

The APLer's Guide to the Raspberry Pi

Introduction

The Raspberry Pi is cheap to buy, fun to explore, and a very practical platform for APL. If you know APL, this document will help you to get going with the Pi as quickly as possible.

Who should read this guide

This guide has been sponsored by Dyalog Limited as a resource for current or veteran APLers who want to experiment with Dyalog's free APL for the Raspberry Pi.

The APL is freely available on the Pi for non-commercial use. Commercial licensing is also available: contact sales@dyalog.com

Getting started with the Pi

Many APLers are familiar with Microsoft Windows running on Intel platforms and with Mac OS. Some are familiar with running Linux on desktop machines.

The Raspberry Pi is different, and the differences can be a bit daunting.

This guide aims to de-mystify the process of getting started with the Raspberry Pi as an APL development platform.

What's covered

The guide covers what to buy, where to get it, how to set it up and how to install Dyalog APL.

What's not covered

The guide assumes you are familiar with APL. It does not cover how to use Dyalog's wonderful RIDE (Remote IDE), as Dyalog already provide documentation on-line.

It does not cover Linux, though there is a link to some relevant learning resources at the end of the document.

Nor does it cover Physical Computing with the Pi (controlling lights and motors, reading buttons and potentiometers etc). I'm planning a separate guide for that.

The 5 minute summary

This section contains the bare minimum of advice you need to get started quickly, but offers no explanation or discussion of alternative approaches.

If you're happy with that just do what I suggest :)

Most APLers don't just blindly follow the herd - that's why they are APLers! If you want to know the assumptions behind my advice and the alternatives available, read the whole document.

If you just want to get going, here's the 5-minute version:

1. Buy a Raspberry Pi 3 Starter Kit and make sure you have the other hardware you'll need
2. Connect the contents and boot it up
3. Install Dyalog APL

Details of each are given below.

Buy a Raspberry Pi 3 starter kit

Raspberry Pi 3 starter kits are widely available in the UK, Europe and the USA.

Look for one which has the following components:

- Raspberry Pi model 3
- 5v 2.5A Power supply which you can plug into your mains supply
- 8Gb SD card with NOOBS or Raspbian Pixel pre-loaded

You'll also need a USB mouse, a USB keyboard (the Dyalog APL keyboard works best but any keyboard will do), a monitor or TV with HDMI support and an HDMI cable to connect it to the Pi.

You may also want to get a case. Some starter kits will include one.

You'll need WiFi or a wired connection in order to install and update the Pi's software.

The Raspberry Pi 3 has built-in WiFi which you can configure once the Pi has booted up. (If you're using an earlier model you can use wired ethernet or get a USB WEiFi dongle). If you decide to use a wired ethernet connection you will also need an ethernet cable.

Some starter kits include a keyboard and mouse; some don't. Check carefully and decide which one to get.

The supplier I normally buy from is a UK company called [Pimoroni](#). If you're based outside the UK Pimoroni ship overseas but a local supplier may be faster and cheaper.

Pimoroni sell a [starter kit for £85](#) (including mouse and keyboard) and an [essentials kit](#) (without mouse or keyboard) for £55.

Connect the hardware

There's a great [guide to setting up your Pi](#) on the Raspberry Pi website. I won't try to reproduce it; just follow their super-clear instructions.

When you're done, come back here.

Install Dyalog following these instructions

As mentioned above, you can run Dyalog APL on all models of the Raspberry Pi, and you can use Dyalog's Remote IDE (the RIDE) to connect, program and debug APL from another computer.

If you've got a Raspberry Pi 2.0 or 3.0 you can also use the RIDE on your Pi.

[Dyalog/RIDE installation on the Pi is covered here.](#)

The documentation for RIDE 4.0 is available [here](#).

Now skip on to the section 'What Next?' at the end of this document.

The alternatives

Alternative models of the Pi

You don't have to get a Raspberry Pi model 3.

You can use a Raspberry Pi zero, zero W, Model 1 or 2 but they are not much cheaper and are much less powerful.

The Raspberry Pi model 3 has 4 cores and 1 Gb of RAM and APL can use them all!

You might consider earlier versions but remember that you won't be able to run the RIDE on a zero, zero W or Pi model 1. Apart from the RIDE, setup and installation are the same as for the Pi 3.

Headless running

It's *possible* to run any model of Raspberry Pi without a keyboard, mouse or monitor. This is sometimes called a *headless* configuration.

You can still program your Pi so long as you have a wired or wireless network connection.

You can even configure a headless Pi from scratch so long as you have a wired connection and are willing to go through a more complex set-up process. There's more information on configuring and using a headless Pi in Appendix A.

What next?

You can now start exploring Dyalog APL on the Pi.

If you're running a Pi model 3 you may be pleasantly surprised at the responsiveness of this very inexpensive setup. [Morten Kromberg was!](#)

The Pi version is a full implementation of Dyalog APL. If you're a current Dyalog user you should find the environment very familiar. If, like me, you're coming back to APL after a period of absence, you might want to take a look at some of the resources described [here](#). You'll also find a quick guide to getting started with RIDE below.

Three ways to RIDE

You'll enjoy using Dyalog APL with the RIDE. It's an intuitive development environment with a lot of powerful features, and version 4.0 runs on Raspberry Pi models 2 and 3. You can also run the RIDE on another computer to connect to an APL interpreter that's running on the Pi.

There are three ways you can use RIDE with APL on the Pi: 1. If you have a Pi model 2 or 3 with a mouse, keyboard and monitor, you can use the RIDE on the Pi. 2. If you have a Pi model 2 or 3 with VNC enabled, you can use the VNC client on a remote computer to run a RIDE session on the Pi, even if the Pi hasn't got a keyboard, mouse or monitor. You can find details of VNC in Appendix A; once VNC access is set up, you'll use the RIDE in the same ways as option 1 above. 3. If you have any model of Pi with network access you can use the RIDE on a networked computer to run Dyalog APL on the Pi. That's a great way to connect to and control a Pi robot.

Running the RIDE on the PI

If you've installed Dyalog APL version 16 and RIDE 4.0 following [this process](#), you should see a couple of new Dyalog Icons when you click on the Pi's Programming Menu.

If you click on the Dyalog Icon labelled *Dyalog APL - not* the one with (TTY) after it - you should see a RIDE session open on the Pi. It should look like this:

As you can see, the RIDE has an area at the top which displays the APL character set, so you can type APL characters without worrying about the PAL keyboard layout. On the Pi, though, APL keyboard support is installed automatically, so you can use the Windows key to type all the supported APL characters.

Running the RIDE remotely

If you've installed Dyalog version 16 on a Pi that's connected to your network via ethernet or WiFi you can connect remotely using RIDE on a Windows, Mac OS or Linux computer. You'll need RIDE 4.0 on the remote computer, and you may also want to enable the APL keyboard.

If the remote machine has been used for APL development it probably has RIDE 4.0 installed and the APL keyboard is probably already enabled.

If not, you can find installation instructions in the [RIDE User Guide](#)

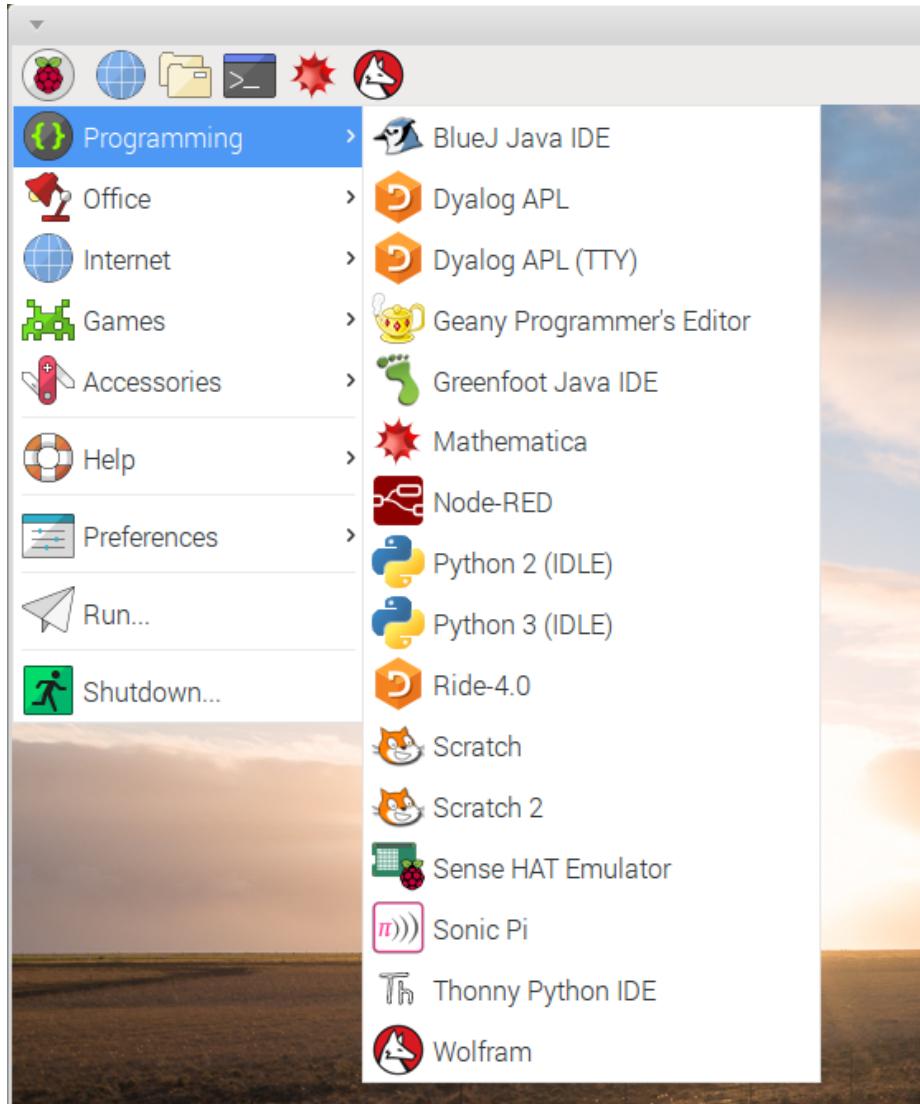


Figure 1: Programming Menu

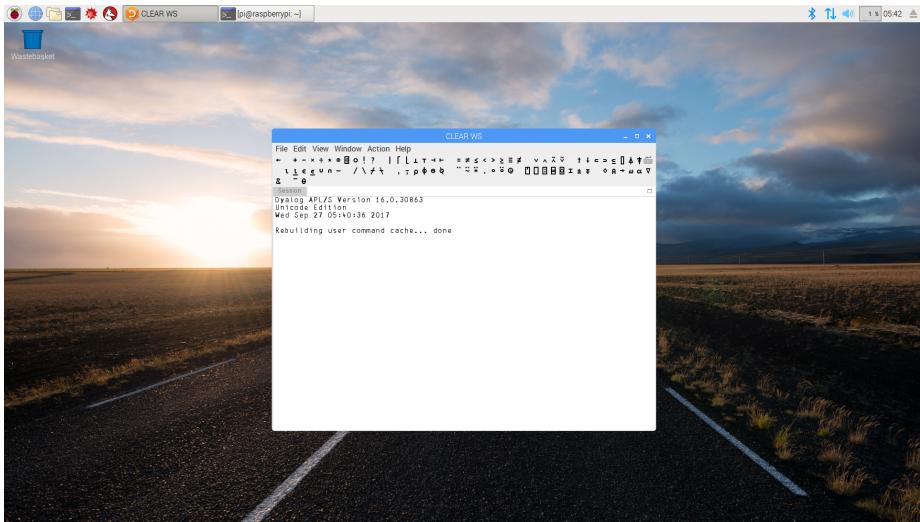


Figure 2: Local RIDE session

Developing with the RIDE

The RIDE is a very capable IDE, and its features are covered in detail in the user guide.

Below you'll find a very brief summary of the features you need to start developing in Dyalog APL.

Immediate Execution Mode

If you type in an APL expression in the RIDE window and press **Enter** the code will be executed and the result (if there is one) will be displayed.

definition of dfns

You can create, use and save direct functions (dfns) in immediate execution mode.

To calculate an average using an anonymous dfn with the traditional* APL algorithm, you can enter the following:

```
{(+/ω)÷ρω} 1 3 5 7 8
```

and you'll see the result displayed

4.8

You can save the dfn by assigning it:

```
average ← {(+/ω)÷ρω}
average 1 3 5 7 8
```

4.8

* These days there's a better version which works with higher-dimensional arrays and uses a *train*:

```
mean ← +/÷#  
mean 1 3 5 7 8
```

4.8

editing functions

You can use the RIDE editor by invoking the `)ed` command, or by double-clicking a name in the RIDE session.

If you defined the function `average` as shown above, you can type

```
)ed average
```

and an editor panel will open in the RIDE.

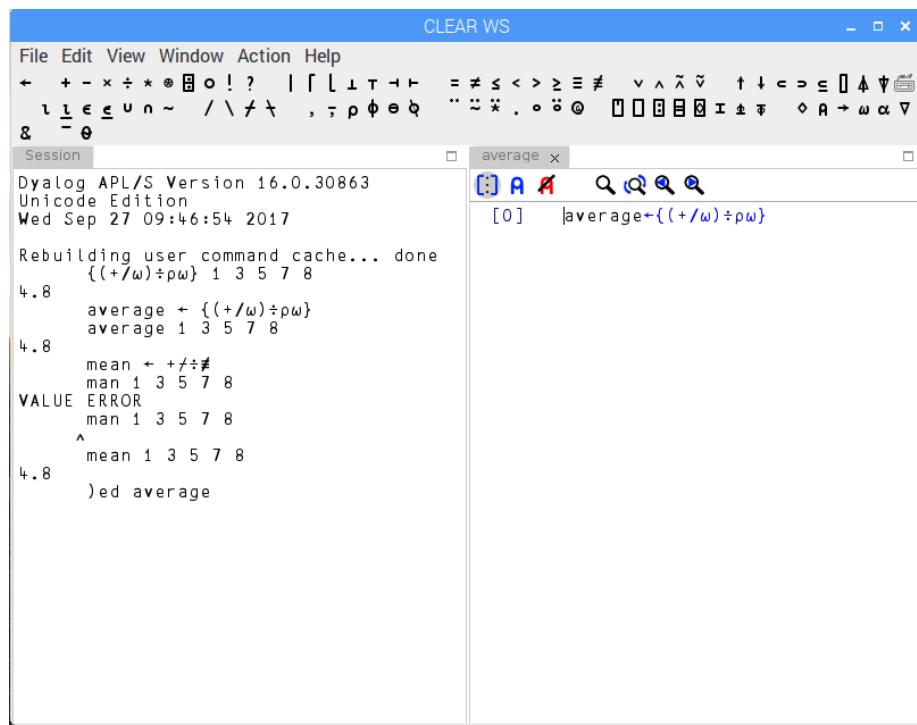


Figure 3: RIDE Editor

You can also use the RIDE editor to create and edit traditional functions, edit data and manage the debugger. You'll find more details in the RIDE USer Guide.

Physical computing

There's not enough space in this guide to cover the ways in which you can do Physical Computing from APL on the Pi (driving LEDs and motors, and sensing the real world). You can read some details on [Morten Kromberg's blog](#), and I hope to publish a guide to the topic if there's enough demand.

Other resources

If you're new to Linux you may also want to get up to speed with Raspbian, the version of Linux that you'll be using with the Pi. There's a good book: Raspberry Pi User Guide by Eben Upton and Gareth Halfacree which you can find on Amazon and elsewhere. Make sure you get the Second Edition!

There's a helpful forum on the [Raspberry Pi website](#), along with a lot of other resources.

There's a monthly magazine called [the MagPi](#). You can get it [free in pdf format](#), but if you subscribe to the printed version you'll get a free Raspberry Pi zero!

Use the slack channel and support forum

There's a slack channel for discussing APL. You need an invitation to join. You can ask for an invitation, or ask questions about Dyalog, on the [Dyalog Support Forum](#). You'll need to register before you can post.

Join the email list

Dyalog have a mailing list which they will use to keep Pi users informed of new resources. You can [join it here](#).

Have fun!

I hope you'll have as much fun with the Pi as I do!

Appendix A

Using a Pi without monitor, mouse or keyboard

Sometimes it isn't practical to connect your Raspberry Pi to a lot of other hardware. You might not have a spare keyboard or monitor available, or you might be using the Pi to control a free-ranging robot. Fortunately you can configure and use a Raspberry Pi without the extra peripherals.

The minimal configuration consists of a Pi zero, model 1, 2 or 3 with a source of power.

The power source is usually an appropriate mains adapter, but any 5v power source will do so long as it can supply enough current. Most robots use LiPo batteries or rechargeable NiMH cells to power their motors and their Pi.

This minimal configuration requires that the Pi already has working software installed. You *can* configure a headless Pi from scratch over a network using another computer, but the process is more complex and is harder to trouble-shoot.

You'll find details below, with links to more detailed information about each step.

The first part of the guide below covers initial configuration of the Pi without mouse, screen or keyboard, which is possible but tricky.

The second part covers subsequent configuration, which is what you'll need if you want to use your Pi in a Robot or a similar stand-alone project.

You will need * a Laptop or computer (which can be an already-configured Raspberry Pi model 2 or 3) from which to configure the target Pi and * a wired network connection for the Pi. (If you're using a Pi zero/zero w, you'll be aware it has no ethernet port, but you can use an inexpensive USB-to-ethernet adapter).

It's also possible to configure a Pi zero or zero W using just a USB connection to your laptop, but this is even trickier and not recommended until you've built up your expertise. If you *really* need to do this, you'll find details [here](#).

There are three main steps to follow, described below: 1. You'll need to prepare an SD card for the Pi. 1. You'll need to use a protocol called `ssh` to connect to the Pi and prepare it for step 3. 1. You'll use a protocol called `VNC` to access the Pi's desktop on your laptop or workstation.

After step 3 you can use your laptop or workstation for carry out further configuration of the Pi. In particular, you can install Dyalog APL on the Pi and use the RIDE to program and debug.

Headless configuration details

Understanding the security risk

The process you'll see below will enable `ssh` access to your Raspberry Pi for anyone on your network. Your Pi will briefly be visible as `raspberrypi.local` and anyone will be able to log in using the default userid `pi` and password `raspberry`.

This is probably safe enough while you're just setting up the Pi, but once setup is complete you'll need to change the login password for `pi`. The process for doing that is covered in the instructions.

Preparing and using the SD card

1. It's best to start with an SD card containing the latest Raspbian image. You'll find details about how to download the image and install it on an SD card [here](#). **Do not** use it to boot the Pi before completing step 2 below!
2. Re-insert the freshly prepared SD card into your card reader. Your operating system should mount the SD card and you should be able to see a partition labelled `boot`. Create a file called `ssh` in the root of the `boot` partition. Its contents do not matter. If you're using Windows, make sure the file is called `ssh` *without any extension*. The `ssh` file tells Raspbian to enable its `ssh` sever when the Pi starts up.
3. Now insert the SD card into the target Pi and connect it to the power source.

Connecting to the Pi via ssh

1. You will need an `ssh` client on your laptop or workstation. Installation depends on the Operation System you're using. *On Linux* you'll need to install the `openssh-client` package if it is not already installed. *MacOS* should have `ssh` support built-in to `Terminal`. *Windows* users will need to install a suitable client: PuTTY (free software, widely used) is a good choice. You'll find installation instructions for PuTTY [here](#).
2. You will need to enable your laptop or workstation to find the target Pi on the network.

By default the Pi will use DHCP to request a dynamic IP address from your network which you will not know. You can sometimes find out the Pi's IP address by connecting to your router, but there is another approach.

Once the Pi is running it will broadcast itself on the network as `raspberrypi.local` using Apple's `bonjour` protocol. Resolving that hostname to an IP address requires suitable software on your Laptop or Workstation.

1. On *Linux* `avahi` should already be installed and running.
2. *MacOS* should also resolve the address without additional software.

- If you're running *Windows*, you can install Bonjour Print Services for Windows, available from the [Apple Website](<https://support.apple.com/kb/DL999?locale=en>). This will enable 'Bonjour' on your Windows computer.

Configuring the Pi

- You will need to log in to the Raspberry Pi from your laptop or workstation. TODO: Update Windows instructions
 - From *Linux* or *MacOs*, open a terminal window and enter `ssh pi@raspberrypi.local`
 - If you're using *PuTTY* on *Windows*, follow the *PuTTY* instructions to connect using `ssh` as user `pi`

Whichever method you use to connect, you are likely to see a couple of warnings.

One will come from your `ssh` client, warning you that you have not connected to the Pi before and asking if you are sure you want to connect. Say yes!

Once you have *connected* to the Pi you will be asked to log in. You'll see another warning that you are connecting via `ssh` to a Pi that still has the default Password set. That's a security risk which you will address in a minute. For now, log in as user `pi` with password `raspberry`.

- Now that you have logged in, you can configure the Pi. You'll need to run a program called `raspi-config`. Since this program can make major changes to your configuration, you will need to run it as `root` (the Linux super-user, equivalent to an Administrator account on Windows).

You can do that using the `sudo` command.

In your `ssh` session window, type `sudo raspi-config`. A menu of options should open.

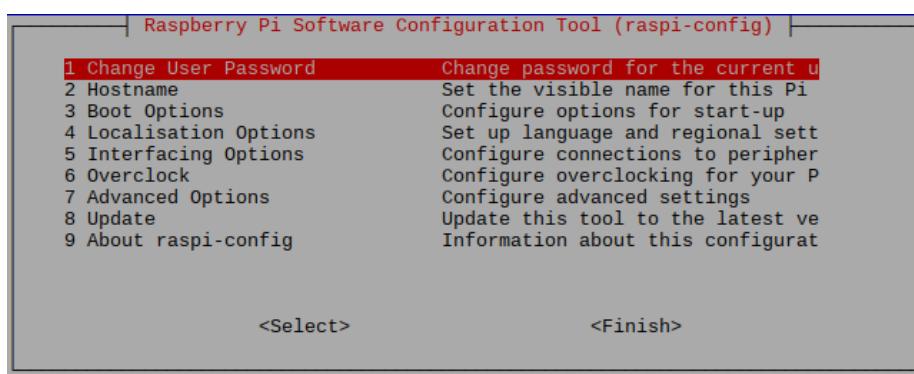


Figure 4: `raspi-config` screenshot

1. Select the first option and change pi's password. This will reduce the risk of some malicious person connecting to your pi using ssh and stealing all your secrets :)

You will be asked to enter the new password twice, to avoid typing errors.

Once you've changed the password you will be taken back to the top level menu.

1. Chose the 5th option.

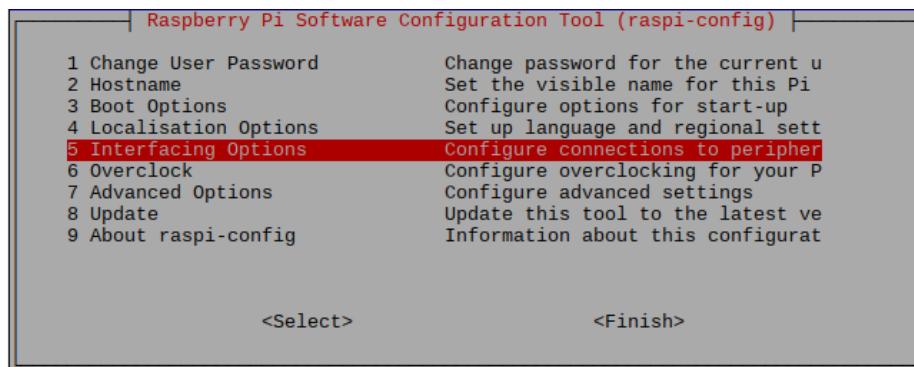


Figure 5: enabling the VNC interface

1. Select option 3 (VNC).

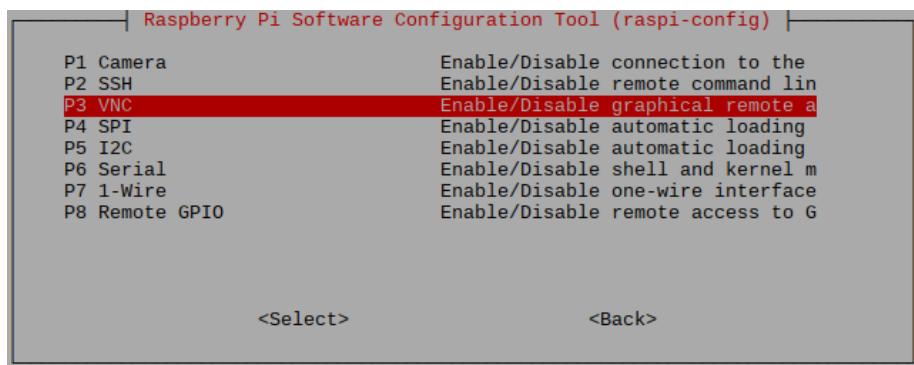


Figure 6: turn on VNC

1. Set the VNC option to Yes

Now you'll be able to use VNC on another computer to connect to the PI.

3. Configure WiFi (*Optional*).

If you're going to use the Pi without a wired ethernet connection to your network, you'll need to use WiFi.

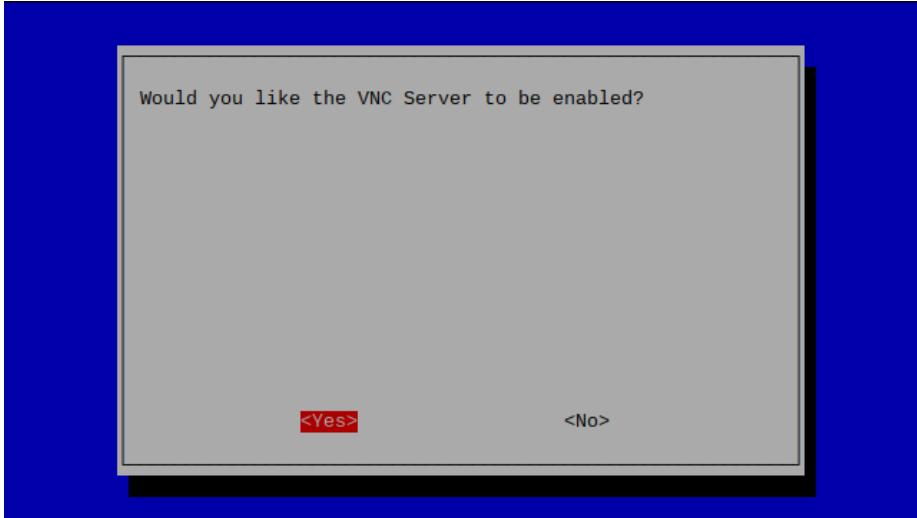


Figure 7: Say yes to VNC

The Pi 3 and zero w have built-in WiFi; other models will need a compatible WiFi dongle.

You can find out how to configure WiFi from the command line from this [Raspberry Pi Website article](#).

Connecting Via VNC

You'll need to install a (free) VNC client (called VNCViewer) on the workstation or computer that you want to connect to the Pi.

You can download the Viewer from the [RealVNC website](#).

Don't worry about the reminder you'll see to make sure you have VNC Connect installed on your target computer. It's part of the Raspbian distribution, and you enabled it in the previous step.

You *probably* want to sign up for a VNC account. It's free, and means that you will be able to access your Pi securely from anywhere on the Internet, but you don't have to if the Pi is on the same network as your computer.

Follow the website's installation instructions for your OS and start VNC Viewer.

You'll see a screen like this (if you didn't create a VNC account it will look a little different):

In the entry field at the top of the screen, enter the address of the Pi you want to connect to. Unless you have changed the hostname of the Pi, that will still be `raspberrypi.local`.

Press enter. You'll see a logon screen like this.

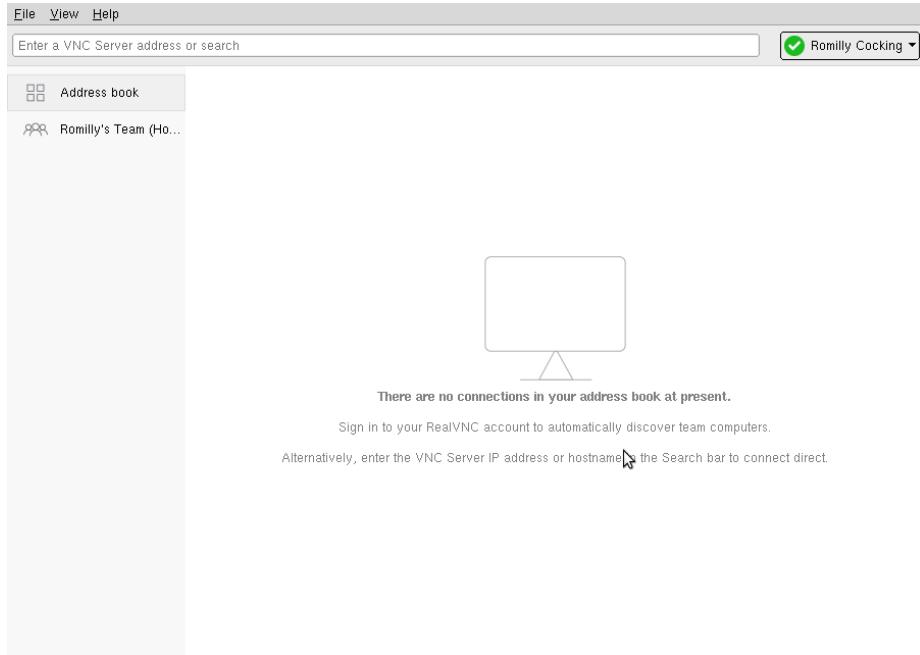


Figure 8: VNC connect

Change the username to `pi` and enter the Pi's logon password.

When you click on OK you should see the Pi's desktop displayed within a window on your machine.

From now on you can control your Pi from your laptop or computer, just as if your keyboard, mouse and monitor were connected to the Pi itself.

Licence

Unless otherwise specified, everything in this repository is covered by the CC BY-SA 4.0 licence:



About the author

Romilly Cocking has been using APL to solve problems since 1974.

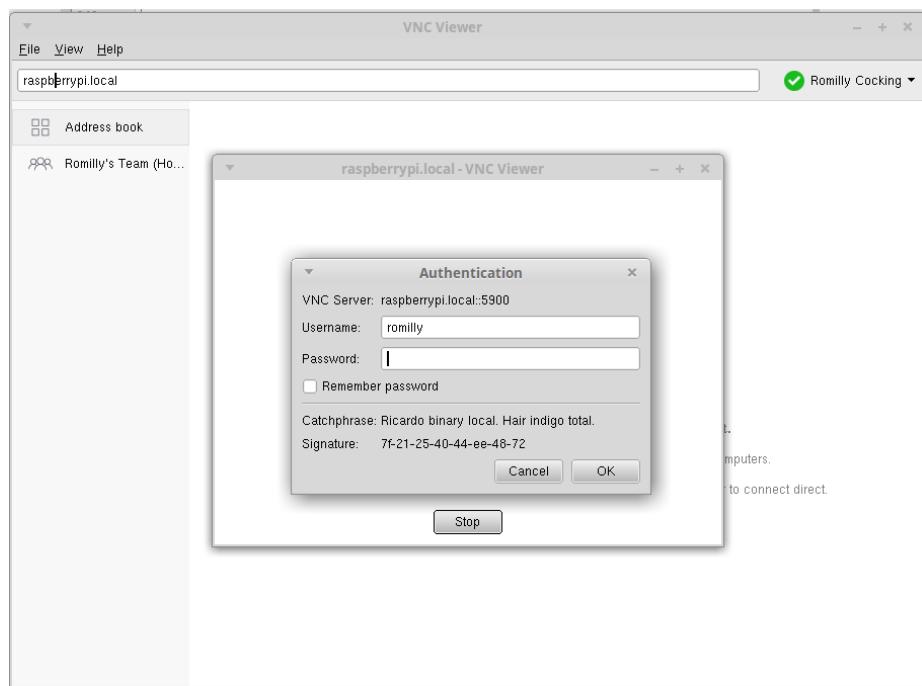


Figure 9: VNC logon

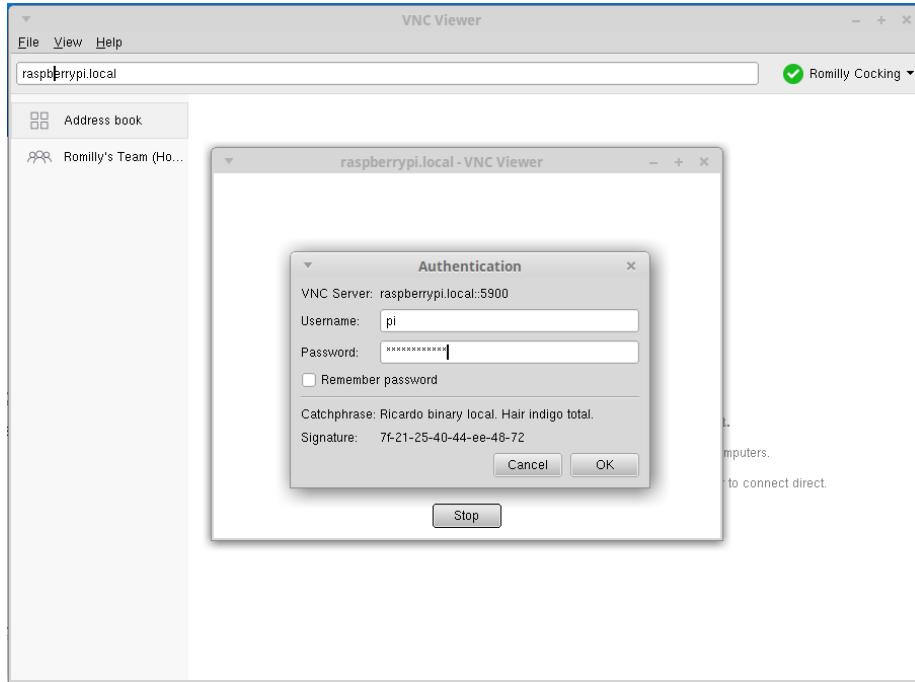


Figure 10: VNC logon step 2

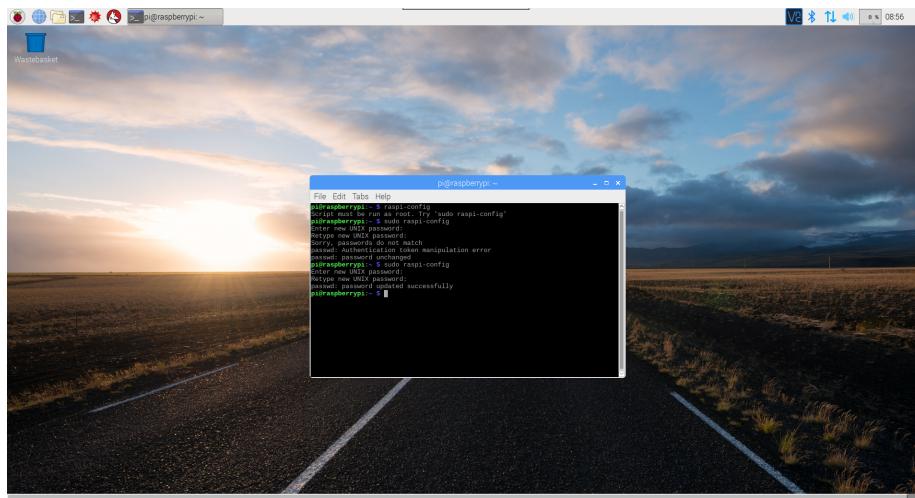


Figure 11: Pi desktop

He founded the British APL Association (then called the UK APL user group) in 1976.

For two decades he helped to run Cocking and Drury, a software house that specialised in APL.

Romilly is currently engaged in Robotics and AI research using APL as the implementation language.

You can contact him at romilly.cocking@gmail.com or follow him on twitter as [[@rareblog](http://twitter.com/rareblog)](<http://twitter.com/rareblog>)

You can read his blog [here](#)



Figure 12: Romilly Cocking