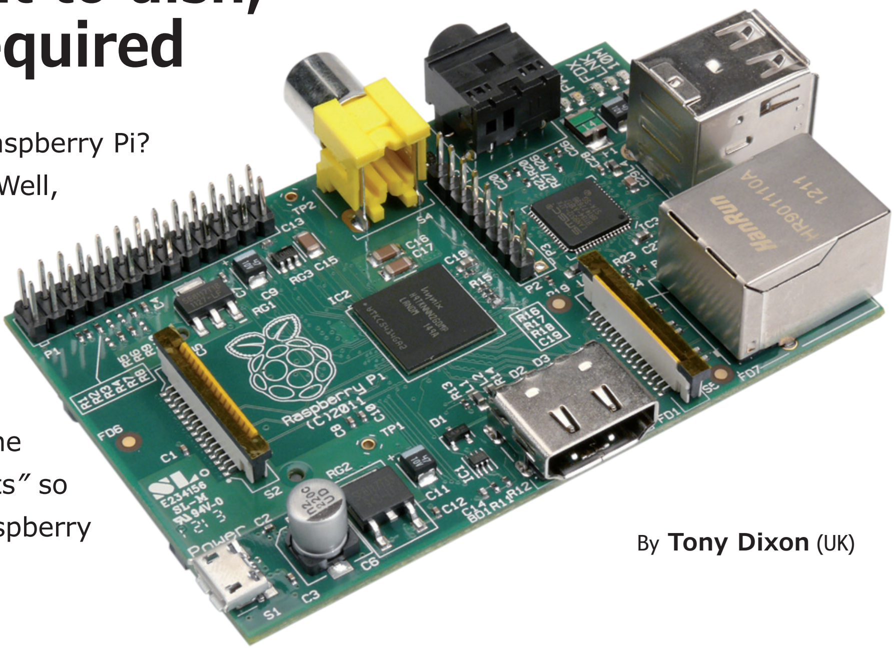
Raspberry Pi Recipes Part #1

From ingredient to dish, and no oven required

So you've got yourself a new Raspberry Pi? Great, but where do you start? Well, you're in luck, welcome to the first part of Elektor's Pi Recipes series published exclusively and monthly in Elektor.POST. Who says engineers are lousy cooks? In this the first part of the series we "gather the ingredients" so we can start configuring the Raspberry Pi and get it up and running.



By **Tony Dixon** (UK)

"Cook-figuring" your Pi

We know the Raspberry Pi [1] is a cool $35 credit card size computer as we already have one or we are thinking of getting one (aren't we?). Unlike most of the laptop or desktop computers we may use, the Raspberry Pi computer is powered by an ARM microprocessor and runs the Linux Operating System (OS).

Before we start configuring things for our Raspberry Pi we have two choices to make. The first choice we have is: which OS version do we use? Table 1 shows what our OS choices are.

Select the distribution that takes your fancy and download it. Be warned, these are big files, so be prepared to wait while it downloads. Alternatively you can use a torrent link if you have a slow broadband connection or want to download it overnight.

The second choice we need to make is what size SD Card do we use? We typically need a SD Card with minimum of 2 GB storage to store our OS. We'll easily get our OS onto a 2 GB card but we won't have much disk space left for our applications and data so we should consider using a 4 GB or bigger SD Card.

Table 1 - OS Distributions

Distribution	Description
Raspbian "wheezy"	Reference file system based on the Raspbian optimized version of Debian. Includes LXDE, Midori and Python development tools.
Soft-float Debian "wheezy"	This is identical to the Raspbian "wheezy" image, but is compiled using the slower software floating point ABI for program compatibility.
Arch Linux ARM	Based on Arch Linux, which is a simple and lightweight distribution, allowing the end-user full control, but not really suited for beginners.
RISC OS	A modern version of the original Acorn Archimedes Computer OS.
Raspbian	A Debian Linux-based operating system optimized for the Raspberry Pi hardware [2].
Occidentalis	Developed by Adafruit from the original wheezy distribution but with additional I2C and SPI hardware support [7].

For our tutorial, we'll use Raspbian [2] "wheezy" as recommended by the Raspberry Pi Foundation. So let's get started and download wheezy from the Raspberry Pi website. We used the latest version at the time of writing this article and this was "2012-12-16-wheezy-raspbian.zip".

Next we need to do is install our OS onto the SD Card, for this we'll use a laptop or desktop computer and SD Card writer.

If you're a Linux user, you'll probably already know how to copy the "wheezy" distribution to the SD card and expand it using DD command. But if you're a Windows user like myself, you'll need to download a small utility program wim32diskimager [3] to help us get our Linux distribution on to our SD Card. Once we've downloaded and installed win32diskimager, we can insert the SD Card into our card read/writer. In win32diskimager we'll select our Linux distribution from where we've saved it to when we downloaded it. Press the write button and press OK again to confirm the write operation, and our OS will be written onto the SD Card, see Figure 1.

While our SD Card is being written to, we can set up our Raspberry Pi. We'll need a USB Mouse and USB Keyboard and either a HDMI compatible LCD screen (or DVI compatible screen/monitor if using a HDMI-DVI adapter or cable), or for a retro 80's feel, a screen with a composite video input. There's no VGA video output on the Raspberry Pi but you can use a HDMI-VGA electronic adapter if you want to purchase one. We also need a Micro USB power supply to power the Pi.

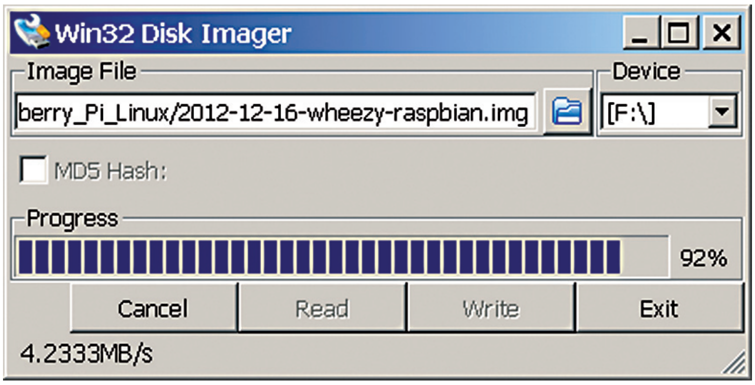


Image File: berry_Pi_Linux/2012-12-16-wheezy-raspbian.img

Device: [F:\]

□ MD5 Hash:

Progress: 92%

4.2333MB/s

Figure 1.
Win32imager in action.

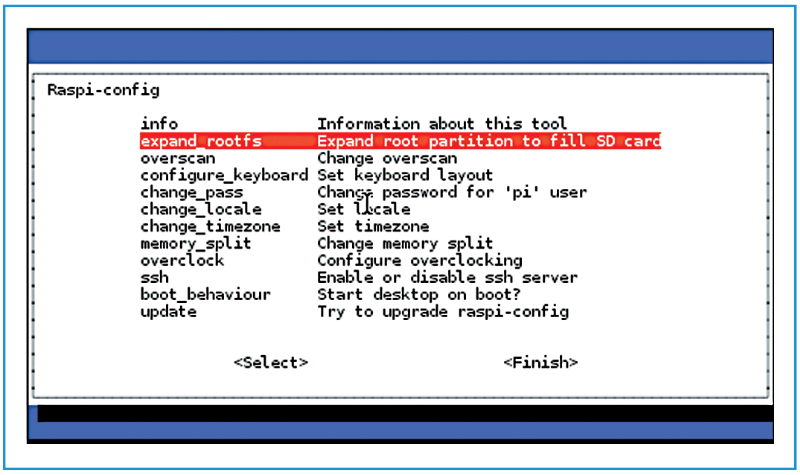
Once we've got our keyboard, mouse and screen connected to our Raspberry Pi, we can insert our SD Card and apply power and we'll start our first Boot. As the Pi boots you'll see a mass of command line instructions flash by as the Pi configures itself. The Pi boots directly from the SD Card so it takes a short time to get started. The beauty of this is, should you want to try a different OS or have a different set of application programs, you only need to change your SD Card with the new one and install your new OS and applications, and your Pi is now a different computer.

More preparations

Once the Raspberry Pi has completed its first time boot, the Raspi-config set-up tool will run automatically, see figure 2.

Table 2 - Raspi-config Options

Option	Description
info	Help on Raspi-config tool
expand_rootfs	Expand root partition to the capacity of the SD-Card.
overscan	Change the Raspberry Pi screen position to best suit your display monitor or screen.
configure_keyboard	Select the keyboard layout that best suits you.
change_pass	Change the default "pi" password to a different one.
change_locale	Change the Raspberry Pi's geographic locality to one that best suits you.
change_timezone	Change the Raspberry Pi's time zone to one that is correct for you.
memory_split	Change the memory split between application RAM and display RAM.
ssh	Enable or disable ssh server
boot_behaviour	Start GUI desktop after boot sequence.
update	Search for and update to the latest version of Raspi-config tool.



```
Raspi-config

         info                Information about this tool
         expand_rootfs       Expand root partition to fill SD card
         overscan            Change overscan
         configure_keyboard  Set keyboard layout
         change_pass         Change password for 'pi' user
         change_locale       Set locale
         change_timezone     Set timezone
         memory_split        Change memory split
         overclock           Configure overclocking
         ssh                 Enable or disable ssh server
         boot_behaviour      Start desktop on boot?
         update              Try to upgrade raspi-config


                 <Select>                 <Finish>
```

Figure 2.
Raspi-config tool.

The Raspi-config tool, which is reminiscent of a PC BIOS screen, gives us many options we can use to configure our Raspberry Pi. We'll use the cursor keys to navigate up and down the list of options and the Return key to make our selection. Table 2 shows the Raspi-config menu items.

Which options should be used is up to you, but configuring the keyboard and changing the location and time zone to ones that best suits you are obvious choices.

But before that, the first option you should select is "expand_rootfs". Using the cursor keys navigate to the option "expand_rootfs" and select it by pressing the return key. What we have done here is to tell the Raspberry Pi to expand the file system on the SD Card to the full capacity of the SD Card. When we originally installed the OS onto the SD Card, the installation only took as much disk space as was needed by the OS. So even if we used a 4GB SD-Card, the file system space available to us was much less than this. The next time we start our Raspberry Pi, a small script program will execute and expand the file system to the full capacity of the SD Card used.

Changing the default password is always a good idea, using the cursor keys navigate to "change_pass" and select. Enter a new password when prompted.

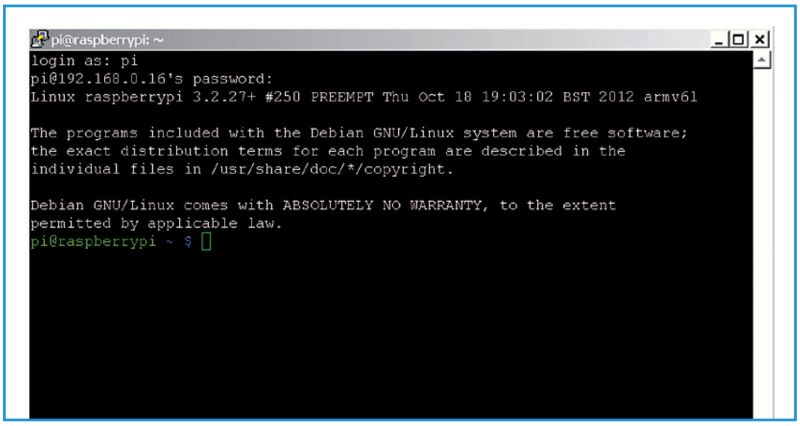
If you intend to use graphic intensive applications or games, then changing ratio the RAM is split between application memory and video memory could be useful, especially if you have a revision-2 Raspberry Pi with 512 MB of RAM to take advantage of the extra 256MB.

Starting the GUI after Boot is another useful selection to choose if you don't intend to use the command-line interface.

One final option to consider is, do you want remote access to your Pi? If you do, you can enable this by selecting the SSH (Secure Shell) server. Then on your laptop or PC use a SSH client such as Putty for Windows [4] and you can create a remote terminal connection to your Pi (Figure 3), wherever it may be physically. You'll need to know your Raspberry Pi's Ethernet IP address though. Checking your router is a good place to start or on your Pi start a LXTerminal screen and type ip addr to find it out.

Once we've finished configuring our Raspberry Pi, we can reset it and let it carry out any further configuration steps we may have opted for.

If we selected "boot_behaviour" to start the GUI after boot, once the boot sequence is complete we'll be started straight into the Debian desktop environment. If we didn't modify "boot_behaviour", after the boot sequence is complete we'll be left at the command line, in which case type in your user name and password (the default user name is raspberry and the default password is pi, or if you change the password then the new password you choose). Once these are accepted, on the command line type startx and press the return key to start the GUI desktop.



```
login as: pi
pi@192.168.0.16's password:
Linux raspberrypi 3.2.27+ #250 PREEMPT Thu Oct 18 19:03:02 BST 2012 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
pi@raspberrypi ~ $
```

Figure 3.
Using Putty for remote access.

We should now have the Raspberry Pi desktop as shown in Figure 4.

When we are in the desktop we can continue to configure our Pi. If we want to connect our Pi to our network or the Internet using WiFi, we can select the WiFi Config icon, Figure 5. Make sure we have plugged in a suitable USB Wifi Dongle into one of the USB ports (we may need to use a powered USB hub to give us extra USB ports). Start WiFi Config and follow the prompts.

What next...

We've got our Raspberry Pi up and running, now what next? Well, Elektor wouldn't be the world's best electronics magazine if we didn't do something electronically with it, so next time we'll be looking at the Raspberry Pi's Expansion Header and how to program our Pi.

Comment — Feedback — need support from the Elektor community? Feel free to join the exchange at Elektor.LABS! [5]. And hey there's a cool Rpi Prototyping Board coming up in Elektor.MAGAZINE's March 2013 edition.

This project can also be accessed online via the Elektor.MAGAZINE website [6].

(130078)

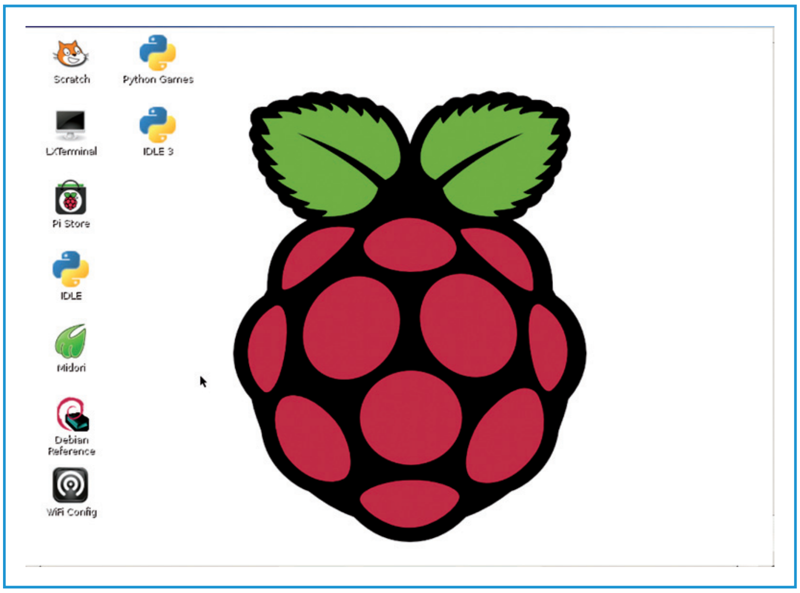


Figure 4.
Raspberry Pi desktop.

Figure 5.
Wifi config icon.

Internet Links

[1] www.raspberrypi.org

[2] www.raspbian.org

[3] launchpad.net/win32-image-writer

[4] www.putty.org

[5] www.elektor-labs.com/ElektorPOST/2013/03

[6] www.elektor-magazine.com/us/post

[7] learn.adafruit.com/adafruit-raspberry-pi-educational-linux-distro/overview