BASICEDU TEAM

HARDWARE SETUP GUIDE

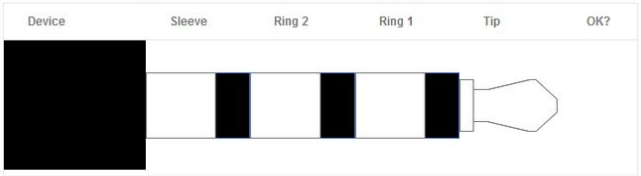
INDEX

List of parts :

1. Raspberry Pi board (Pi Model B+, Pi 2, Pi 3 and Pi 4)
2. Flashed SD card with the OS.
3. Mouse
4. Keyboard
5. CRT Television Set
6. 3.5 mm to RCA Connector Cable / HDMI to HDMI / HDMI to VGA Cable
7. Additional laptop for initial configuration.

NOTE : The Pi Model B+, Pi 2, Pi 3 and Pi 4 features a 4-pole 3.5mm audio jack which also includes the composite video signal. This has allowed for the removal of the composite video socket found on the original Model B.

Raspberry Pi
4-pole 3.5mm Audio/Video Socket



Device	Sleeve	Ring 2	Ring 1	Tip	OK?
	4	3	2	1	
Model B+	Video	Ground	Right	Left	✔
Apple	Video	Ground	Right	Left	✔
Zune	Video	Ground	Right	Left	✔
Camcorders	Right	Ground	Video	Left	⚠
MP3 Players	Ground	Video	Right	Left	✖

The new jack is a 4-pole socket which carries both audio and video signals. It's similar to sockets found on other multimedia devices such as iPods, MP3 players and smartphones. It now used on the A+, B+, Pi 2, Pi 3 and Pi 4. This style of connector is sometimes referred to as "TRRS", which stands for "Tip-Ring-Ring-Sleeve".

The four conductors carry video, left audio, right audio and ground. Cables are easy to obtain but use different configurations so you must be sure how your cable is wired before attempting to use it with the Pi.

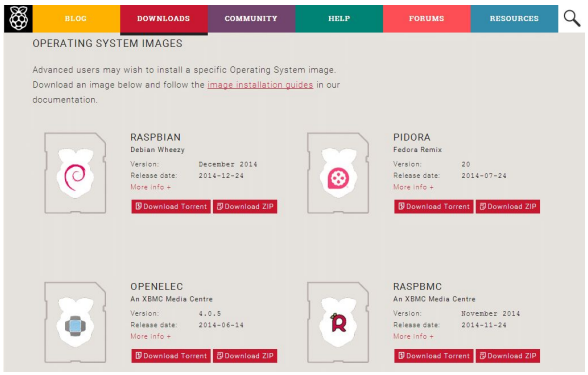
General rule of thumb : While acquiring parts cCables should be avoided where the ground connection appears on any ring other than Ring 2. If you've there's flickering and bad resolution even after following all steps in this guide, you've used the wrong cable.

Note : For certain CRT TV sets, bleeding off the edges has been observed. Using the configuration settings, you can fix most of these problems.

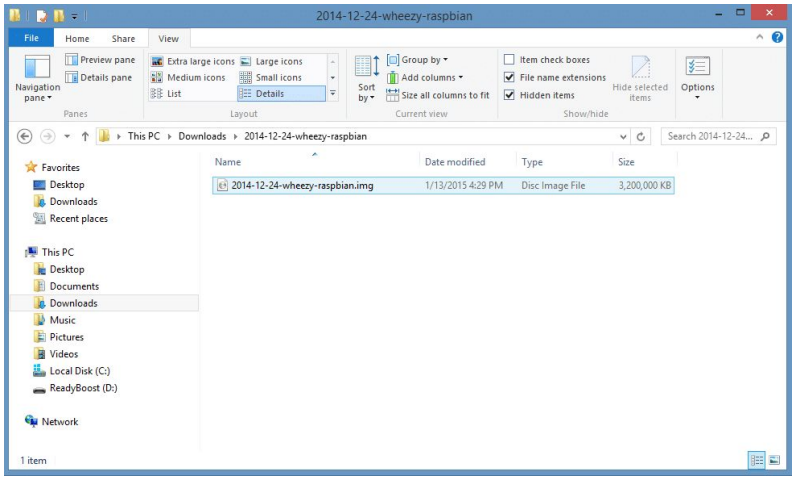
Basic Setup (Headless mode) :

GATHERING THE PIECES

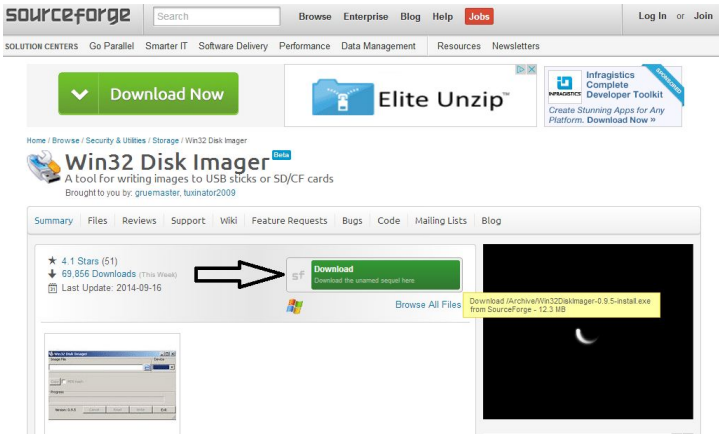
The first step is to download the Raspberry Pi's operating system. I'll be using Raspbian Jessie Lite in this tutorial. You can download it here from the Raspberry Pi Foundation.



Just download the ZIP file and extract it to a new folder. Then you'll have a single .img image file that contains the operating system:

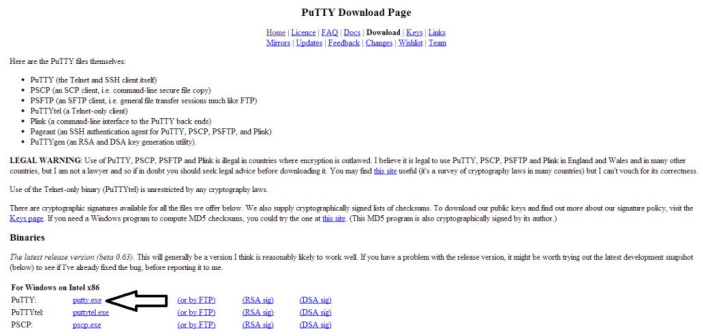


Next we'll need a program that can write the operating system image file onto a microSD card. We'll use Win32DiskImager to do that, so download and install it:

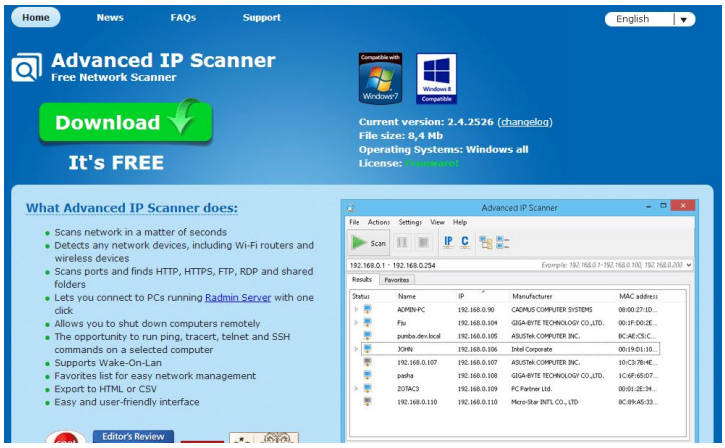


Now we need a way to access the Raspberry Pi's command prompt. We can do this with a remote SSH client called PuTTY. PuTTY establishes an SSH connection between the Raspberry Pi and another computer. It opens up a window that gives you access to the Raspberry Pi's command prompt. Download and install PuTTY to your PC.

If you're using a Mac, you can SSH into the Pi with Terminal. Click here for good tutorial on SSHing via Terminal.



We'll also need a way to find the local IP address of your Raspberry Pi when it's connected to your router via the ethernet cable. There are lots of ways to do this, but Advanced IP Scanner is pretty easy to use. It will scan your home network and list all connected devices and their local IP addresses.



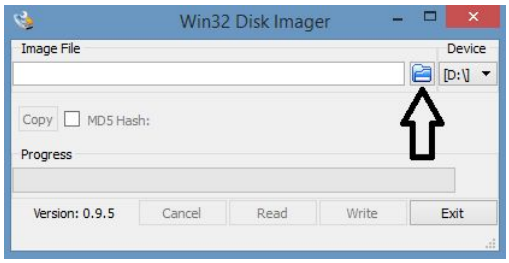
Another way to find your Pi's local IP address is to log into your router's configuration page and get a list of the connected devices. But you need to know your router's username and password to access it. If you want to do it this way, find the router's product number on the back, then search online for the user manual. There should be instructions on how to access the router's configuration page and how to view the connected devices.

NOTE : If you've got a mobile device handy, there are plenty of IP scanners on the play store.

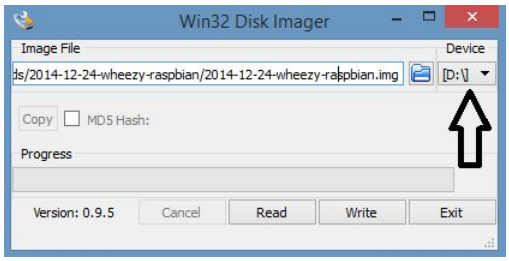
FLASH THE RASPBIAN OS.

Now that we have everything we need, let's write the operating system image to the microSD card.

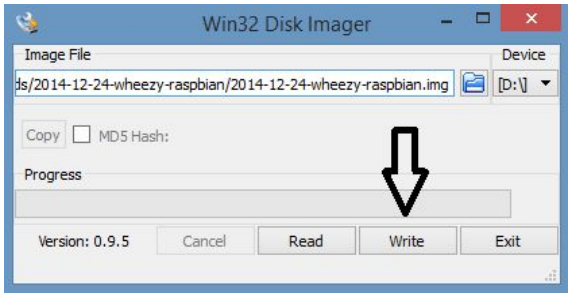
Insert a blank microSD card into your computer using the SD card adapter. Then open Win32DiskImager, and select the operating system image file that was extracted earlier by clicking on the little folder icon:



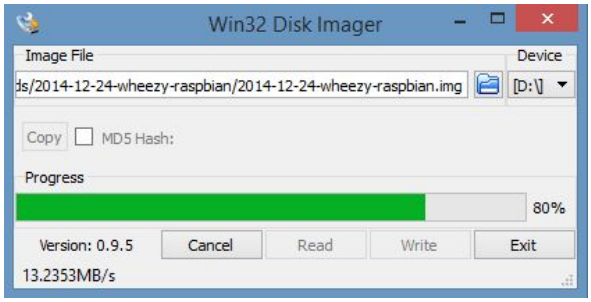
Next, select the drive that has the microSD card from the drop-down list below "Device":



Now click "Write" on the bottom of the window. This will take from 1 to 15 minutes depending on your computer and the size of your microSD card:



Wait for the process to complete:



NOTE : SSH disabled post Nov 2016 by default.

SSH was disabled by default in Raspbian versions released after November 2016. One way to enable it is to access the raspi-config menu and turn it on. However, we're setting it up headless so we don't have access to the command prompt yet. Luckily there's another way to turn on SSH…

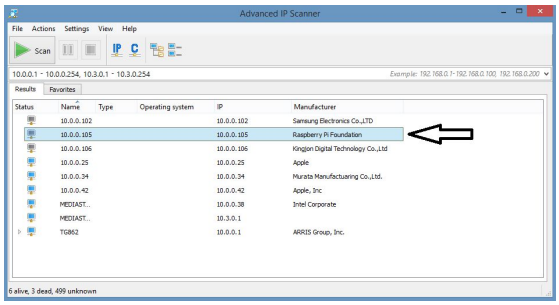
All you need to do is create an empty file called "SSH" and place it in the root partition of the SD card. The file should have no extensions like ".txt" or ".doc". The easiest way to do this is to open up a text editor like Notepad and save the blank page as "SSH.txt". Close the text editor, then rename the file to delete the ".txt" extension. After that, move the file to the top level folder of the SD card (the root partition).

Now you can eject the SD card from your computer and insert it into your Pi.

CONNECTING VIA SSH

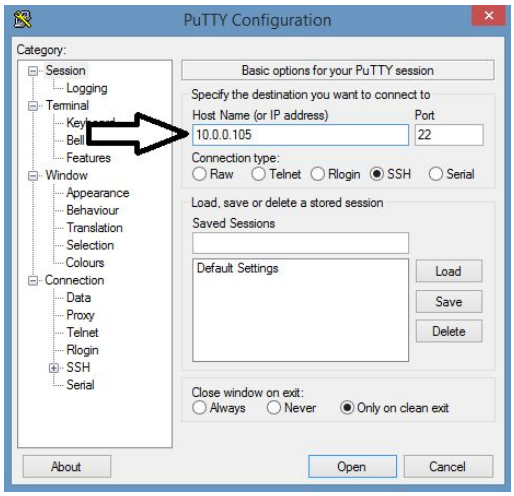
Connect the Pi to your internet router with an ethernet cable, then plug in the power cord so it boots up.

Now we need to find out the local IP address your router has assigned to the Pi. Open up Advanced IP Scanner and click “Scan” in the upper left hand area of the window. A list of the connected devices and their local IP addresses will be generated:



Write down the IP address for the device that says Raspberry Pi Foundation in the “Manufacturer” column. In my case it’s 10.0.0.105.

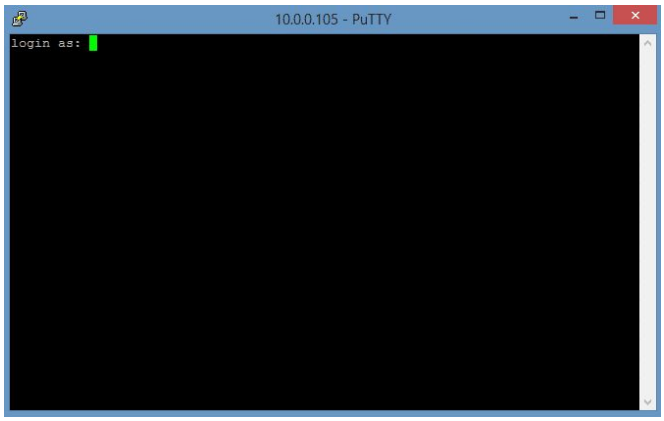
Now it’s time to connect to the Pi. Open up PuTTY, and enter the local IP address into the “Host Name” field:



Now click “Open” to initiate the connection.

On the first connection attempt, a security warning will appear. Just press “Yes”since you’re connecting to your own Pi:

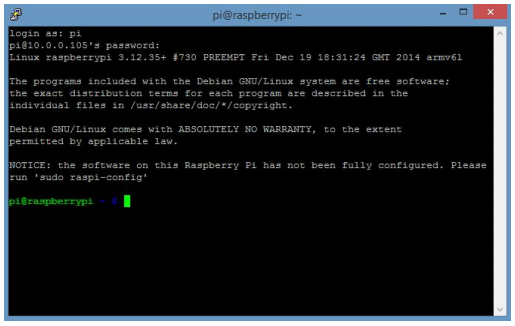
If the SSH connection is successful you’ll be greeted with the login prompt of your Raspberry Pi:



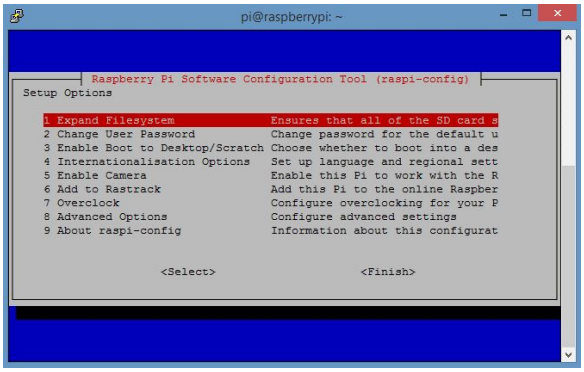
If this is your first login and you haven’t changed the username or password yet, enter pi for the

username, and raspberry as the password.

Now you're logged in to your Raspberry Pi via SSH:



At this point, it's a good idea to configure your Raspberry Pi by entering sudo raspi-config to enter the configuration settings menu:



This is where you can change all of the default settings for Raspbian, and do other useful tasks like expand the file system and overclock the processor.

Now that SSH is set up, the next step is setting up WiFi and establishing a remote desktop connection so you can access the Pi's desktop environment. This tutorial will walk you through that:

Configuration :

Config file :

The Raspberry Pi uses a configuration file instead of the BIOS you would expect to find on a conventional PC. The system configuration parameters, which would traditionally be edited and stored using a BIOS, are stored instead in an optional text file named config.txt. This is read by the GPU before the ARM CPU and Linux are initialised. It must therefore be located on the first (boot) partition of your SD card, alongside bootcode.bin and

start.elf. This file is normally accessible as /boot/config.txt from Linux, and must be edited as root. From Windows or OS X it is visible as a file in the only accessible part of the card. If you need to apply some of the config settings below, but you don't have a config.txt on your boot partition yet, simply create it as a new text file.

Any changes will only take effect after you have rebooted your Raspberry Pi. After Linux has booted, you can view the current active settings using the following commands:

Composite video mode options :

sdtv_mode
The sdtv_mode command defines the TV standard used for composite video output over the yellow RCA jack. The default value is 0.

sdtv_mode	**result**
0	Normal NTSC
1	Japanese version of NTSC – no pedestal
2	Normal PAL
3	Brazilian version of PAL – 525/60 rather than 625/50, different subcarrier
16	Progressive scan NTSC
18	Progressive scan PAL

sdtv_aspect
The sdtv_aspect command defines the aspect ratio for composite video output. The default value is 1.

sdtv_aspect	**result**
1	4:3
2	14:9
3	16:9

sdtv_disable_colourburst
Setting sdtv_disable_colourburst to 1 disables colourburst on composite video output. The picture will be displayed in monochrome, but it may appear sharper.

enable_tvout (Pi 4B only)
On the Raspberry Pi 4, composite output is disabled by default, due to the way the internal clocks are interrelated and allocated. Because composite video requires a very specific clock, setting that clock to the required

speed on the Pi 4 means that other clocks connected to it are detrimentally affected, which slightly slows down the entire system. Since composite video is a less commonly used function, we decided to disable it by default to prevent this system slowdown.

To enable composite output, use the **enable_tvout=1** option. As described above, this will detrimentally affect performance to a small degree.

On older Pi models, the composite behaviour remains the same.

Configuration Steps :

Edit the config.txt file

The config.txt file is located in /boot directory.

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

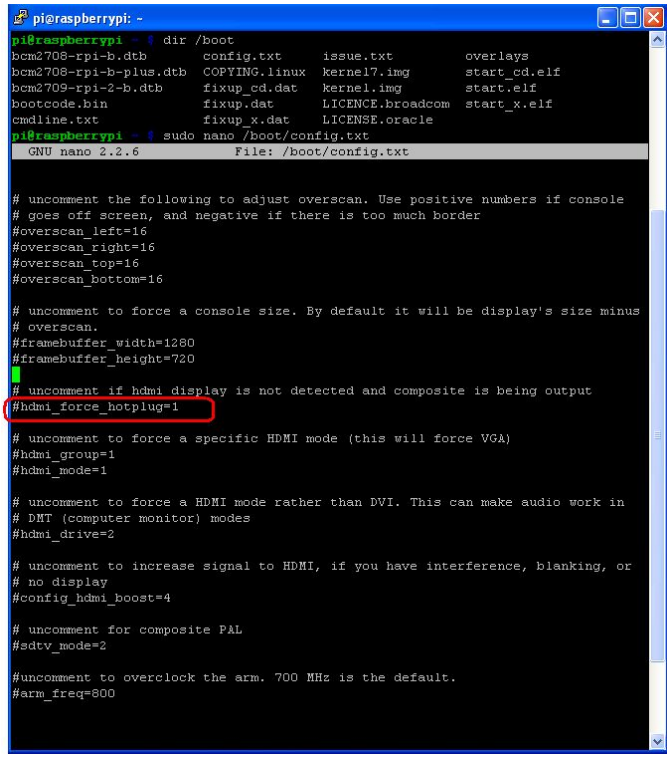
Over there,
`#sdtv_mode=0`

remove the leading 3 and change the number as per the section given above. based on your TV type

Go to following line, and place # at start, if already not there:

`hdmi_force_hotplug=1`
This will make the line like this:

`#hdmi_force_hotplug=1`
Then save the file (Ctrl+o) and exit (Ctrl+x)



Acknowledgements :

We would like to thank our team who've worked tirelessly towards the testing and the compilation of this material. We'd also like to thank the Raspberry Pi Foundation for their help.