

Extracted from:

Raspberry Pi

A Quick-Start Guide

This PDF file contains pages extracted from *Raspberry Pi*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printer versions; the content is otherwise identical.

Copyright © 2012 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

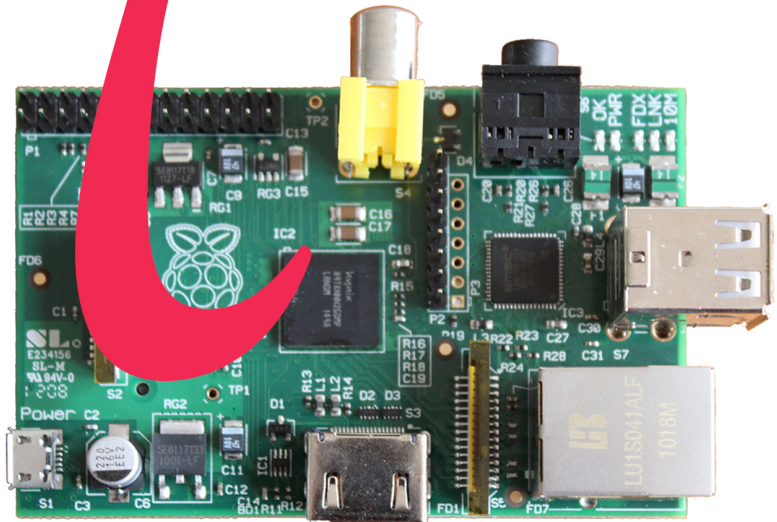
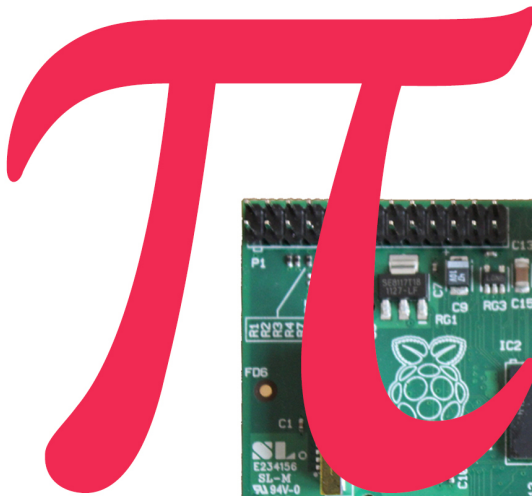
The Pragmatic Bookshelf

Dallas, Texas • Raleigh, North Carolina



Raspberry Pi

A Quick-Start Guide



Maik Schmidt

Edited by Jacquelyn Carter

Raspberry Pi

A Quick-Start Guide

Maik Schmidt

The Pragmatic Bookshelf

Dallas, Texas • Raleigh, North Carolina



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic courses, workshops, and other products can help you and your team create better software and have more fun. For more information, as well as the latest Pragmatic titles, please visit us at <http://pragprog.com>.

The team that produced this book includes:

Jacquelyn Carter (editor)
Kim Wimpsett (copyeditor)
David J Kelly (typesetter)
Janet Furlow (producer)
Juliet Benda (rights)
Ellie Callahan (support)

Copyright © 2012 The Pragmatic Programmers, LLC.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Printed in the United States of America.
ISBN-13: 978-1-937785-04-8
Encoded using the finest acid-free high-entropy binary digits.
Book version: P3.0—April 16, 2013

6.5 Add WiFi to the Pi

Wireless networks are everywhere. Coffee shops, airports, and hotels all offer free WiFi to their customers these days. You probably run a wireless network at home, too, so you can conveniently access your most important devices from your smartphone while you're having a barbeque with the family in the garden. On Windows or Mac OS X you usually do not have to think much about joining wireless networks because the process is nearly seamless.

On the Pi it's different, because its hardware does not support WiFi by default. You need a WiFi stick for the USB port and depending on the stick's type Raspbian might configure it automatically. In some cases you have to configure it manually and in this section you'll learn both ways.

Configuring WiFi Using WiFi Config

The easiest way to configure WiFi on Raspbian is to use a graphical tool named WiFi config. Plug the WiFi stick into a USB port and start the LXDE desktop:

```
pi@raspberrypi ~ $ startx
```

On the desktop you'll find an icon for WiFi Config. Double-click it and you'll see the application shown in [Figure 23, WiFi Config Lets You Scan for Wireless Networks](#).



Figure 23—WiFi Config Lets You Scan for Wireless Networks

Choose wlan0 from the Adapter dropdown menu and then press the Scan button to search for wireless networks. If you've found the network you've been looking for, double-click its name and the configuration dialog in [Figure 24, You Can Change Many Parameters in WiFi Config, on page 7](#) will appear.

Usually, you do not have to change anything here. You only have to enter your WiFi password in the PSK text field. Press the Save button to return to the main menu and then press the Connect button to connect the Pi to the wireless network.

As long as Raspbian and WiFi Config recognize your WiFi stick, WiFi Config certainly is the most convenient way to join wireless networks with the Pi. Unfortunately, WiFi Config doesn't work with all WiFi sticks and sometimes you might need to avoid the desktop system, because you're running a server. In these cases you have to configure WiFi manually.

Configuring WiFi Manually

Configuring WiFi on the command line isn't very convenient, but it isn't rocket science either. Plug your WiFi stick into one of the Pi's USB ports and run the lsusb command to see if the Pi recognizes it properly:

```
pi@raspberrypi ~ $ lsusb
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 002: ID 0424:9512 Standard Microsystems Corp.
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp.
Bus 001 Device 004: ID 050d:0237 Belkin Components F5U237 USB 2.0 7-Port Hub
Bus 001 Device 005: ID 04e8:2018 Samsung Electronics Co., Ltd WIS09ABGN
                      LinkStick Wireless LAN Adapter
Bus 001 Device 006: ID 046d:c312 Logitech, Inc. DeLuxe 250 Keyboard
Bus 001 Device 007: ID 046d:c05a Logitech, Inc. Optical Mouse M90
```

In this case, device 005 is a WiFi stick manufactured by Samsung. You can take a closer look at the Pi's boot message with the dmesg command and see whether the WLAN stick has been initialized properly:

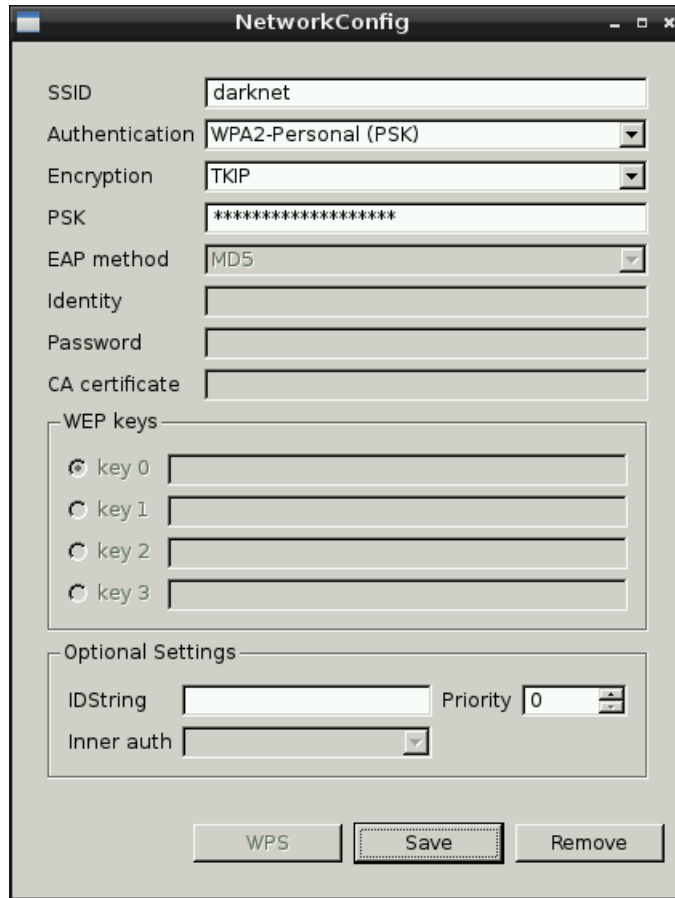


Figure 24—You Can Change Many Parameters in WiFi Config

```
pi@raspberrypi ~ $ dmesg | less
```

```
...
usb 1-1.3.6: new high speed USB device number 5 using dwc_otg
usb 1-1.3.6: New USB device found, idVendor=04e8, idProduct=2018
usb 1-1.3.6: New USB device strings: Mfr=1, Product=2, SerialNumber=3
usb 1-1.3.6: Product: 802.11 n WLAN
usb 1-1.3.6: Manufacturer: Ralink
usb 1-1.3.6: SerialNumber: 1.0
...
```

Press the space bar to go down one page and press 'b' to go up one page. Press 'q' to go back to the shell prompt. As you can see in the current case the Samsung stick uses the WiFi chipset from a company (manufacturer) named Ralink. This chipset is pretty popular, so Debian recognized it out of

the box. If the output of `dmesg` contains any errors right after the initialization of your WiFi stick, check the Pi's Wiki.¹³ Often you have to download the firmware for your WiFi stick manually and reconfigure the Linux kernel.

If no errors occurred, Debian Linux has recognized your WiFi stick successfully. You can use the following command to get the current status of your Pi's wireless network interfaces:

```
pi@raspberrypi ~ $ iwconfig
lo                no wireless extensions.

eth0              no wireless extensions.

wlan0             IEEE 802.11abgn  ESSID:off/any
                  Mode:Managed  Access Point: Not-Associated   Tx-Power=20 dBm
                  Retry  long limit:7   RTS thr:off   Fragment thr:off
                  Power Management:on
```

At the moment the Pi is not connected to a wireless network, but the `wlan0` interface is up and running. The following command searches for a wireless network:

```
pi@raspberrypi ~ $ sudo iwlist scan | grep ESSID
                  ESSID:"darknet"
                  ESSID:"valhalla"
```

In this case, two wireless networks named `darknet` and `valhalla` are within reach. To connect to one of them you have to edit the configuration file `/etc/network/interfaces` using a text editor, such as the nano text editor, for example. To connect to `darknet` add the following lines to the file:

13. http://elinux.org/RPi_VerifiedPeripherals#USB_WiFi_Adapters


```
auto wlan0
iface wlan0 inet dhcp
wpa-ssid darknet
wpa-psk t0p$ecret
```

These lines will activate the wlan0 interface automatically the next time you boot the Pi. Also they'll make the Pi obtain an IP address using DHCP. The Pi will try to join the network named darknet and it'll use the password 't0p\$ecret;. Of course, you have to adjust the network name and the password accordingly.

If you're impatient, you do not have to reboot the Pi. Run the following command to make the Pi join your wireless network:

```
pi@raspberrypi ~ $ sudo ifup wlan0
Internet Systems Consortium DHCP Client 4.2.2
Copyright 2004-2011 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
```

```
Listening on LPF/wlan0/00:12:fb:28:a9:51
Sending on   LPF/wlan0/00:12:fb:28:a9:51
Sending on   Socket/fallback
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 8
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 14
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 14
DHCPPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPOFFER from 192.168.1.1
DHCPACK from 192.168.1.1
bound to 192.168.1.101 -- renewal in 2983 seconds.
```

The Pi has the IP address 192.168.1.101 now and is connected to your network wirelessly (your IP address probably will differ). Use the ping command to check whether you can access a web site like Google, for example:

```
pi@raspberrypi ~ $ ping -c 3 google.com
PING google.com (173.194.69.100) 56(84) bytes of data.
64 bytes from google.com (173.194.69.100): icmp_req=1 ttl=45 time=26.7 ms
64 bytes from google.com (173.194.69.100): icmp_req=2 ttl=45 time=32.3 ms
64 bytes from google.com (173.194.69.100): icmp_req=3 ttl=45 time=34.8 ms

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 26.752/31.338/34.863/3.395 ms
```

As you can see, the Pi is connected to the Internet via WiFi now. Run the following command to get some statistics about the signal strength etc.:

```

pi@raspberrypi ~ $ iwconfig
lo          no wireless extensions.

eth0        no wireless extensions.

wlan0       IEEE 802.11abgn  ESSID:"darknet"
            Mode:Managed  Frequency:2.442 GHz  Access Point: 54:E6:FC:CF:77:8A
            Bit Rate=135 Mb/s   Tx-Power=20 dBm
            Retry  long limit:7   RTS thr:off   Fragment thr:off
            Power Management:on
            Link Quality=40/70  Signal level=-70 dBm
            Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
            Tx excessive retries:1  Invalid misc:6  Missed beacon:0

```

Keep in mind that a computer like the Pi can have more than one IP address. If you connect it via Ethernet and WiFi, for example, your boot message will display something like the following:

```
My IP address is 192.168.2.109 192.168.1.101
```

That means that you've connected your Pi using two network interfaces and for each interface it has a different IP address.