspberry Pi Zero 2 W
- HiFiBerry DAC+ Zero
- Active speakers
- Shairport Sync




Want to add gaming? You can have Kodi and Recalbox on the same machine

### Recalbox

Plot twist. When is a media centre not a media centre? When it's a retro gaming station! Recalbox (**recalbox.com**) is our favourite platform for playing retro games and among its many plug-ins is Kodi, the play-all media centre. So why not have your cake and eat it? Get a case with forward-facing USB (such as the Argon ONE V5) and you can have the nostalgic wired controller experience and have a full-blown media centre at your disposal as well.

# Build a home media centre

## Access your music, movies, and photos anywhere in your home or anywhere in the world, subscription free



**Maker**

**PJ Evans**
PJ is a writer, software engineer, and one of those people who drones on about a director's 'vision', constantly pausing films to discuss the scene composition.
**mastodon.social/@mrpjevans**



A small box that frees your media

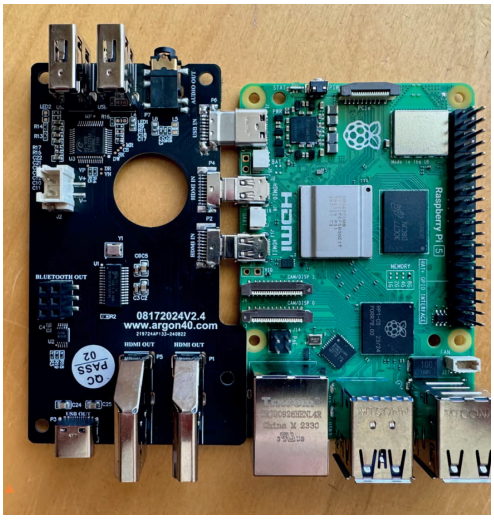
**Raspberry Pi as a media centre has been a classic project since day one.** As new models have been released, the capabilities of this minuscule device have increased exponentially. From a humble audio player to a fully capable video streamer, we are now in the era of the Raspberry Pi as a serious 4K media device. The options for setting up such a system are many, and things can get complicated. So here's a reference guide for setting up a great media system that is subscription-free and, with a bit of configuration, can be accessed anywhere in the world.

# A perfect pairing

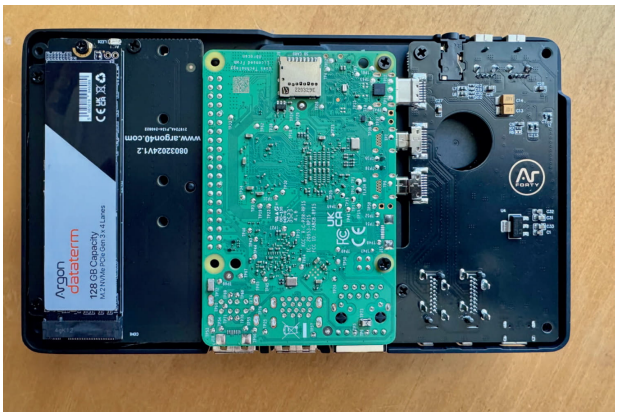
We're going to base our media centre around two great products: Argon's latest case, the Argon ONE V5, and for our software, Jellyfin. Argon has created a bit of a masterpiece with its new product; in fact, in our review we gave it a straight 10 out of 10. This is the perfect case for our project as it not only has built-in support for M.2 NVMe solid-state drives, but also

comes with a DAC as standard. So if you want to use the device as not only a server, but a client as well, you're all set. Our media centre will be equally at home as both a server of images, audio, and video, as well as a 'set-top box' that sits in your living room.

Using an M.2 SSD means speed; and that can be important, especially with video. They are also more reliable than SD cards. We'll start by imaging the standard version of Raspberry Pi OS (64-bit) to our M.2 drive. We're using a 128GB NVMe drive from Argon, but you can use any size you like. To image the M.2 SSD, we recommend getting an inexpensive M.2 to USB adapter. You can then connect the drive just like a USB stick and use Raspberry Pi Imager. Select the recommended operating system and don't forget to configure the hostname, username, and password. You can configure Wi-Fi as well if you wish, but we strongly recommend a wired connection for judder-free viewing.




The Argon ONE V5 case takes longer to assemble than most, but it's worth it




The Argon ONE V5 comes with a board to provide DAC, USB, and full-size HDMI

## Grab the screwdriver

Once your SSD is ready, proceed to build your case. The Argon ONE V5 is a solid piece of engineering and will take you a little time to assemble everything. Luckily, there are good instructions included. Be particularly careful with the wiring for the NVMe connector and the built-in fan. Assembly is a matter of connecting the case's board to the Raspberry Pi, screwing them both to the chassis, adding the M.2 SSD, and then connecting the wiring. Once done, the bottom and top covers can be added. Optionally you can also install the small OLED screen accessory, which can provide rolling status information such as CPU usage, temperature, and IP addresses.

## Typing time

You should now be able to hook up your media centre to a monitor or TV using the full-size HDMI connectors. You also need to decide at this point how to connect to the Raspberry Pi for further configuration. Either connect a keyboard and mouse, or use Secure Shell (SSH) to access Raspberry Pi, as everything can be done from a terminal screen. The system should reboot twice as it sets up and then drop you into the familiar Raspberry Pi desktop. As ever, the first job is to make sure everything is up to date. Open a terminal window (or SSH in) and run:

```
$ sudo apt -y update && sudo apt -y upgrade
```

This will ensure you have the latest software and security patches installed.

Before we install Jellyfin, Argon provides software to help manage the various features of the case, such as the power button and the display. Open a terminal window and run the following command:

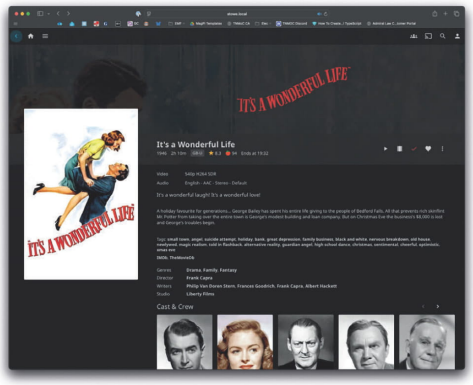
```
$ curl https://download.argon40.com/argon1v5.sh
| bash
```




The optional OLED screen can help you keep a check on your server

This will set up everything for you. If you have the screen installed, it should immediately start showing information. Now you can use the front button to safely switch your computer on and off. If all is well, we can go ahead and install Jellyfin itself. Run the following:

```
$ curl https://repo.jellyfin.org/install-debuntu.sh | sudo
bash
```




Jellyfin automatically populates metadata

This runs a remote script that makes installing Jellyfin a breeze. It'll take care of everything for you. At the end, you'll see confirmation that the Jellyfin service is running and you're ready to configure everything.

The next thing to do is think about is how you want to organise your media. To make things as easy as possible, we recommend creating a root directory as follows:

```
$ sudo mkdir /Jellyfin
```

...and creating directories underneath it such as:

```
$ sudo mkdir /Jellyfin/Music
$ sudo mkdir /Jellyfin/Movies
$ sudo mkdir /Jellyfin/TV
```

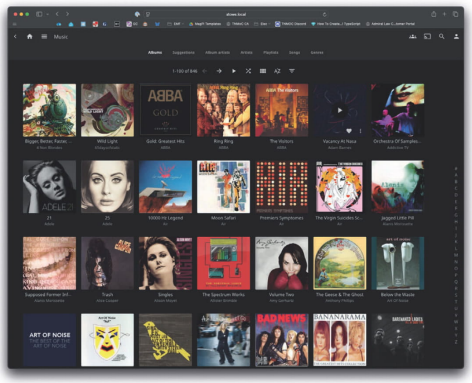
You can choose whatever you want. To get files onto the server, copy them into your home directory and then move them into the Jellyfin directory like this:

```
$ sudo mv /home/pi/Music /Jellyfin/Music
```

Start with a small amount of media as you can add more at any time.



Watch classic movies anywhere, including public domain material from archive.org



Jellyfin provides a beautiful interface for your favourite media

# Scan and go

Jellyfin has installed a local web server which you use to access your media. To find it, open any browser on the same network and go to **http://<yourservername>:8096/** (where <your server name= is replaced by the hostname you gave your server). You now step through the creation of the admin account and the adding of your first media library. Choose a category (such as ‘Movies’) and select the directory you created and populated in the previous step. Jellyfin will help you categorise and populate your media with metadata from various internet sources. You can always make corrections if it gets it wrong.

Once setup is complete, Jellyfin will scan your media in the background, so expect things to look not-quite-right at first, but then you’ll see album covers and film posters as it catches up. Now you have a media server! Any computer with a browser can access your media centre and enjoy the sights and sounds it provides. Better yet, you can install either the official or one

of many third-party Jellyfin clients for both iOS and Android. Just point them at your server and you have all of your media in the palm of your hand.




## Go remote

The final step is to get your media available anywhere. There are different ways of doing this. The simplest is to get either a static IP address (or use a dynamic IP service such as DuckDNS) and then map a port route from your router to your media server. You can then use your new address (as provided by DuckDNS) as your server access point. Your server is now available anywhere. This isn't the most secure approach, though. A better method is to set up Wireguard VPN as a secure way of accessing your home network from afar and then connecting over its encrypted connection. Find out how to do this in issue 145 (**rpimag.co/145**).

*OceanofPDF.com*

# Play Steam games using x86-64 emulation

## Using Box64, we can play modern Intel/AMD PC games from Steam, GOG, Itch, and more



**Maker**

**KG Orphanides**KG is a game developer, writer, and musician with a special interest in retro gaming. They have worked for Kitsune Games, mentioned in this feature.
**rpimag.co/owlbear**

**Raspberry Pi is built on ARM64 architecture, which isn’t compatible with most games, although some have been specifically ported, and retro games can be played via ARM64-compatible rebuilds of their game engines or emulation tools such as DOSBox.**

Although GOG is usually our first port of call, this month we’ve mostly been playing games on Steam, the most popular and Linux-friendly gaming storefront. We’re getting our software from the Pi-Apps repository, which gives you custom builds of popular applications for ARM64 hardware.

> *Manage your expectations when it comes to performance*

Raspberry Pi 5, 500, and 500+, in particular, are powerful enough for everyday use as a desktop PC, but running modern Linux or Windows games can be a power-intensive prospect. We're dealing with multiple compatibility layers in some cases here. Steam is running on top of Box86's emulation of AMD64 CPU architecture; in turn, many games on Steam will use Proton – which allows Windows games to be played on Linux. Despite the performance hit this produces, we've had luck running a number of Steam games.

Manage your expectations when it comes to performance and be prepared to engage in some trial and error. Even if you're only installing the tiniest of games, you'll need to ensure that you have at least 10GB of free disk space to install and run them. Use a high-speed microSD card for your hard disk (or an NVMe HAT) if at all possible if you're not using a 500+.




We were able to get in a round of Vampire Survivors without too much slow-down, although menus lagged

### Change your page size

Unlike previous generations of Raspberry Pi, Raspberry Pi 5/500/500+ default kernels both ship with a 16kB page size set by default – this defines minimum unit size for memory management. A larger page size improves memory access speeds. Unfortunately, none of the emulators of Intel and AMD x86/x64 hardware support 16kB page sizes.

Check your current page size, and then open the config file to manually change it if needed.

```
$ getconf PAGESIZE
$ sudo nano /boot/firmware/config.txt
```

Under `[all]`, add `kernel=kernel8.img`

Press **CTRL+X**, then **Y** to save the file and quit, then reboot your Raspberry Pi.

## The wisdom of crowds

Running Steam games on Raspberry Pi is, for now, a minority interest, but while researching this article, we found a number of users who'd been kind enough to list the titles they'd got working. We didn't have success with all of these, but they'll certainly give you a jumping-off point if you cross-reference them with your Steam library.

Steam games which work on Raspberry Pi 5: **rpimag.co/steampi**

A list of games which run on a Raspberry Pi 5: **rpimag.co/gamepi**

Gaming via x86 Steam with Box86: **rpimag.co/gamebox64**




We used Wine – running via Box64 – to install and run the Windows version of Sunless Sea from GOG

## Run Steam on Raspberry Pi 5/500/500+

The easiest way to get started emulating x86 and AMD64 architectures is to the use Pi-Apps installer (**pi-apps.io**), which provides a repository of popular applications, customised for Raspberry Pi's ARM64 architecture. It will even offer to select the 4kB page size kernel for you when you install an app that requires it.

First, install Pi-Apps. Ophttps://download.argon40.comen a terminal and type:

```
$ wget -qO- https://raw.githubusercontent.com/Botspot/pi-apps/master/install | bash
```

When it's finished installing, it'll point you at further instructions if you need them. You can now close the terminal window. You'll find a Pi-Apps shortcut on your desktop or in the Accessories section of the main menu. Double-click on the desktop icon to launch it, and choose to run it in the terminal – this means you'll be able to see what it's doing in real time. Make sure you have a decent internet connection, as some of the files we're downloading are large and will fail to download if you've insufficient bandwidth.

To install Steam, select the Games menu and then select Steam from the menu. Read the details of how the application works, then click Install. This

will also install Box64, which we can use to run other x86 and x64 applications.

You should also install More RAM from Pi-Apps. This disables the swap file and instead uses on-the-fly compression to squeeze more space out of Raspberry Pi's physical memory.

After rebooting Raspberry Pi, you'll find a shortcut to Steam in the menu under Games. Run the program and it'll download and update the Steam executable. Log into your account using either Steam's 2FA app or your username, password, and the 2FA code that'll be emailed to you. Steam will open on your Friends chat window and game library.

## What Steam games can you run on a Raspberry Pi?

Small games are a good start – roguelite card game hit Balatro works brilliantly, and seeing fans of Mario-inspired platformer Kitsune Tales run it on their own Raspberry Pi hardware is what inspired this feature.

Steam itself is sluggish and slow to respond, and games can take a couple of minutes to launch. Once launched, however, those that'll actually deign to run at all often do so surprisingly well.

You'll find that a number of 2D indie games and older titles work. We were able to run Into The Breach, Vampire Survivors (with a little lag when opening menus), and Long Live the Queen.

Unfortunately, other titles that have been reported as functional in internet forums worked less well for us. Valve's Source engine games can be run, but not in a way that we'd consider playable.




Windows versions of RPG Maker games such as I Have Low Stats But My Class Is “Leader”, So I Recruited Everyone I Know To Fight The Dark Lord, which we downloaded from Itch, also worked well via Wine

Portal launched, but was functionally a sideshow. Although some users on Steam’s forums say they got 3D game performance up to 25–30fps at minimal settings, we were unable to reproduce this by dropping Portal’s resolution. Which isn’t necessarily to say that trial and error won’t produce better results, but we wouldn’t get our hopes up.

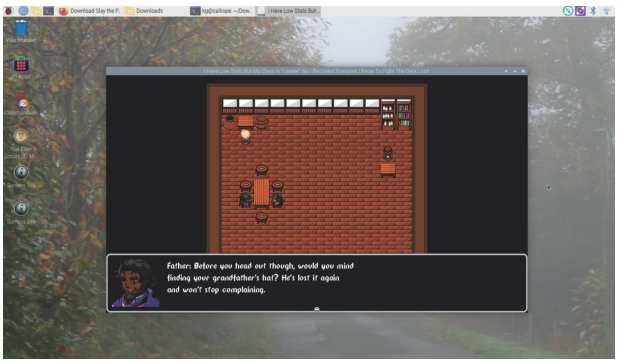
## Beyond Steam

Your Steam install can also stream Steam games installed on another, more powerful computer on your network, but bear in mind that there’s a native Raspberry Pi Steam Link client (**rpimag.co/steamlink**) that will stream your Steam games from another PC without the overhead of x86/x64 emulation.

In some cases, particularly re-releases (rather than remasters) of retro titles, you can install them via Steam, but – once they’re unpacked – you can use an ARM64 native application such as DOSBox-X or ScummVM to run them, without the need to add all that extra weight of compatibility and emulation layers.

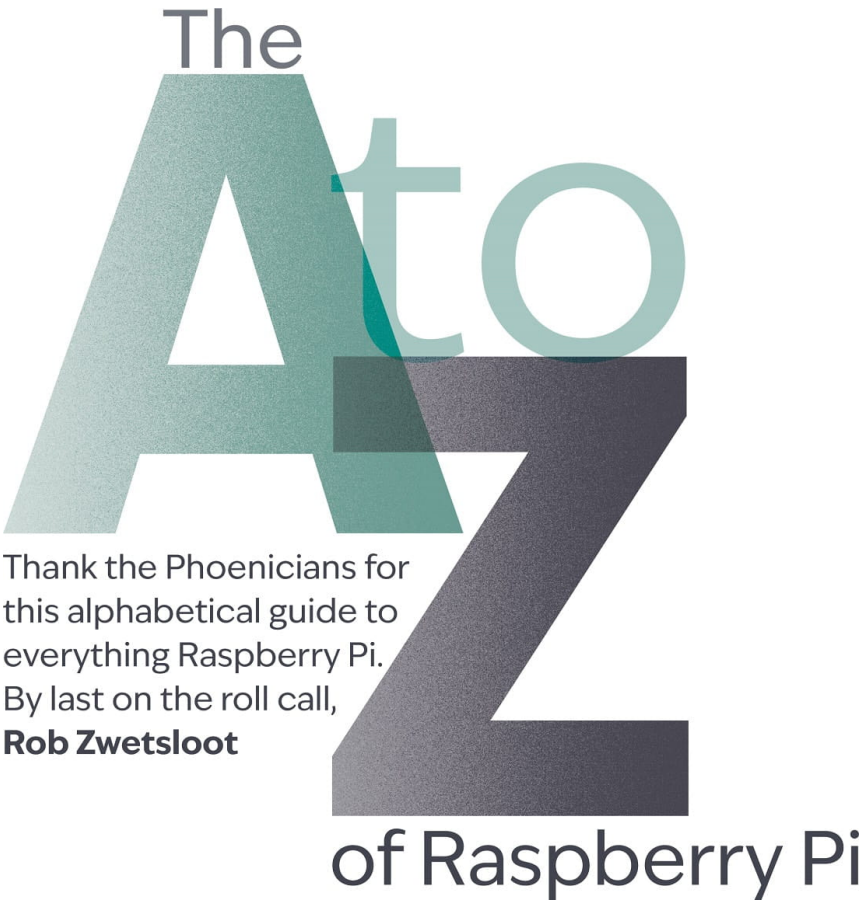
You can also use Box64 to run Wine, allowing you to install and play non-Steam games, such as GOG’s Windows installers and downloadable Windows and Linux games from Itch.io. When installed via Pi-Apps, Wine

automatically runs through Box64. If you want to use a Linux installer or other executable, you can invoke `box64` at the command line before you use `sh` to run it, but we generally had more luck with Windows versions, including Failbetter's Sunless Sea.




Playing the I Have Low Stats… RPG on Raspberry Pi

OceanofPDF.com



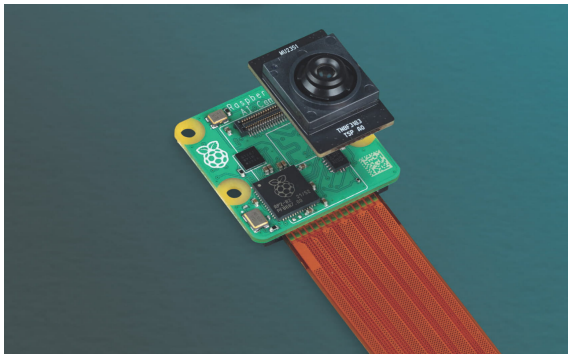

**Raspberry Pi has been around for 14 years, which means there's a lot to learn about it.** Never fear though, as here at Raspberry Pi Official Magazine we have this very definitive A to Z of everything Raspberry Pi. Or at least, 26 things Raspberry Pi. We could not find anything that began with ß.

Think you have a better suggestion for a letter? Or even one for ß? Email us at **magazine@raspberrypi.com** and we'll print the best ones in a future issue.

## AI

## Machine learning

We've covered AI extensively in issue 152 (**rpimag.co/152**), but in brief: Raspberry Pi is becoming big in practical AI applications, with TPUs, AI HATs, AI Cameras and more specifically created to help it with different tasks that can use a tiny Raspberry Pi with advanced recognition abilities.



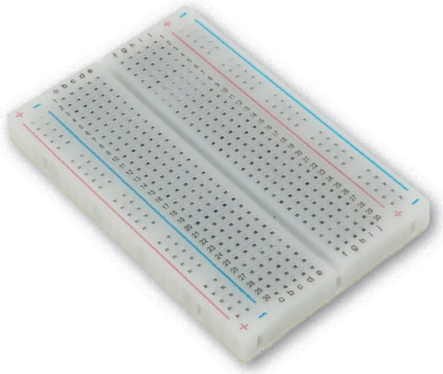
The Raspberry Pi AI Camera uses Sony’s IMX500 Intelligent Vision Sensor

> *Electronic circuits are a huge part of Raspberry Pi*

# Breadboard

## Prototype your circuits

Electronic circuits are a huge part of Raspberry Pi thanks to its GPIO ports allowing you to read sensors and activate components. Breadboards are integral to this, as they allow you to learn electronics and test circuits without any soldering, and with reusable parts.



We like these half-size boards with labelled positive and negative rails at the top and bottom

# Camera Module

## Take photos or analyse video

Raspberry Pi Camera Modules come in a variety of flavours – from your standard point and shoot, to high-quality cameras with interchangeable lenses, global shutter varieties, infrared sensors, and now the AI Camera. People use it for fun and for important vision-based AI tasks like checking production lines.

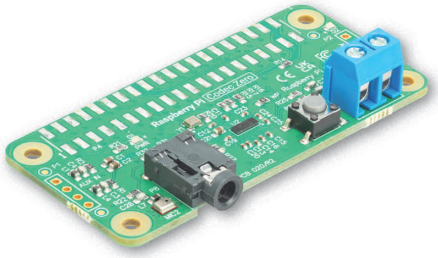


Switch out CS type lenses to get the perfect shot

# DAC

## An aural delight

Turn Raspberry Pi into a hi-fi stereo with an audio DAC (digital to analogue converter) and choose from multiple varieties that connect via RCA cables, stereo connectors, and 3.5 mm headphone jacks to suit your audio needs. Audiophiles would even choose these over the headphone jack in older Raspberry Pi models.



The Codec Zero is Raspberry Pi Zero sized and helps add great audio to the diminutive board

# File server

## Network-attached storage

A very basic but perfect use of Raspberry Pi is as a file server on your own network. Whether you're purely using it as a network hard drive using Samba, a VPN access point, Plex, local web server, or even a home automation server, its low-power consumption and small size means it's one of the best home file server solutions ever.



The Argon EON (**rpimag.co/argoneon**) is a purpose-built NAS case for Raspberry Pi



Contrary to popular belief, Eben is not Jason Statham

# Eben Upton

## The face of Raspberry Pi

Co-creator of Raspberry Pi, and co-founder of the Raspberry Pi Foundation, Eben is our boss and the public face and CEO of Raspberry Pi PLC, the part of Raspberry Pi that actually makes computers (and microcontrollers, etc.). You'll see him at the occasional event and whenever a new Raspberry Pi product is announced. His wardrobe is not filled with black turtle-necks, though.

# GPIO

## Connect to the world

General Purpose Input and Output, these 40 pins (26 on the original Raspberry Pi models) allow for people to expand the capabilities of Raspberry Pi far beyond what is available with the base board, or let you hook up LEDs or buttons so you can experiment with electronics. These pins also support advanced functions like I2C, PWM, SPI, and serial on specific pins.

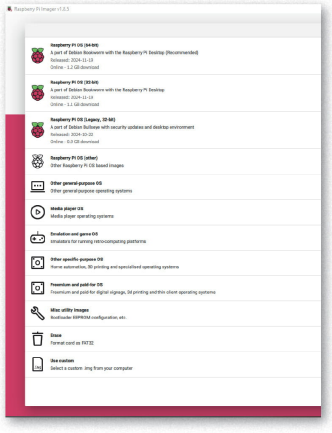


These 40 pins are a powerful part of Raspberry Pi

# Imager

## Install anything to Raspberry Pi

Imager is a multi-purpose app that deals with just about anything related to booting a Raspberry Pi. After selecting a Raspberry Pi model, you can choose from a list of recommended operating systems or install your own image to an SD card or other storage device. It can even do advanced EEPROM configuration and format an SD card.

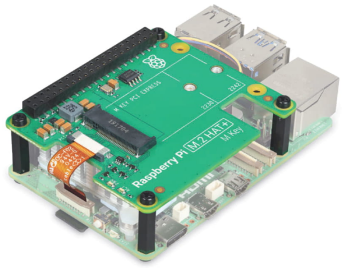


Alternative OSes are filtered by category

# HAT

## Hardware Attached on Top

Get it, because it's like putting a hat on Raspberry Pi? HAT, and HAT+, are actually standards set out by Raspberry Pi (read about them here: **rpimag.co/hatsheet**). They range from simple things like adding more USB ports, to advanced robotics and home automation extras.



The M.2 HAT+ lets you install SSD storage to Raspberry Pi

# Jams

## Worldwide community events

Raspberry jam is a delicious preserve for bread or Victoria sponges, while Raspberry Jam (uppercase J) is the punny name of community events hosted around the world. They also receive some support from Raspberry Pi too, and we list them in the magazine and online – check out each issue for upcoming events, or head to **rpimag.co/events**.

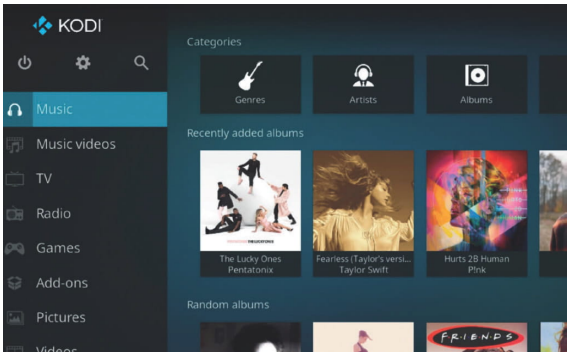
Events come in a variety of names, but Jams are the most recognised

# Kodi

## Media PC software excellent

In another life, Kodi started as a hack for the original Xbox, and is now one of the most popular pieces of HTPC (home theatre PC) software. There are several

operating systems built around it for Raspberry Pi (LibreELEC is a fave), and its lightweight and fast approach to media organising and playback make it a perfect fit.




The interface is very clean and allows for easy indexing of all your media

# Magic mirror

## House of the future

Our original top project from issue 50 (**rpimag.co/50**), as voted for by the community, this is a deceptively easy project that requires either a minimal knowledge of carpentry, or a good frame find at IKEA to fit a monitor. The software is dead easy to set up at (**magicmirror.builders**) and it looks very cool. It's also very practical.



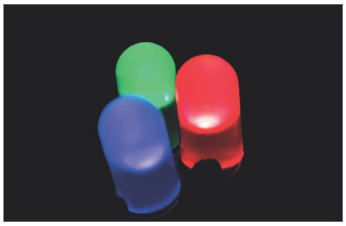
A spare Raspberry Pi, a spare monitor, some reflective glass, and a frame

> *One of the most popular pieces of HTPC software*

# LEDs

## Light up your projects

Whether you're using basic LEDs that just need some power to turn on in your circuit, or you're controlling programmable NeoPixels to create elaborate light displays during a holiday, LEDs are a core part of many Raspberry Pi hobby projects.



The humble LED is a diode, so current only flows in one direction

## Alpha Beta

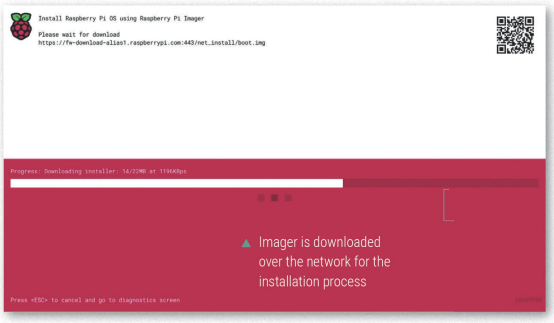



The modern alphabet of many languages can be traced back to the Phoenicians, who created the first true alphabet around 1000 BCE, which itself was descended from Egyptian hieroglyphs. The Latin alphabet to which the English alphabet is most closely related only had 23 letters, so thanks to 2000 years of progress there are three more items on this list.

# Network install

## Get started fast

Plug your brand new Raspberry Pi into your router via an Ethernet cable and you can install Raspberry Pi OS without having to prepare an SD card first. It’s pure tech magic, and even lets you choose different operating systems to install to the SD card. You’ll need a Raspberry Pi 4 or newer for this method, though.

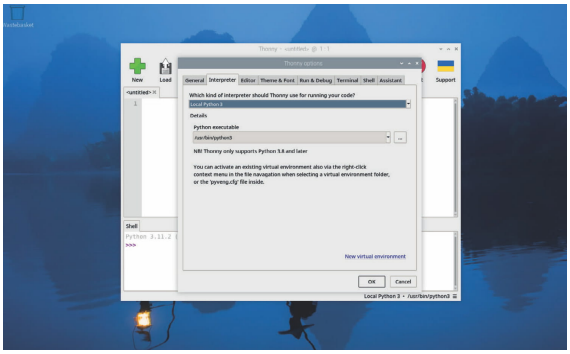



Imager is downloaded over the network for the installation process

# Operating system

## Raspberry Pi’s own OS

Based on the Linux distribution Debian, Raspberry Pi OS is custom-built for Raspberry Pi. It has many optimisations to help it run its very best on various Raspberry Pi computers, a custom graphical interface, a selection of pre-installed software, and it’s also incredibly flexible and customisable by users. It’s perfect for Raspberry Pi.

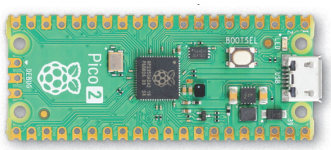


The clean design and recommended software makes it an ideal OS

# Pico

## Microcontroller by Raspberry Pi

Raspberry Pi Pico and Pico 2 are very different from a standard Raspberry Pi. First of all, they're not computers, they're programmable microcontrollers that are better suited for a single task. They draw less power, are very small, and the chip running – RP2040 or RP2350 – can be embedded in other systems too.




Tiny but mighty, Pico's GPIO pins are more versatile than a standard Raspberry Pi

# QWERTY

## Keyboards, peripherals, and more

As well as computers, you can get a Raspberry Pi branded mouse, monitor, USB 3.0 hub, cases, and yes, a keyboard. This also includes the excellent Raspberry Pi 400 and 500, Raspberry Pi built into a keyboard like the home computers of the 1980s. The peripherals tend to be basic, cheap, but high-quality, and pair great with a 500.

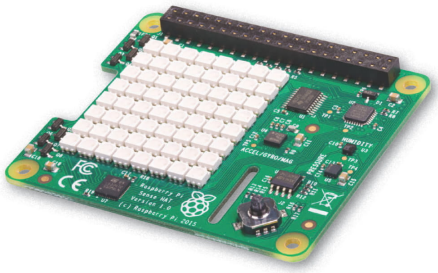


The official Raspberry Pi keyboard is available for multiple regions and keyboard types

# Sense HAT

## Space-worthy sensors

Developed for use on the International Space Station as part of the Astro Pi coding challenge for young students, it packs a series of environmental and motion sensors, a little joystick, and an 8×8 RGB LED matrix. It's great for electronics education, and a new updated version has been released.



There are slots for the camera cable so that you can use the whole suite of Raspberry Pi sensors

> *Developed for use on the International* ***Space Station***

# RetroPie

## Perfect retro gaming

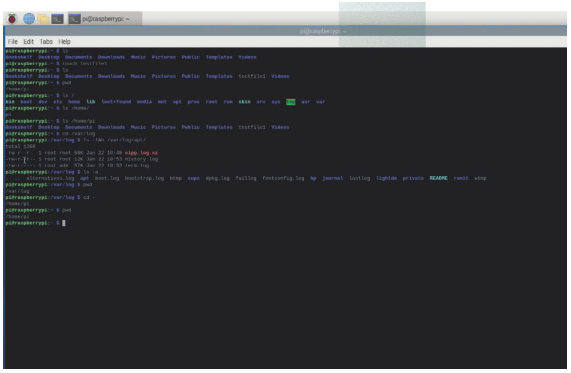
We were once absolutely amazed at how easy RetroPie was to set up – now it seems commonplace to have such easy-to-use retro gaming software on Raspberry Pi. It has a huge range of emulators you can access, compatibility with tons of official hardware, and can boot Kodi. Just bring your own games.

RetroPie uses the popular EmulationStation and RetroArch as front-ends

# Terminal

## Text-based computing

Feel like a cliché movie hacker and control your Raspberry Pi from the command-line interface. You may also use it in a headless scenario, where you have a remote Raspberry Pi that doesn't need to have a desktop interface running. Networking, automatic hard drive mounting, file sharing, and more is also only accessible via a terminal.

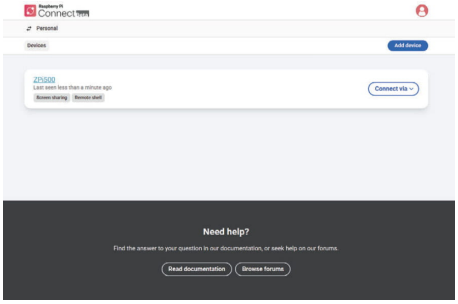


Using cd to move into a directory and saying 'I'm in' is quite fun

# VNC

## Remote control

Raspberry Pi is very small, but monitors and keyboards are not always as small. With VNC you can connect remotely to a Raspberry Pi and its desktop from the comfort of another computer. Raspberry Pi Connect is an official system that makes use of this to securely connect to your Raspberry Pi via a web browser anywhere in the world, with minimal lag.




Raspberry Pi Connect is built into Raspberry Pi OS too, making it easy to set up

# UHD

## 4K video from Raspberry Pi

Raspberry Pi has always been HD, with 1080p output at launch at a time when even the flagship games consoles couldn’t handle it well. Raspberry Pi 4 and 5 both have two Micro HDMI ports that support 4K resolutions with no problem, allowing for the ultimate media PC experience.

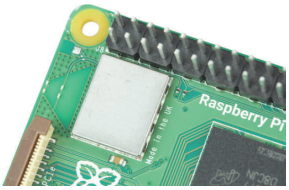



The jump from one 1080p output to dual 4K was huge for Raspberry Pi

# Wi-Fi

## Wireless networking and sharing

While not always standard on Raspberry Pi, or even Pico, Wi-Fi and Bluetooth are now an integral part of many Raspberry Pi models. Raspberry Pi also works great as a wireless access point for when you need a wireless network on the go for portable projects or very old hotels.

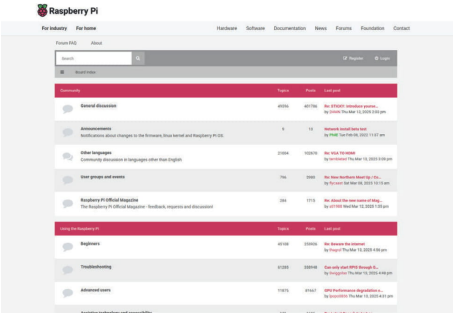



The antenna is printed into the circuit board, right next to the radio chip

# You

## It takes a community

As the community magazine, you might think we're a bit biased, but we think the global community is one of the most important parts of Raspberry Pi. Our magazine is filled with the incredible projects you all create, and you're helping to inspire others to make something.




The Raspberry Pi Forum is a great place to take part in the global maker community

# X HAT

## Arcade machine builder’s dream

A tiny HAT for a tiny computer used in many full-sized arcade machines from a time when the tech had to be huge. Instead of buying expensive and hard-to-maintain cabinets from 40 years ago, why not build your own custom machine that can play more than one game?

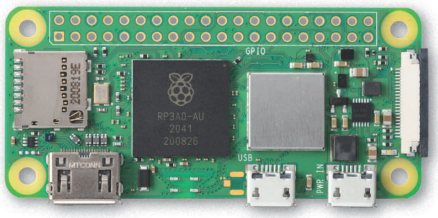



It also helps to power the excellent Picade kit from Pimoroni

# Zero

## The mini Raspberry Pi computer

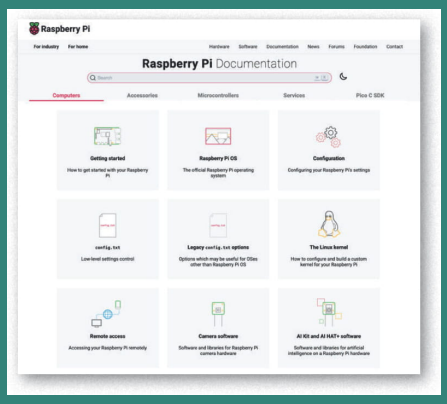
Launched on this very magazine in December 2015, Raspberry Pi Zero and Zero 2 are chewing-gum-stick sized computers that can handle just about anything a Raspberry Pi can. Raspberry Pi Zero 2 has the same power as Raspberry Pi 3, making it very capable despite its size.

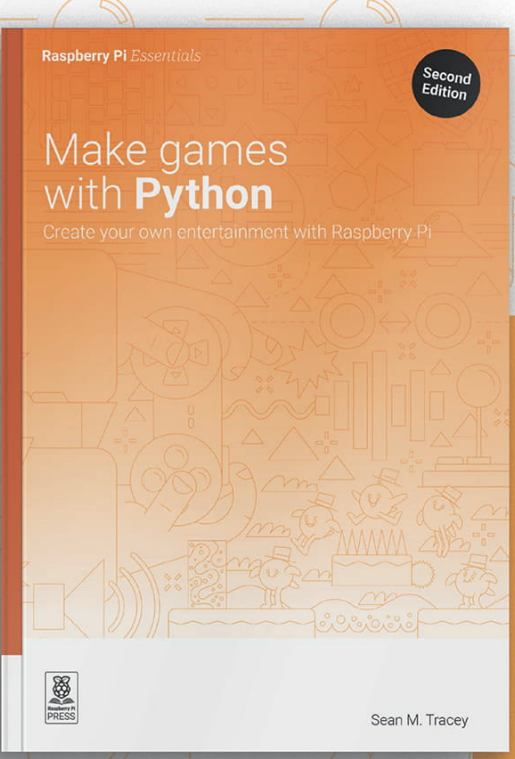


We love Raspberry Pi Zero and we’re always surprised what tiny projects it has been slipped into

# Glossary

For an A to Z of everything you'll ever need to know about using Raspberry Pi, make sure to check out the official documentation (**rpimag.co/docs**). It's easily searchable and covers just about everything you'll ever need to know about using Raspberry Pi.






While millions of us enjoy nothing more than spending hours racking up high scores on our favourite video games, too few are exposed to an even more gratifying way to spend time — making them.

This book teaches Python and Pygame development, helping you to understand the games you play and create almost anything your imagination can come up with.

- ***As you work your way up to creating your own shoot-'em-up game, you'll learn how to:***
  - *Create shapes and paths*
  - *Move sprites and detect collisions*
  - *Handle keyboard, mouse, and gamepad input*
  - *Add sound and music*
  - *Simulate physics and forces*

BUY ONLINE: **rpimag.co/makegamesbook**

OceanofPDF.com

# Create a Sense HAT interactive pixel pet

## Create a pint-sized pocket monster living in the digital world of a Raspberry Pi Sense HAT



**Maker**

**Lucy Hattersley**

Lucy is editor of Raspberry Pi Official Magazine and has a real cat that doesn't like being prodded one bit.

**rpimag.co**

**Using sensors and output devices is a great way to make your computer programs more interactive.** The Raspberry Pi Sense HAT contains a whole set of sensors that can detect movement, and in this activity we'll use them to take a digital pet for a walk.

You'll need to design your pet avatar before you program any actions. There are examples of some famous characters you can make on an excellent sprite sheet created by Johan Vinet, which can be found at **rpimag.co/spritesheet**. You can use this to draw your space pet avatar, as seen in **Figure 1**.

Draw your picture out on squared paper using coloured pencils. You'll need two pet designs, the second preferably being very similar to the first so that you can animate your pet. In **Figure 2**, you can see that our two images are almost identical to each other – only the feet are in a different place.

Later, when you code your animation, this will create the illusion that the pet is walking.

## Labelling each pixel

Think of a letter from the alphabet to represent each colour in your pixel pet image: for example, 'w' for white or 'r' for red. If using squared paper for your design, you can write the letters on top, as shown in **Figure 3**. Note that 'e' stands for empty.

> *Nothing should happen until you shake your Raspberry Pi. Then the pet animation will run*

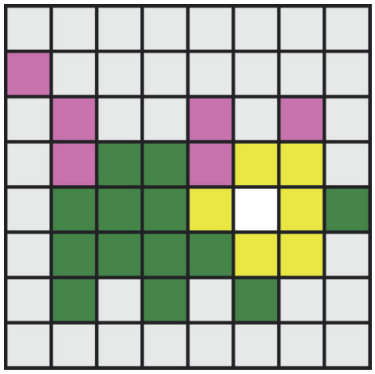


**Figure 1:** The first animation frame

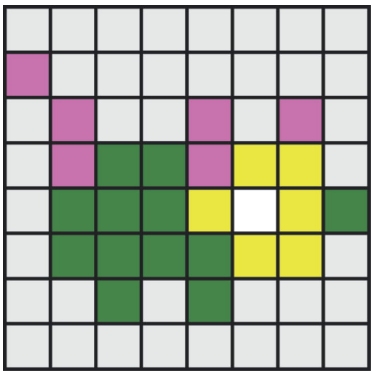


**Figure 2:** The second animation frame

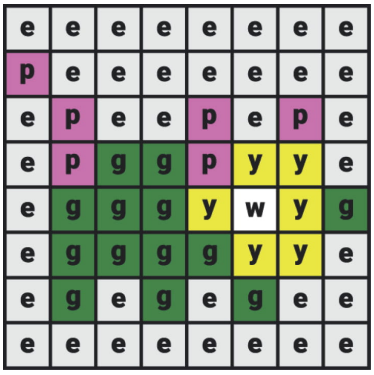



**Figure 3:** The pixel pet with colour labels

**DOWNLOAD**
**THE FULL CODE:**

[rpimag.co/pixelpetpy](rpimag.co/pixelpetpy)

# pixel_pet.py

> Language: **Python**

```
001.  from sense_hat import SenseHat
002.  import time
003.
004.  sense = SenseHat()
005.
006.  p = (204, 0, 204) # Pink
```

```
007.  g = (0, 102, 102) # Dark Green
008.  w = (200, 200, 200) # White
009.  y = (204, 204, 0) # Yellow
010.  e = (0, 0, 0) # Empty
011.
012.  pet1 = [
013.   e, e, e, e, e, e, e, e,
014.   p, e, e, e, e, e, e, e,
015.   e, p, e, e, p, e, p, e,
016.   e, p, g, g, p, w, w, e,
017.   e, g, g, g, w, y, w, y,
018.   e, g, g, g, g, w, w, e,
019.   e, g, e, g, e, g, e, e,
020.   e, e, e, e, e, e, e, e
021.   ]
022.  pet2 = [
023.   e, e, e, e, e, e, e, e,
024.   p, e, e, e, e, e, e, e,
025.   e, p, e, e, p, e, p, e,
026.   e, p, g, g, p, w, w, e,
027.   e, g, g, g, w, y, w, y,
028.   e, g, g, g, g, w, w, e,
029.   e, e, g, e, g, e, e, e,
030.   e, e, e, e, e, e, e, e
031.   ]
032.
033.  def walking():
034.      for i in range(10):
035.          sense.set_pixels(pet1)
036.          time.sleep(0.5)
037.          sense.set_pixels(pet2)
038.          time.sleep(0.5)
039.
040.  sense.clear(0, 0, 0)
041.  while True:
042.      x, y, z = sense.get_accelerometer_raw().values()
043.      if x >= 2 or y >= 2 or z >= 2:
```

```
044.         break
045.  walking()
```

You'll notice that we have eight rows and eight columns of letters, each separated by a comma, to make up the LED matrix on the Sense HAT. Repeat this step for your second pet design so that you end up with two grids of letters.

## Code your pet

Now that you have your designs represented as letters in a grid, you can start to code them in Python. Open the Thonny editor (click the Raspberry Pi menu and select Programming > Thonny); if Thonny is already open, press the New button to start a new program. Save the new, empty file as space-pet.py.

First, you need to import all the modules and libraries required for the project into your code and create a `SenseHat` object:

```
from sense_hat import SenseHat
import time
sense = SenseHat()
```

Note that capital letters, full stops, and commas are very important in Python. Your code might not work if you don't include these. Next, create a variable for each colour label in your pet design, like this:

```
p = (204, 0, 204) # Pink
g = (0, 102, 102) # Dark Green
w = (200, 200, 200) # White
y = (204, 204, 0) # Yellow
e = (0, 0, 0) # Empty
```

The numbers inside the brackets are the RGB (red, green, and blue) values. Mixtures of these colours make different shades. The higher the number, the more of that colour it will contain. For example, (255, 0, 0) would make a solid red, whereas (0, 255, 0) would create a vivid green. You can change these numbers in your code to get the colours that you want.

Next, use a list to store your pixel pet design, like this:

```
pet1 = [
   e, e, e, e, e, e, e, e,
   p, e, e, e, e, e, e, e,
   e, p, e, e, p, e, p, e,
   e, p, g, g, p, w, w, e,
   e, g, g, g, w, y, w, y,
   e, g, g, g, g, w, w, e,
   e, g, e, g, e, g, e, e,
   e, e, e, e, e, e, e, e
   ]
```

Here you have created a variable called `pet1` and stored a list of labels for each colour by using `[` at the start and `]` at the end. Repeat for the second pixel pet design, using a different variable name like `pet2`. Your code should start to look something like the drawing you created earlier.

If you ran your code now, nothing would happen, because so far you have only told the program to store information. To make something happen, you will need to write a command to call on that data and display your colours in the correct order on the Sense HAT's LED matrix. Type this command underneath your lists:

```
sense.set_pixels(pet1)
```

Save your code by clicking Save, then run your program to test it works by clicking Run or pressing F5. Note what happens. Why did only one of your pet designs appear? It's because you only called `pet1` in your command.

Add a delay using the `time.sleep` function, then call the second picture using the same command as before, like this:

```
time.sleep(0.5)
sense.set_pixels(pet2)
```

Save and run your code to see your pet 'move'.

## Animate your pet

So far, your pixel pet only changes once. To animate it fully, you will need to switch repeatedly between the two pictures with a time delay. You could write the commands out over and over again, but it makes more sense to put them in a loop.

Move to the end of your program and locate the `sense.set_pixels(pet1)` line and the two lines that follow. Change them to look like this:

```
for i in range(10):
    sense.set_pixels(pet1)
    time.sleep(0.5)
    sense.set_pixels(pet2)
    time.sleep(0.5)
```

Don’t forget to add the extra `time.sleep(0.5)` on the last line, and remember to indent the lines after `for i in range(10):` to place them inside the `for` loop. This `for` loop with the `range` function will repeat the indented code ten times and then stop.

Save and run your code to watch the animation. You will notice that after the animation has finished, you are left with the same image still displayed on the LED matrix. There is a useful function that you can use to clear the LEDs; add this line above your new loop to clear the LEDs when you first run your program:

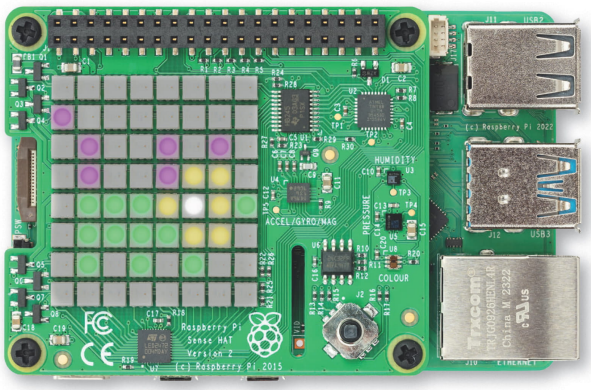


**Figure 4:** The final code running on a Sense HAT

```
sense.clear(0, 0, 0)
```

## Create a walking function

A function is a piece of code that you can use over and over. As the goal is to trigger the walking animation later on, it makes sense for us to put the animation code into a function that can be called when an action has been sensed by the hardware.

To put your code into a function, you simply need to add the line `def walking():` above your `for` loop and indent the lines beneath, like this:

```
def walking():
   for i in range(10):
     sense.set_pixels(pet1)
     time.sleep(0.5)
     sense.set_pixels(pet2)
     time.sleep(0.5)
```

The use of `def` here means that you are defining a function you have named `walking`. Now you need to call the function. So, at the bottom of your code, type:

```
walking()
```

## Shake for more

It's time to use the Sense HAT's movement sensors, particularly its accelerometer, to trigger the walking function and make the project more interactive. Between `sense.clear(0, 0, 0)` and the function call to `walking()`, type:

```
sense.set_pixels(pet1)
while True:
   x, y, z = sense.get_accelerometer_raw().values()
   if x >= 2 or y >= 2 or z >= 2:
```

```
        break
```

The first line shows a (stationary) space pet, and the second creates an infinite loop. The next line obtains current movement readings from the Sense HAT on its x, y, and z axes. As your Raspberry Pi is presumably sitting still on a desk, those readings will have a very low value. The loop continues forever until the Sense HAT moves enough for the statement `x >= 2 or y >= 2 or z >= 2` to evaluate to `True`. When that happens, the `break` statement stops the loop. You can help the Sense HAT take an accelerometer reading above 2 by shaking it! Your final code is in the **pixel_pet.py** listing (previous page). Save your code and then run it. Nothing should happen until you shake your Raspberry Pi. Then the pet animation will run, and your pet will start to walk (**Figure 4**).

Your pet walks for a few seconds, as the `for` loop executes ten times, before the program stops. If you find your pet doesn't move at all, you can try changing the `if` statement to require less movement:

```
    if x >= 1 or y >= 1 or z >= 1:
```

You could even add `x`, `y`, and `z` together to only take action when the sum of the raw values is over a certain amount:

```
    if x + y + z >= 2:
```

Can you think of a way to modify the program so that the pet walks for a bit every time you shake it, rather than just once?

*OceanofPDF.com*

# Pico drum machine

## Play drum samples at the press of a button



**Maker**

**Phil King**

A long-time Raspberry Pi user and tinkerer, Phil is a freelance writer and editor with a focus on technology.

**philkingeditor.com**

### YOU'LL NEED

- Raspberry Pi Pico (any model) with GPIO headers soldered on
- DF Mini Player
- microSD card
- Mini speaker (up to 3W)
- 2 × half-size breadboards (or one full-size)
- 6 × momentary push buttons
- 19 × pin-to-pin jumper wires
- Drum samples

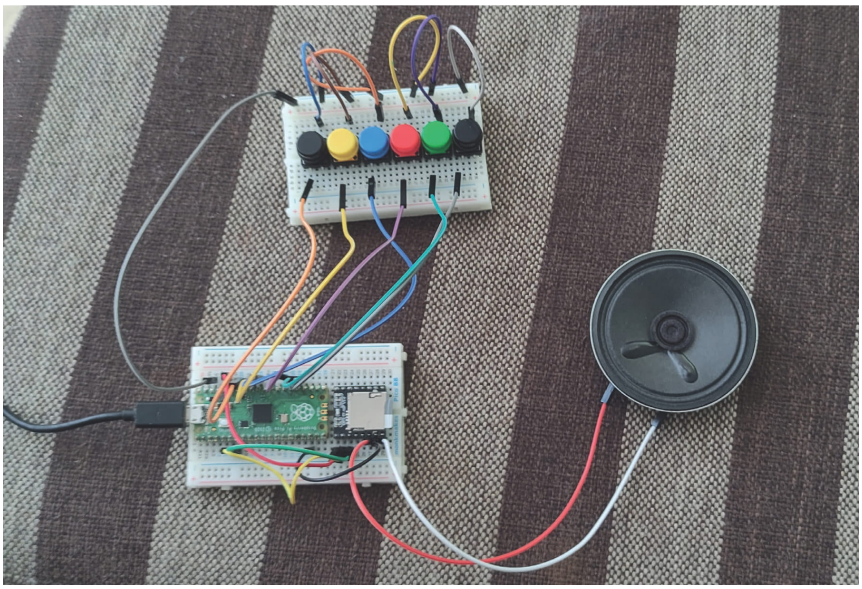
**While the first ever drum machine is considered to be the Rhythmicon, developed by Leon Theremin (yes, he invented that instrument too) in the early 1930s, digital drum machines came to the fore in the 1980s with the likes of the Linn LM-1.** The latter cost the equivalent of over $19,000 today, and yet we can now create a DIY drum machine using a $5 Raspberry Pi Pico.

Inspired by Arnov Sharma's project on Hackster (**rpimag.co/picodrummachine**), your author decided to try to create a Picopowered drum machine using a DF Mini Player to play the drum samples. While Arnov's version uses a custom-created PCB for the buttons, we decided to keep it simple with a breadboard-based design, the downside being a spaghetti of jumper wires to connect everything up. Still, the principle is the same: you press different push buttons to trigger different drum samples played by the DF Mini Player, outputting audio to a mini speaker.

## Building the circuit

To make it easier to wire the buttons to Raspberry Pi Pico, we opted to put the buttons on a second half-size breadboard, as you can see in the **Figure 1** wiring diagram. Each four-legged button spans the division in the middle of the breadboard, with the pins on one side connected to a ground rail. The other side of each button is connected to a GPIO pin on Pico – we used GPIO 28, 27, 26, 21, 20, and 19 – pulled up (in the code). So when you press the button, the GPIO pin is pulled low, and the code senses this and triggers the DF Mini Player to play the relevant sample – more on that later.

The DF Mini Player is placed on the same breadboard as Pico, connected via UART RX and TX pins, along with 3.3V power and ground. We use its speaker output pins to connect to a mini speaker (ours was 2W, 8Ω).



We used two breadboards for our Pico drum machine, although the wiring is still a bit spaghetti-like

> *To test our drum samples were playing correctly, we created a program to play them in sequence*



**Warning!**

**Hot solder**

Soldering irons get very hot, and stay hot for a long time after they're unplugged. Make sure that you put the iron in the stand when you're not using it and don't touch the metal parts – even after it's unplugged.

**rpimag.co/soldering**

To play some sounds, we need some drum samples. There are lots of free, open-source ones available online. We got ours from here: **rpimag.co/drumsamplesgh**. We found that some of the samples were a little long – and the DF Mini Player can only play one file at a time – so we opted to edit them in Audacity (available from **audacityteam.org**). If, after

trimming the end of a sample, you find it ends too abruptly, you can always apply a fade-out effect to the end.

The microSD card must be formatted as FAT32, so we erased it in Raspberry Pi Imager: select Choose OS > Erase, then Choose Storage and select the card. In addition, the files must be named 0001, 0002, 0003, etc., with the relevant suffix – you can use MP3 or WAV files. As we used the latter, ours were named 0001.wav, 0002.wav… 0006.wav. We found that it doesn't matter whether you put them in a folder or not.

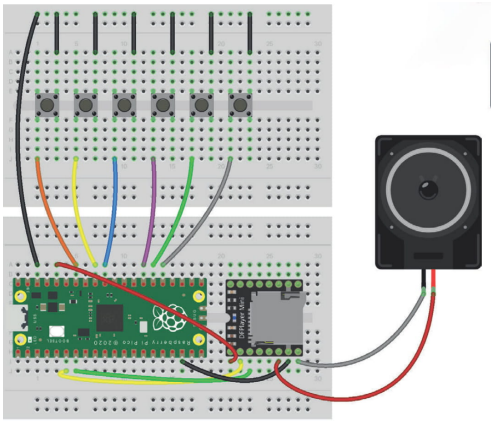
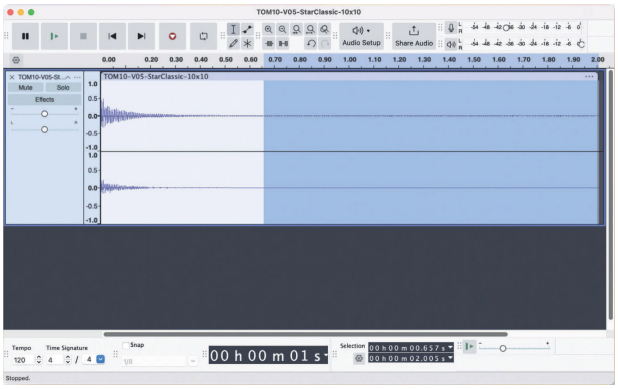


**Figure 1:** The wiring diagram for the Pico drum machine




We edited our drum samples in Audacity to reduce the length of some of the longer ones

## Coding it

Unlike Arnov, who programmed his drum machine in C, we opted to use MicroPython. For this, we made use of Stewart Watkiss's DF Mini Player library at **rpimag.co/dfminilibrarymp**. Just download the **dfplayermini.py**

script from there and then, using the Files tab in Thonny IDE, right-click and upload the file to your connected Pico (which already has MicroPython installed). You can then call the library in your programs.

To test our drum samples were playing correctly, we created a program (**play_drums_seq.py**) to play them in sequence:

```
from dfplayermini import DFPlayerMini
import time

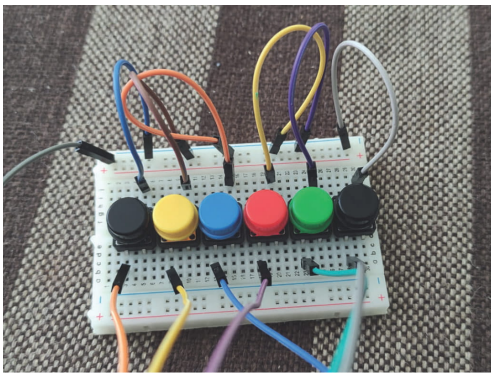
player1 = DFPlayerMini(1, 4, 5)
player1.reset()

print ("Set SD Card")
read_value = player1.select_source('sdcard')

print ("Set Volume 30")
read_value = player1.set_volume(30)

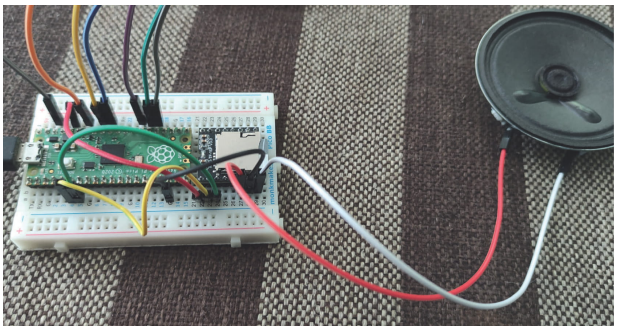
read_value = player1.query_num_files()
print (f"Num files {read_value}")

for i in range(6):
   print("Play",i+1)
   read_value = player1.play(i+1)
   time.sleep(1)
```

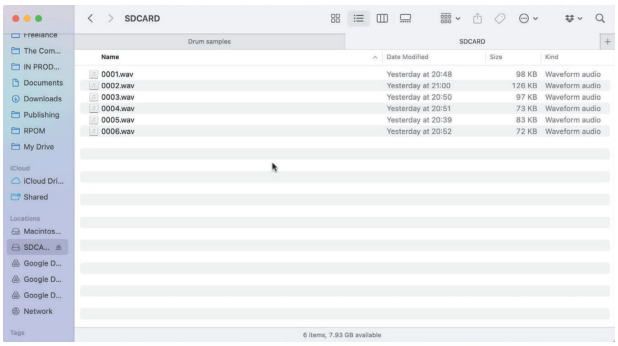


A close-up of the buttons. Each has one pin wired to the ground rail, and a pin on the other side wired to a GPIO pin on Pico

After importing the libraries, we set up a `player1` object to work with UART 1 on GPIO pins 4 and 5. We then reset the DF Mini Player so it's ready to start communicating. We select the SD card as the audio source, then set the volume – we maxed ours up to 30. We then query the number of files on the card (it should be six), before playing each one in turn.



Along with Pico, we placed the DF Mini Player on another breadboard and connected it to a mini speaker




You need to name the files on the microSD card in (four-digit) numerical order for the DF Mini Player to recognise them

Next, we adapted this to create our main program to read the button presses and trigger the sounds accordingly:

```
from dfplayermini import DFPlayerMini
from machine import Pin
import time

player1 = DFPlayerMini(1, 4, 5)
player1.reset()
```

```
print ("Set SD Card")
read_value = player1.select_source('sdcard')

print ("Set Volume 30")
read_value = player1.set_volume(30)

read_value = player1.query_num_files()
print (f"Num files {read_value}")

# Define GPIO pins for buttons
button_pins = [28, 27, 26, 21, 20, 19]

# Initialize input pins
buttons = [Pin(pin, Pin.IN, Pin.PULL_UP) for pin in
button_pins]

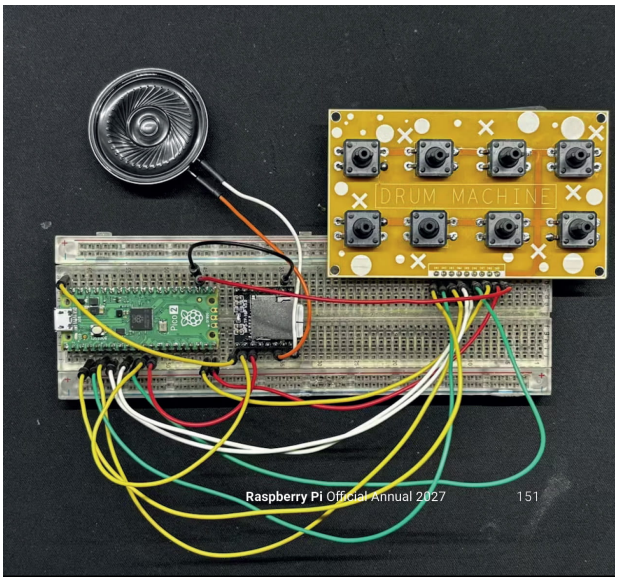
# Main loop
while True:
   for i in range(len(buttons)):
      # Read button state
      if buttons[i].value() == 0:
        player1.stop() # stop current sound
        read_value = player1.play(i+1)
      time.sleep(0.01) # Debounce delay
```

## Speaker soldering

The SPK_1 and SPK_2 pins on the DF Mini Player should be connected to the positive and negative terminals (or wires), respectively, of your mini speaker. While you can get away with wrapping the stripped ends of the jumper wires around the terminals, it’s best to solder them in place for a more secure and reliable connection. The sound from our mini speaker was a little tinny, so you might get more bass out of it by mounting it in some kind of enclosure.

Here, we add a line at the top to import the `Pin` method from the `machine` library. We create a list to set the GPIO pins for the buttons, then initialise them as inputs with the pin pulled up. In the main loop, we read the button state and then, after stopping any currently playing sound, play the relevant drum sample. We added a very short debounce delay to prevent a button press causing multiple triggers.

Hands up: the performance wasn't as good as we'd hoped, with a noticeable lag between pressing a button and the sound being played. As already mentioned, the DF Mini Player can only play one file at a time, which is a major limitation for a drum machine. An alternative would be to use an I2S-based audio board such as the Waveshare Pico-Audio to play the drum sounds. Still, our little Pico drum machine does work and you could use it to trigger other samples, such as spoken phrases or funny noises. Alternatively, using four of the buttons, you could try out Stewart Watkiss's MP3 player project (**rpimag.co/dfminimp3player**) to play songs stored on the microSD card.




This project was inspired by Arnov Sharma's Pico drum machine, for which he created a custom PCB to house the buttons

OceanofPDF.com

# Home Automation
## Made Easy

### Make your home smarter with Raspberry Pi and Home Assistant

**By Phil King**

**With Raspberry Pi, you can build a cost-effective and flexible smart home setup that's easy to control and customise.** A Raspberry Pi paired with Home Assistant – the free, open source home automation platform – becomes the brain of your smart home, enabling routines, voice assistant integration, and remote access, all tailored to your preferences.

First, we show you how to install Home Assistant on your Raspberry Pi and set it up, adding the smart devices on your network. We'll also explain automations and scenes, along with add-ons to get Home Assistant to communicate with more apps and devices. You can even access your Home Assistant system remotely when away from home.

We than explore the vast range of Home Assistant-compatible smart home devices available, ranging from sensors to lights and smart sockets. We also round up some home automation project ideas, including a smart garden.

Finally, we guide you step by step through an example project in Home Assistant: a simple security system with a motion sensor to

trigger an alarm and send notifications to your phone. So, get ready to make your home smarter…

# Get started with Home Assistant

## How to install and access your Home Assistant server

> It supports thousands of devices and services

**YOU'LL NEED**

- Raspberry Pi 4 or 5
- microSD card, 32GB or more
- Ethernet cable
- At least one compatible smart device: **rpimag.co/hadevices**

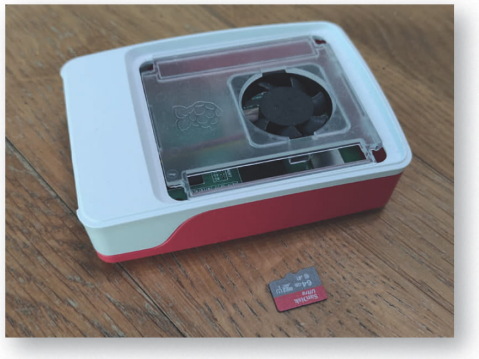
**Home Assistant offers a powerful alternative to other smart home solutions by focusing on privacy, flexibility, and control.** Running locally, it keeps your data private and can still work when the internet is down. It supports thousands of devices and services, enabling seamless integration across brands. Highly customisable, it allows for advanced automations tailored to your needs. As an open-source platform, it's free, constantly evolving through community support, and avoids vendor lock-in, giving you full control over your smart home setup without being tied to a specific company's ecosystem.

Installing it on Raspberry Pi is a quick and painless process. We'll guide you through it and how to start using your new Home Assistant server to monitor and control your smart devices.

## 01 Prepare Raspberry Pi

A Raspberry Pi 4 or 5 model is recommended for use with Home Assistant – other models lack the required RAM to ensure system stability. Along with that, you'll need a microSD card with at least 32GB capacity: this will give you sufficient storage for the operating system, database, logs, and potential add-ons. Ideally, the card should be A2 class, for faster read/write speeds.

For initial setup, you will also need an Ethernet cable to connect Raspberry Pi to your home router. You can subsequently opt to switch to a wireless connection if you want to move your Raspberry Pi somewhere else, away from the router (see 'Go wireless' box, overleaf).

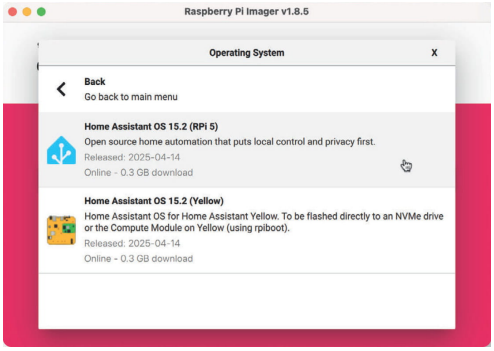


## 02 Install Home Assistant

The Home Assistant OS will be written to your microSD card, so connect the latter to another computer – using a USB card reader adapter if needed. Installation is made easier thanks to the fact that you can use the standard Raspberry Pi Imager tool to write Home Assistant to the card. Download Imager for your computer system (Windows, Mac, or Ubuntu Linux for x86): **rpimag.co/software**.

Open Raspberry Pi Imager, click 'Choose Device' and select your Raspberry Pi model to filter the OS options shown. Next, click 'Choose

OS’, then scroll down and select ‘Other specific-purpose OS’ > ‘Home assistants and home automation’ > Home Assistant. Select the version for your Raspberry Pi model. Finally, click on ‘Choose Storage’, then ‘Next’, and ‘Yes’ when the warning dialog appears. Imager will start writing the Home Assistant OS to the microSD card – it will take two to three minutes. Once it’s written and verified, you’ll get a ‘Write Successful’ dialog; you can now remove the card from the computer.




## 03 Connect to router

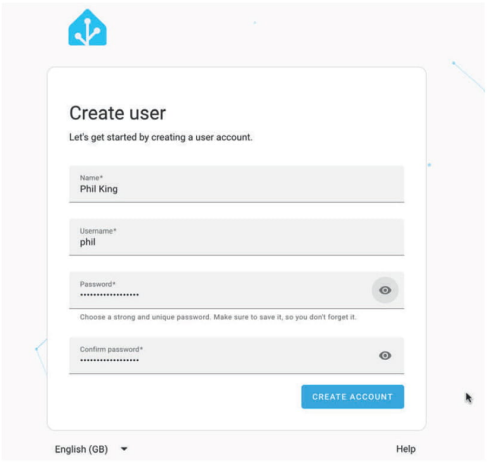
Take the microSD card with Home Assistant OS installed on it and insert it into the card slot on your powered-off Raspberry Pi. Connect an Ethernet cable to Raspberry Pi’s Ethernet port, next to the USB ports, then insert the other end of the cable into an Ethernet port on your home internet router. Power up Raspberry Pi and Home Assistant OS should then start configuring itself.

Give it a few minutes, then open a web browser on another computer or device and point it to **homeassistant/local:8123**. If Home Assistant is still configuring itself, you may see a page with its icon on the top and blue/grey progress bar with three more icons below. It can up to 20 minutes for Home Assistant to be finalise its configuration, but ours finished in about ten minutes. Don’t worry about any red exclamation marks appearing during the process.



## 04 Initial setup

When the Home Assistant server is ready, its web page will prompt you to ‘Create My Smart Home’ (with a couple of restore options below). Click that blue button to start configuring your smart home system. First, you’ll be asked to create a user: enter a name, username, and password. Then click ‘Create Account’.

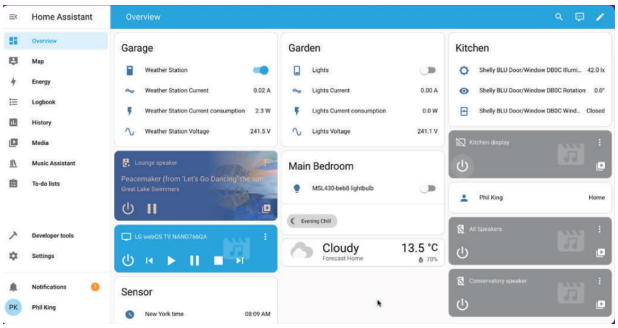



Next, you’re prompted to set your location. While your data is stored locally on your Home Assistant installation, an accurate location may be needed for some cloud-based integrations. Enter a postcode or the start of your address and choose from the list, then click Next.

On the next screen, you can opt to share anonymised data to help improve Home Assistant, if you wish. Finally, a screen will appear that may say Home Assistant has discovered some compatible devices on your network. Click on Finish.

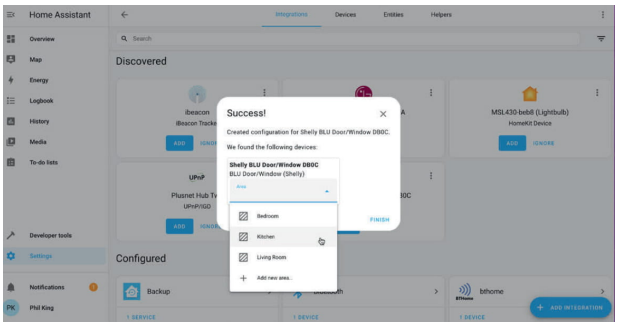
## 05 Web dashboard

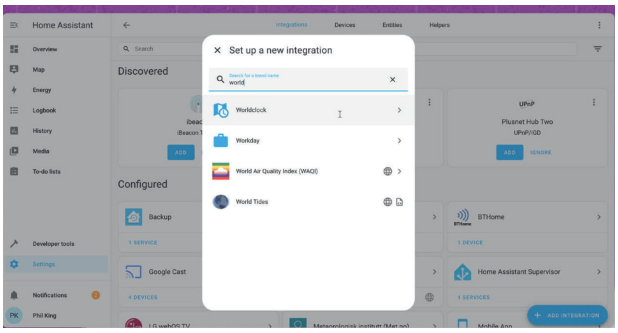
You should now see your Home Assistant dashboard showing any devices that have already been detected automatically on your home network – our Google Nest speakers showed up there, for instance – along with your name and a weather forecast for your location. By default, the dashboard is arranged automatically as you add more devices and entities, although you can create a custom layout later if you want and even change the theme to alter its appearance.




## 06 Add a device

In the left sidebar of the dashboard, you may see a number next to Notifications; click on that and it will tell you new devices have been discovered. Clicking on ‘Check it out’ takes you to the Integrations page, which shows icons and names for any new devices under ‘Discovered’. You can click ‘Add’ for one to add it to your system – or ‘Ignore’ it, which you may want to do for things like your router. Upon clicking ‘Submit’ to add a new device, you’ll see a ‘Success!’ dialog with the option to add it to an area in your home, using a drop-down menu. Click Finish and the device will be added to your dashboard – click on Overview to see it there.






## 07 More integrations

Integrations are the pieces of software that allow Home Assistant to connect to other software and platforms. Some are installed by default, while others will be added automatically when you add a device that uses one – for instance, an LG smart TV uses the LG webOS integration. Others may need to be added manually, while some may only be available via the optional Home Assistant Cloud premium service.

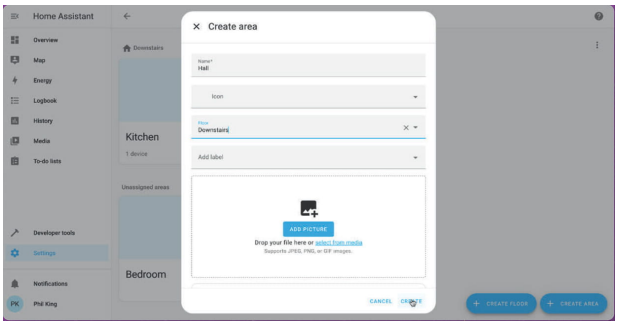
To browse the standard integrations available, click Add Integration on the Integrations page. You can then explore a long list of integrations or type in a search term at the top. To find out more about an integration, select it and then click the '?' icon.

As an example, let's add the Worldclock integration. After selecting it from the integrations list, choose a name (e.g. New York time'), time zone, and time format, then click Submit and Finish. It should now appear on the dashboard (Overview).

## 08 Areas, labels & zones

An area in Home Assistant can be used to represent a room or space, such as the garden, in which you can group devices located there. As well as grouping them together on the dashboard, this enables you to address a set of devices in one room when creating automations or scenes, such as turning on all lights in the living room. To add a new area, go to Settings > 'Areas, labels & zones' and click Create Area, then fill in the details (and add an optional picture). If you live in a multi-storey abode, you can also create floors and assign areas to them.

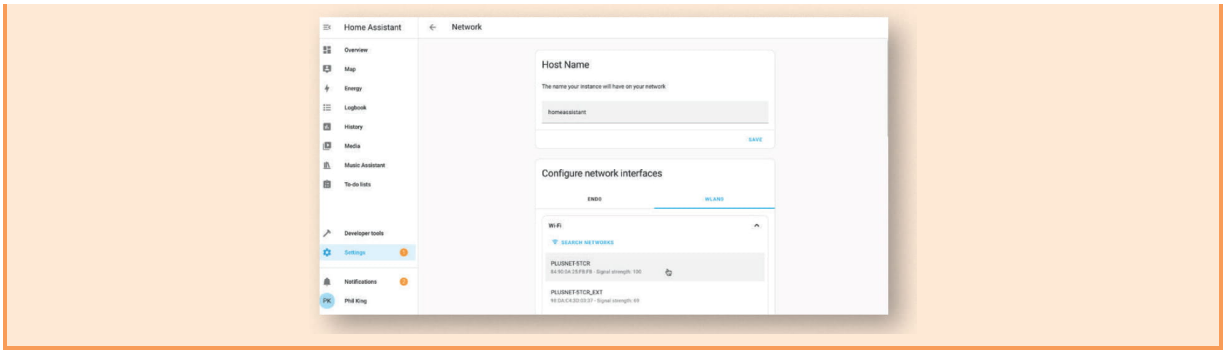
Zones are used to detect whether someone (added to Settings > People) is present in a certain location, typically using their mobile phone to track them. By default, you will have a Home zone – set during initial configuration, but editable by moving the marker on a map – and you can add more zones, perhaps for work or school.

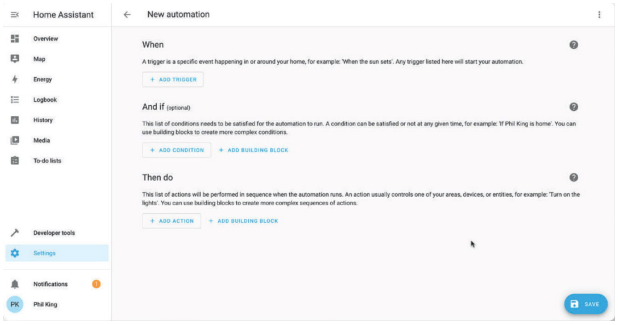



Labels can be used to group related elements – devices, entities, areas, scenes, scripts, and helpers – together, even if they're not of the same type or in the same location. This can be used to filter the list of devices to only those with a certain label.

### Go wireless

If you want to move your Raspberry Pi running Home Assistant away from your router, you can switch it from a wired Ethernet connection to a wireless Wi-Fi one. In Home Assistant, go to Settings > System > Network. Select the WLAN0 tab and Search Networks. Choose your router and security method (typically WPA-PSK), then enter your router password and click Save. Power down, unplug the Ethernet cable from Raspberry Pi, then power it up again. Your Home Assistant should now connect to the router via Wi-Fi.




## 09 Automations and scenes

Home Assistant makes it easy to create automations to determine how devices behave when certain conditions are met. For instance, you could create one to turn on smart lights (or lamps connected to smart sockets) at sunset. To create an automation, go to Settings > ‘Automations & scenes’ and click Create Automation, then select ‘Create new automation’. You then need to add a trigger event, plus optional extra conditions that need to be met, and set one or more actions to take. As an alternative to creating a custom automation from scratch, you can use or adapt one of the preconfigured ‘blueprint’ templates shared by the Home Assistant community.

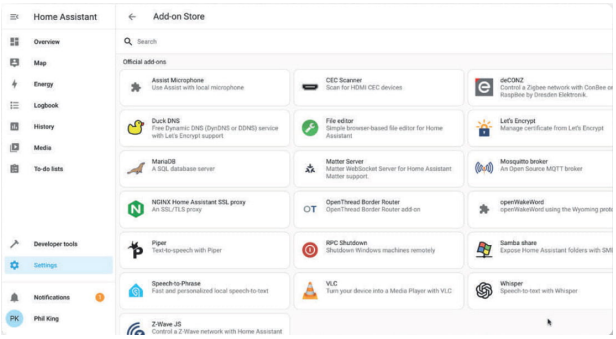
Scenes enable you to set a state for a group of devices or entities. For instance, you could set a scene for dinner time by dimming lights and playing suitable music on a smart speaker. You can then use the scene as an action when creating automations.

In addition, you can create custom ‘scripts’ comprising a sequence of actions and/or conditions. These can be created from the GUI menus or by

editing the YAML files – which you can also do for automations to fine-tune them.

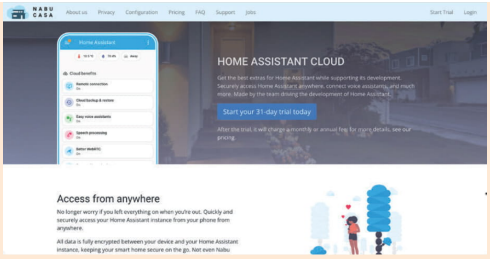
## 10 Installing add-ons

To customise your Home Assistant setup even further, there's a large range of add-ons that you can install to add extra functionality. Go to Settings > Add-ons and click the Add-on Store button to see the full list, arranged into categories. Official add-ons include a Mosquitto broker for sending MQTT messages between devices, Z-Wave JS to use devices with that wireless protocol (communicating via a Z-Wave USB dongle or HAT on Raspberry Pi), and the Music Assistant music library manager for your offline and online music sources. Just select an add-on and click Install to add it to your Home Assistant setup.




**Remote access**

While there are other methods, the easiest way to access your Home Assistant server remotely, when away from your Wi-Fi network, is to use the optional Home Assistant Cloud subscription service. This enables you to securely access your Home Assistant from your phone from anywhere. It also makes it easier to get it to work with voice assistants such as Alexa and Google Home Assistant, or use its alternative Assist privacy-focused voice assistant.




0

# Smart devices

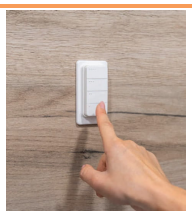
## Just a few of the devices you can add to your smart home

| | |
|---|---|
|  | **Sensors**<br>There are all manner of sensors you can add to a Home Assistant setup. They measure everything from temperature and humidity to soil moisture and light levels. Motion and contact sensors can be used for home security, as in our alarm tutorial over the page. |

| | |
|---|---|
|  | **Smart socket**<br>The great thing about a smart socket is that it can make a ‘dumb’ device smart, enabling you to turn it on and oT âia Koɋe ssistant˙ s well as Wi-Fi sockets, there are Z-Wave and Zigbee models. |

| | |
|---|---|
|  | **Lighting**<br>Many modern homes have at least a few smart lights or bulbs that can be controlled via a phone app or voice assistant. These can usually be connected to Home Assistant and used in automations. |

## Motion and contact sensors can be used for home security

| | |
|---|---|
|  | **Audio/video**<br>Smart speakers, displays, and TVs are ubiquitous these days. Most are easily connected to Home Assistant via integrations. Alexa speakers are a little trickier to set up, unless you use Home ssistant !loud. |
|  | **Raspberry Pi**<br>You can use one or more secondary Raspberry Pi computers linked to Home Assistant, using the Remote GPIO integration to read and control their GPIO pins. There's also a Raspberry Pi !aɑera inteƀration. |
|  | **Input device**<br>There are numerous smart buttons – connected via Bluetooth or another wireless protocol – that you can use to control devices with HA automations, with single or multiple button presses triƀƀerinƀ diꟻerent actions. |

**Wireless protocols**

There are four main protocols used by smart devices to communicate with your Home Assistant server, or each other…

**Wi-Fi:** Offers good range (up to 100m) and high bandwidth, making it ideal for devices such as cameras and smart TVs. The downside is that it's power-hungry – not a problem if the device is mains-connected, however, such as a smart socket.

**Bluetooth LE:** Uses less energy than Wi-Fi, making it ideal for battery-powered devices. The downsides are lower bandwidth and shorter range: typically around 10m, depending on walls and other obstuctions.

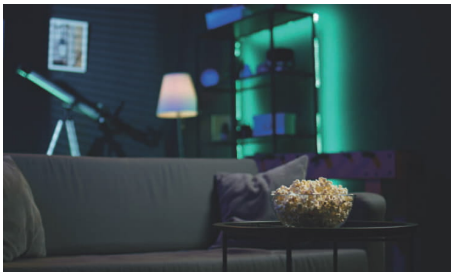
**Zigbee:** Used by certain smart devices, this operates as a mesh network with each device acting as a node, enabling it to cover a large area with energy efficeiency. You'll need a Zigbee USB dongle or HAT for your Raspberry Pi to use it.

**Z-Wave:** Like Zigbee, this works as a mesh network between devices, thereby maximising area coverage and power efficiency. A Z-Wave USB dongle or HAT for your Raspberry Pi is required to communicate with such devices.

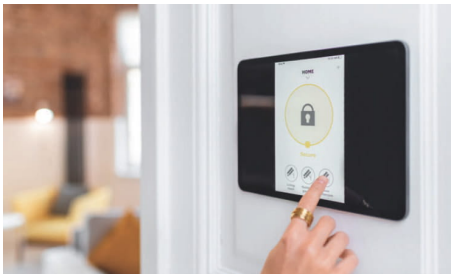
# Automation ideas

## Here are some suggestions for HA automation projects

**Mood Lighting & Sound**
With a few smart lights and a speaker or two, you can set the scene in a room for an occasion such as a romantic meal, party, or movie night. The scenes feature in Home Assistant makes it easy to trigger multiple actions from an automation.



**Automated Motion**
For the ultimate smart home, you'll want to be able to control motorised items such as blinds, curtains, windows, and garage doors. Home Assistant has the integrations needed and can even work with some robotic lawnmowers!

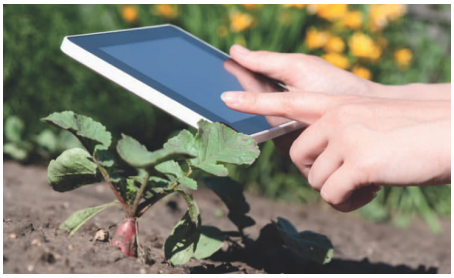


**Home Security**
With a few sensors, plus optional cameras, you can create a DIY home security system using Home Assistant. Or maybe just use a motion sensor to trigger the lights to come on when you enter an area, such as a laundry room.



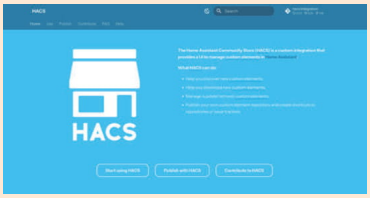
**Smart Garden**

Your smart home can extend into your garden or other outdoor space. With soil moisture sensors and relay-controlled water pumps, you could create an automated irrigation system to keep your plants healthy.



## Community help

Home Assistant has a vibrant community of users who love to share their custom creations: integrations for otherwise unsupported devices, advanced integrations, dashboard cards and themes, scripts, and templates. This is made easier using the HACS (Home Assistant Community Store) custom integration – despite the name, everything in it is free and open source. For more details, visit **hacs.xyz**.



# Create a home alarm

## How to build a simple security system with Home Assistant

**YOU'LL NEED**

- Home Assistant running on Raspberry Pi
- Motion sensor
- Smart speaker, display, or TV
- Window/door sensor (optional)
- Smartphone with Home Assistant companion app

**To showcase how to use the powerful automations feature in Home Assistant, we'll build a simple alarm system using a motion sensor and a smart speaker – plus optional mobile phone notifications for when you're away from home.**

### 01 Add motion sensor

To start with, we'll set up a motion sensor to detect movement in the room we want to secure – our kitchen in this case. To create a more robust home security system, you could expand this with more sensors to cover other areas of the house.

We're using a Shelly BLU Motion Detection Sensor, a passive infrared (PIR) sensor that communicates wirelessly via Bluetooth (**rpimag.co/blumotion**). There are other similar Bluetooth-based sensors, as well as Z-Wave, Zigbee, and Wi-Fi ones. Those longer-range wireless protocols might be preferable if you're having trouble with Bluetooth's

limited range (typically around 10m), although it should prove adequate for most homes.

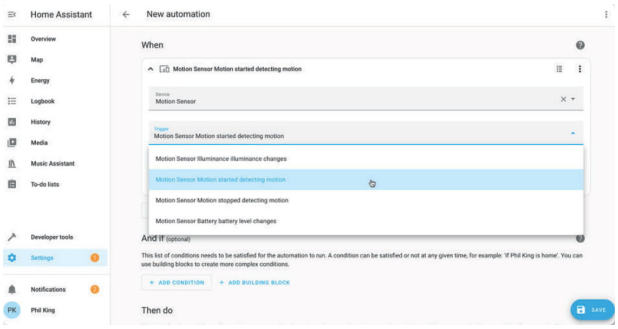
Our Shelly sensor has a beacon mode that broadcasts its status at regular intervals, but for our purposes it's better to use the default regular mode that only does so when the motion detection status changes, as this saves battery power – the sensor uses a standard CR2477 coin cell, which should last a couple of years.

Once you activate the sensor and it detects motion, it should appear as a discovered new device in Settings > 'Devices & services'. Click the Add button to add it to your setup; you can rename it and assign it to an area (in our case, Kitchen). Its illuminance reading and motion detection status should then appear on the dashboard.



## 02 Automation trigger

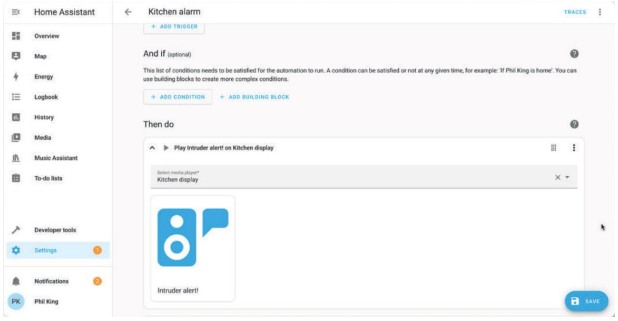
Let's start creating our automation for the intruder alarm. Go to Settings > 'Automations & scenes' and click the blue Create Automation button, then select 'Create new automation' in the dialog. As explained in our Home Assistant setup guide, each automation comprise a trigger and an action, along with optional conditions. So, in this instance, we'll set our sensor detecting motion as the trigger.




Under ‘When’, click the +Trigger button and select Device, then select your motion sensor’s name from the Device drop-down list (we called ours ‘Motion Sensor’). By default, our sensor trigger came up as ‘Motion Sensor Illuminance illuminance changes’, so click on that and select ‘Motion Sensor Motion started detecting motion’.

## 03 Audible alert

For our automation action, we’ll use a Google Nest speaker to play a text-to-speech phrase. Under ‘Then Do’, click the Add Action button. Select Media Player > ‘Play media’. Select the desired media player from the drop-down list. Click ‘Pick media’ > Text-to-speech > Google Translate and then type the message (we chose ‘Intruder alert!’), select the Language (there are several English accents), and click Finish. Finally, click the Save button and name your new automation (e.g. ‘Kitchen alarm’); you can add an optional area, description, etc. if you want.




To test it out, wave a hand over your motion sensor and you should hear the smart speaker say ‘Intruder alert!’ There may be a slight delay the first time, but it should say it almost instantly on subsequent triggerings – note

that you may need to wait 30 seconds for the motion sensor status to revert to ‘Clear’ before it will trigger again.

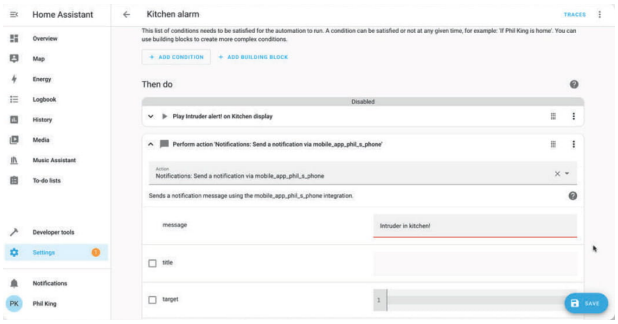
## 04 Send phone notification

An audible alert is all very well if you’re in the vicinity, but what if you’re away from home? For this, you can send a notification to your mobile phone. There are various ways to do this, but possibly the easiest is to use the Home Assistant companion app, available for Android and iOS. Install that on your phone and log in to your HA web server; you can then rename the device (optional) and enable location tracking for presence detection (if you want).

You’ll then be able to view your HA dashboard. Even though you won’t be able to connect to the server when away from your home Wi-Fi zone, you can still receive notifications via the app.

Return to your automation and click Add Action again. Then select ‘Other actions’ > ‘Perform action’. From the Action drop-down list, select ‘Notifications: send a notification via mobile_app_[your_phone_name]’. Then type some text into the ‘message field’. Finally, click Save.

Test it out by triggering the sensor again. Even if you have Wi-Fi disabled on your phone, you should still receive a notification via the Home Assistant app. If not, double-check that notifications are enabled for the app.





## 05 Add extra sensors

Now you have a working alarm system, you could add one or more extra sensors to trigger the same action – or for a new automation.

For our system, we added a Shelly BLU Door/ Window Sensor (**rpimag.co/bluwindow**). This Bluetooth sensor has two parts: one is attached (via a sticky pad) to the outer edge of a window or door, while the other is positioned next to it on the surrounding frame. So, when you open the window or door, the two parts separate and the sensor's status changes from 'Closed' to 'Open'. It can even estimate the angle of opening. After triggering the sensor, HA discovered it and we added it to the dashboard, as before. We then added it as a trigger, with the 'Window opened' option in our existing automation.

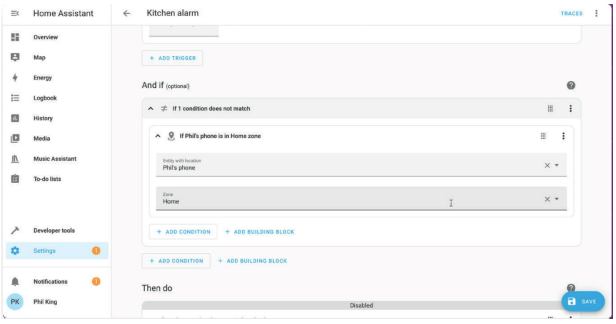
You could add more triggers and actions, as you prefer. Perhaps you might want it to turn on the lights or trigger a camera (HA works with brands such as Ring, as well as Raspberry Pi Camera).

> To fine-tune how your alarm operates, you can add conditions for when it can be triggered

## 06 Add conditions

To fine-tune how your alarm operates, you can add conditions for when it can be triggered. For instance, you might limit it to certain times of the day or when nobody's home (using mobile phone presence detection). in the automation options, click Add Condition, then 'Time and location'. Select 'Time' to limit triggering to before and/or after a certain time, on selected days.

If you want to use presence detection to only trigger the alarm when you're away from home, you'll need to first add a Building Block and set to 'Not'. Now select Add Condition within that block and choose 'Time and location' > Zone. Then select the Entity (your phone) and the Zone (Home).




OceanofPDF.com

# Hack a robot arm

## Upcycle a toy robot arm to add Raspberry Pi smarts and internet remote control



**Maker**

**PJ Evans**

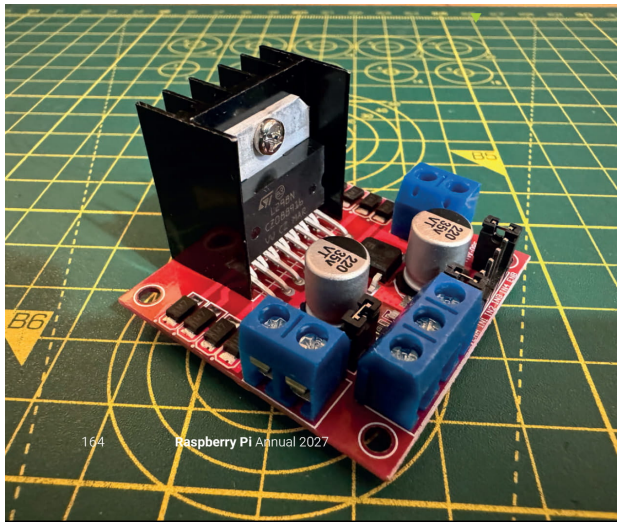
PJ is a writer, software engineer, and very average roboticist. His robot arm enjoys teasing the cat.

**mastodon.social/ @mrpjevans**

**Ever buy a gadget? Does it get played with for a while then consigned to the shelf or cupboard?** Upcycling old gizmos and sprinkling a little Raspberry Pi magic is one of our favourite things. Take the OWI-535 (**rpimag.co/owi535**) robot arm. This fun toy, with several points of articulation and a gripping claw, could be found in electronics shops under various brands. It's still widely available. Although a USB version is now on sale, we wanted to see if we could convert the original remote control version and make it Raspberry Pi powered. Even if you don't have this particular device, this should be a useful guide to motor control.

### How the arm works

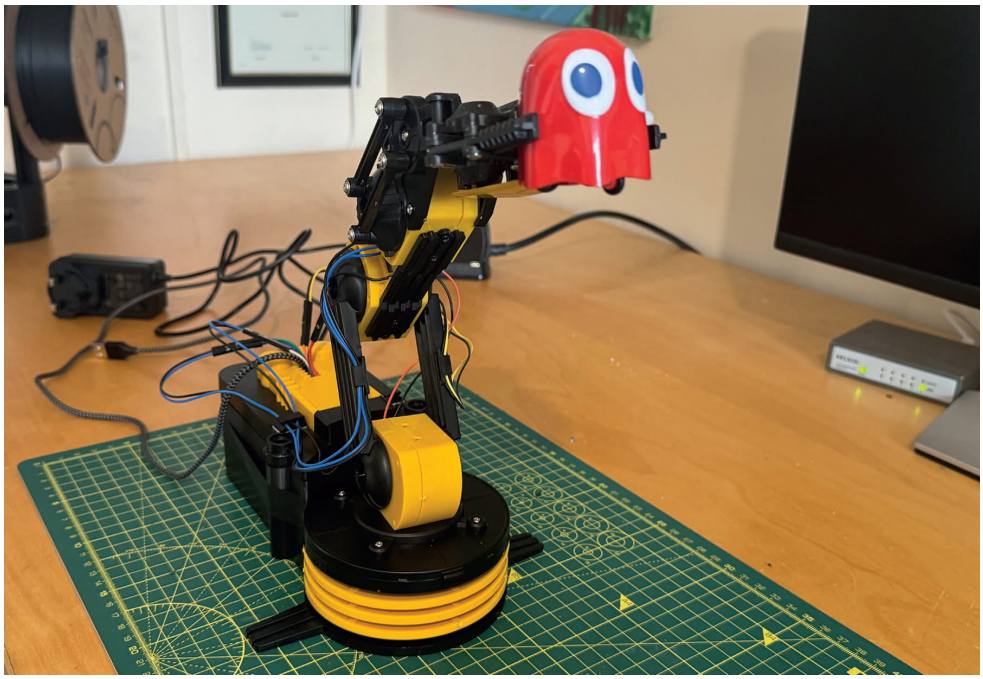
The OWI-535 robot arm is made up of five DC motors, each controlling a different point of articulation. The base rotates, there is a shoulder, elbow, and wrist, then finally the claw. Each point is bidirectional: the base can rotate either way, the joints can move back and forth, and the claw can adjust its grip. Each motor is connected to a control board by standard jumper cables (useful!) and has two wires. To control the arm, we need to be able to power the motors and also be able to reverse the DC current in order to change direction.




The L298N contains two H-bridge circuits on a single board

## Raspberry Pi challenges

The use of two-wire DC motors presents us with a problem. If it were three wires, a ground and a positive per-direction, it would be straightforward, but a motor that will spin based on polarity of the current applied is not something Raspberry Pi's GPIO can handle out of the box. Also, the current required by the motors to operate is well out of the reach of a Raspberry Pi and we would start damaging things. We need an interface between the two that can help us switch the motors. Enter the humble H-bridge.



Robot! Put Blinky down right now!

## H-bridge over troubled motors

The H-bridge is a very common circuit in robotics that solves our problem. It's a wiring method that uses four switches to reverse the current to a motor. See how the diagram (**Figure 1**, overleaf) shows both states, causing the motor to spin one way or another. Using five H-bridge circuits and a separate power supply, we can safely control all the motors. We'll do this using three L298N (**rpimag.co/l298n**) circuit boards. These are cheap and widely available. Each one contains two H-bridge circuits onboard and they are easy to wire up.

> *Take your time and check everything again and again*

## Understanding the L298N

The L298N board can control two motors. Each connects its two wires to the board. Power is provided by an external source connected to the power

terminals. We need 12V at about 1A. Such supplies are widely available. Finally, control is provided by providing a signal to each ‘Enable’ line (this allows us to control speed) and by sending one of two lines to ‘high’, controlling direction. We now have the three-wire setup we originally wanted. The control lines can be safely connected to your Raspberry Pi without risk of damage.

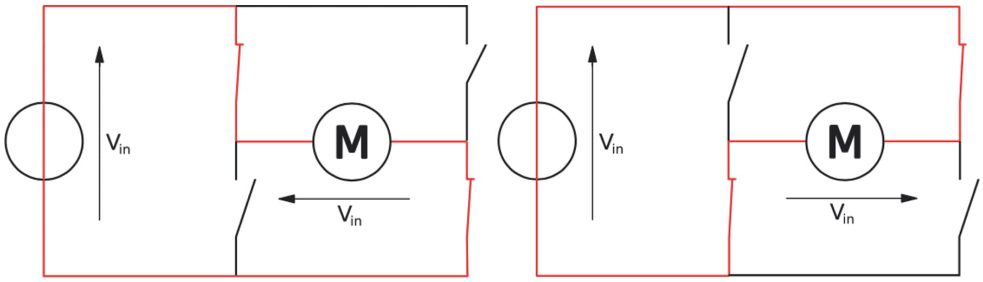



**Figure 1:** This shows how the H-bridge circuit switches polarity (Credit: Cyril Buttay)

## Prototyping

Before doing any irreversible work on your arm, we recommend you do some experimenting first as this will help you understand how everything works. We’re using a Raspberry Pi Pico W for our project. Choose one of the motors – the base is a good start. Using some jumper leads, connect its two wires to one of the motor drivers on an L298N. Hook up a 12V supply to the power terminals (but don’t switch on yet) and then connect EN1 to GPIO 2, IN1 to GPIO 3, and IN2 to GPIO 4. Finally, connect a wire between any GND (ground) pin on Raspberry Pi Pico to the GND terminal where you connected the power supply.

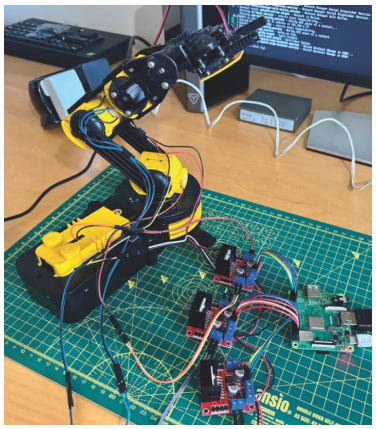
## First moves

Using USB cable, hook your Raspberry Pi Pico W up to another computer that can run Thonny IDE. Ensure the Pico has the MicroPython firmware loaded (**rpimag.co/micropython**). Thonny should be able to see the Pico and its file system. To help us drive the motors, we need the DCMotor library that you can download here: **rpimag.co/dcmotorpy**. Copy it to the Pico and then enter the code in the listing (or download it from **rpimag.co/owi535testpy**). Turn on your 12V supply and run the code. If all

is well, the base should rotate one way for half a second and then the other. If it doesn't, check all your wiring carefully.

## Preparing the arm

We'll come back to the code in the next part of this tutorial. For now, we're going to concentrate on making our project look great, so make sure everything is powered down. The OWI-535 comes with an elongated base designed to take D-cell batteries. That gives us some potential room to house our electronics, but not quite enough. To make everything fit, we decided to 3D-print an extension to the housing, making it tall enough to safely contain the three L298Ns and Raspberry Pi Pico. If you have access to a 3D printer, you can download the STL file here: **rpimag.co/owi535stl**. We then removed the circuit board and carefully removed the stands that held the batteries in place to create plenty of room. With the extra space provided by the 3D print, we are now able to wire up and install the electronics.



Prototyping and testing is an essential part of any project

## Full wire-up

The final part for this month requires a little patience. Following the example for one motor in **Figure 2**, wire everything up as shown in the mapping diagram (**Figure 3**). We recommend a Pico with pre-soldered headers to make connections easier. Every L298N needs to be wired up to the 12V supply and ground by chaining them together, then we need to

connect all the control lines to the GPIO pins of the Pico as shown. Finally, the motors need to be connected to the terminals. Use additional jumper cables between the terminals and the motors so they fit and can easily be changed around. Take your time and check everything again and again. Don't apply any power until you're sure it's OK.

## That's all for now

Next time we'll test everything out, install the circuitry, and then add software to take control of the robot arm. Even if you don't have this particular model, the same principle of driving motors using H-bridge circuits can be applied to many different projects.

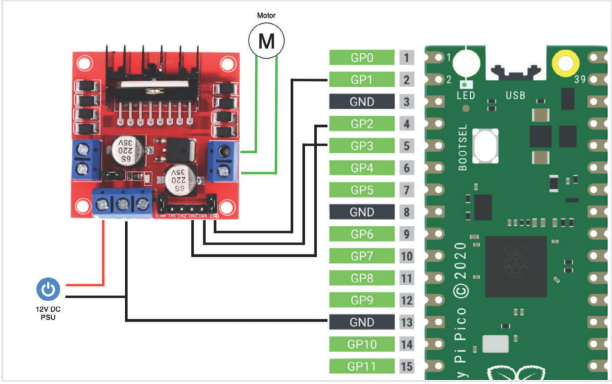



**Figure 2:** It's easier to show a single motor wire-up – repeat this for all five motors

> *Take your time and check everything again and again*

**DOWNLOAD THE FULL CODE:**

**rpimag.co/owi535testpy**

**owi535_test.py**

> Language: **MicroPython**

```
001.  from machine import Pin, PWM
002.  from time import sleep
003.  from dcmotor import DCMotor
004.
005.  # Pins (Motor 1 , Motor 2, Enable)
006.  motor_pins = [2,3,1]
007.  frequency = 1000
008.  speed = 10
009.
010.  motor = DCMotor(Pin(motor_pins[0], Pin.OUT),
      Pin(motor_pins[1], Pin.OUT),
      PWM(motor_pins[2], frequency))
011.
012.  print("Forward")
013. motor.forward(speed)
014. sleep(0.5)
015.
016.  print("Backward")
017.  motor.backward(speed)
018.  sleep(0.5)
019.
020.  print("Stop")
021.  motor.stop()
```

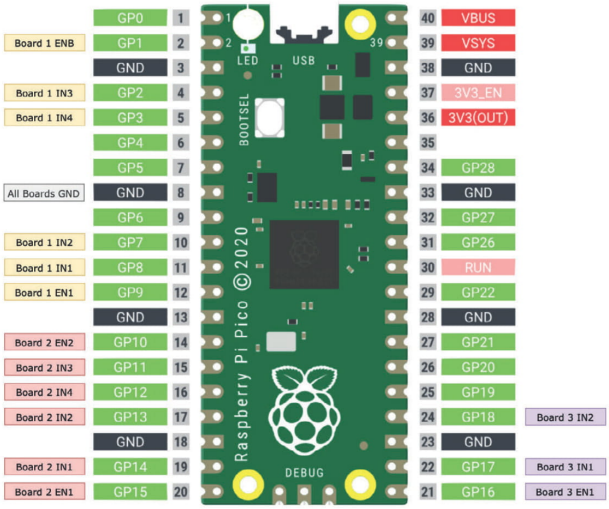



**Figure 3:** This maps the L298N control lines to the Raspberry Pi Pico W – start with one motor and test it

# Hack a robot arm

## Now the wiring is sorted, it’s time to assemble our robot arm and give it some smarts



**Maker**

**PJ Evans**

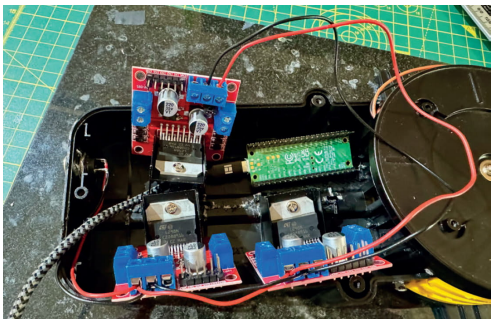
PJ is a writer, software engineer, and maker. The robot arm keeps knocking his coffee over and he’s worried it’s doing it on purpose.

**mastodon.social/@mrpjevans**

**Last time we started the process of upcycling the common OWI-535 robot arm into something a bit more internet connected.** There's still a little assembly to do and we're yet to tackle the software. Although this is a specific project, the examples given here can be transferred to a multitude of different robotics projects. The actual assembly of the robot arm is irrelevant, as the Raspberry Pi Pico has no idea about that; all it's doing is controlling a series of DC motors. So whatever you can think of that uses motors like these, you can automate with Raspberry Pi.

# Last time on robot arm hacking

Follow part one of this tutorial to set up your robotic arm. Hopefully, by now you've been able to wire up the five motors to your Raspberry Pi Pico W. Remember all the motors need to be connected to motor drivers on the H-bridge circuits, the Raspberry Pi Pico must be wired to the IN and EN pins (three per motor) and the 12V supply to the 12V IN on each H-bridge board. Finally, they need to share a common ground, connecting the Raspberry Pi Pico's GND pin to the GND for the PSU. Once you've checked all of this (again, and again) you're ready to continue. A multimeter is very useful here!

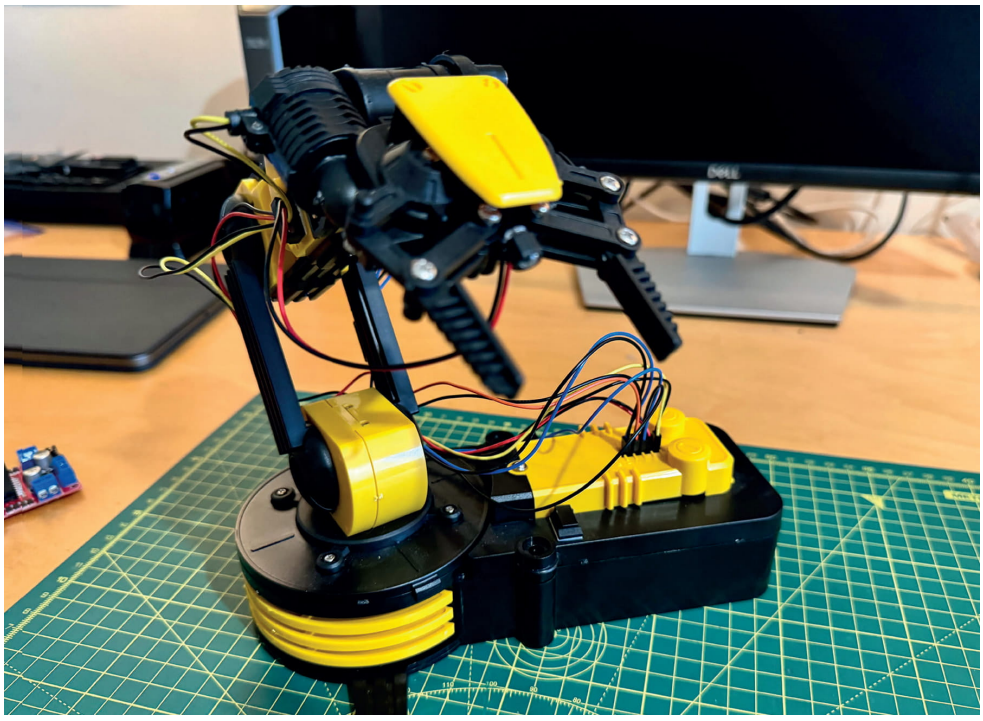


Placing the circuit boards takes time and patience (and a multimeter!)

# Check each motor

Using the MicroPython test script we used in the previous issue, change the wiring values to test each motor individually. If you followed the same

wiring pattern as us, this should be as listed in the code snippet overleaf. One-by-one, update the values and run the script. Each motor should make a small back-and-forth movement. Repeat until each motor is working, and make a note of which motor you have connected to which circuit (the order is not important). Starting from the bottom, we've named them the base, shoulder, elbow, wrist, and grip.



Our goal – an upcycled web-connected robot arm

## Software time

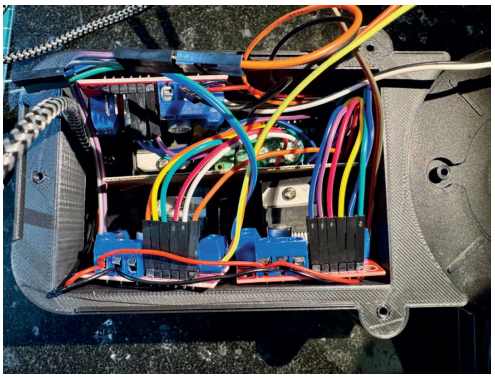
Before we assemble the arm, we should make sure the final software implementation is working. Take your time on this step, there's no rush to get all the circuits in place and it gives you space to debug both the electronics and the software (as you'll still have easy access to your Pico's USB port). In fact, there's no real need to assemble it all if you are not planning on a permanent installation: the arm will work just fine without. For now, make sure your Pico is hooked up to your computer and that the Thonny IDE can see it (see previous issue).

# Just add web

The easiest way to control the arm is using a simple web server. Raspberry Pi Pico W surprisingly has enough power to not only control the arm, but to run a little web server too, so you can use any web browser or even your phone to make the arm move around. We've prepared the server for you, so download the code from GitHub (**rpimag.co/owi535gpioweb**), or clone the whole **owi535gpio.git** repository:

```
git clone git@github.com:mrpjevans/owi535gpio.git
```

> *Control your robot arm from any web browser in your home, even your phone!*



Once fitted, insulate the boards with cardboard or plastic



Add a barrel connector for a neat finish

There are a few files to deal with. Firstly, our control software that runs the web server and handles the commands as they come in, the HTML template that controls the display, and the configuration file **WIFI_CONFIG.py** (you can reuse the same one). For now, you can run the web server in Thonny for testing. See the README if you need more help.

# Remote control

When the server starts, you should see Pico's IP address being used in Thonny's console window. Open up a web browser on your computer and enter **http://** followed by the IP address. You'll then see a simple web interface. Making sure your 12V PSU is powered up, then try pressing the buttons. It's possible they do not match the motors you have connected. You can either swap the motor wiring about or change the wiring assignments in the code to match; it's completely up to you. If everything is working as expected, you can now control your robot arm from any web browser in your home, even your phone.

# Final step

Feel free to tinker with the code. There's lots of scope for making it look better, or changing the default behaviour. You could even have a look at the included MQTT example to see how you could control the arm using an MQTT server over the internet (or even hook it up to Home Assistant!). When you've settled on an approach, the final step is to edit main.py and change it to read:

```
import owi535_web
```

Save **main.py** and transfer it to your Pico. Now the Pico will run the web software whenever power is applied, so it can run off a normal 5V power adapter (you will still need a 12V supply for the motors).

# Tidy-up time

When you are confident that everything is working, it's time to assemble the final project. If you've 3D-printed the extension module from last time (**rpimag.co/owi535stl**), you can place that on top of the main base. Remove all the circuitry in top section to create as much space as you can. The H-bridge circuits can be carefully placed vertically in the 'well' of the base with the Pico alongside, towards the arm itself. Please take your time doing this. Throwing it all in there and jumping up and down on it will almost certainly cause a short circuit. We also recommend placing a piece of cardboard or thin plastic in the centre so the opposing circuits do no touch. For a finishing touch, drill a hole (beforehand!) in the base to mount a barrel connector for your power supply. You can now add the original top of the base, feeding the wires through the hole in the centre.

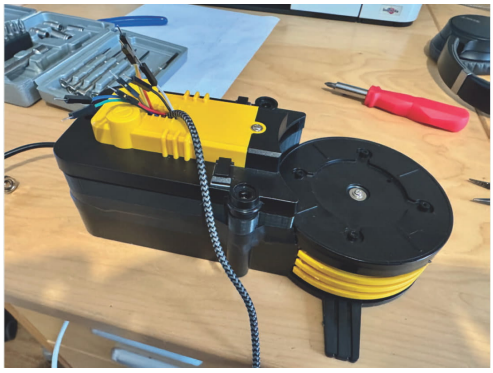
# Over to you

Congratulations, you have a web-connected robot arm, but that's only the beginning! This project is ripe for improvements. Here's some things to try.

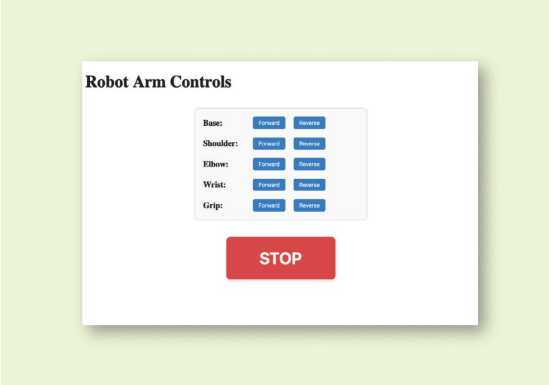
- How about adding timing controls to stop the arm when it travels too far in one direction?

- Better still, add microswitches that stop the arm in certain positions
- You could add cameras and use machine learning to detect objects
- With a more powerful Raspberry Pi, you could add full autonomous movement
- Use MQTT to extend control to other devices using Home Assistant

As ever, the possibilities are endless. Have fun!



The completed base can be hooked up to the arm. Pico needs its own power




A simple web interface, just waiting for customisation

**DOWNLOAD THE FULL CODE:**

**rpimag.co/owi535testpy**

## owi535_test_all.py

> Language: **MicroPython**

```
001.  from machine import Pin, PWM
002.  from time import sleep
003.
004.  from dcmotor import DCMotor
005.
006.  # Pins (Motor 1 , Motor 2, Enable)
007.  base = [2, 3, 1]
008.  shoulder = [7, 8, 9]
009.  elbow = [11, 12, 10]
010.  wrist = [13, 14, 15]
011.  grip = [17, 18, 16]
012.
013.  # Choose the motor to test
014.  motor_pins = base
015.
016.  frequency = 1000
017.  speed = 10
018.
019.  motor = DCMotor(Pin(motor_pins[0], Pin.OUT),
      Pin(motor_pins[1], Pin.OUT), PWM(motor_
      pins[2], frequency))
020.
021.  print("Forward")
022.  motor.forward(speed)
023.  sleep(0.5)
024.
025.  print("Backward")
026.  motor.backward(speed)
027.  sleep(0.5)
028.
029.  print("Stop")
030.  motor.stop()
```

OceanofPDF.com

# Extreme Raspberry Pi

Take Raspberry Pi to its very limits with these incredible projects.

By the pretty pedestrian **Rob Zwetsloot**

**Raspberry Pi is a very powerful, very small, very customisable device and we have seen it be used for so many things over the years because of this.** Whether folks are slipping them into 3D-printed classic console cases or simply hiding them away as home file servers, we've covered many of the cool things the community have done with Raspberry Pi.

Which begs the question – how can you push the limits with a Raspberry Pi? Well, with 13 years of projects and Raspberry Pi models, you'll be surprised just how far/ fast/high/deep Raspberry Pi has gone. We're taking Raspberry Pi to the extreme.

**Have a more extreme use of Raspberry Pi? An updated record? Email us at magazine@raspberrypi.com**

# Longest-running Raspberry Pi

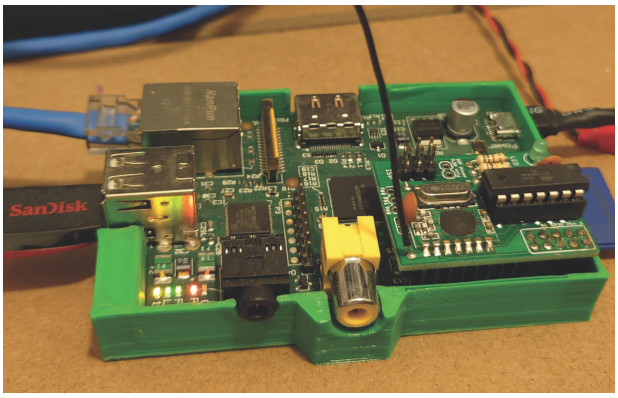
## 2331 days of uptime for the Model B that could

## rpimag.co/uptime

“This Raspberry Pi has been running for ten years,” says Reddit user KerazyPete, although not without a few reboots. At the time of posting this in late 2023, his Raspberry Pi 1 Model B (revision 0002 with 256MB of RAM no less, one of the first production versions) had been ‘up’ for 2331 consecutive days – since July 2017.

It’s the longest uptime we’ve seen for a Raspberry Pi – in fact, it's pretty impressive with occasional power cuts, power surges, and accidental unplugs. Our file servers and media centres have undergone hardware upgrades as well.

You can easily check how long your Raspberry Pi has been on yourself by opening a terminal and typing uprecords – maybe you’ll be surprised. Although we do recommend upgrading the operating system on your Raspberry Pi whenever a new OS gets released – the security updates can be very important.




The system in question. The original Raspberry Pi 1 design is quite nostalgic

```
pi@node-pi ~ $ uprecords
     #               Uptime | System                                     Boot up
----------------------------+---------------------------------------------------
->   1   1389 days, 14:22:4 | Linux 4.1.17+            Sun Feb  9 16:58:36 2020
     2    531 days, 01:18:11 | Linux 4.1.17+            Tue Jul 11 22:51:32 2017
     3    371 days, 17:17:30 | Linux 4.1.17+            Sat Feb  2 23:38:37 2019
     4     39 days, 10:36:09 | Linux 4.1.17+            Mon Dec 24 22:16:50 2018
----------------------------+---------------------------------------------------
NewRec   858 days, 13:04:31 | since                    Sat Jul 24 19:16:46 2021
    up  2331 days, 19:34:3 | since                    Tue Jul 11 22:51:32 2017
  down     0 days, 13:55:14 | since                    Tue Jul 11 22:51:32 2017
   %up              99.975 | since                    Tue Jul 11 22:51:32 2017
pi@node-pi ~ $ sudo tune2fs -l /dev/root | grep 'Filesystem created:'
Filesystem created:       Sat Nov 30 20:00:32 2013
pi@node-pi ~ $ cat /proc/cpuinfo | grep 'Revision'
```

The uprecords command shows how long the system has been running

# Largest
## Raspberry Pi

Jumbo working Raspberry Pi 3

**rpimag.co/largest**

There are a few huge, working Raspberry Pi computers. A 10× Raspberry Pi 3 was shown off at Maker Faire Bay Area in 2016, and Raspberry Pi's own Toby Roberts built a 6× Raspberry Pi 4 for a little exhibition in the shopping centre where the official Raspberry Pi Store resides. This 12× working Raspberry Pi takes the crown, though, designed and 3D-printed by Zach Hipps.

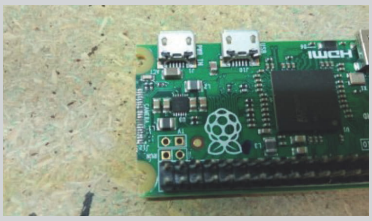
The PCB is made from plywood, the GPIO pins are made from aluminium tubing, and over 5kg of PLA filament was used for the other large-scale components. "I connected my Raspberry Pi to all the large-scale connectors with extension cables," Zach told us when we spoke to him a few years ago. "I plugged in a monitor and keyboard and everything fired right up!"

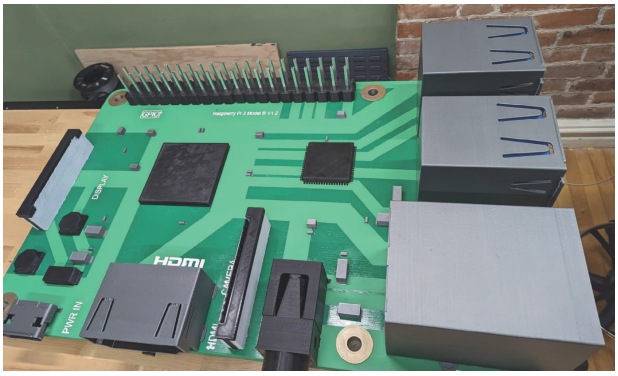


Raspberry Pi 3; human being for scale

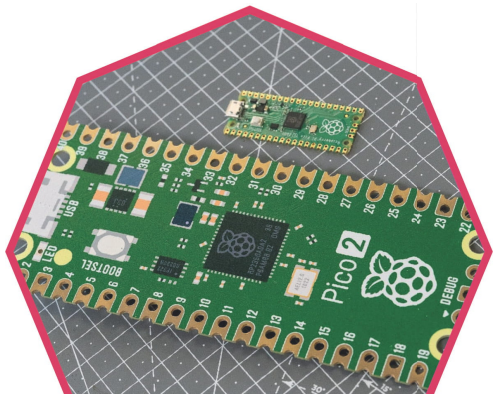
# Smallest Raspberry Pi

While some tiny RP2040/RP2350 boards are technically the smallest Raspberry Pi products, the smallest standard Raspberry Pi computer is Raspberry Pi Zero. However, you can make it smaller still – 5mm smaller as it goes. Raspberry Pi Zero v1.3 has a camera connector which can be removed from the board and, with no circuits in that section of the board, you can trim it. Check out the forum post about it: **rpimag.co/smallest**.






Most Raspberry Pi devices are a little less conspicuous than this

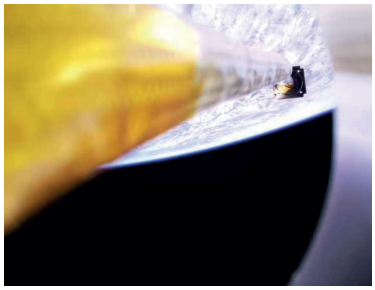



The Pico Jumbo is an adapter for Raspberry Pi Pico 2 that makes it 3.5× larger: **rpimag.co/picojumbo**

# Pi in the Sky

During the dawn of Raspberry Pi, and for many years afterwards, high-altitude balloonist Dave Akerman would regularly send balloons about 40km into the mesophere (which is above the stratosphere, Queen fans) with a Raspberry Pi attached to take photos and other telemetry. In 2016 he broke the world record for the highest live image sent down from an amateur balloon, at 41,837 metres. That probably makes it the highest a Raspberry Pi has gone without the use of a rocket.





A successful mission photo




CubeSats are very unassuming, but this one did go into space with a Raspberry Pi aboard

# Highest Raspberry Pi

# GASPACS

## 428km over the Earth

## rpimag.co/highest

Is it cheating to go to space so your Raspberry Pi can claim the altitude record for the computer? Not when there are several Raspberry Pi computers in space – the current highest are the Astro Pi boxes up on the International Space Station. However, GASPACS beat their record by a mere six kilometres on its 117-day mission in 2022.

The GASPACS (Get Away Special Passive Attitude Control Satellite) was a 1U CubeSat built by students at Utah State University to test aerobraking with an inflatable 'AeroBoom'. Since the Earth is not a perfect sphere, the CubeSat's orbital altitude was 416km (258 miles) at its perigee, and 428km (267 miles) at its apogee.




The first pair of Astro Pi boxes went up for Tim Peake's mission

# Fastest Raspberry Pi

## Astro Pi

## 17,100mph is easier with no friction

## astro-pi.org

While Astro Pi on the ISS doesn't hold the altitude record for a Raspberry Pi, due to the peculiarities of orbital physics it is the fastest. Very basically, having a lower orbit means you need to go faster so that you don't fall to Earth.

The ISS is clocked at 17,100mph (27,520km/h) – that's 4.77 miles per second (7.67km/s), and 22.5 times the speed of sound at ground level. Those numbers are difficult to conceptualise, but it orbits the Earth in just shy of 93 minutes, meaning it orbits 15.5 times a day. Pretty quick!

Astro Pi in orbit has various sensors thanks to a Sense HAT, including motion sensors, which are used by schoolchildren to run experiments via code. Unlike GASPACS, these Raspberry Pi computers are specially hardened for the extended period of time in space.




A humble Raspberry Pi living on the ISS

# Deepest Raspberry Pi

## Maka Niu

## 1500m underwater – and possibly able to do 6000m – is no mean feat

## rpimag.co/deepest

Going deep underwater is hard. The deeper you go, the more the weight of the water above you becomes a huge issue, and human-made devices need to withstand the pressure in the depths. It's why your watch may only be rated to 100m underwater – stuff will then start to break.

A descent of 1500 metres – nearly a mile underwater – is very, very far. At this depth in salt water, pressure is 148 atmospheres, or 148 times normal air pressure at sea level.

Maka Niu is a very special system then, able to safely contain a Raspberry Pi Zero and Camera Module V2 in a low-cost device – opening up citizen science to more folks, and helping to explore the largely unknown deep sea.



It may just be a tube, but it's a very sturdy one



They're called sea urchins because hedgehogs used to be called urchins

# Raspberry Pi submersible at 300m depth

While unable to make it to depths of 1500m, this pilotable Raspberry Pi craft can go to 300m, which is still very impressive. The price tag of $4250 (•3288) is a little eye-watering, but it's at least a factor of ten cheaper than other commercial ROVs of its type. Find out more at **bluerobotics.com**.



# Highest clock speed

## Reaching a limit of 3.6GHz

## rpimag.co/fastesthz

‘Fastest Raspberry Pi’ can really mean two things, which is why we’ve awkwardly titled this ‘highest clock speed’. Every new numbered model of Raspberry Pi is slightly faster than the last, but the question for some people is: how fast?

Overclocking computer hardware – the process of telling it to run faster than it was designed to, i.e. increase its clock speed – is an age-old tradition amongst tech nerds. The caveat is, this tends to make the hardware hotter and can damage it in the long-run. In normal overclock scenarios, this is solved with liquid cooling or other cooling solutions, but if you want to take a chip to its absolute limits, you need something really cold: liquid nitrogen.

With a specialised tube over Raspberry Pi 5, Pieter-Jan Plaisier poured liquid nitrogen directly onto the chip as it ran at 3.6GHz. Can it go higher? Supposedly not – trying at 3.7 GHz the system crashed, and not because of the heat.



**Warning!**

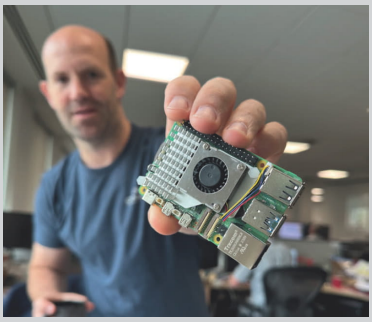
**Nitrogen**

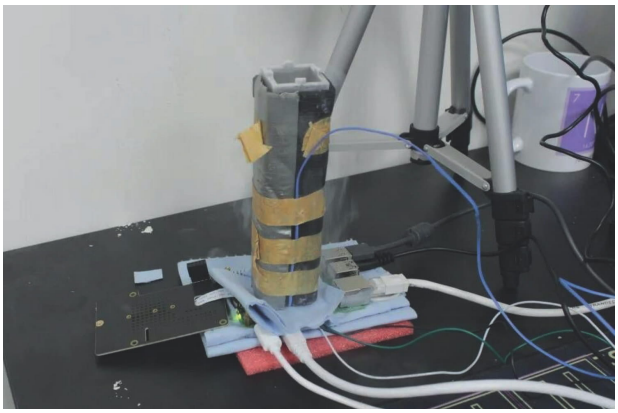
Be careful when handling liquid nitrogen. Wear protective clothing and eyewear and follow all instructions. Work in a well-ventilated area.

**rpimag.co/nitrosafety**

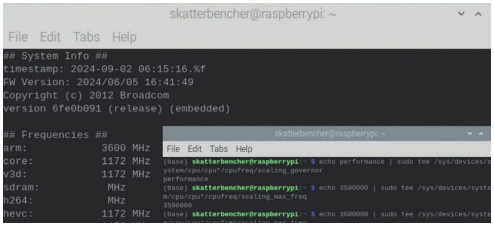
# Hottest Raspberry Pi?

What is the hottest environment a Raspberry Pi operates in? Unfortunately, we couldn't find an answer – some have operated in deserts that regularly clock 40°C (104°F). The real trick here is that with good enough cooling, you can have a Raspberry Pi operate in very high temperature environments, and the BCM2712 chip that runs Raspberry Pi will operate at temperatures as high as 80–85°C (176–185°F) before throttling is enabled.





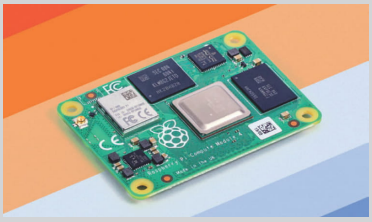
It works but it's not very practical – it's just a test after all




Proof from the terminal of the amazing feat

# Raspberry Pi operating temperatures

Recently, Compute Module 4 got an extended temperature variant, giving it an even lower operating temperature of -40°C (-40°F). Many parts of the world, not just Antarctica, can get to below -20°C (-4°F), which is the standard lowest operating range of other Raspberry Pi products. 85°C/185°F is the highest operating temperature of Raspberry Pi.





The photos were used for time lapses



The solar-powered station taking photos of some feathered friends

**Coldest Raspberry Pi**

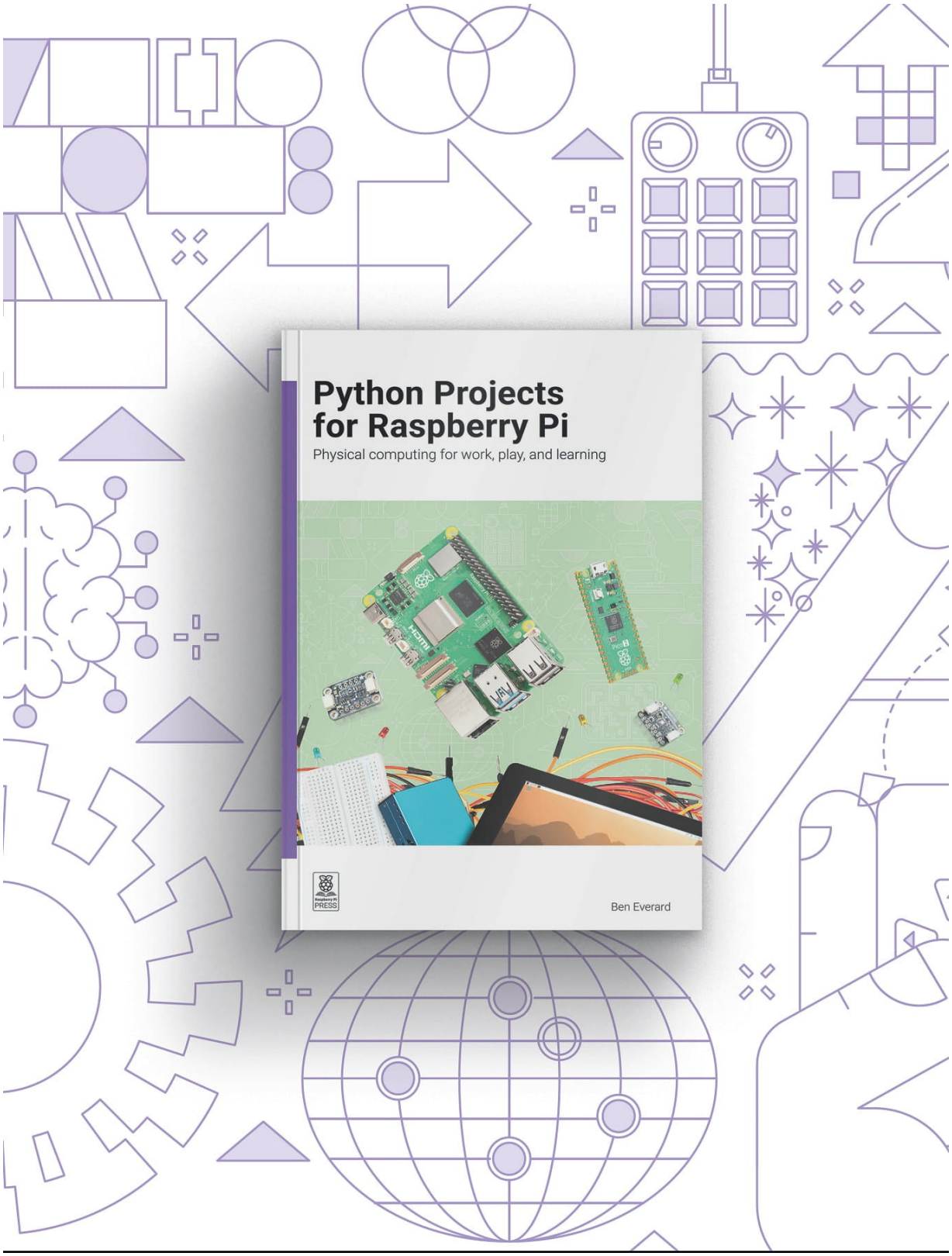
Arribada Penguin Monitoring

It gets to -60°C in the Antarctic winters

## rpimag.co/coldest

Sending any equipment to Antarctica is tricky – it gets very, very cold there. When your tech accidentally stays there for three years, and remains working, that is quite the feat. The Arribada Penguin Monitoring project (accidentally) managed that, being unable to pick up the camera at the end of 2019 and then, well, the world shut down in 2020 during the start of the Covid-19 pandemic.

While the temperature regularly gets below -30°C (-22°F), it can reach -60°C (-76°F) in the winter. Antarctica is very large though and depending on the placement of the camera, the temperature will be different. Having said that, after three years in the frigid wasteland, the Arribada camera returned safely with 32,764 photos for the researchers to go through.


Python Projects
for Raspberry Pi
Physical computing for work, play, and learning
Ben Everard

# Python Projects for **Raspberry Pi**

Physical computing for work, play, and learning

Computers are embedded into almost everything we own. Our doorbells, kitchen gadgets, vacuum cleaners, and media players are all powerful computing devices running software that someone has written.

Using the flexible Python programming language, *Python Projects for Raspberry Pi* shows you how to get the most out of the Pico range of microcontrollers and Raspberry Pi computers like Raspberry Pi 5.

- ***Learn how to:***
  - *Interact with sensors and motors*
  - *Build a weather station*
  - *Make your own games console*
  - *Add web interfaces to your projects*
  - *Recognise gestures with machine learning*
  - *Get the most out of your Raspberry Pi Pico with MicroPython*
  - *Extend Python with C*

BUY ONLINE: **rpimag.co/pythonprojectsbook**

[OceanofPDF.com](OceanofPDF.com)

# Reviews and Resources

**Argon ONE V5**
A solid case, literally

**Pironman 5 Mini**
Mini desktop case

**RP2350 Pico W5**
5GHz Wi-Fi-enabled

**Dream HAT+**
Radar on Raspberry Pi

**CrowPi 3**
Electronics learning lab

**Adafruit Fruit Jam**
Pico as a computer

**10 amazing airborne projects**
Take to the skies

**10 amazing astronomy projects**
Look to the stars

**10 amazing Raspberry Pi garden projects**
Down in the allotment

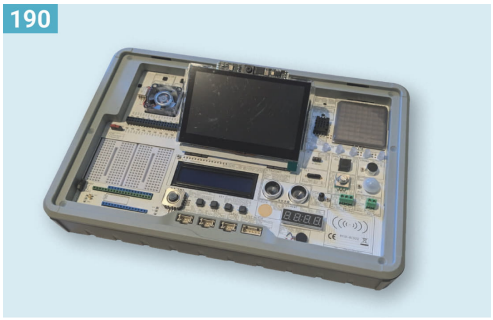
**10 amazing Raspberry Pi cyberdecks**

[Hack the planet](#)





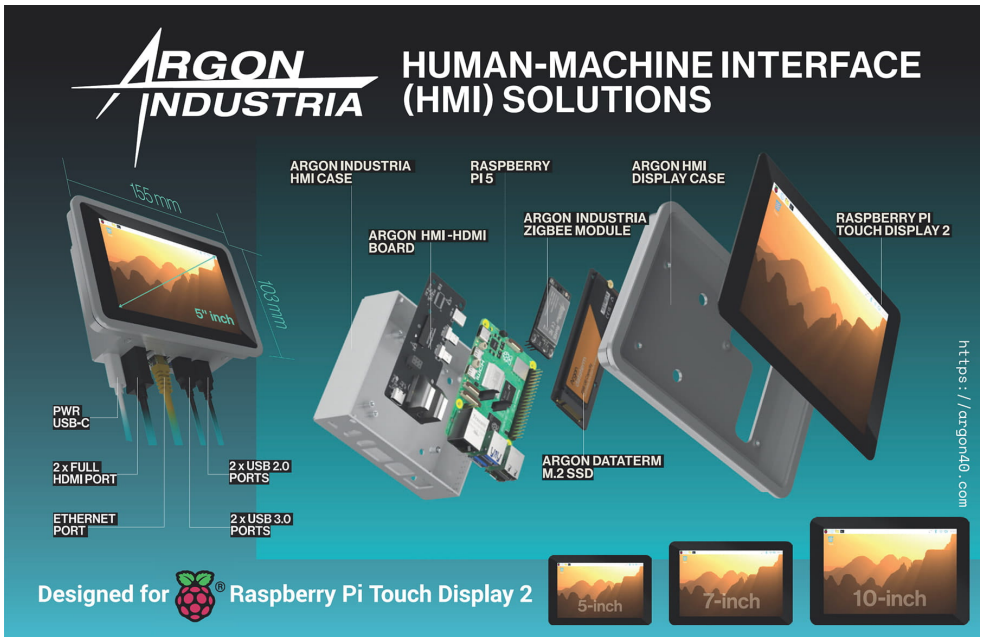













OceanofPDF.com

# Argon ONE V5

## Another month, another case. As **PJ Evans** finds out, this one means business

Argon 40 [rpimag.co/argononev5](rpimag.co/argononev5) From £28 / $35

**SPECS**

**PORTS:**
6 × USB, USB-C, 2 × full-size HDMI, 3.5 mm audio

**ADD-ONS:**
Up to 2 × NMVe M.2 SSD (2230, 2242, 2260, 2280), Zigbee module with antenna, OLED screen, UPS

**COOLING:**
Passive aluminium, active 30 mm PWM fan

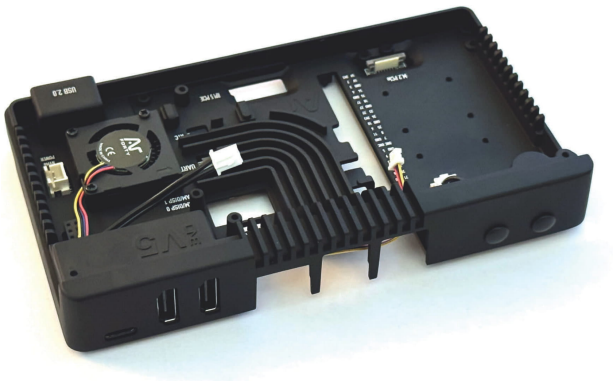


A much more industrial design from Argon

**Argon 40 is not messing about.** After an eyebrow-raising detour into the budget case market with its perfectly serviceable Poly+ range, it is back to what it is known for with the latest in its premier ONE range.

There has, however, been a shift in approach. Whereas the original ONE and a Raspberry Pi 5-compatible V3 version sported a sleek sports car-like styling, the new V5 is an absolute unit. Its roots can be found in the industrial design of the NEO range, and this is clearly an evolution of that style. So what makes this new case so different? Let's start with the basics.

The first thing that strikes you is the sheer size. This is not a small case, and that's because it has a few tricks under its tough exterior. Argon is clearly intending this case to be your powerhouse server. Removing the solid aluminium lid, we find space for your Raspberry Pi 5 along with some impressive passive heatsinking and a mounted PWM 30 mm fan. On-board ports – including the GPIO header – are exposed within the housing for projects and there's a few millimetres of clearance for adding your own bits and pieces. Argon's signature daughterboard design is here too, with a PCB that plugs into the USB-C and HDMI ports of your Raspberry Pi, breaking them out to full-size HDMI and an additional two front-mounted USB-A connectors. Slotting it all together onto the plastic base creates a solid unit for your next project.



There is excellent heat dissipation and port access inside

## Features and upgrade options

There is a very good reason this case is so much larger than standard. Alongside the Raspberry Pi, there is space for two M.2 NVMe slots for up

to full-sized 2280 solid-state drives. This case then becomes a NAS with SSD RAID capability. You can purchase the add-on PCBs for each M.2 separately or pre-mounted, offering flexibility on price and future upgrade options. Then, nestling next to the front-mounted USB ports, there's an audio-out port. This case comes with a DAC built-in, which upgrades it further to a potential media server.

Home automation enthusiasts may be tempted by the optional Zigbee add-on which plugs into the daughterboard and comes complete with an antenna which mounts to the case externally. If you're using the case headless and want to see some info at a glance, the £9 OLED module adds a tiny screen to the top of the case for you to use as you wish. Finally, another option is to add a dedicated uninterruptible power supply (UPS), which sits neatly under the case. Argon has created an entire ecosystem around the simple concept of a case.

> *Argon is clearly intending this case to be for your powerhouse server*



All the usual rear ports plus full-size HDMI and antenna mount points

We were supplied with a manual, thermal pads for the various Raspberry Pi chips, and additional thermal pads for protecting the M.2 SSDs. Assembly was easy and the result was formidable. It may not look as pretty as its predecessors on your living room cabinet, but you'll probably be able to drop it from some height without causing it much damage.

## Verdict

If you are looking to build a home server for media, automation, or just data storage, this is a well-priced option that will keep your Raspberry Pi cool and protected with a wealth of upgrade options.

**10**/10

OceanofPDF.com

# Pironman 5 Mini

## Compact and bijou, this Raspberry Pi 5 case looks cool. By **Phil King**

Sunfounder  rpimag.co/pironman5mini  £34 / $46

**SPECS**

**FEATURES:**
HAT+ with M.2 SSD slot, RGB fan, PWM active cooler, power button, RTC battery

**DIMENSIONS:**
67.8 × 98.6 × 101.5mm

## Verdict

While not quite as eye-catching or feature-packed as its siblings, it's a quality case with effective cooling.

**8**/10



The partly assembled case with its power button connected to the HAT+

**We reviewed the original Pironman 5 tower PC-style case for Raspberry Pi 5 back in issue 145 and were very impressed by its looks and cooling performance.** Since then, it has been joined by a Max version with room for two NVMe SSDs, and now the Pironman 5 Mini.

As the name suggests, this is a more compact case – with a volume of 0.68 litres vs its siblings' 1.03 litres. This, along with a lower price point, has resulted in several compromises such as the lack of a mini OLED and a single RGB fan rather than two. The tower cooler has also been replaced with an (unofficial) active cooler that sits on Raspberry Pi 5 via thermal pads. Even so, the combination of fans provides good airflow and cooling.

> *The GPIO header is rerouted to the side via the supplied HAT+ board*

## Wear a HAT

There are case cutouts for Raspberry Pi 5's main ports (and camera cable), rather than them being rerouted via an adapter board – so no full-size

HDMI ports. The GPIO header is rerouted to the side via the supplied HAT+ board, which also houses an RTC battery and optional M.2 NVMe SSD, and has connections for the RGB fan and a power button on the case exterior.

With the case itself comprising just a single square aluminium piece and two clear acrylic panels for the sides, assembly is fairly straightforward, aided by good online documentation. As with the other models, installing the Pironman 5 software enables control of the RGB fan/lighting settings, safe shutdown with the power button, and system monitoring via a web dashboard.



The RGB-lit case looks cool and the rerouted GPIO header is clearly labelled

OceanofPDF.com

# RP2350 Pico W5

## It’s Raspberry Pi Pico 2, but with a lot more memory. By **Rob Zwetsloot**

Elecrow [rpimag.co/picow5](rpimag.co/picow5) £5 / $7

**SPECS**

**DIMENSIONS:**
51mm × 21mm, same as Raspberry Pi Pico

**CONNECTIVITY:**
USB Type-C, wireless LAN 802.11a/b/g/n 2.4GHz/5GHz, Bluetooth, 40-pin GPIO

**STORAGE:**
8MB flash

## Verdict

A great alternative to your standard Raspberry Pico 2 W at the same price point, but you can’t quite slip it into every Pico-specific use-case – just most of them.

**9**/10

**In order to ensure maximum compatibility with Raspberry Pi Pico use-cases, Pico 2 and the rest of its line was shipped with a USB Micro B port for connectivity and power.** As the chip that powers Pico 2, RP2350,

is available for anyone to use as they see fit, one of the most common alternatives we see for Pico 2 is a board with USB-C for power and data.

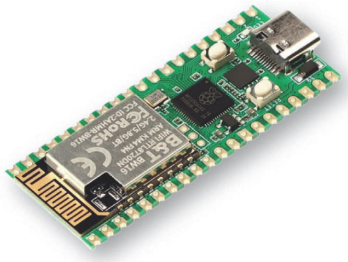
Elecrow has decided to go a little further, and a little different, by creating a version of Pico 2 that not only has a USB-C port, but also has a bump in the amount of storage it uses – doubling the 4MB of onboard storage to a whopping 8MB. Due to the way Pico is used as a microcontroller, you don't need much storage; however, it's always nice to have a slightly larger limit to work to.

> *With RP2350 powering it, all code is compatible*

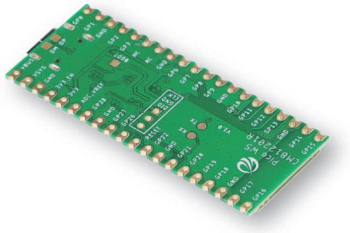
## Chunky wireless

This Pico 2 alternative uses a different wireless chip, too. While the standard Pico 2 W adds a noticeable chip to the board for wireless radio, the version here is a fair bit bigger – although it does offer 5GHz connectivity. So while it keeps just about the same dimensions as your standard Pico 2, the extra bulk on the PCB is something to bear in mind.

Otherwise it works like a Pico. With RP2350 powering it, all code is compatible and so is any hardware you can connect to it. W5 also comes with a handy reset button, which we always like to see. It's definitely an elevated version of Pico 2, and with an excellent price point to match.



Even the PCB is green, making it look very similar to a Pico 2



The pins are the same as on a regular Pico; pads are accessible on the rear

OceanofPDF.com

# DreamHAT+

## Discover high-frequency mmWave radar using the Infineon BGT60TR13C. By **Lucy Hattersley**

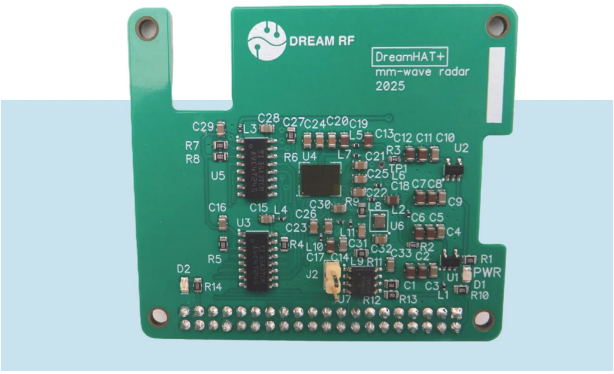
Dream RF  [dreamboards.co.uk](dreamboards.co.uk)  £100 / $135

**SPECS**

**OPERATION:**
Frequency at 58–63.5GHz; Bandwidth at 5GHz

**RANGE:**
Distance: 0.1–15m; Resolution: 3cm, Field of view: 40° horizontal, 65° vertical

**INTEGRATED ANTENNAS:**
1 × TX antenna, 3 × RX antennas, 5dBi (max) gain




The DreamHAT+ board with the Infineon BGT60TR13C

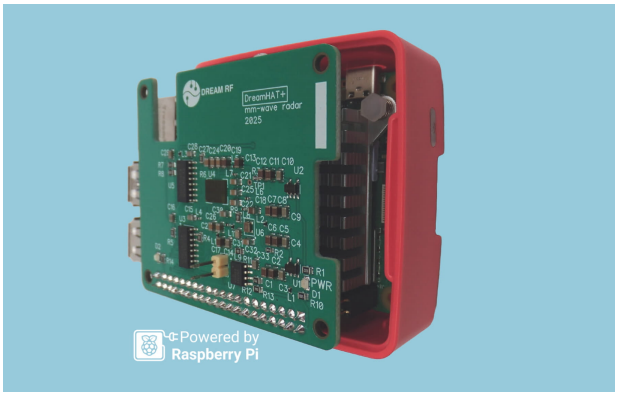
**DreamHAT+ is a Hardware Attached on Top (HAT) kit that gives Raspberry Pi a 60GHz millimetre-wave radar for you to play with.**

The full-size HAT packs an Infineon BGT60TR13C chip featuring a single transmitter antenna and three receivers (TX/RX), connecting to Raspberry Pi 4 or 5 via GPIO pins and SPI.

Dream RF also manufactures ApRESS (Autonomous Phase-sensitive Radio Echo Sounder), a device that uses Frequency-Modulated Continuous Wave (FMCW), as used to monitor ice loss in Greenland. While that device operated on the 2–8GHz frequency, DreamHAT+ is operating on a much shorter wavelength: 60GHz. This gives it a much higher resolution, making it ideal for human detection and gesture recognition-type projects.

As well as using it to research personal radar (a perfectly fun and relatively low-key technology), it can be used to add situational awareness to robotics, security systems, and wildlife monitors. There are also health and industrial uses (mostly involving monitoring breathing patterns or equipment vibrations).

The latter use case is beyond our scope, but we had a lot of fun experimenting with DreamHAT+.




DreamHAT+ assembled inside a case. We found it best positioned horizontally for testing

## Get started

Setup – like most HATs - was relatively simple. We connected DreamHAT+ to a test Raspberry Pi 5 and set up the software. Dream RF supplied us with a preconfigured microSD card, which you can flash yourself using the link on Dream RF's GitHub (**rpimag.co/dreamhatgit**).

You can also start from a Raspberry Pi OS installation and follow the instructions to install the various packages and download the example projects. We recreated both approaches and found flashing the microSD card easier.

On the resources section of the DreamBoards site (**dreamboards.co.uk**) is a setup guide, product brief, and schematics. You can also look at some real-world examples on Dream RF's YouTube channel (**rpimag.co/dreamrfvid**). There are currently the 2D Cartesian Plot and Range Doppler examples. It's a good idea to read the documentation and watch the videos first so you get an understanding of the screen output.

## Software stack

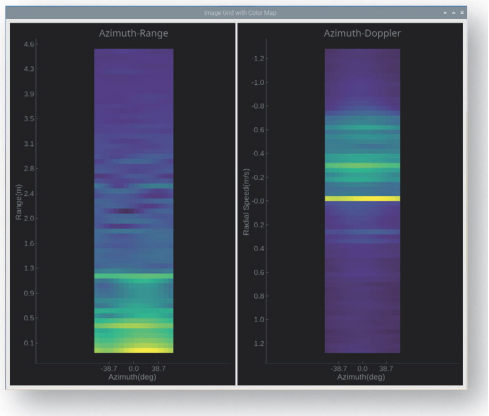
There are a range of example projects in the GitHub repository. These include a range-Doppler plot (the classic radar visualisation), an xy plot with tracking and persistence, a Doppler-azimuth plot (that detects objects around the device), and an offline processing example.

Beyond that, you can look at the Python modules to start your own implementations:

- **BGT60TR13C.py**: This module manages SPI communication with the Infineon BGT60TR13C chip.
- **udp_streaming.py**: This streams radar data over User Datagram Protocol (UDP).
- **gui.py**: Provides a graphical user interface for visualising radar data, including range-Doppler plots and heatmaps, facilitating intuitive analysis.
- **offline_processing.py**: For post-processing of recorded radar data.

While these modules provide a solid Python-based application programming interface API, the documentation around the API isn't as solid as the setup. "This is more than enough to get you started and on the way to creating your own unique 60GHz radar project!" states the GitHub page, somewhat optimistically. Your mileage may vary.

> *We had a lot of fun experimenting with DreamHAT+*




The DreamHAT+ Doppler-azimuth script

## Verdict

> We thoroughly enjoyed our time experimenting with DreamHAT+ and the Infineon BGT60TR13C. The hardware and example projects are first-rate. The documentation is in development and a more detailed API reference guide would be helpful for moving beyond the initial projects.
>
> **8**/10

OceanofPDF.com

# CrowPi 3

## An electronics playground with added AI lessons for Raspberry Pi, Pico, and more. **Rob Zwetsloot** gets stuck in

Elecrow [rpimag.co/crowpi3](rpimag.co/crowpi3) £168 / $229

**SPECS**

**DIMENSIONS**
285mm × 185mm × 38mm

**SCREEN**
4.3-inch 800×480 IPS screen

**COMPATIBILITY**
Raspberry Pi 5, Raspberry Pi Pico (all models), Arduino Nano, micro:bit




A fairly packed electronic lab for people learning to code and make

**Educational and electronics starter kits for Raspberry Pi have been around since, well, Raspberry Pi has existed.** We've covered a large variety over the years – from small kits that cover specific use cases, to a newer breed of portable electronics labs. These latter types are always fascinating, cramming as many different functions as possible into a briefcase-shaped computer that looks a little like a prop from a 1990s hacker movie. Unlike its laptop-esque predecessors, CrowPi 2 and L, the CrowPi 3 returns to this lab form factor with more built-in components than you can shake a Sense HAT at.

The really clever thing is, once you install a Raspberry Pi 5 and load up the included OS, you are presented with an interface to start learning how to use these components – and this is where the new flagship feature comes in: working with AI.

## Educational intelligence

These AI-based lessons allow you to make use of LLMs (large language models), voice commands with the built-in microphone, image recognition using computer vision on the built-in camera, and have it connect all of this up to the components on the board. Some of the example lessons will enable you to control some of the included LEDs with your voice as well as tracking a ping-pong ball or similar in your hand across the included screen.



A successful Kickstarter campaign for CrowPi 3 just finished

> *It all works well, with very clear and useful labelling on all the installed components*

These pair well with the multitude of lessons on the various components and how to use them with, for example, Python for Raspberry Pi. The AI examples can then be modified and grown using these lessons, making it a great way to extend the uses of this playground. Elecrow says there are over 180 individual lessons at the moment, and the number seems to be growing every time we talk to the firm as well.

## Hands-on learning

Setting up the whole system is extremely easy – you basically just need to plug in a Raspberry Pi 5 – and you'll be ready to go in about ten minutes. We tested the basic kit, which doesn't come with any batteries; however, you can power it with just the provided PSU, and easily add the correct batteries later to make it portable. All the physical components and doodads are supported by a decent-sized bag of extra components for use in the breadboard, motor connectors, etc. and really add to the tactile feel of making something on the system.



There are a ton of additional components that come in the box

Teaching ‘Hello world!’ to people is all well and good, but making your Python code control some lights or beep faster if your hand is closer to an ultrasonic sensor gives much better feedback and reward. And it all works well, with very clear and useful labelling on all the installed components so you can add or remove them from your code as you please, even beyond the pre-installed lessons.

It’s a very cool piece of kit which should keep many curious folks busy for hours and days, and impart some genuinely useful making and coding skills as they go.

## Verdict

A very complete learning kit for people wanting to get into coding and making, and also find out what all the fuss is about with the current AI trends in computing.

**10**/10

*OceanofPDF.com*

# Adafruit Fruit Jam

## If you use it like a computer and program it like a computer, is it a computer? By **Ben Everard**

Adafruit rpimag.co/fruitjam $39.99

**SPECS**

**DIMENSIONS:**
86 × 54 mm

**STORAGE:**
16MB flash, 8MB PSRAM, SD card slot

**I/O:**
DVI port, STEMMA QT, STEMMA Classic, EYESPI, 5 × NeoPixels, 3 × buttons, Wi-Fi adapter

## Verdict

Adafruit's Fruit Jam is well thought out on both the software and hardware sides, to deliver a machine that's easy and fun to hack on.

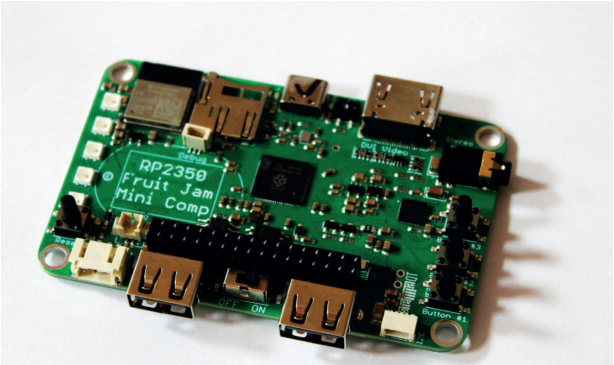
**10**/10

**What is a computer? What is a microcontroller? And where do we draw the line between the two?** There are some technicalities that can try to point us in one direction or another, such as whether or not it includes a Memory Management Unit (or MMU). However, this sort of technical distinction misses the point. We're not really interested in dictionary definitions here, but more the question of what the word 'computer' means to you.

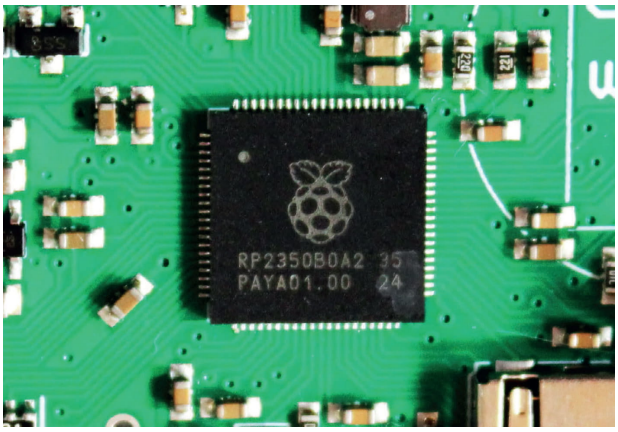
A computer – to most people – is something with a screen, and at least one user input device such as a mouse, keyboard or touchpad, that runs a range of interactive software. Take any feature of that away and you have something that we'd probably call a microcontroller.

The Adafruit Fruit Jam, then, is a computer based on an RP2350. However, the RP2350 is usually considered a microcontroller. If this all sounds a bit contradictory, it's because the Adafruit Fruit Jam only just makes it through our definition of computer by cramming in extra hardware and squeezing all the performance it can out of the RP2350.

The two features that make it feel most like a computer are the HDMI output (technically a DVI port that works with an HDMI cable) and USB input, both provided by RP2350 using HSTX and PIO respectively. This is coupled with extra RAM and an SD card slot for storage to allow the software to behave something a bit like a late 1980s or early 1990s computer where you can use a mouse to launch different bits of software from a menu.




At approximately the size of a business card, the Fruit Jam squeezes a lot into a small space




The Adafruit Fruit Jam uses the larger RP2350B package which has 48 GPIOs (as opposed to 30 on the RP2350A package)

## Expansion options

There are also a few features that you're more likely to find on a microcontroller: STEMMA QT and STEMMA Classic ports are included, both of which let you connect sensors and other hardware. There's an EYESPI-compatible GPIO header that lets you add a TFT screen if you don't want to use the DVI port. In other words, it's easy to expand with more hardware.

Why, in a world where there are 2025's computers, would you want a microcontroller dressed up akin to a computer from 30 years ago? There are three things that we think the Adafruit Fruit Jam really excels at…

> *Given the hardware, we'd expect to see some emulators of classic computers popping up*

**Nostalgia** – keeping retro computers running takes work and they can take up a lot of space as they're often specific about which keyboards and monitors they accept. The Adafruit Fruit Jam gives a similar-ish experience, but lets you use modern hardware. For many people, they'll already have all the peripherals and just need the board.

**Understanding** – This is arguably the most understandable computer on the market today. It's completely conceivable for a reasonably technical

person to fully understand how this works, and that's probably not possible with any other machine that has HDMI output and USB input. It's not simple – a huge amount of work has gone into making this possible – but it is limited in scope and open source.

**Fun** – This is an easy computer to mess around with. There's not much to go wrong, it's all Python, and you can very quickly re-flash it if it all goes wrong. There's a low bar to tinkering with it, yet still plenty of power in there to do interesting things.

These are perhaps all slightly different ways at looking at the same reason. It's a hackable little computer that's easy to mess about with.

## Programming and OS features

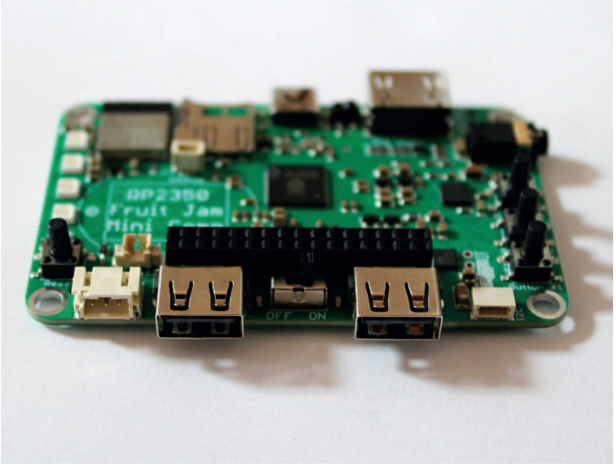
You can program the Fruit Jam in just about any way that you can program any other RP2350 board, though Adafruit has put a lot of work into getting CircuitPython well supported. As well as a module to make it easy to access the underlying hardware, there's Fruit Jam OS which adds a launcher and a way of managing apps.

The default apps include a text editor, IRC client, and some classic games (including versions of Snake, Minesweeper, and Breakout). There's PyDOS that gives you a DOS-like interface for managing files. As well as fun little apps, we can see this being useful for controlling hardware in situations where you want a monitor and mouse or keyboard, but not the complexities of a full OS.

Given the hardware, we'd expect to see some emulators of classic computers popping up – Lady Ada has shown a Macintosh 128K emulator running on a Fruit Jam on YouTube. You can download the code from **rpimag.co/picomac**. The RP2350 at the heart is certainly capable of emulating more modern machines, provided someone is willing to invest the time to port emulators to this hardware.

Documentation was limited at the time of review, but updated resources will likely be available at **learn.adafruit.com** by the time you read this.

There's nothing unique about the features on the Fruit Jam. You can add DVI and USB ports to Pico 2 using various add-ons. However, the Fruit Jam does bring together a very useful set of hardware at a very reasonable price.




It’s rare to see one USB port on a microcontroller board, but the Fruit Jam has two

OceanofPDF.com

# 10 amazing:

# airborne projects

## Take to the skies with Raspberry Pi and a dream (or helium)

**Is it a bird?** Is it a plane? Is it a tired cliché? No, it's a Raspberry Pi and/or Raspberry Pi Pico flying high up in the sky! These projects defy gravity and send your electronics soaring – sometimes you control it, sometimes you track it, and sometimes you have to get a telescope out to see the vessel it's located in. Grab your aviator goggles and let's take off.

### 01. Pico 2 Drone

Miracle flyer

**rpimag.co/pico2drone**

Created by maker intern Louis Wood, who also made the LEGO card shuffler, this drone started with a 3D-printed body before moving to a less-brittle CNC-routed carbon fibre. It also has a Raspberry Pi Zero aboard.

### 02. Pi0drone

Zero can fly

**rpimag.co/pi0drone**

This smart drone build uses a Raspberry Pi Zero, the PXFmini board made specifically for drone use, and comes in under $200, which is pretty good.

### 03. RC plane OSD

Bird's-eye view

**rpimag.co/rcplaneosd**

Update your RC plane with not only a POV camera, but also flight telemetry, including battery voltage so that you know when you need to bring the plane back home.

### 04. Pi in the Sky

High-altitude ballooning

**rpimag.co/piinthesky**

Within a few months of Raspberry Pi launching, Dave Akerman was sending balloons up with Raspberry Pi to take photos of the upper atmosphere. He even broke several records

doing it.

### 05. Astro Pi

Raspberry Pi in space

**astro-pi.org**

Sitting aboard the International Space Station is Astro Pi, a special hardened version of Raspberry Pi with a Sense HAT attached. School students can enter a competition to get their code running on it.

### 06. DIY Quadcopter Drone

Pico 1 powered

**rpimag.co/diyquadcop**

Built around a Raspberry Pi Pico 1 and its RP2040, this cool project does a little less than the Pico 2 Drone; still, it's very impressive that it works at all.

### 07. Remote-Controlled-Airplane

Raspberry Pi 3 remote

**rpimag.co/rcairplane**

This is an open-source project (it's on GitHub) for creating an RC airplane with a Raspberry Pi 3B+ as the brains of the operation – although you can probably upgrade it if you need to.

### 08. GASPACS CubeSat

Orbital Raspberry Pi

**rpimag.co/gaspacs**

Designed to test aerobraking through an inflatable AeroBoom in low Earth orbit, this undergraduate-developed Raspberry Pi-powered CubeSat was in space for 117 days.

### 09. Hovercraft

Technically airborne

**rpimag.co/hovercraft**

This Wiimote-controlled hovercraft was showcased at Coolest Projects 2016, using a Raspberry Pi alongside the Wii Remote to move the cardboard hovercraft around.

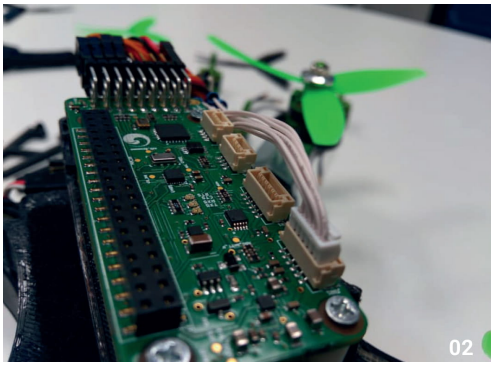
### 10. Pi Zero Plane

Fixed-wing Zero

**rpimag.co/zeroplane**

Instead of being elevated with rotors, this one uses aerodynamic lift to fly. As the name suggests, it features a Raspberry Pi Zero, along with the PXFmini drone add-on.


01


02

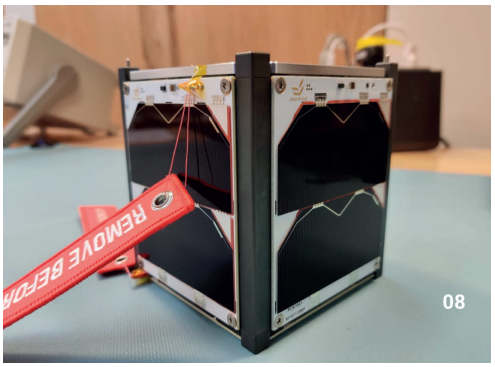

BATT: 10.94V RSSI: 82 AD: 99 T 12:57:09
ALT: 195.1 VEL: 9.0 DIST: 177.8 DIST G: 154.5
LAT:
LONG:
HEAD: 195
03


04


Astro Pi VIS
05

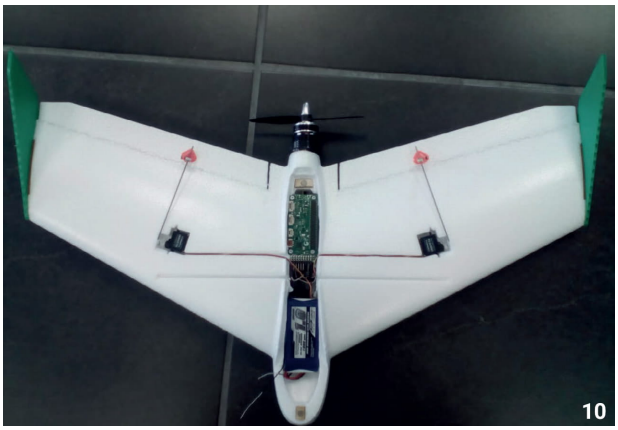

06


07



08


09




OceanofPDF.com

# 10 amazing:

# Astronomy projects

## Look to the stars with a Raspberry Pi

**Writing about projects in space for our Extreme Projects feature got us thinking about other ways Raspberry Pi can be used with the vast cosmos.** It's one of our favourite uses and we always love seeing what folks are doing when it comes to space observation and Raspberry Pi.

### 01. ALBATROS radio astronomy

Listening for signals

**rpimag.co/albatros**

An array deployed in the 'sub-Antarctic' and then the Arctic and is being used to map out the universe – and find out how old it is.

### 02. Mini Observatory

Table-top miniature

**rpimag.co/miniobs**

This 3D-printed observatory replica uses a Raspberry Pi 4, RP2040, and 50mm SLR lens to capture beautiful photos with a beautiful project.

### 03. Hubble Pi

Earth-bound telescope

**rpimag.co/hubblepi**

Not the one up in space, but a telescope modified to use a Raspberry Pi HQ Camera for wonderful, and customisable, astrophotography.

### 04. Fireballs Aotearoa

Great balls

**rpimag.co/aotearoa**

Make sure no meteor goes undetected around the world (and specifically New Zealand here) with this fantastic citizen science project.

### 05. DeMoor Orrery planetarium

Track the planets

**rpimag.co/demoor**

This project inspired by an 1781 orrery is a beautiful piece of art you can hang on your ceiling. It also includes astrological labels.

### 06. Raspberry Pi Pico solar system

Cheaper orrery

**rpimag.co/picosolar**

This tiny clock shows the current orientation of the solar system and, with a bit of editing, can show you how the planets aligned in the past or will do in the future.

### 07. Astroberry

Track the stars

**astroberry.io**

A system for Raspberry Pi that can hook up to computer-controlled telescopes and other similar devices for tracking celestial objects. Perfect for long-exposure photos of far-away targets.

### 08. Raspberry Pi Sunspot & Solar Eclipse Camera

Total coverage

**rpimag.co/eclipsecam**

Remember, never look directly at an eclipse, have a Raspberry Pi do it for you. Then you can convert it, like creator CapeGeek, into a solar camera for stunning shots of our very own star.

### 09. PiFinder

Pre-made tracking

**pifinder.io**

$485 / £375

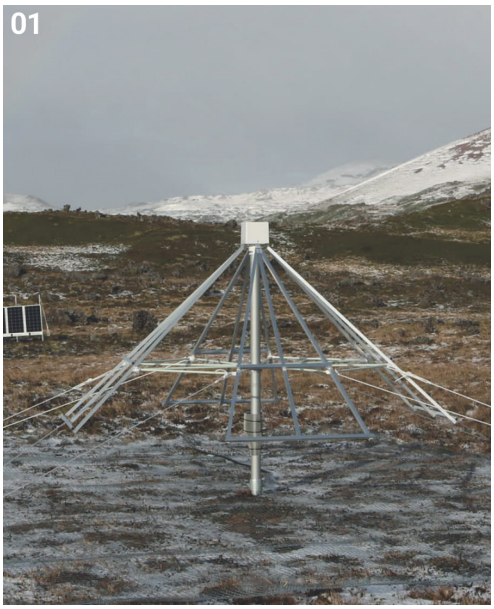
Built around a Raspberry Pi, this product is already fully set up to help control your telescope (or other observation equipment) to track celestial objects.

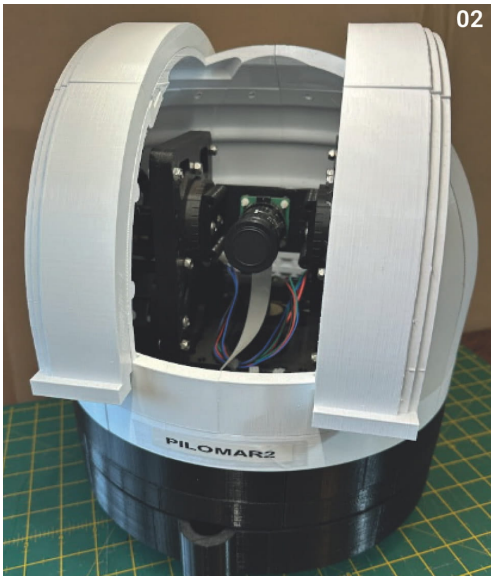
### 10. Cheap All Sky Camera

Time-lapse on a budget

**rpimag.co/allskycam**

Using a plant pot and an acrylic dome, you too can create an all-weather camera on a tight budget that can capture meteor showers and other cosmic events.


01

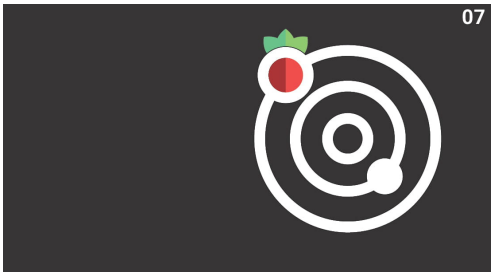

02
PILOMAR2

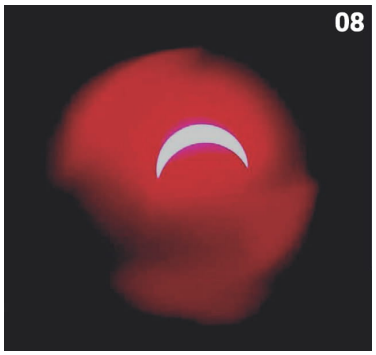

03


04


05


06
pico display
A
B
X
Y
RGB
27 JUL
2022
WEDNESDAY
13:33
pimoroni.com/picodisplay


07


08


09
PiFinder
PiFinder by BBLabs




OceanofPDF.com

# 10 amazing:

# Raspberry Pi projects in the garden

## Raspberry Pi is green, and so is grass. Coincidence? Yes

**We're British, so we do love a nice well-kept garden, or back yard as you might call it in other English dialects.** We're also fans of automation and data, so using a Raspberry Pi to help improve our gardening is an exciting prospect. Here's just a few excellent ways to do so…

### 01. Watering and time lapse

See it grow

**rpimag.co/waterlapse**

This very simple project uses a moisture sensor to make sure the plant has enough water. And a camera for the time-lapse video.

### 02. Szerafin mushroom farm

Fungi optimisation

**rpimag.co/szerafin**

Temperature and humidity are measured by this system to keep the environment perfect for the mushrooms to grow.

### 03. Freight farms

Professional hydroponics

**rpimag.co/freightfarms**

There have been a few Raspberry Pi farm ideas in the past, but none have been created to work in an actual shipping container at such high volume.

### 04. Pollination monitor

Track the bees

**rpimag.co/pollination**

How do you know your strawberries have pollinated? Using Raspberry Pi and computer vision to track bees visiting your plants, of course.

### 05. Remote agricultural monitoring

Tracking fruit flies

**rpimag.co/remoteagri**

Invasive fruit flies can destroy crops if left unchecked, This Raspberry Pi trap sees what insects are showing up in case action is needed.

### 06. Open Weed Locator

Save on spray

**rpimag.co/openweed**

A large method of checking weeds in large areas of greenage using computer vision and many Raspberry Pi boards attached to a weed-spraying robot.

### 07. PiMowBot

Robotic lawn trimming

**rpimag.co/pimowbot**

Roombas are well-established robots that hoover your house. What if there was an equivalent for your grass? Meet PiMowBot. But be careful of its spinning blades.

### 08. Hydroponic garden

Grow at home

**rpimag.co/hydroponics**

Hydroponics is a fancy word for just using nutrient-filled water to grow plants, and is great for growing your own veggies using Raspberry Pi automation.

### 09. MudPi

Smart watering system

**rpimag.co/mudpi**

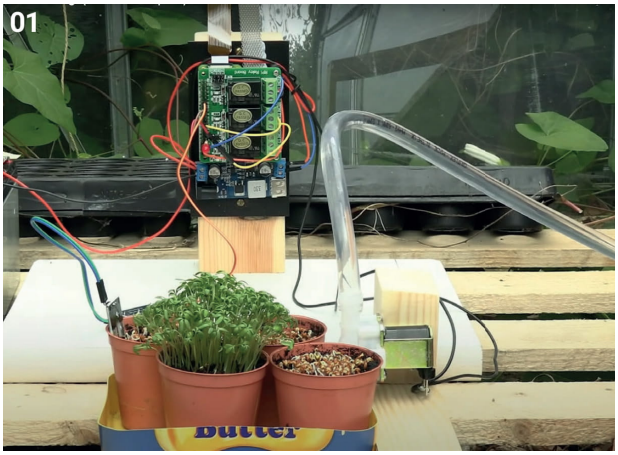
Using various sensors and automated irrigation systems, you too can keep your garden perfectly watered all year round with MudPi.

### 10. Garden Lighting

Flower bed illumination

**rpimag.co/dmxlight**

Using a Pico and a garden bollard, you can easily create custom lights to make your garden look very pretty during the moonlit hours.


01

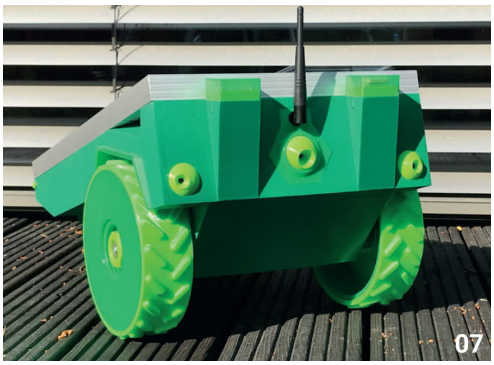

02


03
©Freight Farms

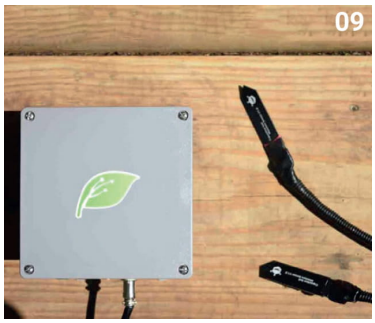

04


05


06


07


08


09




OceanofPDF.com

# 10 amazing:

# Raspberry Pi cyberdecks

## Preparing for the cyberpunk future with these rad custom PCs, choom

**Raspberry Pi is small and compact, making it much smaller than most accessories you need to plug in to make a computer.** Utilising this size and portability makes it perfect for your own custom cyberdeck, built to your exact liking, and there are plenty of folks who have done so. Let's go off the grid with just ten of them…

### 01. Backpack Cyberdeck

Lab with straps

**rpimag.co/backpackcyber**

A mobile workstation for wireless communications analysis and pentesting, this backpack has a special hand-crafted frame so that everything fits. It also has a ton of antennas.

### 02. Cyberdeck portable entertainment system

Plug and play

**rpimag.co/cyberes**

Not every cyberdeck needs to be able to hack into the mainframe, sometimes you just need to take a break and play some Street Fighter II.

### 03. Clamshell BlackBerry cyberdeck

Cyberdeck for ants

**rpimag.co/berrycyber**

BlackBerry was once the platform of choice for corpos. And now with a Raspberry Pi installed, it can be your netrunner terminal! Think we're using these cyberpunk terms correctly…

### 04. Extremely mini cyberdeck

Extremely small PC

**rpimag.co/minicyber**

Friend of the magazine Michael Horne has been working on this a while – it's the smallest cyberdeck you'll probably see, and it has a whole suite of sensors too.

### 05. Raspberry Pi Recovery Kit

No Wi-Fi required

**rpimag.co/recoverkit**

Need a hardened box to last the post-apocalypse? You might be able to use this specific Raspberry Pi build. It can still be used pre-apocalypse, though.

### 06. CyberPlug

Handheld prototyping

**rpimag.co/cyberplug**

Many cyberdecks are handheld and aren't always in a typical laptop computer form factor, much like this CyberPlug, which its creator loves to use to experiment with different components.

### 07. Pilet

Commercial cyberdeck

**rpimag.co/pilet**

This modular cyberdeck can be modified to be used how you wish, with a retro computer style straight from the late 1970s and early 1980s.

### 08. PiDex

Waterproof deck for Raspberry Pi 5

**rpimag.co/pidex**

Using a big, red switch, you can turn on this 'rugged and waterproof' cyberdeck. Although the box is probably the thing that's waterproof.

### 09. Vertical Runner

Working sci-fi prop

**rpimag.co/verticalrunner**

You can read about the full project earlier in the book (p100), but this cool-looking movie prop will actually work if you need it to.

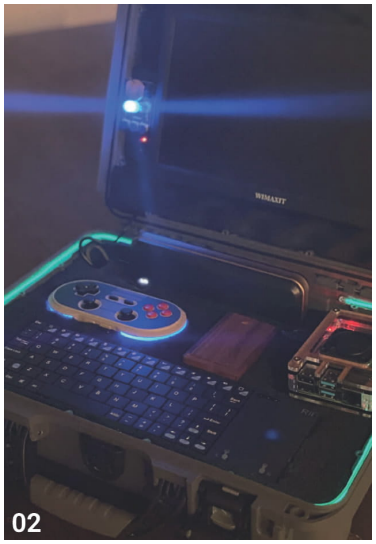
### 10. Mega six-screen cyberdeck

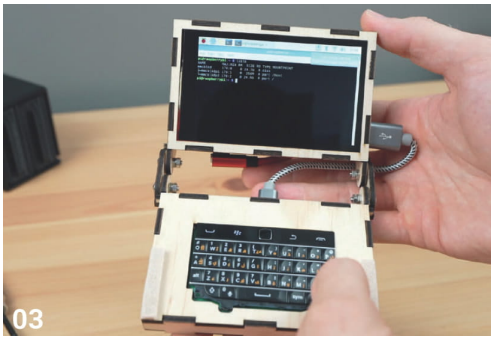
I'm seeing triple

**rpimag.co/sixcyber**

This six-screen Raspberry Pi-powered cyberdeck easily folds down into a special case – perfect for making a run to the next safe house.


01


02


03


04


05


06


07



08


09


10


PROJECTS FOR MAKERS AND HACKERS
Raspberry Pi
BOOK OF MAKING 2027
Raspberry Pi
Book of Making
2027
LED origami | Smart $CO_2$ sensor | Make your own PCBs | Underwater rover | Wordsearch clock
Build a jukebox | Control a robotic arm | And more

# Raspberry Pi

# BOOK OF MAKING 2027

**Technology isn't just something you buy — you can build it yourself! We've rounded up a few of our favourite makes previously published in Raspberry Pi Official Magazine, which should keep you entertained for an afternoon, a weekend, or longer.**

- Build a simple sensor robot from scratch
- Experiment with long-range communication platform Meshtastic
- Turn a toy skeleton into a jumping, dancing robot
- Design, prototype and manufacture a custom PCB
- Control a robot arm
- Add sensors and LEDs to pretty much anything you want!

Buy online
rpimag.co/bookofmaking2027

[OceanofPDF.com](OceanofPDF.com)

# Raspberry Pi
# Official Annual
# *2027*

**Dive into the world of Raspberry Pi with this huge book of tutorials, project showcases, guides, product reviews, and much more from the pages of *Raspberry Pi Official Magazine*.**

Whether you're new to the world of Raspberry Pi or want to broaden your maker horizons, we have a bit of everything for everyone in this, the first Raspberry Pi Official Annual. That's 200 pages of Raspberry Pi 500+, Raspberry Pi 5, Raspberry Pi Pico, Raspberry Pi Zero 2, and more to help you get inspired for your next – or first! – project.

## Look inside to find:

**A get started guide** covering every Raspberry Pi

**Everything you need** to know about Raspberry Pi 500+

**Troubleshooting tips** for your latest build

**Inspiring projects** to give you your next project idea

**Tutorials on how to build** the ultimate media centre

**Pushing Raspberry Pi computers** to extreme limits

This bumper book is your definitive guide to everything Raspberry Pi. It's essential for any maker with big dreams and a thirst for knowledge.

**UK** £14 | **US** $24.99

ISBN 978-1-916868-57-1

9 781916 868571

Raspberry Pi PRESS

OceanofPDF.com