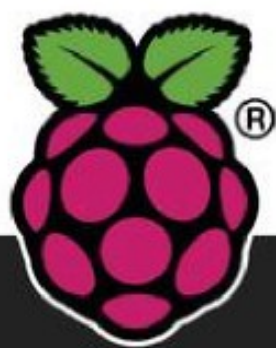


Actualizada  
Segunda Edición



# Raspberry Pi®

## Guía del Usuario



**Eben Upton**

Co-creador de la Raspberry Pi

**Gareth Halfacree**



# Índice de Contenido

## Parte I: Conexión de la Placa

### CAPÍTULO 1

<b>Conozca la Raspberry Pi.....</b>	<b>1</b>
Un Recorrido por la Placa.....	2
El Modelo A.....	3
El Modelo B.....	4
Una Historia de las Revisiones del PCB del Modelo B.....	5
La Revisión 1.....	5
La Revisión 2.....	6
Un Poquito de Historia.....	6
ARM versus x86.....	6
Windows versus Linux.....	7

### CAPÍTULO 2

<b>Comenzando con la Raspberry Pi.....</b>	<b>8</b>
Conexión de la Pantalla.....	9
Video Compuesto.....	9
Video HDMI.....	10
Video DSI.....	11
Conexión del Audio.....	11
Conexión del Teclado y el Ratón.....	12
Instalando NOOBS en una Tarjeta SD.....	14
Conexión del Almacenamiento Externo.....	15
Conexión de la Red.....	16
Red Cableada.....	17
Red Inalámbrica.....	18
Conexión de la Fuente de Alimentación.....	19
Instalación del Sistema Operativo.....	20
Instalación a través de NOOBS.....	20
Instalación Manual.....	22

### CAPÍTULO 3

<b>Administración del Sistema Linux.....</b>	<b>26</b>
Linux: Una Descripción General.....	27
Comandos Básicos de Linux.....	28
Introducción a Raspbian.....	29
Sobre el Padre de Raspbian, Debian.....	32
Alternativas a Raspbian.....	33
Utilizando Dispositivos de Almacenamiento Externo.....	33
Creación de una Nueva Cuenta de Usuario.....	34
Sistema de Archivos.....	35
Distribución Lógica.....	36
Distribución Física.....	37
Instalación y Desinstalación de Software.....	37
Obtener Software desde la Pi Store.....	37
Obtener Software desde Otros Lugares.....	39
Encontrar el Software que Desea.....	40
Instalación del Software.....	41
Desinstalación del Software.....	42
Actualización del Software.....	43
Apagando la RasPi de Manera Segura.....	43

## CAPÍTULO 4

<b>Solución de Problemas.....</b>	<b>44</b>
Diagnóstico del Teclado y el Ratón.....	45
Diagnóstico de la Fuente de Alimentación.....	46
Diagnóstico de la Pantalla.....	48
Diagnóstico del Arranque.....	48
Diagnóstico de la Red.....	49
El Kernel de Emergencia.....	51

## CAPÍTULO 5

<b>Configuración de la Red.....</b>	<b>53</b>
Red Cableada.....	54
Redes Inalámbricas.....	57
Instalación del Firmware.....	57
Utilizando wpa_gui para Conectarse a una Red Inalámbrica.....	60
Utilizando la Terminal para Conectarse a una Red Inalámbrica.....	63

## CAPÍTULO 6

<b>La Herramienta de Configuración del Software de la Raspberry Pi.....</b>	<b>70</b>
Ejecución de la Herramienta.....	71
La Pantalla de Opciones del Sistema.....	71
1 Expand Filesystem.....	72
2 Change User Password.....	72
3 Enable Boot to Desktop.....	72
4 Internationalisation Options.....	73
5 Enable Camera.....	74
6 Add to Rastrack.....	75
7 Overclock.....	75
8 Advanced Options.....	76
9 About raspi-config.....	79

## CAPÍTULO 7

<b>Configuración Avanzada de la Raspberry Pi.....</b>	<b>80</b>
Edición de los Archivos de Configuración a través de NOOBS.....	81
Configuraciones Hardware – config.txt.....	83
Modificación de la Pantalla.....	84
Opciones de Arranque.....	86
Overclockeando la Raspberry Pi.....	86
Desactivación de la Caché L2.....	89
Activación del Modo de Prueba.....	90
Particionamiento de la Memoria.....	90
Configuraciones Software - cmdline.txt.....	91

## Parte II: Construyendo un Centro Multimedia, Máquina Productiva o Servidor Web

## CAPÍTULO 8

<b>La RasPi como una PC Teatro en Casa.....</b>	<b>95</b>
Reproduciendo Música desde la Consola.....	96
HTPC Dedicado con Raspbmc.....	98
Streaming Multimedia por Internet.....	99
Streaming Multimedia por la Red Local.....	101
Configurando Raspbmc.....	103

## CAPÍTULO 9

<b>La RasPi como una Máquina Productiva.....</b>	<b>106</b>
Utilizando Aplicaciones Basadas en la Nube.....	107



Utilizando LibreOffice.....	109
Edición de Imágenes con The Gimp.....	110

## CAPÍTULO 10

<b>La RasPi como un Servidor Web.....</b>	<b>114</b>
Instalación de la Pila LAMP.....	115
Instalación de WordPress.....	119

## Parte III: Programando con la Raspberry Pi

## CAPÍTULO 11

<b>Una Introducción a Scratch.....</b>	<b>125</b>
Presentando a Scratch.....	126
Ejemplo 1: Hola Mundo.....	126
Ejemplo 2: Animación y Sonido.....	130
Ejemplo 3: Un Juego Sencillo.....	132
Robótica y Sensores.....	137
Sensibilidad con la Picoboard.....	138
Robótica con LEGO.....	138
Lectura Sugerida.....	138

## CAPÍTULO 12

<b>Una Introducción a Python.....</b>	<b>139</b>
Presentando a Python.....	140
Ejemplo 1: Hola Mundo.....	140
Ejemplo 2: Comentarios, Entradas, Variables y Bucles.....	144
Ejemplo 3: Juego con pygame.....	148
Ejemplo 4: Python y la Gestión de Redes.....	154
Lectura Sugerida.....	159

## Parte IV: Hardware Hacking

## CAPÍTULO 13

<b>Aprendiendo a Hackear Hardware.....</b>	<b>164</b>
Equipo Electrónico.....	165
Interpretación de los Códigos de Colores de las Resistencias.....	167
Fuente de Componentes.....	169
Fuentes en Línea.....	169
Fuentes Fuera de Línea.....	170
Hobby Especialistas.....	170
Subiéndose a la Breadboard.....	171
Una Breve Guía de Soldadura.....	173

## CAPÍTULO 14

<b>El Puerto GPIO.....</b>	<b>179</b>
Identificando la Revisión de su Placa.....	180
Diagramas Pinout el Puerto GPIO.....	180
Características del GPIO.....	182
El Bus Serie UART.....	182
El Bus I <sup>2</sup> C.....	182
El Bus SPI.....	183
Utilizando el Puerto GPIO en Python.....	183
Salida GPIO: Parpadeo de un LED.....	183
Entrada GPIO: Lectura de un Botón.....	188

## CAPÍTULO 15

<b>El Módulo de Cámara de la Raspberry Pi.....</b>	<b>192</b>
¿Por Qué Utilizar el Módulo de Cámara?.....	193
Instalación del Módulo de Cámara.....	194
Activación del Módulo de Cámara.....	197
Captura de Imágenes Fijas.....	198
Grabación de Video.....	200
Fotografía de Lapso de Tiempo desde la Línea de Comandos.....	201

## CAPÍTULO 16

<b>Placas de Expansión.....</b>	<b>206</b>
Slice of Pi de Ciseco.....	207
Prototyping Pi Plate de Adafruit.....	210
Gertboard de Fen Logic.....	213

## Parte IV: Apéndices

### APÉNDICE A

<b>Python Recetas.....</b>	<b>220</b>
Raspberry Pi Serpiente (Capítulo 12, Ejemplo 3).....	221
IRC User List (Capítulo 12, Ejemplo 4).....	222
Entrada y Salida GPIO (Capítulo 14).....	224

### APÉNDICE B

<b>Rápida Referencia del Módulo de Cámara.....</b>	<b>225</b>
Opciones Compartidas.....	226
Opciones de Raspistill.....	228
Opciones de Raspivid.....	229
Opciones de Raspiyuv.....	230

### APÉNDICE C

<b>Modos de Pantalla HDMI.....</b>	<b>231</b>
------------------------------------	------------





Parte I

# Conexión de la Placa

<b>Capítulo 1</b>	Conozca la Raspberry Pi
<b>Capítulo 2</b>	Comenzando con la Raspberry Pi
<b>Capítulo 3</b>	Administración del Sistema Linux
<b>Capítulo 4</b>	Solución de Problemas
<b>Capítulo 5</b>	Configuración de la Red
<b>Capítulo 6</b>	La Herramienta de Configuración del Software de la Raspberry Pi
<b>Capítulo 7</b>	Configuración Avanzada de la Raspberry Pi



# Capítulo 1

## Conozca la Raspberry Pi

**Su placa Raspberry Pi** es una maravilla en miniatura, aguada en su interior un importante poder de computo en un tamaño no más grande que el de una tarjeta de crédito. Es capaz de realizar cosas extraordinarias, pero antes de comenzar a utilizarla hay unas cuantas cosas que necesita conocer primero.

#### CONSEJO

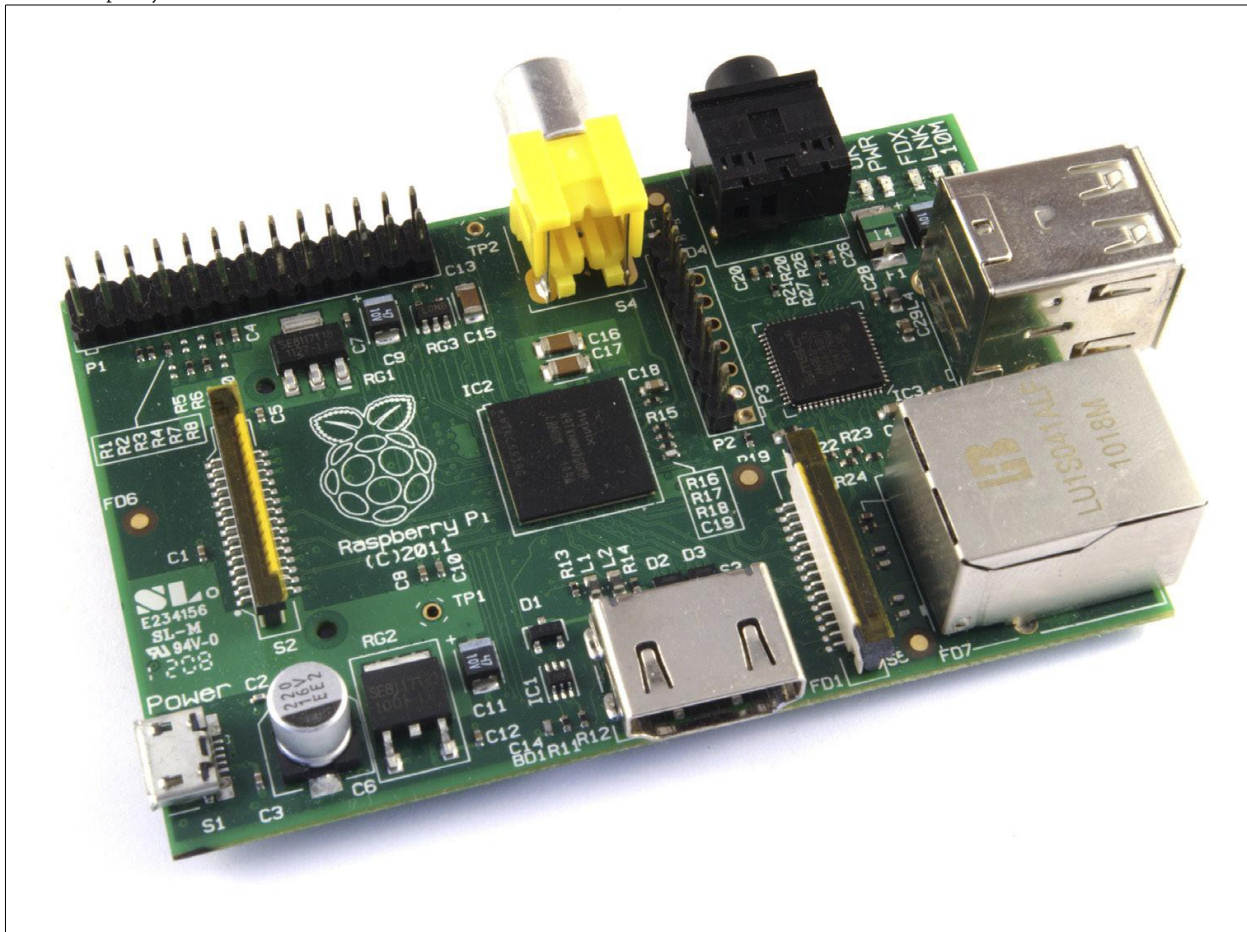
Si está impaciente por comenzar, avance un par de páginas más para que averigüe cómo conectar su teclado, ratón y pantalla a la Raspberry Pi, así como para conocer cómo instalar un sistema operativo e ir directamente a utilizar la RasPi.

## Un Recorrido por la Placa

La Raspberry Pi se encuentra actualmente disponible en dos modelos diferentes, conocidos como el Modelo A y el Modelo B. Aunque existen diferencias (con el Modelo A sacrificando algo de funcionalidad en razón de disminuir su costo y los requerimientos de energía) ambos comparten un montón de similitudes como se dará cuenta en este capítulo. La Figura 1-1 muestra una Raspberry Pi Modelo B.

Figura 1-1

La Placa Raspberry Pi



En el centro de todas las placas Raspberry Pi se encuentra un *semiconductor* cuadrado, más comúnmente conocido como *circuito integrado* o *chip*. Este es el *módulo system-on-chip* (SoC) Broadcom BCM2835, el encargado de proporcionarle a la RasPi sus capacidades de procesamiento de propósito general, de renderización de gráficos y de entrada/salida. Apilado sobre este chip se encuentra otro semiconductor, este es el encargado de proveer la *memoria* a la RasPi para el almacenamiento temporal de datos mientras sus programas se encuentran en ejecución. Este tipo de memoria es conocida como *memoria RAM* (*memoria de acceso aleatorio*), ya que la computadora puede



leer o escribir en cualquier parte de la memoria y en cualquier momento. La RAM es *volátil*, lo que significa que cualquier cosa almacenada en esta memoria se eliminará cuando la RasPi pierda su energía o sea desenchufada.

Por la parte de arriba y de abajo del SoC se encuentran las salidas de video de la RasPi. El conector plata (abajo) es un puerto *HDMI (High Definition Multimedia Interface)*, es del mismo tipo de conector que se puede encontrar en los reproductores de video y muchos decodificadores de cable y satélite. Cuando es conectado a un televisor o monitor moderno, el puerto HDMI proporciona video en alta resolución y audio digital. El conector amarillo (arriba) es un puerto de *video compuesto* y está diseñado para conectarse a televisores antiguos que no cuentan con conexión HDMI. La calidad del video es más baja que la que está disponible a través del HDMI y no transmite audio; en cambio, el audio es suministrado como una señal analógica a través del *conector de audio de 3.5mm* a la derecha del conector del video compuesto.

Los pines en la parte superior izquierda de la RasPi componen el header o *conector con entradas-salidas de propósito general (GPIO)* que puede utilizarse para conectar la RasPi a otros hardware. El uso más frecuente de este puerto es para conectar una *placa de expansión (Add-On)*. Una selección de estas placas es descrita en el Capítulo 16, “Placas de Expansión”. El puerto GPIO es extremadamente poderoso, pero también frágil; cuando manipule la RasPi, siempre evite tocar estos pines, así como también nunca conectar nada en ellos mientras la RasPi se encuentra encendida.

El conector de plástico y metal debajo del puerto GPIO es el puerto *DSI (Display Serial Interface)*, que sirve para conectar sistemas de pantalla plana controladas digitalmente. Estas pantallas rara vez son utilizadas excepto por los desarrolladores profesionales en sistemas embebidos, ya que el puerto HDMI es más flexible. Un segundo conector de plástico y metal, se encuentra a la derecha del puerto HDMI, es el puerto *CSI (Camera Serial Interface)*, que proporciona una conexión de alta velocidad para el módulo de cámara de la Raspberry Pi u otros sistemas de cámaras con conexión CSI compatibles con la RasPi. Para más detalles sobre el puerto CSI, ver el Capítulo 15, “El Módulo de Cámara de la Raspberry Pi”.

En la esquina inferior de la placa está la *toma de alimentación* de la RasPi. Esta es una toma *micro-USB*, la misma que está presente en la mayoría de los modernos teléfonos inteligentes (smartphones) y tablets. La conexión de un cable micro-USB a un adaptador de energía adecuado (detallado en el Capítulo 2 “Comenzando con la Raspberry Pi”) encenderá a la Raspberry Pi; a diferencia de una PC o laptop, la RasPi no cuenta con un botón de encendido y arranca inmediatamente cuando el cable de la alimentación es conectado.

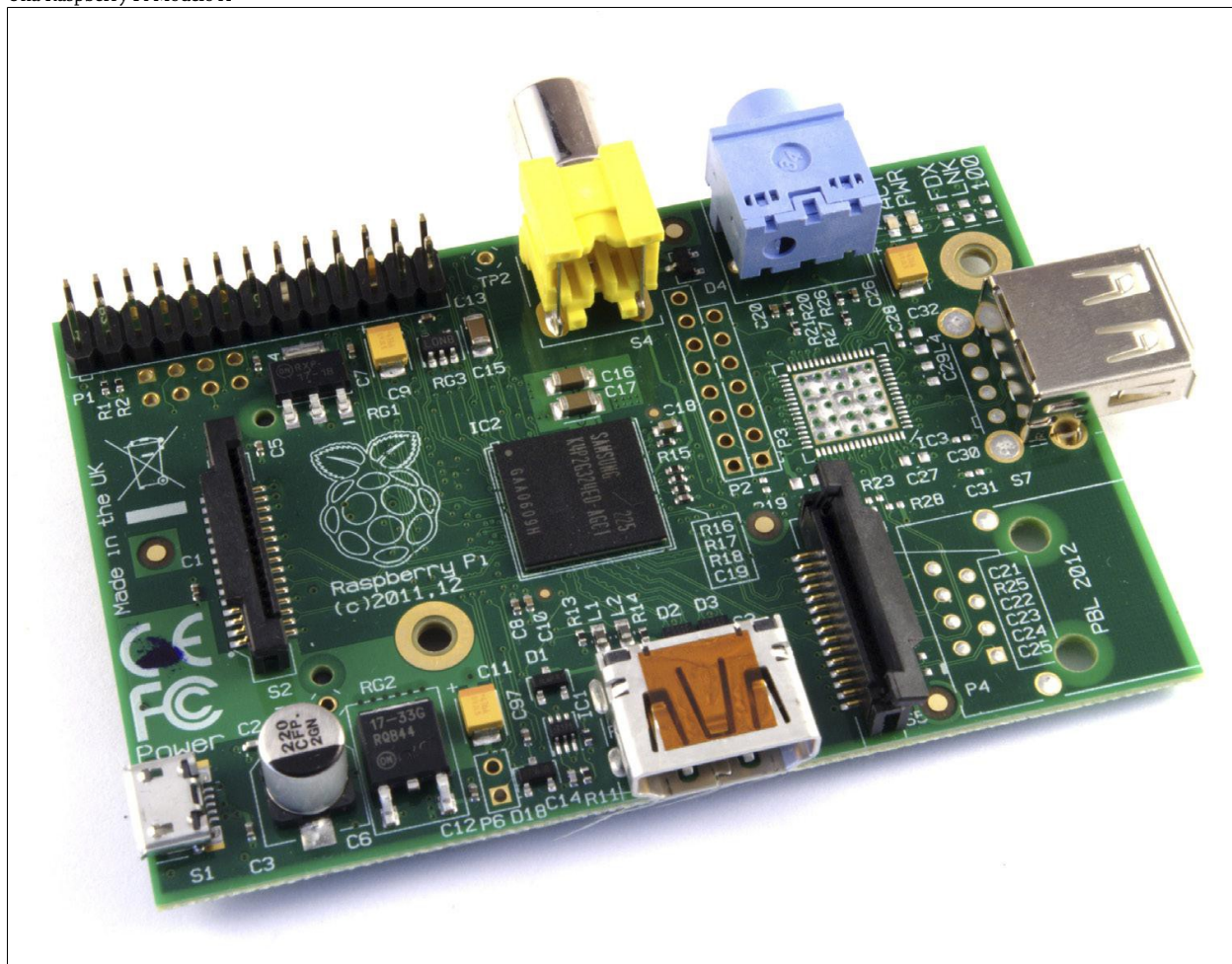
En la parte inferior de la placa Raspberry Pi, a mano izquierda, se encuentra una *ranura para tarjetas SD*. Una tarjeta de memoria SD (Secure Digital) proporciona el almacenamiento para el sistema operativo, los programas, datos y otros archivos, y *no es volátil*; a diferencia de la RAM volátil, la memoria SD conservará su información incluso cuando la energía se pierda. En el Capítulo 2, “Comenzando con la Raspberry Pi”, aprenderá cómo preparar una tarjeta SD para su uso con la RasPi, así como también cómo instalar un sistema operativo (un proceso conocido como *flashing* o *flasheo*).

El borde a mano derecha de la RasPi tendrá diferentes conectores dependiendo de qué modelo de Raspberry Pi tenga, el Modelo A o el Modelo B. Arriba de éstos se encuentran una serie de *LEDs (Light Emitting Diodes)*. Los dos primeros LEDs que están etiquetados con ACT y PWR proporcionan una notificación de actividad y de energía respectivamente, y están presentes en todas las placas.

## El Modelo A

Es la más económica de las Raspberrys, el Modelo A que se muestra en la Figura 1-2 está diseñado para ser asequible pero aún flexible. Aparte de ser económica si se compara al Modelo B, el Modelo A consume menos energía y es una buena elección para proyectos que utilizan la energía solar, eólica o de batería. Aunque el SoC BCM2835 del Modelo A es tan poderoso como el del Modelo B, este viene sólo con 256MB de memoria. Esto es una consideración importante al momento de decidir qué modelo comprar, ya que puede ocasionar que las aplicaciones más complejas se ejecuten lentamente (en particular, aquellas aplicaciones que convierte la RasPi en un servidor, como se describe en el Capítulo 10, “La RasPi como un Servidor Web”).

Figura 1-2  
Una Raspberry Pi Modelo A

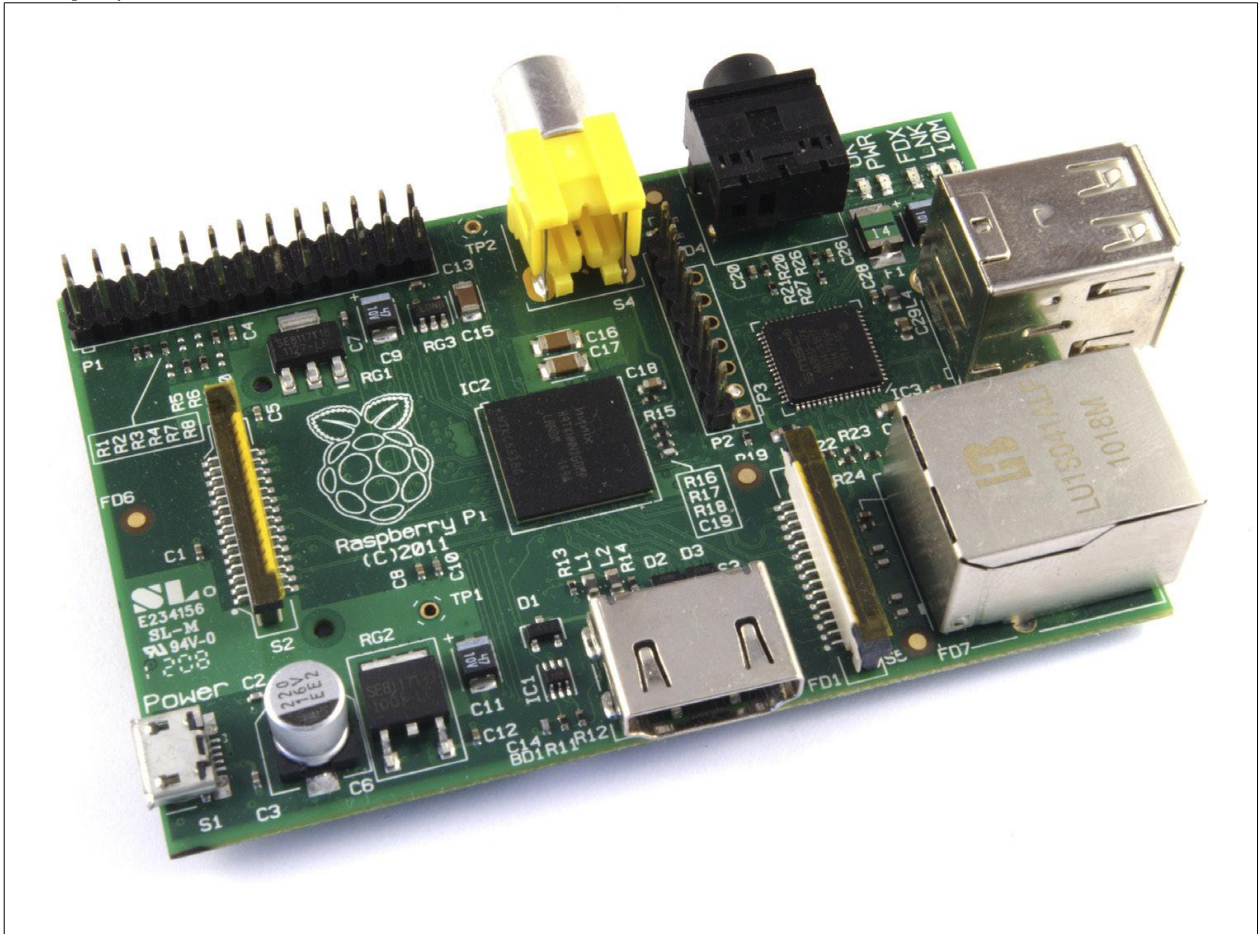


El Modelo A sólo tiene un puerto *USB* (*Universal Serial Bus*) en su borde a mano derecha. Este es el mismo tipo de puerto que se puede encontrar en las computadoras de escritorio y laptops, y permite que la RasPi pueda ser conectada a casi cualquier periférico USB compatible. Por lo general, el puerto USB es utilizado para conectar un teclado y poder interactuar con la RasPi. Si además desea utilizar un ratón al mismo tiempo, necesitará comprar un *hub USB* para agregar más puertos al Modelo A o como alternativa, comprar un teclado con la funcionalidad del ratón integrada.

## El Modelo B

La Raspberry Pi Modelo B que se muestra en la Figura 1-3 es más costosa que la Modelo A, pero viene con algunas ventajas considerables. En su interior incluye el doble de memoria (512MB), mientras que el exterior hay puertos adicionales que no están disponibles en el modelo económico. Para muchos usuarios, el Modelo B es una inversión que vale la pena; sólo aquellos con requisitos particulares de peso, espacio o consumo energético deben considerar adquirir el Modelo A para su uso general.

Figura 1-3  
Una Raspberry Pi Modelo B



El Modelo B tiene dos puertos USB en el borde derecho de la placa, permitiendo conectar un teclado y un ratón, o aún más, permite disponer de dos puertos libres de repuesto para accesorios adicionales tales como los dispositivos de almacenamiento externo o interfaces de hardware. Además, incluye un puerto *Ethernet* para conectar la RasPi a una red cableada, lo que permite tener acceso a Internet y hace posible que otros dispositivos en la red puedan acceder a la RasPi (suponiendo que saben el nombre de usuario y contraseña o que la RasPi se haya configurado como un servidor como se describe en el Capítulo 10, “La RasPi como un Servidor Web”).

## Una historia de las Revisiones del PCB del Modelo B

Aunque la Raspberry Pi Modelo B hoy tenga 512MB de memoria y dos puertos USB, esto no fue siempre así. El Modelo B disponible hoy en día es conocido como la *Revisión 2*, ya que es la segunda placa en hacerse con la designación Modelo B. Si usted ha comprado la Raspberry Pi modelo B de segunda mano o la compró hace ya algún tiempo, es posible que tenga una placa Revisión 1 que cuenta con algunas diferencias.

### La Revisión 1

La placa Revisión 1 original del modelo B de la Raspberry Pi sólo tiene 256MB de RAM. Además tiene una header o pines GPIO un poco diferentes, visualmente idénticos a las revisiones posteriores pero que tienen ciertas características asignadas a diferentes pines, como se explicará en el Capítulo 14, “El Puerto GPIO”. Esta es la diferencia más importante: todas las demás revisiones y modelos de Raspberry Pi comparten el mismo diseño de GPIO, así que si tiene una Modelo B original revisión 1 puede que necesite hacer ajustes en las instrucciones y programas antes de que puedan funcionar satisfactoriamente.

## La Revisión 2

Introducida poco antes del lanzamiento del Modelo A, la Revisión 2 de la Raspberry Pi incluye el doble de memoria (512MB) que la original. También introduce el nuevo header GPIO estandarizado y en común con el Modelo A. Un header adicional, que también se encuentra presente en las placas Modelo A (etiquetado con P5) y ubicado justo debajo del header GPIO, es un signo seguro de que la placa Modelo B es la más reciente Revisión 2.

## Un poquito de historia

Antes de pasar al Capítulo 2, “Comenzado con la Raspberry Pi”, es una buena idea que primero se familiarice con algunos detalles de fondo de la RasPi así como de su concepción. Aunque la RasPi pueda ser utilizada como una computadora de propósito general, capaz de realizar los mismos trabajos que cualquier otra computadora de escritorio, laptop o servidor (aunque más lentamente) ha sido diseñada como una *computadora dentro de una sola placa* enfocada a los aficionados y usos educativos, y como tal, se diferencia de una computadora “normal” en un par de aspectos importantes.

## ARM versus x86

El procesador en el interior de su Raspberry Pi es un procesador multimedia Broadcom BCM2835 system-on-chip (SoC). Esto quiere decir que la mayor parte de los componentes del sistema, incluidos la unidad central de procesamiento y la de gráficos junto con el audio y el hardware de comunicaciones, se encuentran integrados dentro de aquel único componente oculto ubicado justo debajo del chip de memoria al centro de la tarjeta.

No es sólo el diseño del SoC lo que hace al BCM2835 diferente del procesador de su PC o laptop. Lo que lo hace también diferente es que utiliza una *arquitectura de conjunto de instrucciones (Instruction Set Architecture, ISA)* distinta, conocida como ARM.

Desarrollada por Acorn Computers años atrás a finales de 1980, la arquitectura ARM es relativamente poco conocida en el mundo de las computadoras de escritorio. En donde destaca, sin embargo, es en los dispositivos móviles: el teléfono en su bolsillo es casi seguro que cuente con al menos un núcleo de procesamiento basado en ARM escondido en su interior. La combinación de la arquitectura *RISC (Simple Reduced Instruction Set)* y su bajo consumo energético lo convierten en la opción perfecta frente a los chips de computadoras de escritorio que demandan altos consumos y arquitecturas *CISC (Complex Instruction Set)*.

El BCM2835 basado en ARM es el secreto que explica cómo la Raspberry Pi es capaz de funcionar con tan sólo una fuente de alimentación de 5V 1A suministrada por el puerto micro-USB a bordo. Es también la razón por la cual no encontrará ningún disipador térmico sobre el dispositivo: el bajo consumo de energía del chip se traduce directamente en muy poco calor residual, incluso durante las tareas de procesamiento más complejas.

Esto, sin embargo, significa que la Raspberry Pi no es compatible con el software de las PC tradicionales. La mayoría del software para computadoras de escritorio y laptops se construyen teniendo en cuenta la arquitectura de conjunto de instrucciones x86, presente en los procesadores como AMD, Intel y VIA. Por consiguiente, este software no funciona en la Raspberry Pi que se basa en la arquitectura ARM.

El BCM2835 utiliza una generación del diseño del procesador ARM conocida como ARM11, que a su vez está diseñada en torno a una versión de la arquitectura de conjunto de instrucciones conocida como ARMv6. Vale la pena recordar que ARMv6 es una arquitectura ligera y potente, pero tiene un rival en la arquitectura más avanzada ARMv7 utilizada por la familia de procesadores ARM Cortex. El software desarrollado para la ARMv7, al igual que el desarrollado para la x86, es por desgracia incompatible con el BCM2835 de la Raspberry Pi, aunque los desarrolladores generalmente pueden convertir el software para adecuarlo, un proceso conocido como *porting*.

Esto no quiere decir que va a estar limitado en sus opciones. Como descubrirá más adelante en el libro, hay un montón de software disponible para el conjunto de instrucciones ARMv6, y en tanto que la popularidad de la Raspberry Pi siga aumentando, el software disponible seguirá creciendo. En este libro, también aprenderá cómo crear su propio software para su RasPi aun si no tiene experiencia alguna con la programación.

## Windows versus Linux

Otra diferencia importante entre la Raspberry Pi y su PC de escritorio o laptop, aparte de su tamaño y su costo, es el sistema operativo (el software que permite controlar su computadora) que utiliza.

La mayoría de las PCs y laptops disponibles hoy en día funcionan con alguno de estos dos sistemas operativos: Microsoft Windows o Apple OS X. Ambas plataformas son de *código cerrado*, creados en un ambiente reservado utilizando técnicas patentadas.

Estos sistemas operativos son conocidos como de código cerrado por la naturaleza de su *código fuente*, es decir, la receta en lenguaje de computadora que le dice al sistema que hacer. En el software de código cerrado, esta receta es mantenida como un secreto muy bien guardado. Los usuarios pueden obtener el software terminado, pero nunca ver cómo está hecho.

La Raspberry Pi, por el contrario, está diseñada para ejecutar un sistema operativo llamado *GNU/Linux* (de ahora en adelante nos referiremos a éste simplemente como Linux). A diferencia de Windows u OS X, Linux es de *código abierto*. Esto quiere decir que es posible descargar el código fuente del sistema operativo por completo y hacer los cambios que uno desee. Nada es ocultado, y todos los cambios hechos están a la vista del público. Este espíritu de desarrollo de código abierto ha permitido a Linux rápidamente ser modificado para poder ejecutarse sobre la Raspberry Pi. Al momento de escribir estas líneas, varias versiones de Linux (conocidas como *distribuciones*) han sido portadas al chip BCM2835 de la Raspberry Pi, incluyendo Raspbian, Pidora y Arch Linux.

Las distintas distribuciones atienden diferentes necesidades, pero todas ellas tienen algo en común: son de código abierto. Además, por lo general, todas son compatibles entre sí: el software escrito en un sistema Debian funcionará perfectamente bien en uno con Arch Linux y viceversa.

Linux no es exclusivo de la Raspberry Pi. Cientos de diversas distribuciones están disponibles para PCs de escritorio, laptops e incluso dispositivos móviles; la popular plataforma Android de Google está desarrollada en encima de un núcleo Linux. Si disfruta la experiencia de utilizar Linux en la Raspberry Pi, podría considerar añadir otros dispositivos informáticos suyos para que lo utilicen también. Estos coexistirán felizmente con su sistema operativo actual, permitiéndole disfrutar de las ventajas de ambos sistemas mientras le brindan un entorno familiar cuando su RasPi no se encuentre disponible.

Al igual que la diferencia que existe entre las arquitecturas ARM y x86, hay un punto clave que hace la diferencia práctica entre Windows, OS X y Linux: el software escrito para Windows u OS X no funciona en Linux. Afortunadamente, hay un montón de alternativas compatibles para la gran mayoría de los productos de software comunes y lo mejor, casi todos son de libre uso y de código abierto como lo es el propio sistema operativo, y incluso se pueden instalar en Windows y OS X para proporcionar una experiencia familiar en las tres plataformas.

# Capítulo 2

Comenzando con la Raspberry Pi

**Ahora que ya** tiene la comprensión básica de cómo la RasPi se diferencia de los otros dispositivos de cómputo, es momento de comenzar. Si acaba de recibir su RasPi, sáquela de su bolsa protectora antiestática y colóquela sobre una superficie plana y no conductora de electricidad antes de continuar con este capítulo.

Para utilizar su RasPi, necesitará algunos periféricos adicionales. Una pantalla le permitirá ver lo que está haciendo, mientras que un teclado y ratón serán sus dispositivos de entrada. En este capítulo, aprenderá cómo conectar éstos a la RasPi, así como hacer una conexión de red en el caso del Modelo B. También aprenderá cómo descargar e instalar un sistema operativo a la RasPi.

## Su experiencia puede variar

La información y las instrucciones en este libro le dan todo lo que necesita para poner en funcionamiento su Raspberry Pi y aprovechar al máximo sus capacidades. Tenga en cuenta que algunos de los programas para la Raspberry Pi están evolucionando tan rápidamente que puede que lo que ve en su pantalla sea ligeramente diferente de lo que ve en algunas de las imágenes de este libro, como pudieran ser nuevas características y opciones que apenas comenzaron a estar disponibles.

## Conexión de la Pantalla

Antes de empezar a usar su Raspberry Pi, necesita conectar una pantalla. La RasPi es compatible con tres salidas de video diferentes: video compuesto, video HDMI y video DSI. El video compuesto y video HDMI son fácilmente accesibles para el usuario final, como se describe en esta sección; mientras que el video DSI requiere de un hardware especializado.

### Video Compuesto

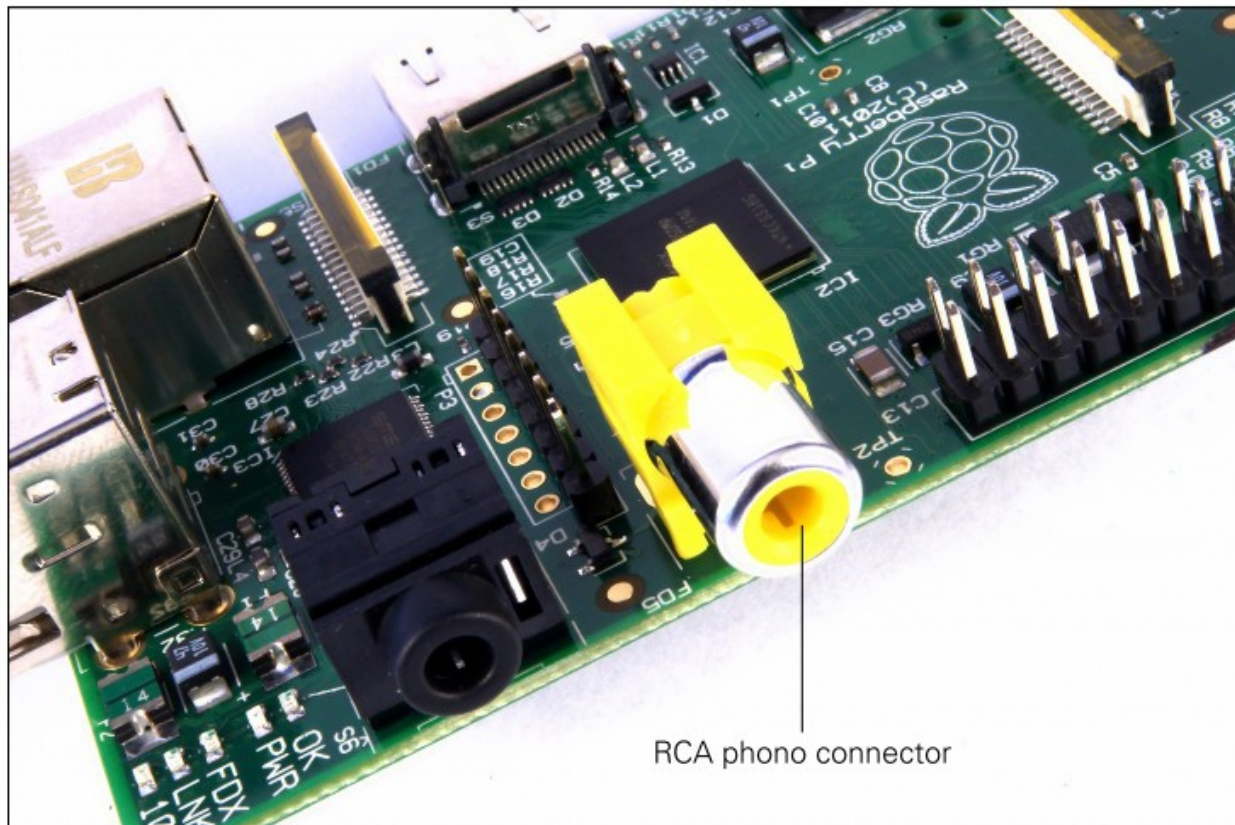
El video compuesto se encuentra disponible a través del puerto color amarillo y plata ubicado en la parte superior de la RasPi conocido como *conector fono RCA* (ver la Figura 1-2), está diseñado para conectar la Raspberry Pi a los viejos dispositivos de pantalla. Como su nombre lo indica, el conector crea una compuesto de colores que se encuentran dentro de una imagen (rojo, verde y azul) y lo envía por un sólo cable al dispositivo de visualización, comúnmente un viejo televisor de tubo de rayos catódicos (CRT).

Cuando no hay otro dispositivo de visualización disponible, una conexión de video compuesto le ayudará a comenzar con la RasPi. La calidad, sin embargo, no será la mejor. Las conexiones de video compuesto son significativamente más propensas a la interferencia, faltas de claridad y funcionan a una *resolución* limitada, es debido a esto que caben muy pocos iconos y líneas de texto en la pantalla a la vez.



Figura 2-1

El conector fono RCA amarillo, para la salida de video compuesto



### Video HDMI

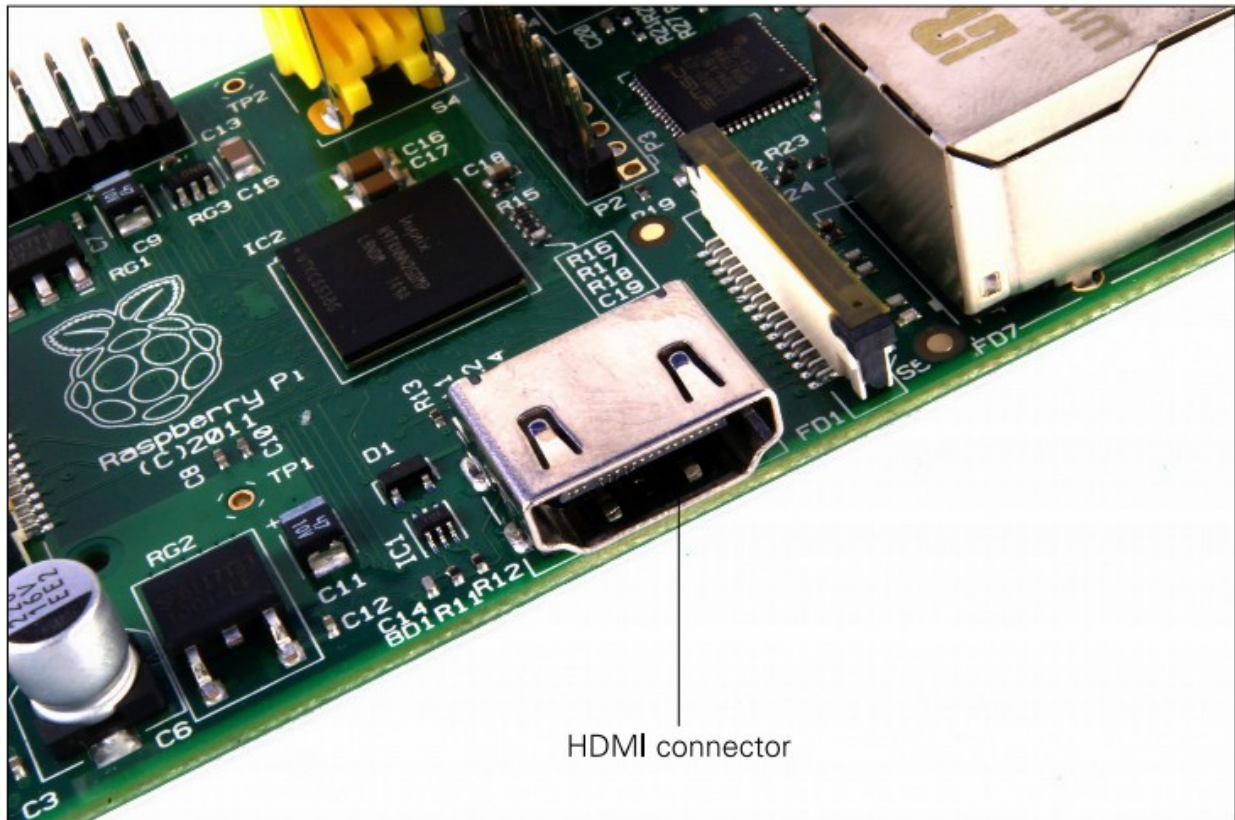
Una mejor calidad de imagen puede obtenerse usando el conector *HDMI* (*High Definition Multimedia Interface*), el único puerto HDMI se encuentra en la parte inferior de la RasPi (ver Figura 2-2). A diferencia de la conexión analógica de video compuesto, el puerto HDMI proporciona una conexión digital de alta velocidad para mostrar imágenes de píxeles perfectos tanto en monitores de computadora como en televisores de alta definición. Al utilizar el puerto HDMI, la RasPi puede desplegar imágenes a la resolución de 1920x1080 Full HD como lo hacen los sistemas HDTV más modernos. A esta resolución, el detalle sobre la pantalla es significativamente superior.

Si cuenta con un monitor de computadora y está esperando poder utilizarlo con la RasPi, es posible que su pantalla no disponga de la entrada HDMI. Esto no es mucho problema: las señales digitales presentes en el cable HDMI pueden convertirse a un estándar común de computadora llamado *DVI* (*Digital Video Interconnect*). Comprando un cable HDMI a DVI, será capaz de conectar el puerto HDMI a un monitor con conectividad DVI-D.

Si su monitor tiene una entrada VGA (un conector con forma de D con 15 pines, comúnmente en color azul y plata) la Raspberry Pi no podrá conectarse. Para utilizar este tipo de monitor, necesitará comprar un *adaptador*; busque modelos que conviertan HDMI a VGA y que en particular mencionen que son compatibles con la Raspberry Pi, y simplemente conecte el conector HDMI del adaptador a la RasPi y el cable de su monitor al otro extremo del adaptador.



Figura 2-2  
El conector plata HDMI, para la salida de video de alta definición.



### Video DSI

La última salida de video que tiene la RasPi puede localizarla encima de la ranura para la tarjeta SD en la parte superior de la placa de circuito impreso (es un pequeño conector de cinta protegido por una capa de plástico). Esta salida es para un estándar de video conocido como *DSI (Display Serial Interface)*, que se utiliza en los monitores de pantalla plana de las tablets y los smartphones. Las pantallas con un conector DSI raramente se encuentran disponibles para su venta al por menor, y por lo general están reservadas para los ingenieros que buscan crear un sistema compacto y autónomo. Una pantalla DSI puede conectarse mediante la inserción de un cable de cinta en el conector correspondiente de la RasPi, para los principiantes, el uso de pantallas compuestas y HDMI es los más recomendado.

### Conexión del Audio

Si está usando el puerto HDMI de la Raspberry Pi, obtener el audio es sencillo: cuando está configurado apropiadamente, el puerto HDMI transporta ambas señales, la de video y la de audio. Esto significa que puede conectar un único cable a su pantalla para disfrutar del sonido y las imágenes.

Asumiendo que está conectando la RasPi a una pantalla HDMI estándar, no habrá otras cosas más que hacer por el momento. Así que, por ahora, sólo basta con conectar el cable.

Si está utilizando la RasPi con un monitor DVI-D a través de un adaptador o cable, el audio no será incluido. Esto destaca la principal diferencia que existe entre HDMI y DVI: mientras que HDMI puede llevar las señales de audio, DVI no puede, y en cambio sólo se usa exclusivamente para las señales de video.

Para aquellas personas con monitores DVI-D, o que utilizan la salida de video compuesto, una *ranura de audio de 3.5 mm* en color negro se ubica en la parte superior del borde de la RasPi junto al conector amarillo RCA para proporcionar salida analógica de audio (ver Figura 2-1). Este es el mismo conector utilizado en los audífonos y micrófonos de los equipos de audio de consumo, y se conecta de la misma forma. Si lo desea, puede simplemente conectar un par de audífonos a este puerto para acceder lo más pronto posible al audio.

---

**CONSEJO**

Mientras los audífonos estén conectados directamente a la Raspberry Pi, podrá encontrar que el volumen es un poco escaso. Si le es posible, en su lugar, conecte un par de altavoces con alimentación propia. El amplificador en el interior de los altavoces le ayudará a aumentar la señal a un nivel más audible.

---

Si está buscando algo más permanente, puede utilizar unos altavoces estándar de PC que cuentan con conector 3.5 mm o puede comprar algunos cables adaptadores. Para los usuarios de video compuesto, un cable *3.5 mm a fono RCA* les resultará útil. Este cable proporciona las dos conexiones fono RCA (blanco y rojo) que se sitúan justo al lado de la conexión de video, cada uno llevando un canal de la señal de audio estéreo a la TV.

Para aquellos que conectan la Raspberry Pi a un amplificador o equipo de música, necesitarán de un cable 3.5 mm a fono RCA o un cable 3.5 mm a 3.5 mm, dependiendo de las conexiones libres que tengan en su sistema. Ambos cables son muy económicos y son fáciles de encontrar en las tiendas de electrónica de consumo, o pueden adquirirse incluso más baratas en las tiendas en línea como Amazon.

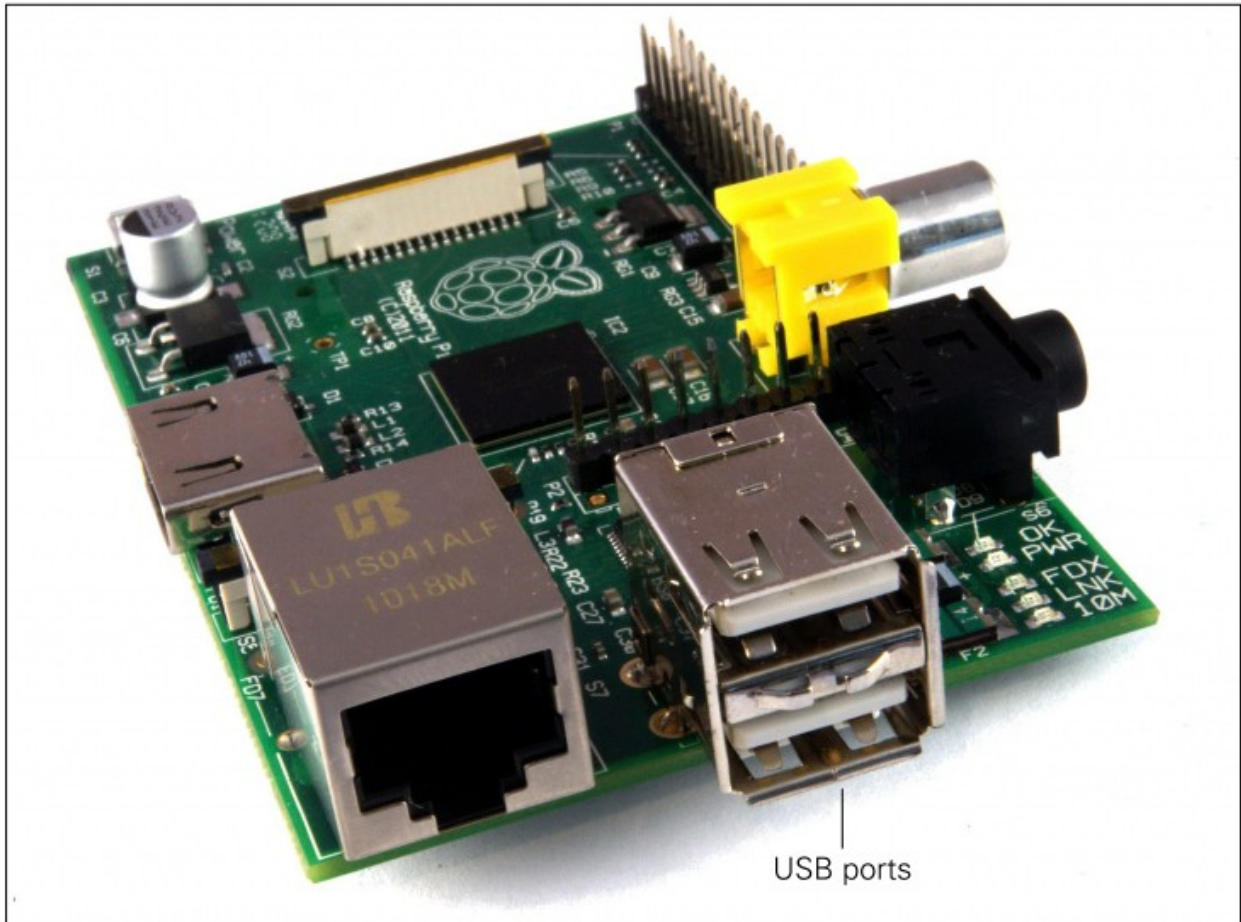
## Conexión del Teclado y el Ratón

Ahora que ya tiene los dispositivos de salida de la Raspberry Pi bien clasificados, es el momento de pensar en la entrada. Como mínimo, va a necesitar un teclado, y para la mayoría de los usuarios, un ratón o trackball también será necesario.

Primero, algunas malas noticias: si tiene un teclado y un ratón con conector PS/2 (un conector redondo con una matriz de pines en forma de herradura) entonces tendrá que salir y comprar otros. La antigua conexión PS/2 ha sido reemplazada y la RasPi espera que sus periféricos se conecten a través del *puerto USB (Universal Serial Bus)*. Una alternativa es adquirir un *adaptador USB a PS/2*, aunque hay que tener en cuenta que algunos teclados particularmente viejos podrían no funcionar correctamente mediante un adaptador de este tipo.

Dependiendo de si ha adquirido el Modelo A o el Modelo B, va a tener uno o dos puertos USB disponibles sobre el lado derecho de la RasPi (ver Figura 2-3). Si está utilizando el Modelo B, puede conectar el teclado y el ratón directamente a esos puertos. Si utiliza el Modelo A, necesitará comprar un *hub USB externo* para conseguir conectar los dos dispositivos USB simultáneamente.

Figura 2-3  
Los dos puertos USB del Modelo B



Un hub o concentrador USB es una buena inversión para cualquier usuario de RasPi: aun si tiene el Modelo B, ocupará ambos de sus puertos con tan sólo conectar el teclado y el ratón, no dejando nada libre para dispositivos adicionales tales como alguna unidad óptica externa, un dispositivo de almacenamiento o un joystick. Asegúrese de comprar un concentrador USB con alimentador propio (powered USB Hub): los modelos pasivos son más baratos y pequeños, pero no tienen la capacidad de hacer funcionar los dispositivos que consumen mucha energía tales como las unidades de CD y los discos duros externos.

---

**CONSEJO**

Si desea reducir el número de tomas de corriente utilizadas, conecte el cable de alimentación USB de la Raspberry Pi al concentrador USB con alimentador propio. De esta forma, la RasPi puede obtener su energía directamente desde el concentrador, en vez de tener su propia toma de corriente dedicada y su adaptador principal enchufado al socket. Esto sólo funcionará en los concentradores USB con una fuente de alimentación capaz de suministrar 700mA al puerto USB de la RasPi, junto con toda la energía necesitada por los otros periféricos.

---

La conexión del teclado y el ratón es tan sencilla como enchufarlas a los puertos USB, ya sea directamente en el caso de un Modelo B o a través de un concentrador USB en el caso del Modelo A.

## Una Nota sobre el Almacenamiento

Como se habrá dado cuenta, la Raspberry Pi no tiene un disco duro tradicional. En su lugar utiliza *una tarjeta de memoria SD (Secure Digital)*, un sistema de almacenamiento de estado sólido que se usa habitualmente en las cámaras digitales. Casi cualquier tarjeta SD funcionará con la Raspberry Pi, pero debido a que alberga todo el sistema operativo, es necesario que la tarjeta sea de al menos 4 GB de capacidad para almacenar todos los archivos requeridos.

Las Tarjetas SD con el sistema operativo precargado se encuentran disponibles en la tienda oficial de la Raspberry Pi así como en muchos otros sitios en Internet. Si ha comprado una de éstas, o recibió una en paquete con su RasPi, simplemente conéctela en la ranura para tarjetas SD que está en la parte inferior a mano izquierda.

Algunas tarjetas SD funcionan mejor que otras y algunos modelos se niegan en absoluto a trabajar con la Raspberry Pi. Para ver una lista actualizada de los modelos de tarjetas SD que funcionan con la Raspberry Pi, visite la página eLinux Wiki: [http://www.elinux.org/RPi\\_SD\\_cards](http://www.elinux.org/RPi_SD_cards)

## Instalando NOOBS en una Tarjeta SD

La Fundación Raspberry Pi distribuye una herramienta de software para la RasPi conocida como *NOOBS (New Out-Of-Box Software)*. Esta herramienta, ha sido diseñada para hacer lo más fácil posible el comenzar a utilizar la RasPi, y se encuentra precargada en algunas tarjetas SD incluidas con la placa Raspberry Pi, así como por separado, en forma de descarga gratuita. NOOBS ofrece una selección de diferentes sistemas operativos listos para instalarse en la RasPi así como algunas herramientas que permiten modificar su configuración.

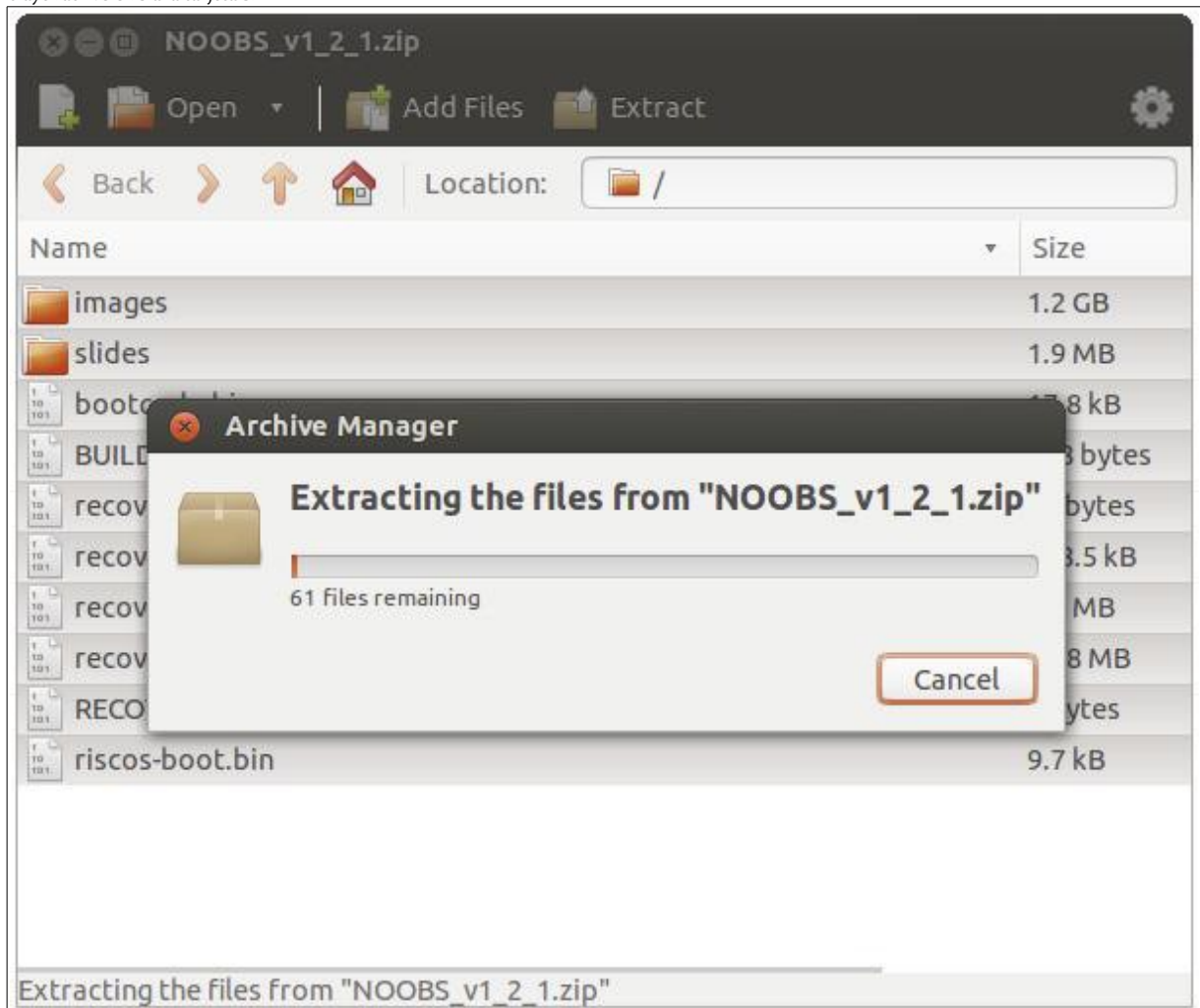
Si ha comprado una tarjeta SD con NOOBS ya instalado, no necesita hacer nada más en este momento. Si no fue así, descargue la última versión del software NOOBS desde el sitio oficial de la Fundación Raspberry Pi [www.raspberrypi.org/downloads](http://www.raspberrypi.org/downloads). Tenga en cuenta que el archivo es de gran tamaño y puede tomarle un tiempo considerable descargarlo; si cuenta con una conexión a Internet con un límite máximo de navegación de aproximadamente 1GB mensuales, no será capaz de descargar el archivo. En este caso, adquiera una tarjeta SD con NOOBS precargado en cualquiera de las tiendas que venden Raspberrys.

Para utilizar NOOBS, necesitará una tarjeta SD de al menos 4 GB de capacidad, y preferiblemente de al menos 8GB para dar espacio a la instalación de programas adicionales que utilizará con su RasPi. También necesitará una computadora con lector de tarjetas SD, ya sea que este integrado como en algunos modelos de laptops o como un dispositivo adicional. Para comenzar, inserte la tarjeta SD en el lector de tarjetas. Si ha usado su tarjeta SD anteriormente con otros dispositivos (por ejemplo, con una cámara digital o consola de videojuegos) siga el enlace Downloads en la página de la Raspberry Pi para descargar la herramienta de formateado de la *SD Card Association* y utilícela para *formatear* la tarjeta SD, preparándola así para la posterior instalación de NOOBS. Si la tarjeta es nueva, puede omitir este paso.

El software NOOBS se proporciona como un *archivo Zip*. Este es un formato de archivo donde los datos se encuentran *comprimidos*, por lo que ocupa menos espacio y es más rápida su descarga. En la mayoría de los sistemas operativos, hacer doble clic sobre el archivo debería ser suficiente para abrirlo; si no es así, descargue una utilidad de compresión como 7Zip ([www.7-zip.org](http://www.7-zip.org)) e inténtelo de nuevo.

Cuando haya abierto el archivo, utilice la función de *extracción* o *copiado* del software de compresión para transferir los archivos desde el archivo zip hacia su tarjeta SD (ver Figura 2-4). Esta tarea puede tardar algo de tiempo en llevarse a cabo, debido al número y tamaño de los archivos involucrados. Sea paciente y cuando la extracción haya finalizado y la luz de actividad (si la hay) se haya apagado, utilice la opción Expulsar de su sistema operativo para retirar la tarjeta SD y luego insértela en el lector de tarjetas SD de la RasPi.

Figura 2-4  
Extrayendo NOOBS a la tarjeta SD



## Conexión del Almacenamiento Externo

Debido a que la Raspberry Pi usa una tarjeta SD como su dispositivo de almacenamiento principal (conocido como *dispositivo de arranque*) es posible que experimente limitaciones de espacio muy pronto. Aunque las tarjetas SD con mayor tamaño son capaces de albergar 32 GB, 64 GB o más, a menudo son muy costosas.

Afortunadamente, existen dispositivos que proveen a cualquier computadora de un disco duro adicional cuando son conectadas a través de un cable USB. Conocidos como dispositivos *USB de Almacenamiento Masivo* (UMS – *USB Mass Storage*), estos dispositivos pueden ser discos duros físicos, unidades de estado sólido (SSD) o incluso memorias USB de bolsillo (ver Figura 2-5).

Casi todos los dispositivos USB de Almacenamiento Masivo pueden ser leídos por la RasPi, tengan o no contenido alguno. Para que la RasPi pueda acceder a estos dispositivos, las unidades deben ser *montadas* (un proceso que aprenderá en el Capítulo 3, “Administración del Sistema Linux”). Por ahora, sólo basta con conectar las unidades de disco a la RasPi para que estén preparadas.



**Figura 2-5**

Dos dispositivos USB de Almacenamiento Masivo: una memoria USB y un disco duro externo.



## Conexión de la Red

Si bien la mayoría de las instrucciones de configuración se aplican por igual tanto para la Raspberry Pi Modelo A como para el Modelo B, la configuración de redes es una excepción. Para mantener el número de componentes (y por lo tanto el costo) lo más bajo como sea posible, en el Modelo A no se incorpora característica alguna de red sobre la placa. Afortunadamente, eso no significa que no se pueda conectar el Modelo A en red, sólo que necesitará de algún equipo adicional para hacerlo.

## Configuración de Redes en el Modelo A

Para dotar al Modelo A de las mismas capacidades de red que su contraparte más cara el Modelo B, necesitará conectarle un adaptador USB Ethernet. Dicho adaptador se conecta en un puerto USB libre de la Raspberry Pi o en un hub USB que tenga conectado y proporciona una conexión Ethernet cableada con un conector RJ45, el mismo que se encuentra disponible en el Modelo B.

Un adaptador USB Ethernet 10/100 (con sus números refiriendo su modo de dos velocidades, 10 Mb/s y 100/Mbs) puede adquirirse en las tiendas en línea a un costo muy económico. Cuando compre un adaptador Ethernet, asegúrese de comprobar que el sistema operativo Linux está listado como un sistema compatible con el adaptador. Algunos modelos sólo funcionan con Microsoft Windows y son incompatibles con la Raspberry Pi.

No caiga en la tentación de ir por un adaptador clase gigabit, el cual es reconocido como adaptador USB Ethernet 10/100/1000. Los puertos USB estándar que utiliza la Raspberry Pi, no pueden hacer frente a la velocidad de conexión Ethernet gigabit, así que no conseguirá ninguna ventaja con un adaptador más caro.

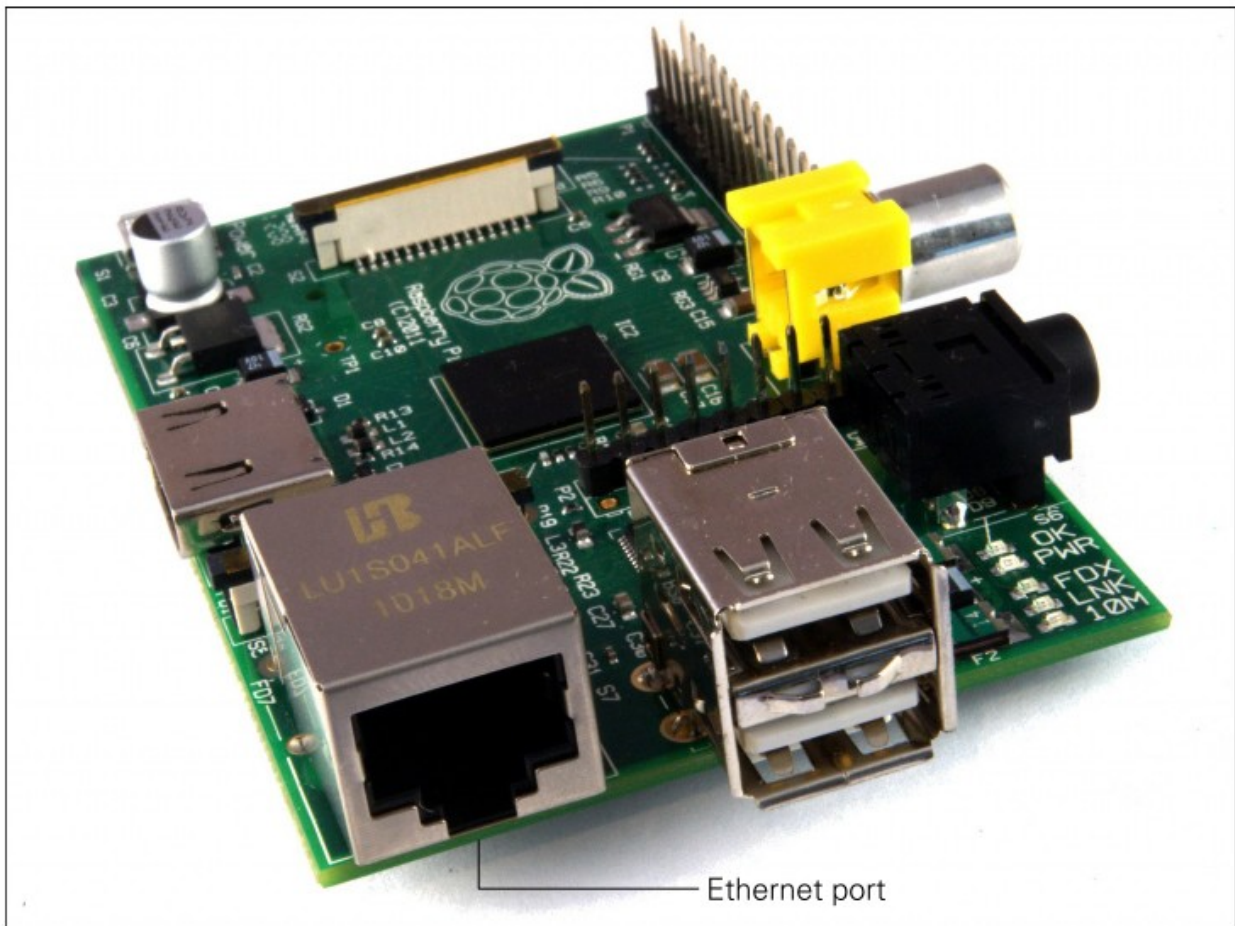
### Red Cableada

Para conectar su Raspberry Pi en Red, necesitará conectar un *cable Ethernet RJ45* entre la RasPi y el concentrador (switch), router o hub. Si no cuenta con un router o hub puede conseguir comunicar su PC o laptop con la RasPi conectando ambos equipos de manera directa con un cable Patch Cord.

Por lo general, conectar juntos dos clientes de red de esta manera requiere de un cable especial conocido como *cable cruzado*. En un cable cruzado, la recepción y transmisión de pares se intercambian de modo que los dos dispositivos no se interfieren el uno con el otro (una tarea usualmente llevada a cabo por el conmutador o concentrador).

La Raspberry Pi sin embargo es más astuta. El puerto RJ45 a un costado de la RasPi (ver Figura 2-6) incluye una característica conocida como *auto-MDI*, lo que le permite reconfigurarse automáticamente. Por consiguiente, puede utilizar cualquier cable RJ45 (cruzado o no) para conectar la RasPi a la Red y establecer su configuración en la forma que le corresponda.

Figura 2-6  
El puerto Ethernet de la Raspberry Pi Modelo B



### CONSEJO

Si conecta la RasPi directamente a la PC o laptop no podrá conectarse a la red de Internet predeterminada. Para hacerlo, necesitará configurar su PC para crear un **puente** entre el puerto Ethernet alámbrico y la otra conexión (comúnmente inalámbrica). Llevar a cabo esta tarea está fuera del alcance de este libro, pero si no encuentra forma alguna de conectar la RasPi a la Internet, puede intentar buscar la frase "bridge network" en el archivo de ayuda de su sistema operativo para obtener más información al respecto.

Con un cable de red conectado, la RasPi automáticamente recibirá todos los detalles necesarios para acceder a Internet a través del *DHCP (Protocolo de Configuración Dinámica de Host)* al momento de cargarse el sistema operativo. DHCP asigna a la RasPi una dirección *IP (Internet Protocol)* sobre la red y le indica la puerta de entrada (gateway) que tiene que utilizar para acceder a Internet (normalmente la dirección IP de su router o módem).

En algunas redes, no existe un servidor DHCP que pueda proporcionarle a la RasPi una dirección IP: Cuando se conecte a una red de éstas, la RasPi necesitará ser configurada manualmente. Aprenderá más sobre esto en el Capítulo 5, “Configuración de la Red”.

### Red Inalámbrica

Los modelos actuales de la Raspberry Pi no cuentan con la característica integrada para gestionar redes inalámbricas, pero (igual que con el cable Ethernet añadido en el Modelo A) es posible añadir soporte Wi-Fi a cualquier RasPi utilizando un adaptador USB para red inalámbrica (ver Figura 2-7).

**Figura 2-7**

Dos adaptadores USB para red inalámbrica, adecuados para utilizarse con la Raspberry Pi.





Utilizando tal adaptador, la RasPi puede conectarse a una amplia gama de redes inalámbricas, incluyendo aquellas que se funcionan bajo el último estándar 802.11n de alta velocidad. Antes de adquirir un adaptador USB para red inalámbrica, tenga en cuenta lo siguiente:

- Asegúrese que Linux se lista como un sistema operativo compatible con el adaptador. Algunos adaptadores inalámbricos se ofrecen sólo con drivers para Windows y OS X, por lo tanto son incompatibles con la Raspberry Pi. Una lista de adaptadores Wi-Fi que se conocen trabajan con la Raspberry Pi puede encontrarse en el siguiente sitio web: [http://elinux.org/RPi\\_USB\\_Wi-Fi\\_Adapters](http://elinux.org/RPi_USB_Wi-Fi_Adapters)
- Cerciórese que su tipo de red Wi-Fi es compatible con el adaptador inalámbrico. El tipo de red se muestra en las especificaciones como un número seguido de una letra. Por ejemplo, si su tipo de red es 802.11a, un adaptador inalámbrico de red 802.11g no funcionará.
- Compruebe las frecuencias soportadas por la tarjeta. Algunos estándares de redes inalámbricas como el 802.11a admiten más de una frecuencia. Si su adaptador USB de red inalámbrica está diseñado para trabajar sobre una red de 2.4GHz no se conectará a una red de 5GHz.
- Compruebe el tipo de cifrado utilizado por su red inalámbrica. Los adaptadores USB inalámbricos más modernos soportan todas las formas de encriptación o cifrado, pero si compra uno de segunda mano o un viejo modelo podría no conectarse a la red. Los tipos de cifrado más comunes son el viejo WEP y los más modernos WPA y WPA2.

La configuración de la conexión inalámbrica se realiza estando dentro de Linux, por lo que por el momento basta con solo conectar el adaptador a la RasPi (lo ideal sería mediante un concentrador USB). Aprenderá cómo configurar la conexión en el Capítulo 5, “Configuración de la Red”.

## Conexión de la Fuente de Alimentación

La Raspberry Pi es alimentada por el pequeño *conector micro-USB* ubicado en la parte inferior izquierda de la placa de circuito. Este conector es el mismo que se encuentra presente en la mayoría de los smartphones y algunas tablets.

Muchos cargadores diseñados para smartphones funcionarán con la Raspberry Pi, pero no todos. La RasPi consume más energía que la mayoría de los dispositivos micro-USB y requiere de al menos 700mA para funcionar. Algunos cargadores sólo pueden suministrar hasta 500mA, ocasionando problemas intermitentes en el funcionamiento de la RasPi (ver Capítulo 4 “Solución de Problemas”).

Es posible conectar la RasPi al puerto USB de una PC o laptop, pero no es recomendable. Al igual que con los cargadores de menor potencia, los puertos USB de una computadora no pueden suministrar la energía necesaria para que la RasPi funcione adecuadamente. Sólo conecte la fuente de alimentación micro-USB cuando se encuentre listo para comenzar a utilizar su RasPi. Al no contar con un botón de encendido, la RasPi comenzará a trabajar en el mismo instante en que la fuente de alimentación sea conectada.

Para apagar con seguridad la Raspberry Pi, envíe un comando *shutdown* desde la consola o terminal escribiendo:

```
sudo shutdown -h now
```

Para más información sobre el uso de la terminal, vea el Capítulo 3, “Administración del Sistema Linux”.

Si ha preparado o comprado su tarjeta SD con la herramienta NOOBS, como se ha descrito anteriormente en este capítulo, la RasPi cargará esta herramienta y esperará sus instrucciones; si no sucede así, ha encendido la RasPi con una tarjeta SD vacía lo que resultará en una pantalla en blanco o coloreada con un patrón de prueba de color. En este caso, desconecte el cable de alimentación del dispositivo y retire la tarjeta SD antes de seguir con las instrucciones de la instalación manual en la siguiente sección, “Instalación del Sistema Operativo”.

## Instalación del Sistema Operativo

Si compró su Raspberry Pi junto con una tarjeta SD con un sistema operativo precargado, o siguió las instrucciones para instalar NOOBS anteriormente en este capítulo, sólo tiene que insertar la tarjeta en la ranura para tarjetas SD en la parte inferior de la RasPi. Si adquirió únicamente la RasPi, necesitará instalar un sistema operativo a la tarjeta SD antes de que pueda utilizar su Raspberry Pi.

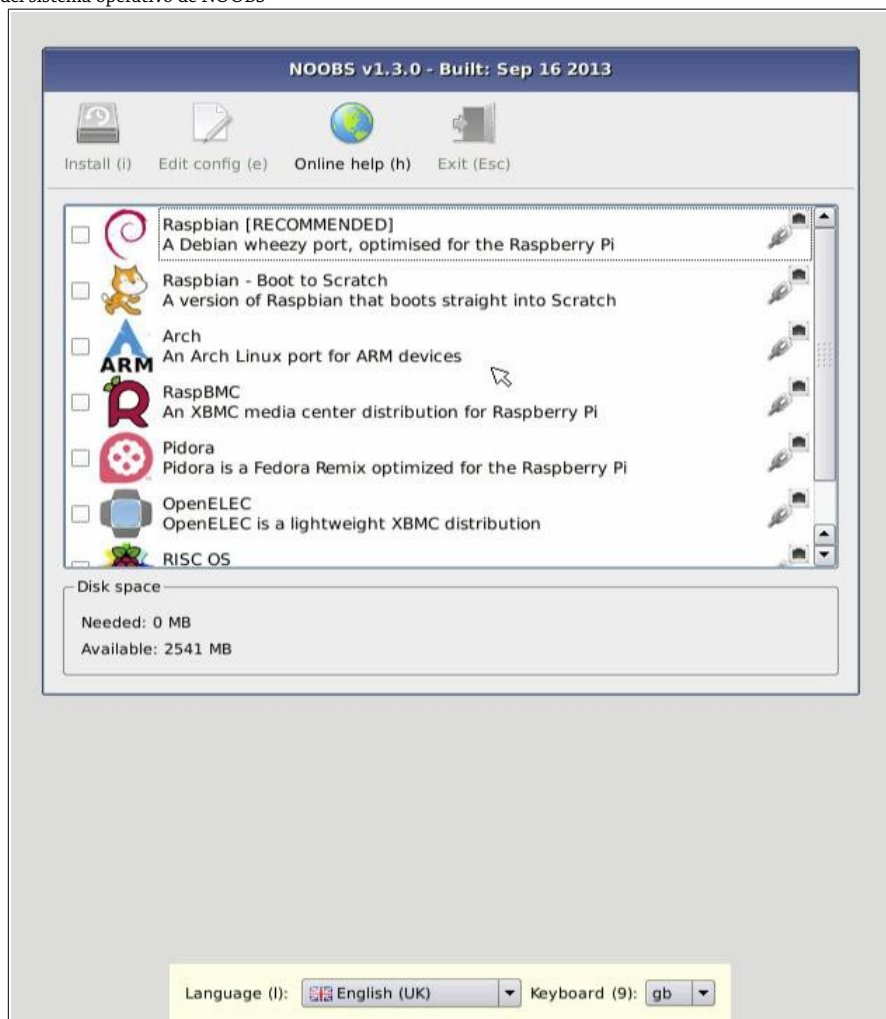
## Instalación a través de NOOBS

Si tiene instalado NOOBS en su tarjeta SD o compró una con NOOBS precargado, al encender la Raspberry Pi se mostrará un menú (ver Figura 2-8). Este menú muestra una lista de sistemas operativos adecuados para la RasPi, alguno o varios de los cuales pueden ser instalados. También puede elegir cambiar el idioma de la interfaz de usuario haciendo clic en la flecha que está en seguida de Language en la parte inferior de la pantalla, o escoger una distribución distinta del teclado utilizando la flecha que está en seguida de Keyboard.

Si es la primera vez que ejecuta NOOBS en aquella tarjeta SD, habrá un retardo en el arranque mientras la *partición* de la tarjeta se redimensiona para hacer espacio a su sistema operativo seleccionado; no desconecte la alimentación de la RasPi mientras se encuentra en este proceso, ya que corré el riesgo de dañar su tarjeta SD.

Figura 2-8

El menú de selección del sistema operativo de NOOBS



**CONSEJO**

Si sólo ve una pantalla en blanco, pero las luces ACT y PWR de la RasPi están encendidas, puede que tenga que elegir un modo de pantalla diferente. Presione 1 en el teclado para activar el modo HDMI estándar, 2 para el modo "seguro" con una resolución más baja, 3 si está utilizando el puerto compuesto en una región PAL o 4 si está utilizando el puerto compuesto en una región NTSC. Si no está seguro de cuál es el adecuado, traté todas las opciones hasta encontrar uno que le funcione. El modo de pantalla elegido será también pasado automáticamente al sistema operativo instalado.

Utilizando el teclado o el ratón, navegue por la lista de los sistemas operativos y haga clic sobre su nombre para marcarlo para su instalación. Note que puede instalar múltiples sistemas operativos si su tarjeta SD es lo suficientemente grande: simplemente marque dos o más sistemas operativos de la lista. Para los principiantes, recomendamos el sistema operativo Raspbian. El resto de este libro fue escrito con Raspbian en mente, pero mucho de lo que aprenderá es aplicable a casi todos los sistemas operativos basados en Linux, tanto en la RasPi como en otros dispositivos.

Para comenzar el proceso de instalación, haga clic en el icono Install en la parte superior izquierda de menú y cuando se le pregunte, confirme que la tarjeta SD puede ser reescrita. Al igual que con la instalación del propio NOOBS, este proceso suele tardar mucho tiempo en completarse; sea paciente, observe la barra de progreso y la presentación (ver Figura 2-9) y no retire el cable de alimentación o tarjeta SD de la RasPi hasta que haya terminado. Cuando el sistema operativo está instalado, haga clic en el botón OK para reiniciar y cargar el sistema operativo elegido.

Figura 2-9

Instalación de un sistema operativo a través de NOOBS



Si ha instalado más de un sistema operativo, NOOBS le mostrará un menú preguntándole cuál sistema operativo desea arrancar: si no toma una elección, el último sistema operativo elegido (o el primero en la lista, si no ha arrancado un sistema antes) automáticamente arrancará después de 10 segundos.

Para más información sobre el uso de NOOBS después de que el sistema operativo haya sido instalado, ya sea para instalar un sistema operativo diferente o para modificar las variadas configuraciones de la RasPi, vea el Capítulo 7, “Configuración Avanzada de la Raspberry Pi”.

## Instalación Manual

La instalación manual de un sistema operativo es un procedimiento más complicado a diferencia de sólo utilizar la herramienta NOOBS, pero algunas veces es más preferible. Al instalar el software manualmente (un proceso conocido como *flashear*) se tiene la oportunidad de poder instalar otros sistemas operativos que no se encuentran disponibles a través de NOOBS así como instalar las versiones más recientes.

Para que su tarjeta SD vacía esté preparada para su uso con la Raspberry Pi, necesitará flashear un sistema operativo en la tarjeta utilizando su PC o laptop. Aunque esto es un poco más complicado que simplemente arrastrar y soltar los archivos a la tarjeta, no tomará más que unos cuantos minutos en completarse.

En primer lugar, necesita decidir qué distribución de Linux desea usar con su Raspberry Pi. Cada una tiene sus ventajas y desventajas. No se preocupe si cambia de opinión más tarde y desea probar una versión diferente de Linux: una tarjeta SD puede ser flasheada nuevamente en cualquier momento con otro nuevo sistema operativo.

La lista más actualizada con las versiones de Linux que son compatibles con la RasPi se encuentra disponible en el sitio web oficial en <http://www.raspberrypi.org/downloads>

La Fundación pone a su disposición enlaces *BitTorrent* para cada distribución. Estos enlaces BitTorrent son pequeños archivos que pueden ser utilizados con programas BitTorrent para descargar los archivos desde otros usuarios. Utilizar estos enlaces es una forma eficiente y rápida de distribuir archivos de gran tamaño, y así evitar que los servidores de descarga de la Fundación se sobrecarguen.

Para usar un enlace BitTorrent, necesitará tener instalado un *cliente* BitTorrent compatible. Si no lo tiene, descargue uno e instálelo antes de tratar de descargar la distribución de Linux para la Raspberry Pi. µTorrent es un cliente para Windows, OS X y Linux, se puede descargar desde <http://www.utorrent.com/downloads>.

¿Qué distribución descargar? Eso depende de usted. Las instrucciones en el resto del libro se basarán sobre la distribución Raspbian para la Raspberry Pi (una buena elección para los principiantes). Cuando sea posible, daremos instrucciones también para las demás distribuciones.

Las distribuciones Linux para la Raspberry Pi se ofrecen como un sólo *archivo de imagen* comprimida para hacer más rápida su descarga. Una vez que haya descargado el archivo Zip (un archivo comprimido, el cual tarda menos en descargarse que los archivos sin compresión) de su distribución elegida, necesitará descomprimirla en algún lugar de su sistema. En la mayoría de los sistemas operativos, puede hacer simplemente doble clic al archivo para abrirlo y luego seleccionar Extraer o UnZip para recuperar el contenido.

Una vez que haya descomprimido el archivo, terminará obteniendo dos archivos separados. El archivo que termina en sha1 es un *hash*, que se utiliza para verificar que la descarga no se haya corrompido durante el trayecto. El archivo con extensión img contiene una copia exacta de una tarjeta SD producida por los creadores de la distribución en un forma que la Raspberry Pi comprende. Este es el archivo que necesita ser flasheado a la tarjeta SD.

---

### ADVERTENCIA

A continuación, va a utilizar una herramienta de software llamada dd. Utilizada incorrectamente dd felizmente escribirá la imagen en su disco duro principal, borrando su sistema operativo y todos sus datos. Asegúrese de leer las instrucciones detenidamente en cada sección y observe con cuidado el nombre del dispositivo de su tarjeta SD. ¡Lea dos veces, escriba una vez!

---

### Flasheando desde Linux

Si su PC actual trabaja con alguna variante de Linux, puede utilizar el comando `dd` para escribir los contenidos del archivo de imagen a la tarjeta SD. `dd` es un programa de interfaz de texto que trabaja desde la línea de comandos, conocida en la jerga Linux como una *terminal*. Siga estos pasos para flashear la tarjeta SD:

1. Abra una terminal desde el menú de aplicaciones de su distribución.
2. Inserte la tarjeta SD vacía dentro del lector de tarjetas de su PC.
3. Escriba `sudo fdisk -l` para ver un listado de los discos. Localice su tarjeta SD reconociéndola por su tamaño y tome nota del nombre del dispositivo (`/dev/sdX`, en donde `X` es una letra que identifica al dispositivo de almacenamiento. Algunos equipos con lectores de tarjeta SD integrados podrían utilizar un formato alternativo `/dev/mmcblkX`, si este fuera su caso, recuerde cambiar el nombre del dispositivo de destino en las siguientes instrucciones en donde corresponda).
4. Utilice `cd` para cambiarse al directorio que contiene el archivo `.img` que ha extraído desde el archivo Zip.
5. Escriba `sudo dd if=nombrearchivoimagen.img of=/dev/sdX bs=2M` para grabar el archivo `nombrearchivoimagen.img` a la tarjeta SD vinculada al nombre del dispositivo del paso 3. Reemplace `nombrearchivoimagen.img` por el nombre real del archivo extraído desde el Zip. ¡Este paso tomará algo de tiempo, sea paciente! Durante el flasheo, nada se mostrará en la pantalla hasta que el proceso termine por completado (ver Figura 2-10).

Figura 2-10

Flasheando la tarjeta SD en Linux con el comando `dd`

```

blacklaw@xerxes-linux: /media/Data/raspberrypi/debian6-19-04-2012
File Edit View Terminal Tabs Help

blacklaw@xerxes-linux: ~
blacklaw@xerxes-linux: /media/Data/raspberrypi...

Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x24282427

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1  *           1         12748    102398278+   7   HPFS/NTFS
/dev/sda2             12749        25496    102398310   5   Extended
/dev/sda3             25497        77825    420332692+   7   HPFS/NTFS
/dev/sda5             12749        12997      2000061   82   Linux swap / Solaris
/dev/sda6             12998        25496    100398186   83   Linux

Disk /dev/sdb: 3965 MB, 3965190144 bytes
49 heads, 48 sectors/track, 3292 cylinders
Units = cylinders of 2352 * 512 = 1204224 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1             4          3293     3868160    b   W95 FAT32
blacklaw@xerxes-linux: /media/Data/raspberrypi/debian6-19-04-2012$ sudo dd if=debian6-19-04-2012.img of=/dev/sdb bs=2M

```

### Flasheando desde OS X

Si su PC actual es una Mac que trabaja con Apple OS X, le alegrará saber que las cosas son tan simples como con Linux. Gracias a un parecido, tanto Linux como OS X cuentan con la utilidad `dd`, la cual puede utilizar para flashear la imagen del sistema a su tarjeta SD de la siguiente forma:

1. Seleccione Utilidades desde el menú Aplicación, y luego haga clic en la aplicación Terminal.
2. Inserte la tarjeta SD vacía en el lector de tarjetas de su Mac.
3. Escriba `diskutil list` para ver listado de los discos. Localice su tarjeta SD reconociéndola por su tamaño y tome nota del nombre del dispositivo (`/dev/diskX`, en donde X es una letra que identifica al dispositivo de almacenamiento).
4. Si la tarjeta SD fue montada automáticamente y se muestra sobre el escritorio, escriba `diskutil unmountdisk /dev/diskX` para desmontarla antes de proceder.
5. Utilice `cd` para cambiarse al directorio que contiene el archivo `.img` que ha extraído desde el archivo Zip.
6. Escriba `dd if=nombrearchivoimagen.img of=/dev/diskX bs=2M` para grabar el archivo `nombrearchivoimagen.img` a la tarjeta SD vinculada al nombre del dispositivo del paso 3. Reemplace `nombrearchivoimagen.img` por el nombre real del archivo extraído desde el Zip. ¡Este paso tomará algo tiempo, sea paciente!

### Flasheando desde Windows

Si su PC actual trabaja con Windows, las cosas son ligeramente más complicadas que con Linux u OS X. Windows no cuenta con una utilidad parecida a `dd`, debido a esto un software de terceros es necesario para conseguir flashear el archivo de imagen en la tarjeta SD. Aunque es posible instalar una versión de `dd` compatible con Windows, hay una forma más fácil de hacerlo: utilizando el programa Image Writer. Diseñado específicamente para grabar imágenes de distribuciones Linux en dispositivos USB o tarjetas SD, Image Writer cuenta con una interfaz gráfica de usuario muy sencilla que hace que la creación de una tarjeta SD sea muy fácil.

La última versión de Image Writer puede encontrarla en su página oficial:

<https://sourceforge.net/projects/win32diskimager/>. Siga los siguientes pasos para descargar, instalar y utilizar el software Image Writer para preparar la tarjeta SD para su uso con la RasPi:

1. Haga clic en el botón verde Download para descargar el archivo Zip de Image Writer y descomprímalo en una carpeta de su computadora.
2. Inserte la tarjeta SD vacía en el lector de tarjetas de su PC.
3. Haga doble clic en el archivo `Win32DiskImager.exe` para abrir el programa y haga clic en el icono de la carpeta azul para abrir un cuadro de diálogo de selección de archivos.
4. Busque el archivo `nombrearchivoimagen.img` que extrajo desde el archivo de la distribución, reemplace `nombrearchivoimagen.img` por el nombre real del archivo que ha extraído desde el archivo Zip y luego haga clic en el botón “Open”.
5. Seleccione la letra de unidad que corresponda a su tarjeta SD desde la lista desplegable “Device”. Si no está seguro de la letra a elegir, abra Equipo o el Explorador de Windows para asegurarse.
6. Haga clic en el botón “Write” para flashear el archivo de imagen a la tarjeta SD. ¡Este proceso tomará algo tiempo, sea paciente!

---

### ADVERTENCIA

Sin importar en qué sistema operativo esté grabando la imagen, es importante dejar la tarjeta SD conectada hasta que la imagen sea completamente grabada. Si no lo hace así, es posible que la RasPi no arranque cuando la tarjeta SD se conecte. Si esto ocurre, comience el proceso de nuevo.

Cuando la imagen haya sido flasheada en la tarjeta SD, sáquela de la computadora e insértela en la ranura de la tarjeta SD de su Raspberry Pi, ubicada en la parte de abajo de la placa de circuito. La tarjeta SD debe ser insertada con la etiqueta boca abajo y empujada completamente hacia dentro para asegurar una buena conexión.

---



# **Capítulo 3**

## Administración del Sistema Linux



**La mayoría de** las modernas distribuciones Linux son fáciles de utilizar, gracias a sus *interfaces gráficas de usuario (GUI)* que brindan una forma fácil de realizar las tareas comunes. Sin embargo, son muy distintas a la de Windows y OS X, de manera que si desea aprovechar al máximo su Raspberry Pi, necesitará una rápida revisión de cómo utilizar su sistema operativo.

## Linux: Una Descripción General

Como brevemente se explicó en el Capítulo 1, “Conozca la Raspberry Pi”, Linux es un proyecto de código abierto que fue originalmente fundado para crear un *núcleo* que sería libre de utilizar por cualquier persona. El *núcleo* o *kernel* es el corazón de un sistema operativo y se ocupa de la comunicación entre el usuario y el hardware.

Aunque el término “Linux” es correctamente empleado para referirse al “núcleo”, es a menudo utilizado para referirse a una colección de diferentes proyectos de código abierto de distintas compañías. Esas colecciones de proyectos se unen para formar diferentes *sabores* de Linux, conocidas como *distribuciones* o *distros*.

La versión original de Linux fue combinada con una colección de herramientas creadas por un grupo denominado GNU. El sistema resultante fue conocido como *GNU/Linux* y era muy básico pero potente. A diferencia de los otros sistemas operativos de la época, éste ofrecía facilidades como: múltiples cuentas de usuario, en donde varios usuarios podían compartir una misma computadora. Aquello fue algo que los otros sistemas operativos rivales de código cerrado tomaron abordo, tanto Windows como OS X ahora ya soportan múltiples cuentas de usuario sobre un mismo sistema. Esta característica continúa presente hoy en día en Linux y proporciona seguridad y protección al sistema operativo.

En Linux, pasará la mayor parte de su tiempo trabajando en un cuenta de usuario *limitada* o *restringida*. Esto no significa que está siendo limitado en lo que puede hacer. En vez de eso, trabajar con una cuenta de usuario limitada evita que accidentalmente estropee el software en su Raspberry Pi. También previene que los *virus* y otros *malware* infecten el sistema bloqueándoles el acceso a los archivos y directorios críticos del sistema.

Antes de empezar, vale la pena familiarizarse con algunos de los términos y conceptos utilizados en el mundo Linux, tales como se definen en la Tabla 3-1. Aun si cuenta con experiencia en otros sistemas operativos, no es mala idea revisar esta tabla antes de arrancar la RasPi por primera vez.

**Tabla 3.1 Rápido Glosario de Linux**

Término	Definición
Bash	El <i>shell</i> o intérprete de comandos más popular, utilizado en la mayoría de las distribuciones Linux.
Bootloader	Software responsable de cargar el núcleo Linux. El más común es GRUB.
La consola	Una versión de la terminal que siempre está disponible y es la primer cosa que ve en la Raspberry Pi.
Entorno de escritorio	Software que hace la GUI más bonita. GNOME y KDE son los entornos de escritorio muy populares.
Directorio	El término Linux para lo que Windows denomina como <i>carpetas</i> en donde los archivos se almacenan.
Distribución	Una versión particular de Linux. Pidora, Arch y Raspbian son distribuciones.
Ejecutable	Un archivo que puede ejecutarse como un programa. Los archivos en Linux deben ser marcados a fin de que puedan ser ejecutados.
EXT2/3/4	El sistema de archivos EXTendido, el formato más común utilizado en Linux.
Sistema de archivos	La forma en que un disco duro o dispositivo de almacenamiento es formateado para quedar listo para almacenar archivos.
GNOME.	Uno de los entornos de escritorio Linux más comunes.
GNU	Un proyecto de software libre que proporciona muchas de las herramientas que se utilizan en las distribuciones Linux.
GRUB	El Gestor de Arranque Múltiple (GRan Unified Bootloader), creado por GNU y utilizado para cargar el núcleo Linux.
GUI	Una interfaz gráfica de usuario, con la que el usuario controla la computadora a través de un ratón o dispositivo táctil (touch).
KDE	Otro entorno de escritorio Linux muy popular.
Linux	Definido exactamente, es el núcleo utilizado por GNU/Linux. Definido popularmente, es un sistema operativo de código abierto.
Live CD (CD Vivo)	Una distribución Linux que se ofrece como un CD o DVD que no requiere de instalación.
Paquete	Una colección de archivos requeridos para la ejecución de una aplicación, comúnmente administrados por el gestor de paquetes.
Gestor de paquetes	Una herramienta que sigue la pista del software ya instalado y de las nuevas instalaciones.
Partición	Una sección de un disco duro que se encuentra lista para tener un sistema de archivos dedicado al almacenamiento de información.
Root	La cuenta de usuario principal en Linux, equivale a la cuenta de <i>administrador</i> en Windows. También denominada <i>superuser</i> .
Shell	Un intérprete de comandos basado en texto, cargado dentro de una terminal.
sudo	Un programa que permite a los usuarios limitados o restringidos ejecutar un comando como el usuario root.
SuperUsuario	Ver Root.
Terminal	Un intérprete de comandos basado en texto dentro del cual el usuario interactúa con un programa <i>shell</i> .
X11	El sistema de Ventanas X, es un paquete que proporciona un interfaz gráfica de usuario (GUI)

## La Terminal y la GUI

Al igual que en Windows y OS X, existen normalmente dos principales formas de llevar a cabo una determinada tarea: a través de la interfaz gráfica de usuario (GUI) y a través de la línea de comandos (conocida en la jerga Linux como la *consola* o la *terminal*).

La apariencia de las distintas distribuciones Linux pueden ser completamente diferentes, dependiendo sobre el entorno de escritorio que utilicen. En este libro, la distribución recomendada a utilizar es la Raspbian, pero la mayoría de los comandos que va a aprender aquí son introducidos desde la *terminal* y casi siempre son los mismos para todas las distribuciones.

Se le darán métodos alternativos para realizar el mismo objetivo para aquellas distribuciones en donde difieran los comandos.

## Comandos Básicos de Linux

Aunque existen cientos de diferentes distribuciones Linux, todas ellas comparten un conjunto de herramientas en común conocidos como *comandos*. Esas herramientas son puestas en funcionamiento a través de la terminal y son análogas a herramientas similares presentes en Windows y OS X. Para comenzar, necesitará aprender los siguientes comandos:

- **ls**-Abreviación para *listing* (listado), **ls** proporciona un lista de los contenidos del directorio actual. Opcionalmente puede ser invocado con un argumento que indica el directorio a ser listado. Como un ejemplo, al escribir **ls /home** se visualizará una lista de los contenidos del directorio **/home**, independientemente del directorio actual en el que se encuentre. El comando equivalente en Windows es **dir**.
- **cd**-Las siglas para *change directory* (cambio de directorio), **cd** le permite navegar a través del sistema de archivos. Escribiendo únicamente **cd** lo colocará devuelta a su directorio inicial de partida. Si escribe el comando junto con el nombre del directorio al que desea moverse, por consecuencia, lo cambia hacia aquel directorio. Considere que los directorios pueden ser absolutos o relativos: **cd boot** lo llevará al directorio llamado **boot** bajo el directorio actual, pero **cd /boot** lo llevará directamente al directorio **/boot** desde cualquier lugar en donde usted se encuentre.
- **mv**-El comando *move* (mover) tiene dos propósitos en Linux: permitir que un archivo sea trasladado de un directorio a otro, y también permitir que los archivos sean renombrados. Esta última característica puede parecer fuera de lugar, pero en términos de Linux, el archivo se ha movido de un nombre a otro. El comando es invocado en la siguiente forma: **mv viejoarchivo nuevoarchivo**.
- **rm**-Abreviación para *remove* (remover), **rm** elimina archivos. Cualquier archivo (o lista de archivos) escrito después del nombre del comando será eliminado. Su equivalente en Windows es **del**, y los dos comandos comparten un mismo requisito, que es el cuidado que debe tomarse para asegurarse de eliminar el archivo correcto.
- **rmdir**-Por regla general, el comando **rm** no puede eliminar directorios por sí mismo. Por consiguiente, el comando **rmdir** es proporcionado para borrar los directorios una vez que éstos han quedado vacíos de archivos debido al comando **rm**.
- **mkdir**-Es el opuesto a **rmdir**. El comando **mkdir** crea nuevos directorios. Por ejemplo, escribir **mkdir micarpeta** en la terminal, creará un nuevo directorio llamado **micarpeta** bajo el directorio actual de trabajo. Al igual que **cd**, los directorios que se proporcionen al comando pueden ser relativos o absolutos.

## Introducción a Raspbian

Raspbian es el nombre dado a una variante personalizada de la popular distribución Debian Linux. Debian es una de las distribuciones Linux más antiguas y se enfoca en la alta compatibilidad y un rendimiento excelente incluso sobre el hardware modesto, convirtiéndose en un gran socio para la Raspberry Pi. Raspbian toma a Debian como su base o *distribución padre* y agrega herramientas y software para hacer que la Raspberry Pi sea lo más fácil de utilizar.

Para mantener el tamaño del archivo de descarga al mínimo posible, la imagen de la Raspberry Pi para Raspbian sólo incluye un subconjunto del software que encontraría en una versión normal de escritorio. Esta imagen incluye herramientas para navegar en la web, programar en Python y utilizar la RasPi con una GUI. El software adicional puede instalarse rápidamente utilizando el gestor de paquetes *apt* de la distribución, o adquiriéndolo a través de la Raspberry Pi Store con el enlace sobre el escritorio. Raspbian incluye un entorno de escritorio conocido como *LXDE* (*Lightweight X11 Desktop Enviroment*). Diseñado para ofrecer una interfaz de usuario atractiva utilizando el software del Sistema de Ventanas X (*X Windows*), LXDE ofrece una interfaz familiar “apunta y has clic”, la cual puede ser rápidamente accesible por cualquiera persona que haya utilizado en el pasado Windows, OS X u otros sistemas operativos basados en GUI.

---

### CONSEJO

La GUI no se carga por defecto en la mayoría de las distribuciones que existen para la Raspberry Pi y Raspbian no es la excepción. Para cargarla rápidamente y salir de la consola basada en texto, inicie sesión, escriba `startx` y presione la tecla Intro (Enter). Para volver a la consola, mientras deja la GUI ejecutándose en segundo plano, mantenga presionado `Ctrl + Alt` y presione `F1` antes de soltar las tres teclas.

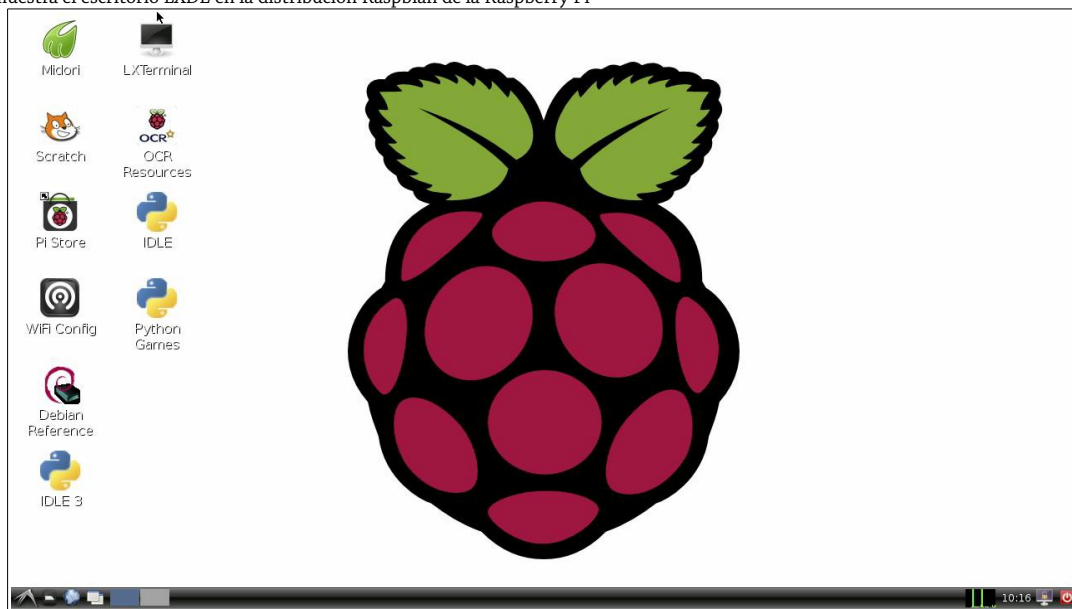
---

Si está utilizando la distribución Raspbian, notará que cuenta con un montón de software preinstalado listo para ser usado. Si bien éste, es apenas un pequeño ejemplo del software que se encuentra disponible para la RasPi de entre miles de *paquetes*, son una buena introducción que precisa lo que el sistema puede hacer.

El software que se proporciona con la distribución Raspbian se encuentra dividido en categorías temáticas. Para ver estas categorías, puede hacer clic con el botón izquierdo sobre el icono del menú, en la parte inferior izquierda de la pantalla en LXDE (ver Figura 3-1).

Figura 3-1

Como se muestra el escritorio LXDE en la distribución Raspbian de la Raspberry Pi



En los siguientes listados se describen los paquetes de software, agrupados por categorías. También hay una categoría no cubierta aquí, denominada “Otros”, en la que se agrupan numerosas herramientas del sistema. Si ha instalado un programa y no lo encuentra por ningún lado, pruebe en el menú Otros.

### Accesorios

- **Debian Reference**-Una guía de referencia integrada, en ella se proporciona una explicación detallada de la distribución Debian Linux y cómo los programadores pueden contribuir a su desarrollo.
- **Calculator**-Una calculadora científica de código abierto, que ofrece una variedad de funciones para realizar cálculos rápidos y complejos.
- **Gestor de archivos**-El administrador de archivos PCManFM proporciona una interfaz gráfica para navegar en los archivos almacenados en la RasPi o en cualquier dispositivo de almacenamiento conectado.
- **Leafpad**-Este es un editor de texto muy sencillo, utilizado para crear rápidas anotaciones o escribir sencillos programas.
- **LXTerminal**-La terminal de LXDE le permite utilizar la consola de línea de comandos Linux dentro de una ventana sin tener que salir de la interfaz gráfica de usuario.
- **Terminal de =Root=**-Similar a LXTerminal, la Terminal de Root automáticamente lo inicia en la sesión con la cuenta de *Super-usuario* (*root*) para poder llevar a cabo tareas de mantenimiento en el sistema que son imposibles desde una cuenta de usuario normal.
- **Visor de Imágenes**-GPicView le permite ver imágenes, tales como aquellas almacenadas en su cámara digital o en cualquier dispositivo de almacenamiento conectado.
- **Xarchiver**-Si necesita crear o extraer archivos comprimidos, como los archivos Zip, ésta es la herramienta indicada para eso.

### Educación

- **Scratch**-Un lenguaje de programación gráfico destinado a los niños. Aprenderá más sobre Scratch y sus capacidades en el Capítulo 11, “Una introducción a Scratch”.
- **Squeak**-La plataforma sobre la cual Scratch se ejecuta. Raramente necesitará utilizar esta entrada del menú, y en su lugar deberá utilizar la entrada Scratch anterior.

### Internet

- **Dillo**-Uno de los navegadores web más rápidos disponibles para la RasPi, pero algo básico comparado con Midori.
- **Midori**-Un navegador web rápido y ligero, Midori es el equivalente a Internet Explorer en Windows o a Safari en OS X.
- **Midori Private Browsing (Midori Navegación Privada)**-Haciendo clic en esta entrada del menú se cargará el navegador web Midori en modo privado, lo que significa que los sitios que visite no serán guardados en el historial del navegador.
- **NetSuf Web Browser**-Una alternativa a Midori, NetSuf puede tener mejor rendimiento en ciertos tipos de páginas web. Probar ambos le permitirá experimentar y encontrar el que mejor le funciona a usted.
- **Wps\_gui**-Una interfaz gráfica de usuario que permite configurar un adaptador USB inalámbrico (opcional) con la Raspberry Pi y conectarse a una red cifrada WPS (Wireless Protected System).

### Programación

- **IDLE**-Un entorno de desarrollo integrado (IDE) escrito específicamente para Python. Aprenderá en el Capítulo 12, “Una Introducción a Python” cómo utilizarlo para escribir sus propios programas en Python.
- **IDLE 3**-Haciendo clic en esta entrada se cargará un IDLE configurado para utilizar el lenguaje programación Python 3 más reciente, en vez del Python 2.7 por defecto. Ambos son en gran medida compatibles entre sí, pero algunos programas pueden necesitar características particulares de Python 3.
- **Scratch**-Este acceso directo abre el lenguaje de programación educativo Scratch y es el mismo Scratch que

se encuentra en la categoría Educación. Cualquiera puede utilizarse para iniciar el programa.

- **Squeak**-Al igual que Scratch, este es un acceso directo duplicado del que se encuentra en la categoría Educación. Rara vez tendrá que hacer clic en éste y en vez de ello debería utilizar el acceso directo Scratch.

### Herramientas del Sistema

- **Administrador de tareas**-Es una herramienta que le permite ver la cantidad de memoria libre disponible en la RasPi, la carga de trabajo actual del procesador y con la que puede cerrar los programas que se hayan colgado o que de alguna forma no respondan.

### Preferencias

- **Ajustes del monitor**-Desde aquí la resolución en la que trabaja el monitor o televisor conectado a la RasPi puede cambiarse, aunque las configuraciones avanzadas requerirán de la modificación de los archivos de configuración. Aprenderá sobre esto en el Capítulo 7, “Configuración Avanzada de la Raspberry Pi”.
- **Aplicaciones preferidas**-Una herramienta para cambiar las aplicaciones que abrirán determinados tipos de archivo. Si elige utilizar un navegador web alternativo, el navegador predeterminado del sistema puede cambiarse aquí.
- **Configuración de la sesión del escritorio**-Una herramienta para modificar el funcionamiento del sistema cuando el usuario inicia sesión, incluyendo qué programas serán cargados automáticamente y qué gestor de ventanas (el software que dibuja los bordes y las barras de título de las ventanas) será utilizado.
- **Gestor de configuración de Openbox**-La GUI LXDE utiliza un entorno de escritorio llamado *Openbox*, el cual puede configurarse desde aquí. Con esta herramienta, puede aplicar nuevos temas para cambiar la apariencia de la GUI, o alterar el funcionamiento de ciertos aspectos de la interfaz.
- **Personaliza apariencia y comportamiento**-Un conjunto de herramientas para modificar la apariencia de la GUI, incluyendo el estilo y el color de las ventanas.
- **Teclado y Ratón**-Una herramienta para configurar los dispositivos de entrada. Si el teclado está escribiendo caracteres incorrectos para ciertas teclas o el ratón es demasiado sensible, es aquí donde ese tipo de cosas pueden ser configurados.

## Consiguiendo Ayuda

Linux está diseñado para ser lo más amigable posible para los nuevos usuarios, incluso, en la terminal de línea de comandos. Aunque aprenderá las formas más comunes de utilizar cada comando en este capítulo, no todas las opciones serán cubiertas (hacerlo requeriría un libro mucho más grande).

Si se encuentra atorado, o si desea aprender más sobre cualquiera de las herramientas que se describirán en las siguientes páginas, hay un comando que debería aprender: `man`.

Cada aplicación Linux viene con un archivo de ayuda, conocido como *man page* (abreviatura en inglés para “página del manual”). Este archivo le proporciona un preámbulo sobre el software así como también los detalles sobre lo qué hacen sus opciones y cómo emplearlas.

Para acceder a la página de manual de una determinada herramienta, simplemente escriba `man` seguido por el nombre del comando. Para ver la página del manual de `ls` (una herramienta que muestra un listado del contenido de los directorios) simplemente escriba `man ls`.

## Sobre el Padre de Raspbian, Debian

Raspbian está basado en una de las distribuciones originales de Linux, Debian. Llamada así por su creador y su novia (Ian y Deb). Debian es una distribución popular gracias a méritos propios. Sin embargo, es común que en el

mundo del software abierto los proyectos comiencen basados sobre la refinación y personalización de proyectos ya existentes a lo que se conoce como *bifurcación (forking)*. Raspbian es un *fork* o *bifurcación* de Debian, pero no es la única: Ubuntu Linux de Canonical está también basado en Debian, mientras que Linux Mint, una de las distribuciones más populares para PCs de escritorio y laptops, está basada en torno a Ubuntu.

El proceso de bifurcar y bifurcar nuevamente es una característica única del software abierto. Con paquetes de código cerrado, como Micro Windows, no es posible personalizar el paquete a sus requerimientos individuales. Esta es una de las grandes fuerzas del software de código abierto y se demuestra brillantemente con la facilidad en la que Raspbian ha sido personalizado a los requerimientos de la Raspberry Pi.

## Alternativas a Raspbian

Aunque Raspbian es la distribución Linux recomendada para la Raspberry Pi, existen otras alternativas. Las más populares se encuentran disponibles en la página de descargas de la Fundación Raspberry Pi en <http://www.raspberrypi.org/downloads>, y la mayoría pueden ser instaladas utilizando NOOBS, como se describió en el Capítulo 2, Comenzando con la Raspberry Pi”.

Después de Raspbian, las distribuciones más instaladas comúnmente son RaspBMC y OpenELEC, las cuales convierten la Raspberry Pi en un sistema de teatro en casa dedicado, como será mostrado en el Capítulo 8, “La RasPi como una PC Teatro en Casa”. El siguiente más popular es Pidora, una distribución basada en el proyecto Fedora Linux, el cual a su vez, tiene a Red Hat como su distribución Padre. Por último, se encuentra Arch Linux que está diseñada para aquellas personas que ya están familiarizados con Linux; a diferencia de las otras en la lista, esta distribución no incluye una interfaz gráfica de usuario por defecto.

Una sistema operativo en la lista NOOBS que no es en lo absoluto una variante de Linux, es RiscOS. Originalmente producida por Acorn Computers a finales de los años 80 para su gama de computadoras personales Archimedes (que al igual que la Raspberry Pi, estaban basados en un procesador ARM). RiscOS es un sistema operativo rápido y fácil de utilizar con un aspecto limpio en su interfaz de usuario. Si bien la disolución de Acorn en 1998 trajo la popularidad de RiscOS al declive, la plataforma todavía conserva sus fans, quienes rápidamente añadieron soporte a la Raspberry Pi.

La ejecución de RiscOS sobre la Raspberry Pi resulta en un ambiente que es significativamente más rápido que cualquier otro sistema operativo de los que se ofrecen, gracias a sus orígenes como una plataforma diseñada específicamente para la arquitectura del conjunto de instrucciones ARM. Desafortunadamente, la velocidad tiene un costo: RiscOS sólo puede ejecutar aplicaciones escritas específicamente para RiscOS, las cuales existen en menor cantidad que las que están escritas en Linux.

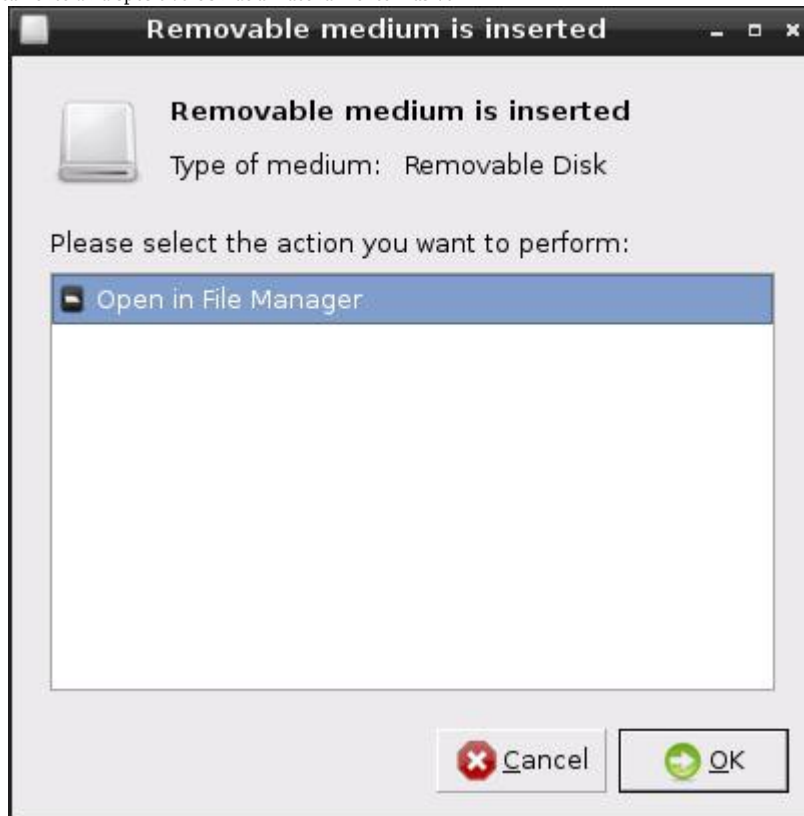
## Utilizando Dispositivos de Almacenamiento Externo

La tarjeta SD que almacena todos los archivos y directorios de la RasPi, no es muy grande. La tarjeta SD más grande disponible al momento de escribir estas líneas es de 256 GB, la cual es pequeña comparada con los 4,000 GB (4 TB) disponibles en los discos duros de mayor tamaño de las PC de escritorio.

Si está empleando la RasPi para reproducir archivos de video (ver Capítulo 8 “La RasPi como una PC Teatro en Casa”) necesitará más espacio para el almacenamiento que el que le puede ofrecer una tarjeta SD. Como aprendió en el Capítulo 1, “Conozca la Raspberry Pi”, es posible conectar dispositivos USB de Almacenamiento Masivo (UMS) a la RasPi a fin de obtener más espacio de almacenamiento.

No obstante, antes de que estos dispositivos puedan ser accedidos, el sistema operativo necesita conocer sobre ellos. En Linux, este proceso es conocido como *montaje*. Si está ejecutando una versión de Linux con un entorno de escritorio cargado (como LXDE de la distribución Raspbian, cargado desde la consola con el comando `startx`) este proceso es automático. Simplemente conecte el dispositivo a un puerto USB libre de la RasPi o del concentrador USB (hub USB), y el dispositivo y sus contenidos serán inmediatamente accesibles (ver Figura 3-2).

Figura 3-2  
LXDE montando automáticamente un dispositivo USB de almacenamiento masivo



Desde la consola, las cosas son sólo un poquito más difíciles. Para conseguir que un dispositivo sea accesible en Linux cuando el entorno de escritorio no se encuentre cargado, siga los siguientes pasos:

1. Conecte el dispositivo USB de almacenamiento a la RasPi directamente o a través de un concentrador USB conectado.
2. Escriba `sudo fdisk -l` para obtener una lista de las unidades conectadas a la RasPi y localice el dispositivo USB de almacenamiento identificándolo por su tamaño. Tome nota del nombre del dispositivo: `/dev/sdXN`, en donde *X* es la letra de unidad y *N* es el número de partición. Si es el único dispositivo conectado a la RasPi, éste vendrá a ser `/dev/sda1`.
3. Antes de que el dispositivo USB de almacenamiento sea accesible, Linux necesita un punto de montaje para este dispositivo. Cree este punto de montaje escribiendo `sudo mkdir /media/unidadexterna`.
4. Hasta ahora, el dispositivo es accesible sólo por el usuario root. Para hacerlo accesible para todos los usuarios, escriba en una sola línea lo siguiente:  
`sudo chgrp -R users /media/unidadexterna && sudo chmod -R g+w /media/unidadexterna`
5. Escriba el siguiente comando para montar el dispositivo USB de almacenamiento y conseguir acceder al dispositivo y a su contenido:  
`sudo mount /dev/sdXN /media/unidadexterna -o=rw`

## Creación de una Nueva Cuenta de Usuario

A diferencia de otros sistemas operativos que están diseñados en gran medida para ser utilizados por una sola persona, Linux es el fondo un sistema operativo social, diseñado para alojar a muchos usuarios. De forma predeterminada, Raspbian está configurado con dos cuentas de usuario: *pi*, que es la cuenta de usuario normal, y *root*, que es una cuenta de superusuario con privilegios adicionales.



**CONSEJO**

No se sienta tentado a iniciar sesión como *root* todo el tiempo. Usar una cuenta de usuario sin privilegios le protege de arruinar accidentalmente su sistema operativo y de los estragos de los virus y otros programas maliciosos descargados de la Internet.

Aunque es posible utilizar la cuenta *pi* sin ningún problema, sería mucho mejor si se creará su propia cuenta de usuario dedicada. Además, podría también crear cuentas adicionales para los amigos o miembros de la familia que deseen trabajar con la RasPi.

El procedimiento para crear una nueva cuenta es sumamente sencillo, y es el mismo para todas las distribuciones, lo único que varía es el nombre de usuario y la contraseña de inicio utilizada para iniciar sesión en la RasPi. Para crear una nueva cuenta sólo siga los siguientes pasos:

1. Inicie sesión en la RasPi utilizando una cuenta de usuario existente (usuario *pi* y contraseña *raspberry*, si está usando la distribución Raspbian recomendada).
2. Escriba lo siguiente en una sola línea, sin espacios después de las comas:

```
sudo useradd -m -G adm,dialout,cdrom,audio,plugdev,users,lpadm,lpadmin,smbshare,vchiq,powervv nombre_usuario
```

Lo anterior crea una nueva cuenta de usuario en blanco, llamada *nombre\_usuario*.

3. Para establecer una contraseña a la nueva cuenta, escriba `sudo passwd nombre_usuario` seguido de la nueva contraseña cuando se le solicite.

Explicación de lo que acaba de suceder: el comando `sudo` le dice al sistema operativo que el comando que está escribiendo se debe ejecutar como si se hubiera iniciado sesión con la cuenta de usuario *root*. El comando `useradd` le dice a Linux que desea crear una nueva cuenta de usuario. La parte `-m` (conocida como *bandera* u *opción*) le dice al programa `useradd` que cree un directorio de inicio (home) en donde el nuevo usuario podrá almacenar sus archivos. La larga lista después de la opción `-G` es la lista de grupos de los cuales el usuario será un miembro.

## Usuarios y Grupos

En Linux, cada usuario tiene tres atributos principales: su *ID de Usuario (UID)*, su *ID de Grupo (GID)* y una lista complementaria de los grupos de los cuales forma parte, es decir, es miembro. Un usuario puede ser miembro de tantos grupos como le plazca, aunque sólo uno puede ser el grupo primario o principal del usuario. Este grupo es por lo general un grupo que se autodenomina con el mismo nombre del usuario.

La pertenencia a grupos es algo importante. Si bien los usuarios pueden tener acceso directo a los archivos y a los dispositivos en el sistema, es más común que un usuario obtenga acceso a éstos a través de la pertenencia a grupos. El grupo `audio`, por ejemplo, concede a todos sus miembros la capacidad para acceder al hardware de reproducción de sonido de la RasPi. Sin esa membresía, el usuario no sería capaz de escuchar nada de música.

Para ver las pertenencias a grupos de un usuario escriba `groups nombre_usuario` en la terminal. Si utiliza este comando con el usuario *pi* predeterminado, verá la lista de grupos a los que cualquier miembro debería unirse para hacer uso de la RasPi. Aquí es donde se encuentra toda la información utilizada en el paso 2 del anterior procedimiento.

## Sistema de Archivos

El contenido de la tarjeta SD es conocido como el *sistema de archivos*, y se encuentra dividido en varias secciones, cada una con un propósito particular. Aunque no sea necesario para usted entender lo que hace cada sección a fin de utilizar la Raspberry Pi, le puede resultar útil contar con conocimientos previos por si algo no marcha bien.

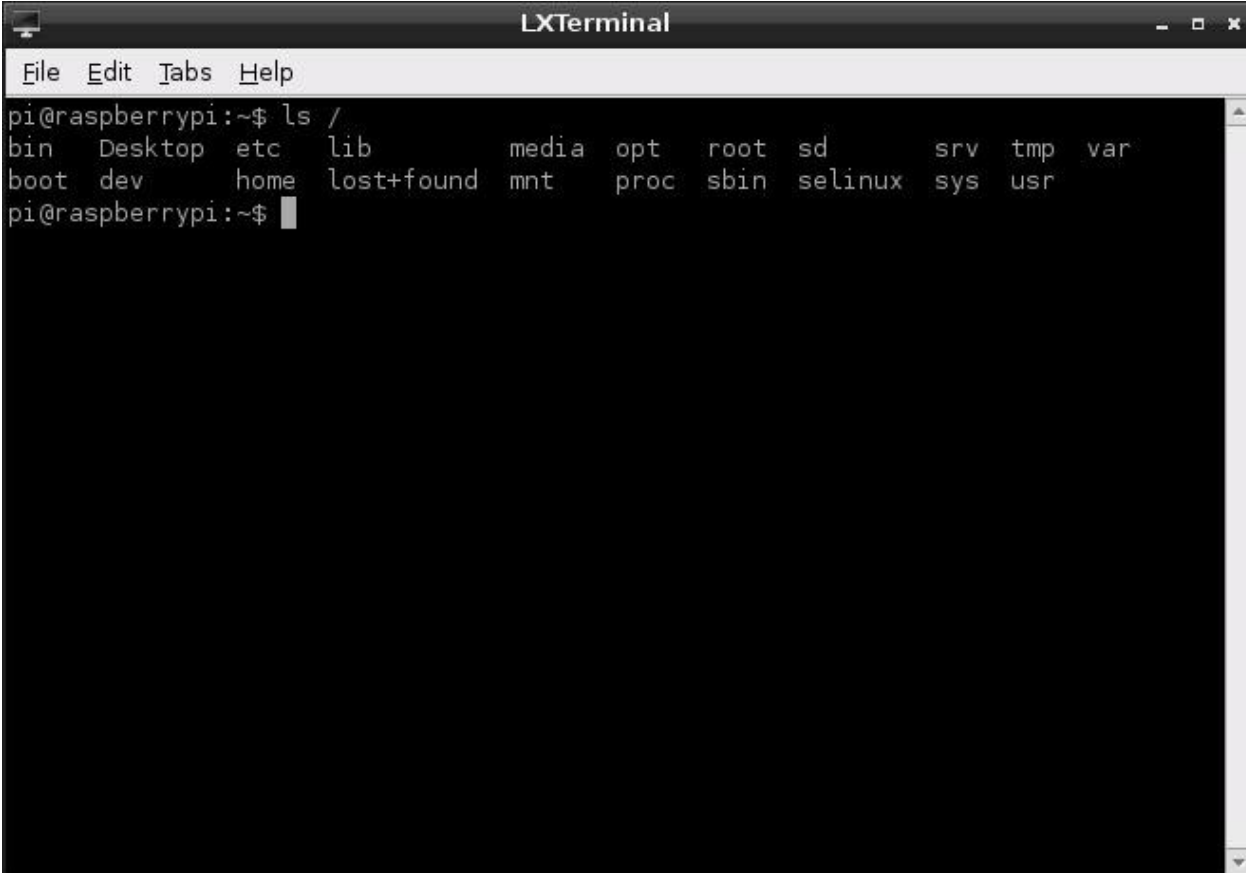
## Distribución Lógica

La manera en que Linux gestiona las unidades, archivos, carpetas y dispositivos es un poco diferente a la de otros sistemas operativos. En vez de tener varias unidades etiquetadas con una letra, todo se presenta como una ramificación hacia abajo conocida como *sistema de archivos raíz*.

Si inicia sesión en su RasPi y escribe `ls /` verá varios directorios desplegarse (ver Figura 3-3). Algunos de éstos son regiones de la tarjeta SD para el almacenamiento de archivos, mientras que otros son *directorios virtuales* para acceder a diferentes partes del sistema operativo o hardware.

Figura 3-3

Un listado de directorios para el sistema de archivos raíz de la RasPi



```
pi@raspberrypi:~$ ls /
bin  Desktop  etc  lib      media  opt  root  sd      srv  tmp  var
boot dev      home lost+found mnt   proc sbin  selinux sys  usr
pi@raspberrypi:~$
```

Los directorios visibles en la distribución Raspbian por defecto son los siguientes:

- **boot**-Este directorio contiene el núcleo Linux y otros paquetes necesarios para iniciar la RasPi.
- **bin**-Los archivos binarios relacionados con el sistema operativo, como por ejemplo, los archivos requeridos para ejecutar la GUI son almacenados aquí.
- **dev**-Este es un directorio virtual, que no existe en realidad sobre la tarjeta SD. Todos los dispositivos conectados al sistema (incluyendo los dispositivos de almacenamiento, la tarjeta de sonido y el puerto HDMI) pueden accederse desde aquí.
- **etc**-Este directorio almacena diversos archivos de configuración, incluyendo la lista de los usuarios y sus contraseñas cifradas.
- **home**-Cada usuario tiene un subdirectorio bajo este directorio para almacenar todos sus archivos personales.

- **lib**-Este directorio es un espacio de almacenamiento para las *bibliotecas* o *librerías*, que son trozos de código compartido requeridos por numerosas aplicaciones.
- **lost+found**-Este es un directorio especial en donde *fragmentos de archivos* se almacenan por si el sistema se bloquea.
- **media**-Este es un directorio especial para los dispositivos de almacenamiento extraíbles, como las memorias USB o las unidades de CD externas.
- **mnt**-Esta carpeta es utilizada para *montar* manualmente los dispositivos de almacenamiento, tales como los discos duros externos.
- **opt**-Este directorio almacena el software opcional que no forma parte del sistema operativo en sí. Si instala un nuevo software en la RasPi, por lo general vendrá a caer aquí.
- **proc**-Este es otro directorio virtual, que contiene información sobre los programas en ejecución que en Linux son conocidos como *procesos*.
- **selinux**-Los archivos relacionados con la *Seguridad Mejorada de Linux (Security Enhanced Linux)*, una suite de utilidades originalmente desarrolladas por la Agencia de Seguridad Nacional de EE.UU.
- **sbin**-Este directorio almacena archivos binarios especiales, principalmente utilizado por la cuenta root (superusuario) para mantenimiento del sistema.
- **sys**-En este directorio es donde se almacenan los archivos especiales del sistema operativo.
- **tmp**-Los archivos temporales se almacenan aquí automáticamente.
- **usr**-Este directorio proporciona almacenamiento para los programas accesibles por el usuario.
- **var**-Este es un directorio virtual que utilizan los programas para almacenar valores que cambian o variables.

## Distribución Física

Aunque la lista anterior muestra como el sistema de archivos aparece en el sistema operativo Linux, no es así como se encuentra distribuida físicamente sobre la tarjeta SD. Para la distribución Raspbian por defecto, la tarjeta SD se encuentra organizada en dos secciones principales, conocidas como *particiones*, porque dividen el dispositivo en diferentes regiones en una forma muy parecida a la que los capítulos de este libro ayudan a organizar el contenido del mismo.

La primera partición en el disco es una pequeña partición (75 MB aprox.) con formato VFAT, el mismo formato de partición que utiliza Microsoft Windows para las unidades extraíbles. Ésta se *monta* o hace accesible por Linux en el directorio `/boot` y contiene todos los archivos necesarios para configurar la Raspberry Pi y para cargar Linux en sí.

La segunda partición es más grande y está formateada como EXT4, un sistema de archivos nativo de Linux diseñado para conseguir acceso de alta velocidad a los datos y para garantizar la seguridad de los mismos. Esta partición aloja la parte principal de la distribución. Todos los programas, el escritorio, los archivos del usuario y cualquier software que instale por su cuenta se almacenan aquí. Esta partición ocupa la mayor parte de la tarjeta SD.

## Instalación y Desinstalación de Software

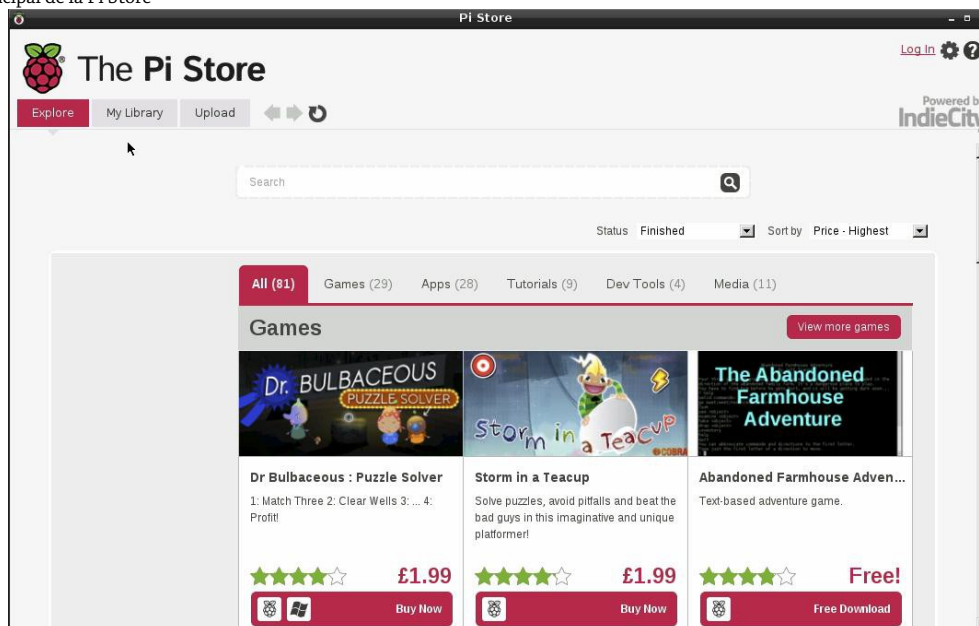
El software por defecto instalado con la distribución Raspbian es suficiente para empezar, pero lo más probable es que desee personalizar su RasPi de acuerdo a sus propias necesidades.

## Obtener Software desde la Pi Store

La instalación de nuevo software en la RasPi es muy sencilla, gracias a la inclusión de la Raspberry Pi Store. Esta es una tienda en donde los programadores pueden ofrecer su software para su rápida descarga e instalación. Para empezar, haga doble clic en el icono Pi Store en el escritorio. La ventana que aparece le permite acceder a todo el software que actualmente se encuentra disponible a través de la Pi Store (ver Figura 3-4).

Figura 3-4

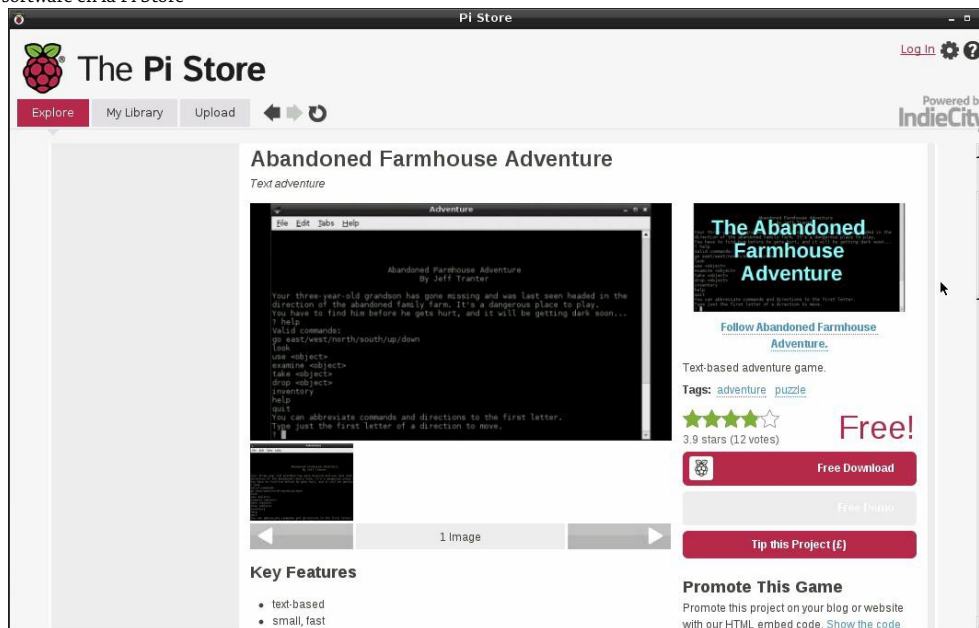
La ventana principal de la Pi Store



El software en la Pi Store está dividido en categorías que van desde juegos hasta tutoriales (inclusive videos, libros y revistas). Al hacer clic en alguna categoría de la parte superior se visualizará todo el software disponible en esa categoría dentro de una cuadrícula, en la cual se puede desplazarse utilizando el teclado o ratón. Al hacer clic en alguna pieza de software accederá a su información detallada así como a un enlace para descargar e instalar automáticamente el software (ver Figura 3-5).

Figura 3-5

Visualizando el software en la Pi Store

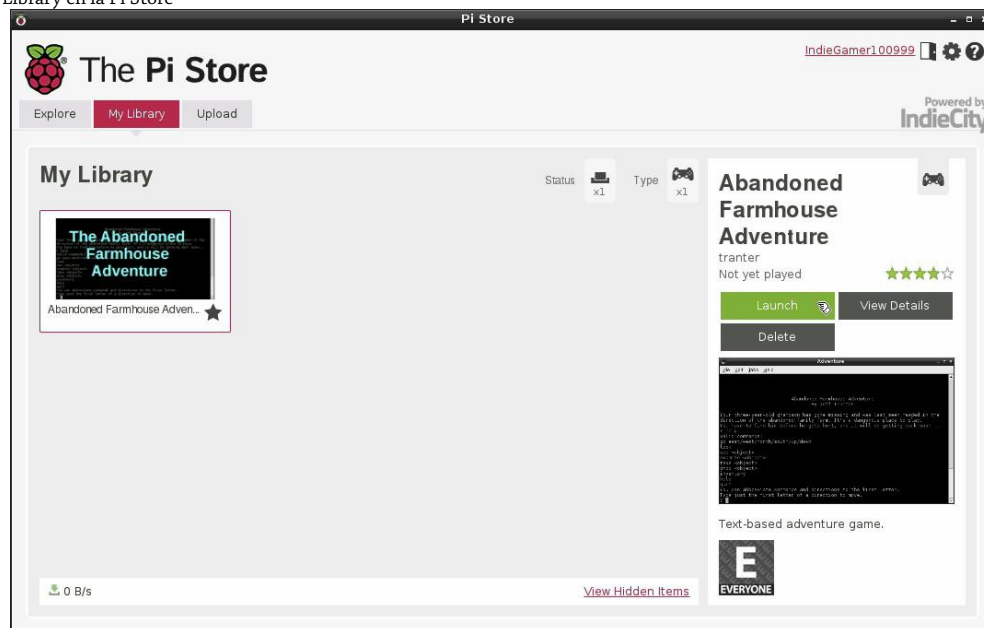


Antes de poder instalar el software, sin embargo, es necesario que registre una cuenta. La Pi Store funciona gracias a un especialista de la distribución de contenidos digitales, IndieCity; si ya tiene una cuenta en IndieCity, puede utilizarla haciendo clic en la opción *Log In* en la parte superior derecha de la ventana de la Pi Store e introducir su correo electrónico y contraseña. Si no la tiene, cuando trate de descargar algún software se le solicitará que se registre: simplemente haga clic en el botón *Register* y complete su dirección de correo electrónico, en seguida escriba la contraseña para su cuenta de usuario en las dos casillas correspondientes.

Algunos programas en la Pi Store son de paga. Si este fuera el caso, al tratar de instalar el software se le preguntará su nombre, dirección y los detalles para el pagos. Estos detalles serán transmitidos mediante una *conexión cifrada*, lo que significa que no pueden ser vistas durante el tránsito y son utilizadas por el operador de IndieCity de la Pi Store para tomar el pago del artículo. Una vez realizado el pago, el software se puede descargar de forma normal.

Todo el software que instale a través de la Pi Store se coloca en la sección *My Library* (ver Figura 3-6). Esto le permite tener una forma fácil de desinstalar el software o de reinstalar el software que ya ha utilizado anteriormente pero que elimino. Las descargas están vinculadas a su cuenta de usuario; si inicia sesión en la Pi Store con una Raspberry Pi diferente, puede utilizar *My Library* para buscar e instalar su software favorito rápidamente.

**Figura 3-6:**  
La pantalla My Library en la Pi Store



Cuando aprenda a utilizar su Raspberry Pi, y sobre todo cuando aprenda a programar, es posible que pueda escribir su propio software. Ya sea que se trate de un juego, una aplicación productividad o incluso un tutorial, podrá fácil y rápidamente ponerlo a disposición en los listados de descarga de la Pi Store haciendo clic en la opción *Upload* y completando el formulario que aparece.

Si tiene algún problema con la Pi Store al momento de la descarga o durante la publicación del software, haga clic sobre el icono con el signo de interrogación en la parte superior derecha de la ventana para acceder a la ayuda.

## Obtener Software desde Otros Lugares

Para contar con una mayor selección de paquetes, la distribución Debian incluye una herramienta llamada *apt*, la cual es un poderoso *gestor de paquetes*. Paquetes, es el término que emplea Linux para nombrar a las piezas de software o alguna colección de diferentes piezas de software diseñadas para trabajar juntas.

Aunque **apt** está diseñada para ser utilizada desde la línea de comandos, es muy fácil de usar y aprender. Existen interfaces gráficas (GUIs) para **apt**, tales como el popular gestor de paquetes Synaptic, pero a dichos gestores a menudo les cuesta un poco ejecutarse sobre la RasPi debido a la falta de memoria. Por consiguiente, recomendamos que el software se instale desde la terminal.

## Otras Distribuciones

Debian y las distribuciones basadas sobre Debian, por lo general utilizan **apt** como su gestor de paquetes. Pero **apt** no es la única herramienta en ese ámbito y otras distribuciones cuentan con diferentes alternativas. Pidora, por ejemplo, utiliza la herramienta **pacman**.

**Pacman** no es más difícil de utilizar que **apt**, pero su sintaxis (la forma que ésta espera de usted las frases de instrucciones para instalar el nuevo software o quitar el software existente) es diferente. Para conocer los detalles de cómo utilizar **pacman** en vez de **apt**, escriba `man pacman` en la terminal de Pidora.

Otras distribuciones pueden utilizar el gestor de paquetes **yum**. Si está probando una distribución que usa **yum**, simplemente escriba `man yum` en la terminal para ver las instrucciones.

El trabajo del gestor de paquetes es hacer un seguimiento de todo el software instalado en el sistema. No solamente instalar nuevo software, sino que también mantener el control del software que está actualmente instalado, permitir que el software antiguo sea eliminado e instalar actualizaciones cuando se encuentren disponibles.

La gestión de paquetes es una de las áreas en donde Linux difiere enormemente de los sistemas operativos como Windows u OS X. Aunque se puede descargar manualmente el nuevo software a instalar, es más común utilizar las herramientas integradas para la gestión de paquetes.

---

### CONSEJO

Antes de tratar de instalar el nuevo software o actualizar el software existente, es necesario asegurarse que la *caché* de **apt** está actualizada. Para hacerlo, basta con escribir el comando `sudo apt-get update`.

---

## Encontrar el Software que Desea

El primer paso para instalar una nueva pieza de software es averiguar cómo se llama. La forma más sencilla de hacerlo es buscar en la *caché* de paquetes de software disponible. Esta *caché* es una lista de todo el software disponible que se puede instalar a través de **apt**, almacenada en servidores de Internet conocidos como *repositorios*.

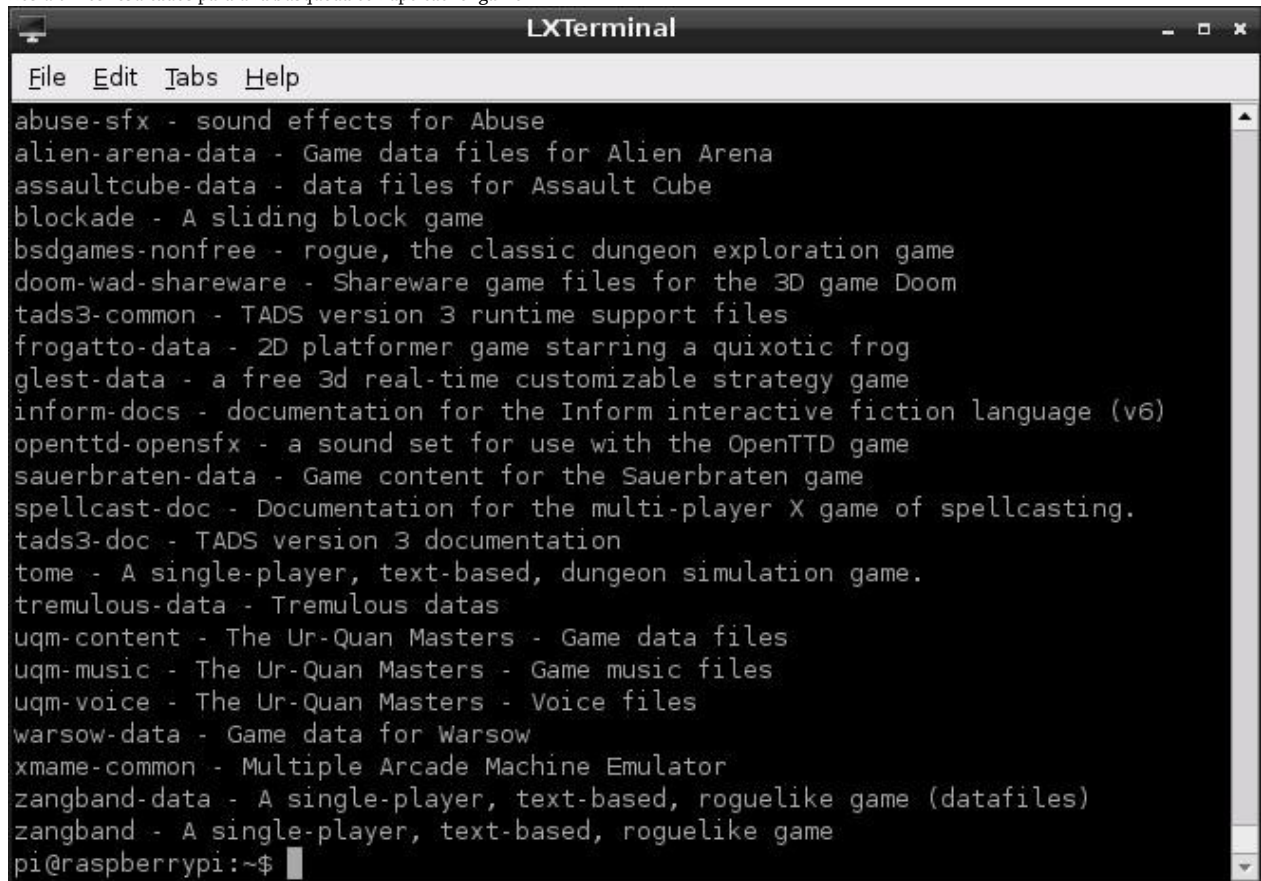
El software **apt** incluye una herramienta para la gestión de esta *caché*, llamada **apt-cache**. Este software le permite realizar una búsqueda por todos los paquetes de software disponibles en busca de una determinada palabra o frase.

Por ejemplo, para encontrar un juego, escriba el siguiente comando:

```
apt-cache search game
```

Esto le dice a **apt-cache** que busque en su lista de software disponible por cualquier cosa que tenga la palabra “game” en su título o descripción. Para los términos de búsqueda muy comunes, podría terminar obteniendo una lista de software muy larga (ver Figura 3-7), así que trate de ser lo más específico posible.

Figura 3-7  
Los últimos resultados para una búsqueda con apt-cache "game"



```

LXTerminal
File Edit Tabs Help
abuse-sfx - sound effects for Abuse
alien-arena-data - Game data files for Alien Arena
assaultcube-data - data files for Assault Cube
blockade - A sliding block game
bsdgames-nonfree - rogue, the classic dungeon exploration game
doom-wad-shareware - Shareware game files for the 3D game Doom
tads3-common - TADS version 3 runtime support files
frogatto-data - 2D platformer game starring a quixotic frog
glest-data - a free 3d real-time customizable strategy game
inform-docs - documentation for the Inform interactive fiction language (v6)
openttd-opensfx - a sound set for use with the OpenTTD game
sauerbraten-data - Game content for the Sauerbraten game
spellcast-doc - Documentation for the multi-player X game of spellcasting.
tads3-doc - TADS version 3 documentation
tome - A single-player, text-based, dungeon simulation game.
tremulous-data - Tremulous datas
uqm-content - The Ur-Quan Masters - Game data files
uqm-music - The Ur-Quan Masters - Game music files
uqm-voice - The Ur-Quan Masters - Voice files
warsow-data - Game data for Warsow
xname-common - Multiple Arcade Machine Emulator
zangband-data - A single-player, text-based, roguelike game (datafiles)
zangband - A single-player, text-based, roguelike game
pi@raspberrypi:~$

```

#### CONSEJO

Si el término de búsqueda nos devuelve demasiados paquetes diferentes imposibles de verlos todos juntos en una sola pantalla, puede decirle a Linux que desea hacer una pausa entre cada pantalla llena *canalizando* la salida de apt-cache a través de una herramienta llamada less. Simplemente cambie el comando a apt-cache search game | less y utilice las teclas de dirección para desplazarse por la lista. Presione la letra Q del teclado para salir.

## Instalación del Software

Una vez que conoce el nombre del software que desea instalar, modificaremos el comando **apt-get** a fin de instalarlo. La instalación del software es un privilegio permitido sólo para el usuario *root*, ya que afecta a todos los usuarios de la RasPi. Por consiguiente, los comandos necesitan ser precedidos con **sudo** para indicarle al sistema operativo que debe de ejecutarse como el usuario *root*.

Por ejemplo, para instalar el paquete **nethack-console** (un juego de rol basado en consola que se genera aleatoriamente), debe simplemente utilizar el comando **install** junto con **apt-get** como se muestra a continuación:

```
sudo apt-get install nethack-console
```

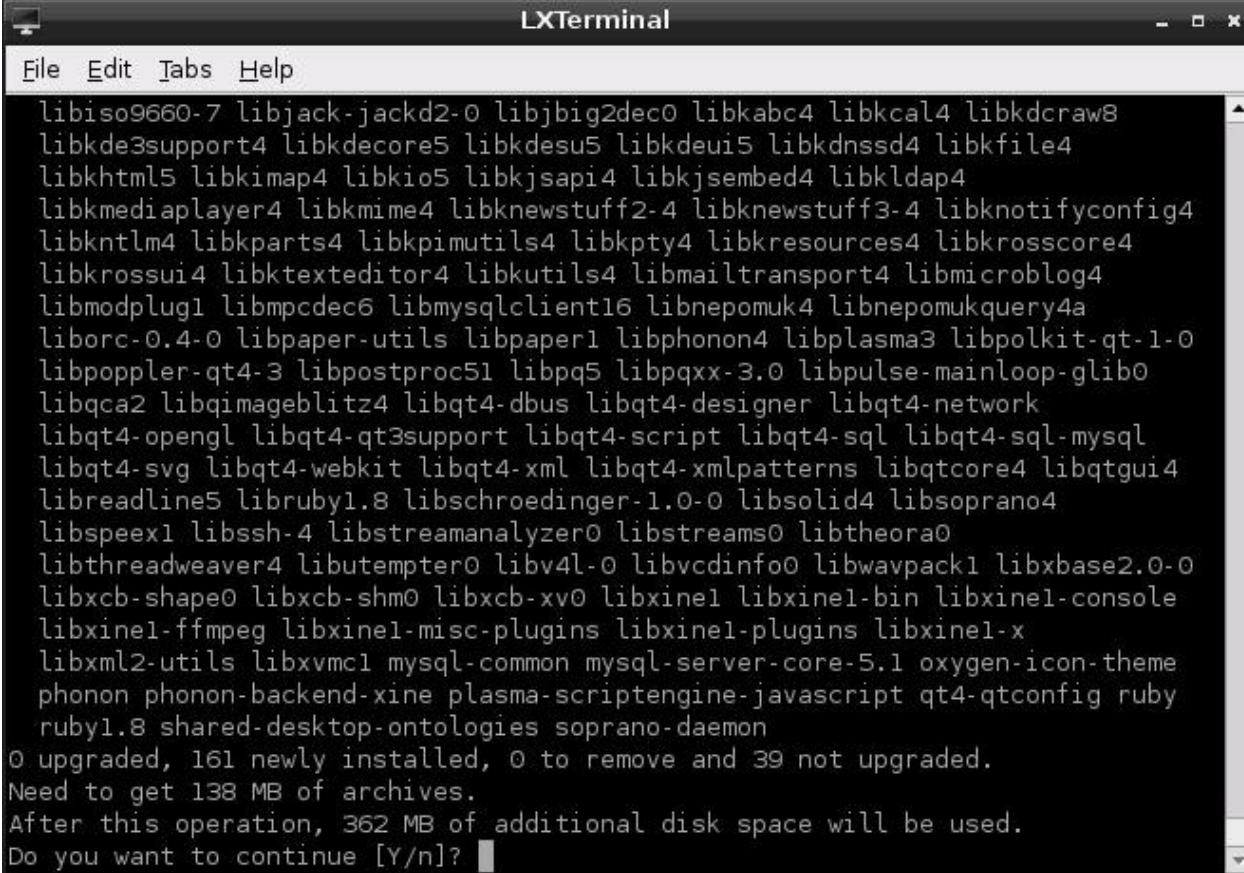
Algunos paquetes dependen de otros paquetes a fin de poder funcionar. Un lenguaje de programación puede depender de un compilador, un motor de juego de los archivos gráficos, o un reproductor de audio de los codecs para poder reproducir diferentes formatos. Estos paquetes son conocidos en términos de Linux como *dependencias*.



Las dependencias son una de las principales razones por las que es mejor utilizar un gestor de paquetes como **apt** en lugar de instalar el software manualmente. Si un paquete depende de otros paquetes, **apt** automáticamente los encontrará (ver Figura 3-8) y los preparará para su instalación. Si éste es el caso, se mostrará un mensaje preguntándole si desea continuar con la instalación. Si lo desea, escriba la letra Y y presione la tecla Intro.

Figura 3-8

Apt listando las dependencias para el paquete OpenOffice.org



```

LXTerminal
File Edit Tabs Help

libiso9660-7 libjack-jackd2-0 libjbig2dec0 libkabc4 libkcal4 libkdcraw8
libkde3support4 libkdecore5 libkdesu5 libkdeui5 libkdnssd4 libkfile4
libkhtml5 libkimap4 libkio5 libkjsapi4 libkjsembed4 libkldap4
libkmediaplayer4 libkmime4 libknewstuff2-4 libknewstuff3-4 libknotifyconfig4
libkntlm4 libkparts4 libkpimutils4 libkpty4 libkresources4 libkrosscore4
libkrossui4 libktexteditor4 libkutils4 libmailtransport4 libmicroblog4
libmodplug1 libmpcdec6 libmysqlclient16 libnepomuk4 libnepomukquery4a
liborc-0.4-0 libpaper-utils libpaper1 libphonon4 libplasma3 libpolkit-qt-1-0
libpoppler-qt4-3 libpostproc5.1 libpq5 libpqxx-3.0 libpulse-mainloop-glib0
libqca2 libqimageblitz4 libqt4-dbus libqt4-designer libqt4-network
libqt4-opengl libqt4-qt3support libqt4-script libqt4-sql libqt4-sql-mysql
libqt4-svg libqt4-webkit libqt4-xml libqt4-xmlpatterns libqtcore4 libqtgui4
libreadline5 libruby1.8 libschroedinger-1.0-0 libsolid4 libsoprano4
libspeex1 libssh-4 libstreamanalyzer0 libstreams0 libtheora0
libthreadweaver4 libutempter0 libv4l-0 libvcdinfo0 libwavpack1 libxbase2.0-0
libxcb-shape0 libxcb-shm0 libxcb-xv0 libxine1 libxine1-bin libxine1-console
libxine1-ffmpeg libxine1-misc-plugins libxine1-plugins libxine1-x
libxml2-utils libxvmc1 mysql-common mysql-server-core-5.1 oxygen-icon-theme
phonon phonon-backend-xine plasma-scriptengine-javascript qt4-qtconfig ruby
ruby1.8 shared-desktop-ontologies soprano-daemon
0 upgraded, 161 newly installed, 0 to remove and 39 not upgraded.
Need to get 138 MB of archives.
After this operation, 362 MB of additional disk space will be used.
Do you want to continue [Y/n]?

```

## Desinstalación del Software

Si decide que ya no necesita una pieza de software, **apt-get** también incluye un comando **remove** para eliminar el software, el cual desinstala limpiamente el software junto con cualquiera de las dependencias que ya no sean requeridas. Cuando utiliza una tarjeta SD de poca capacidad con su RasPi, la posibilidad de probar el software y eliminarlo rápidamente resulta de gran utilidad.

Para eliminar **nethack-console**, simplemente abra la terminal y escriba el siguiente comando:

```
sudo apt-get remove nethack-console
```

El comando **remove** tiene un hermano más poderoso en la forma del comando **purge**. Que al igual que **remove**, el comando **purge** se deshace del software que ya no necesita. Mientras que el comando **remove** deja los archivos de configuración del software intactos, **purge** lo elimina todo. Si se metió en un lío al personalizar un paquete y ya no funciona, **purge** es el comando a utilizar. Por ejemplo, para purgar **nethack-console**, escriba lo siguiente:

```
sudo apt-get purge nethack-console
```

## Actualización del Software

Además de instalar y desinstalar paquetes, **apt** puede utilizarse para mantenerlos actualizados. La actualización de un paquete a través de **apt** le garantiza que ha recibido las últimas actualizaciones, correcciones de errores y parches de seguridad.

Antes de intentar actualizar un paquete, asegúrese que la caché de **apt** sea la más reciente llevando a cabo una actualización:

```
sudo apt-get update
```

Al actualizar el software, tiene dos opciones: puede actualizar de una vez todo en el sistema o actualizar los programas individualmente. Si sólo desea mantener su distribución actualizada, la forma de conseguirlo es escribiendo lo siguiente:

```
sudo apt-get upgrade
```

Para actualizar un paquete individual, basta con decirle a **apt** que lo vuelva a instalar. Por ejemplo, para instalar una actualización de **nethack-console**, debe escribir lo siguiente:

```
sudo apt-get install nethack-console
```

Si el paquete ya se encuentra instalado, **apt** en su lugar lo tratará como una actualización. Si ya cuenta con la última versión disponible del software, **apt** sencillamente le dirá que no puede actualizar dicho software y saldrá.

---

### CONSEJO

Para obtener más información sobre la administración de paquetes con el comando **apt** (en particular, cómo hacer que ciertos paquetes pueden ser “resguardados” y excluidos de las actualizaciones) escriba `man apt` en la terminal.

---

## Apagando de Forma Segura la RasPi

Aunque la RasPi no tiene un interruptor de encendido como una computadora ordinaria, no significa que simplemente tenga que desconectar el cable de la alimentación cuando haya usted terminado. Aun cuando no parezca estar haciendo nada, la RasPi a menudo lee y escribe en la tarjeta SD, por tal motivo, si pierde la energía mientras esto ocurre los contenidos de la tarjeta pueden volverse *corruptos* e ilegibles.

Para apagar la RasPi utilizando la terminal, escriba el siguiente comando:

```
sudo shutdown -h now
```

La RasPi cerrará todas las aplicaciones, avisándole que guarde sus archivos abiertos si aún no lo ha hecho y procederá a apagarse a sí mismo, en ese momento la pantalla se quedará en negro y la luz ACT se apagará. Una vez que la luz se haya apagado, es seguro retirar el cable micro-USB de la RasPi o desconectar el alimentador de corriente de la toma en la pared.

Cuando quiera volver a encender la RasPi o si la apago por accidente, simplemente vuelva a conectar el cable de la alimentación y se iniciará el arranque automáticamente.

# **Capítulo 4**

## Solución de Problemas

**En ocasiones, las cosas no** van del todo bien. Entre más complejo sea el dispositivo, más complejos son los problemas que pueden surgir y la RasPi es de cierto un dispositivo extremadamente complejo.

Afortunadamente, los problemas más frecuentes son fáciles de diagnosticar y de corregir. En este capítulo, conocerá algunas de las razones más comunes por las que la RasPi puede comportarse mal y cómo corregirlas.

## Diagnóstico del Teclado y el Ratón

Quizás el problema más común que los usuarios experimentan con la Raspberry Pi es cuando el teclado repite ciertos caracteres. Por ejemplo, si el comando `startx` aparece en la pantalla como `stttttttttartxxxxxxxxxxx` será comprensible que no funcione cuando la tecla Intro sea presionada.

Normalmente, existen dos razones por las que un teclado USB no funciona correctamente cuando se conecta a la Raspberry Pi: está absorbiendo demasiada energía o su chipset interno se encuentra en conflicto con los circuitos USB de la RasPi.

Consulte la documentación de su teclado o la etiqueta en la parte inferior del mismo, para averiguar si se indica una potencia expresada en *miliamperios (mA)*. Esta es la cantidad de energía que su teclado intenta sacar del puerto USB cuando se encuentra funcionando.

Los puertos USB de la RasPi tienen un componente conectado a ellos denominado *polyfuse*, el cual protege a la RasPi en caso de que un dispositivo intente extraer demasiada energía. Cuando se dispara o activa este polyfuse, provoca que el puerto USB se apague cercano a los 150 mA. Si su teclado consume mucha energía podría funcionar de manera extraña (o no hacerlo en absoluto). Esto puede ser un problema con los teclados que tienen LEDs de iluminación integrados, los cuales requieren mucha más energía para operar que un teclado estándar.

Si se da cuenta que su teclado USB consume demasiada energía, trate de conectarlo a un Hub USB en vez de conectarlo directamente a la RasPi. Esto permitirá al teclado obtener su energía desde la fuente de alimentación del concentrador USB, en lugar de la propia RasPi. Opcionalmente, puede cambiar su teclado por un modelo que demande menos energía. El problema de la repetición de letras también puede deberse a una fuente de alimentación inadecuada para la RasPi, la cual se abordará en la siguiente sección, “Diagnóstico de la Fuente de Alimentación”.

La cuestión de la compatibilidad, por desgracia, es más difícil de diagnosticar. Si bien la gran mayoría de los teclados funcionan muy bien con la RasPi, otros pocos presentan algunos extraños síntomas. Estos síntomas van desde la respuesta intermitente del teclado, el síndrome de las teclas repetitivas o incluso bloqueos que impiden que la RasPi trabaje. En ocasiones, estos problemas no aparecen sino hasta que otros dispositivos USB son conectados a la RasPi. Si su teclado funciona bien hasta que otro dispositivo USB, particularmente un adaptador USB inalámbrico, es conectado, puede que tenga un problema de incompatibilidad.

Si le es posible, cambie su teclado por otro modelo. Si el nuevo teclado funciona, quiere decir que el anterior teclado es incompatible con la RasPi. Para conocer una lista de teclados no compatibles visite la wiki eLinux:

[http://elinux.org/RPi\\_USB\\_Keyboards](http://elinux.org/RPi_USB_Keyboards)

El mismo consejo sobre la comprobación de la compatibilidad aplica para los problemas experimentados con el ratón: la mayoría de los ratones USB y trackballs funciona bien, pero unos pocos presentan incompatibilidad con los circuitos USB propios de la RasPi. Esto por lo general se traduce a síntomas como un puntero de ratón que avanza con dificultades o que no responde, pero en ocasiones también que puede llevar a que la RasPi fracase al momento de cargar o que se bloquee en intervalos aleatorios. Si está buscando adquirir un ratón nuevo, una lista actualizada de los modelos conocidos que funcionan con la RasPi se encuentra disponible en la wiki eLinux:

[http://elinux.org/RPi\\_USB\\_Mouse\\_Devices](http://elinux.org/RPi_USB_Mouse_Devices)

## Diagnóstico de la Fuente de Alimentación

Muchos de los problemas experimentados con la Raspberry Pi pueden deberse a una fuente de alimentación inadecuada. El Modelo A requiere una alimentación de 5 V capaz de suministrar una corriente de 500 mA, mientras que los componentes adicionales del Modelo B aumentarán este requerimiento de corriente a 700 mA. No todos los adaptadores de corriente USB están diseñados para suministrar tanta energía, aunque en sus etiquetas afirmen lo contrario.

---

### CONSEJO

El estándar USB formal declara que los dispositivos no deben consumir más de 500 mA, incluso si algún dispositivo requiriera ese nivel de corriente sólo se podría disponer de aquella después de pasar por un proceso llamado *negociación*. Dado que la RasPi no negocia la potencia, es poco probable que funcione si se conecta directamente a los puertos USB de una PC de escritorio o laptop.

---

Si está experimentando problemas intermitentes con la RasPi (específicamente si el sistema funciona hasta que usted conecta algo en el puerto USB o inicia una operación que hace uso intenso del procesador como por ejemplo reproducir video) lo más probable es que la fuente de alimentación utilizada no es la adecuada. La RasPi ofrece una forma relativamente fácil de comprobar si éste es el caso, llevando a cabo una *prueba de voltaje entre dos puntos*.

Para realizar una prueba de voltaje entre los puntos, necesita un *voltímetro* o *multímetro* con la capacidad para medir voltaje de corriente continua (DC). Si su medidor cuenta con múltiples entradas para diferentes voltajes, utilice un ajuste apropiado.

---

### ADVERTENCIA

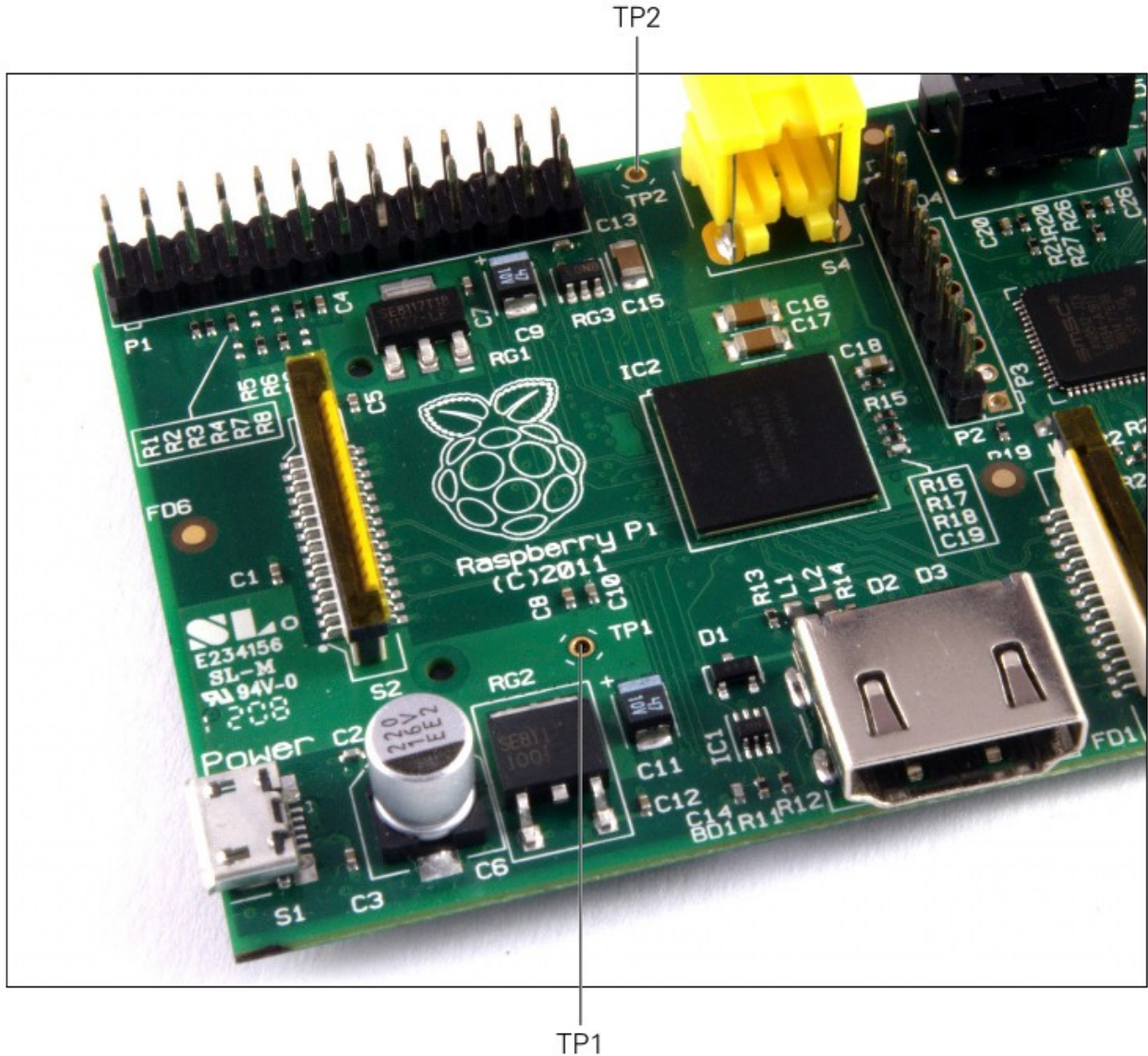
Evite que las puntas de prueba del voltímetro toquen cualquier cosas que no esté etiquetada como un punto de prueba. Es posible puentear el suministro de 5 V que entra a la RasPi al suministro interno de 3.3 V, creando así un corto circuito que puede dañar su dispositivo. Sea especialmente cuidadoso al dejar expuestos los pines del header.

---

Los dos puntos de prueba son pequeños orificios, bañados en cobre conocidos como *vías*, que se encuentran conectadas a los circuitos de 5 V y tierra de la RasPi. Coloque el cable de prueba positivo (rojo) sobre TP1, ubicado a la izquierda de la placa, justo por encima de un pequeño componente negro llamado *regulador* etiquetado con RG2. Conecte el cable de prueba negro (negativo) a TP2, localizado entre los pines de cobre GPIO y el conector fono RCA amarillo en la parte superior izquierda de la placa (ver Figura 4-1).

Figura 4-1

Los dos puntos de prueba de voltaje, etiquetados con TP1 y TP2



La lectura del voltímetro debe mostrarse entre 4.8 V y 5 V. Si es inferior a 4.8 V, indica que la RasPi no está recibiendo la suficiente potencia. Intente cambiar el alimentador de energía micro-USB por un modelo distinto y verifique que la etiqueta diga que puede suministrar 700 mA o más. Se recomienda un modelo de 1A, pero tenga cuidado con los modelos baratos (en ocasiones tienen el etiquetado incorrecto y fallan en suministrar la corriente que prometen). Los cargadores originales de los teléfonos móviles rara vez tienen problema, pero los dispositivos baratos sin marca (a menudo vendidos como *adaptadores compatibles*) deben evitarse.

Si su voltímetro lee un número negativo, no se preocupe: esto le indica que tiene las puntas de prueba positiva y negativa colocadas en el lugar equivocado. Cámbielas de lugar o simplemente ignore el signo negativo cuando anote su lectura.



## Diagnóstico de la Pantalla

Aun cuando la RasPi está diseñada para funcionar con casi cualquier dispositivo de pantalla HDMI, DVI o video compuesto, podría suceder que cuando la conecte no trabaje como se esperaba. Por ejemplo, es posible que la imagen se muestre cargada hacia un lado o completamente fuera de la pantalla, o que sólo se visualice con el tamaño de una estampilla postal recortada situada en el centro de la pantalla o en blanco y negro (o incluso no mostrarse en absoluto).

Lo primero que hacer es comprobar el tipo de dispositivo al cual está conectada la RasPi. Esto es de suma importancia cuando se está utilizando la conexión RCA compuesta para conectar la RasPi a la TV. Los distintos países usan diferentes estándares de vídeo para el televisor, lo que significa que una RasPi que está configurada para un país podría no funcionar en otro. Esta es la explicación normal para una RasPi que muestra el video en blanco y negro. Aprenderá como ajustar esta configuración en el Capítulo 7, “Configurando Avanzada de la Raspberry Pi”.

Cuando utiliza la salida HDMI, el tipo de pantalla por lo general es detectado automáticamente. Sin embargo, si utiliza un adaptador HDMI a DVI para conectar la RasPi a un monitor de computadora, en ocasiones podría no funcionar como se esperaba. Los síntomas más comunes incluyen: el ruido (nieve estática), partes de imagen desaparecidas o de plano ningún tipo de visualización. Para solucionar estos problemas, tome nota de la resolución y la tasa de refresco de su pantalla conectada y salte al Capítulo 7, “Configuración Avanzada de la Raspberry Pi” para averiguar cómo configurarla de forma manual.

Otro caso a parte es una imagen que se presenta demasiado grande o demasiado chica, lo que ocasiona que porciones de la imagen quedan fuera del borde de la pantalla o que las imágenes se muestren situadas en el centro de la pantalla rodeadas por un gran borde negro. Esto es provocado por una configuración conocida como *overscan*, que es utilizada cuando la RasPi es conectada a los televisores para evitar la impresión a las porciones de la pantalla que pueden quedar ocultas bajo un bisel. Como en los anteriores ajustes relacionados con la pantalla, aprenderá cómo configurar (o incluso deshabilitar por completo) el overscan en el Capítulo 7.

## Diagnóstico del Arranque

La causa más común por la que una RasPi no arranca se debe a un problema relacionado con la tarjeta SD. A diferencia de una PC de escritorio o laptop, la RasPi depende de los archivos almacenados en la tarjeta SD para todo. Si la RasPi no puede hablar con la tarjeta, no mostrará nada sobre la pantalla o ni dará señales de vida en absoluto.

Si se enciende el LED POWER de su RasPi al momento de conectar la fuente de alimentación micro-USB, pero no pasa nada y el LED OK permanece apagado, el problema es de la tarjeta SD. Primero, asegúrese de que la tarjeta funcione cuando la conecte en una PC, y que muestre las particiones y los archivos que se esperan de una tarjeta correctamente flasheada. (Para más detalles, ver Capítulo 3, “Administración del Sistema Linux”, particularmente la sección titulada “Sistema de Archivos” de ese capítulo).

Si la tarjeta funciona sobre una PC pero no en la RasPi, puede tratarse de un problema de compatibilidad. Algunas tarjetas SD (especialmente las tarjetas de alta velocidad marcadas con *Class 10* en sus etiquetas) no funcionan correctamente cuando son conectadas al lector de tarjetas SD de la RasPi. Una lista de las tarjetas compatibles con la RasPi puede encontrarse en la wiki eLinux: [http://elinux.org/RPi\\_SD\\_Cards](http://elinux.org/RPi_SD_Cards)

Lamentablemente, si tiene una tarjeta incompatible, puede que necesite reemplazarla por una tarjeta diferente a fin de conseguir que la RasPi funcione. Sin embargo, aun cuando el software base de la RasPi está desarrollado, se sigue trabajando para asegurar que una gama amplia de tarjetas trabajen correctamente con la RasPi. Antes de renunciar por completo a una tarjeta de alta velocidad, vea si se encuentra disponible una versión más actualizada de su distribución elegida (ver Capítulo 1, “Conozca la Raspberry Pi”, para obtener más información sobre de las distribuciones).



Si modificó la velocidad de su Raspberry Pi a través del overclocking (ver Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi”), esto también puede impedir que la RasPi arranque adecuadamente. Para deshabilitar el overclock y ejecutar la RasPi a su velocidad por defecto, mantenga presionada la tecla Shift mientras encienda la RasPi.

## Diagnóstico de la Red

La herramienta más utilizada para diagnosticar problemas con la red es `ifconfig`. Si está utilizando una conexión de red inalámbrica salte al Capítulo 5, “Configuración de la Red”, para obtener más información sobre una herramienta similar para esos dispositivos. De lo contrario, siga leyendo.

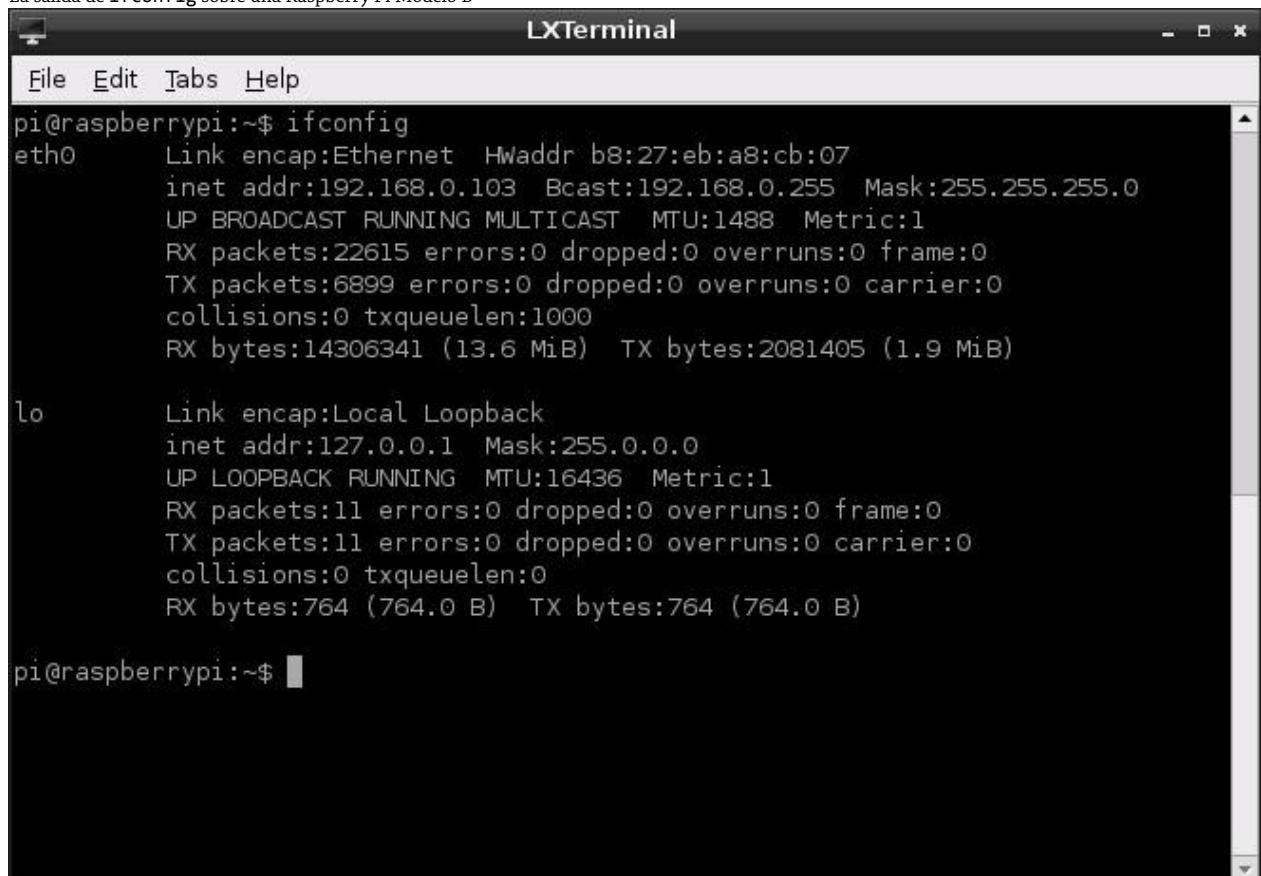
Diseñada para proporcionar información sobre los puertos de red conectados, `ifconfig` es una poderosa herramienta para controlar y configurar los puertos de red de la RasPi. Para utilizarla en su forma más básica, simplemente escriba el nombre de la herramienta en la terminal:

### `ifconfig`

Invocada de esta forma, `ifconfig` proporciona información sobre todos los puertos de red que pueda encontrar (ver Figura 4-2). Para la Raspberry Pi Modelo B estándar, hay dos puertos: el puerto Ethernet físico al lado derecho de la placa y una interfaz de red virtual *loopback* que permite a los programas sobre la RasPi comunicarse unos con otros.

Figura 4-2

La salida de `ifconfig` sobre una Raspberry Pi Modelo B



```
pi@raspberrypi:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:a8:cb:07
          inet addr:192.168.0.103  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1488  Metric:1
          RX packets:22615 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6899 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:14306341 (13.6 MiB)  TX bytes:2081405 (1.9 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:11 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:764 (764.0 B)  TX bytes:764 (764.0 B)

pi@raspberrypi:~$
```

La salida de `ifconfig` está dividida en las siguientes secciones:

- **Link encap**-El tipo de *encapsulación* empleado por la red, que sobre un Modelo B leerá Ethernet para el puerto de red físico o Local Loopback para el adaptador virtual loopback.
- **Hwaddr**-La *dirección MAC (Media Access Control)* de la interfaz de red, escrita en hexadecimal. Este dirección es única para cada dispositivo sobre la red, y cada RasPi tiene su propia dirección MAC, la cual es establecida en la fábrica.
- **inet addr**-La *dirección IP (Internet Protocol)* de la interfaz de red. Esta dirección es con la que usted identifica a la RasPi sobre la red si la está utilizando para ejecutar algún servicio accesible por la red, como por ejemplo un servidor web o servidor de archivos.
- **Bcast**-La *dirección de difusión (broadcast)* de la red en la que se encuentra conectada la RasPi. Cualquier tráfico enviado a esta dirección será recibido por todos los dispositivos sobre la red.
- **Mask**-La *máscara de red*, es la que controla el tamaño máximo de la red en la cual está conectada la RasPi. Para la mayoría de los usuarios domésticos, este leerá 255.255.255.0.
- **MTU**-El tamaño de la *unidad máxima de transferencia*, el cual indica cuan grande puede ser un sólo paquete de datos antes de que el sistema necesite dividirlo en varios paquetes.
- **RX**-Esta sección proporciona información sobre el tráfico de red recibido, incluyendo el registro de número de errores y paquetes perdidos. Si comienza a ver que aparecen errores en esta sección, hay algo mal con su red.
- **TX**-Este proporciona la misma información que la sección RX, pero para los paquetes transmitidos. Una vez más, los errores registrados aquí indican un problema con la red.
- **collisions**-Si dos sistemas sobre la red tratan de hablar al mismo tiempo, ocurre una *colisión* que requiere que los sistemas retransmitan sus paquetes. Un pequeño número de colisiones no es un problema, pero un gran número aquí indica un asunto de red que atender.
- **txqueuelen**-La longitud de la *cola de transmisión*, que normalmente se establece a 1000 y que rara vez necesita ser modificada.
- **RX bytes, TX bytes**-Un resumen de la cantidad de tráfico que la interfaz de red ha pasado.

Si tiene problemas con la red sobre la RasPi, primero debe deshabilitar la interfaz de red y luego volverla a habilitar. La forma más fácil de hacer esto es con las herramientas `ifup` y `ifdown`.

Si la red está construida, pero no funciona correctamente (por ejemplo, si `ifconfig` no muestra nada en la sección `inet addr`) comience por deshabilitar el puerto de red. Desde la terminal, escriba el siguiente comando:

```
sudo ifdown eth0
```

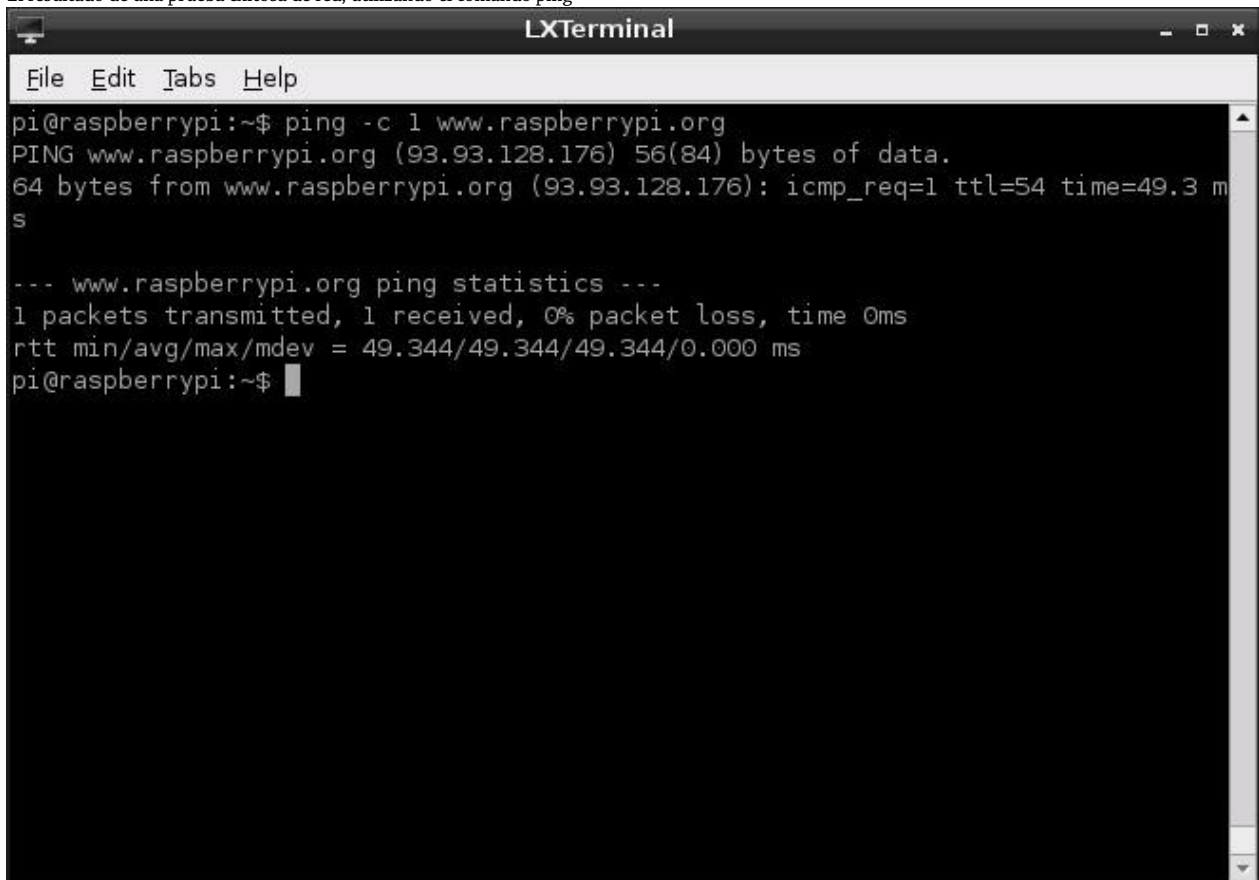
Una vez que la red ha sido deshabilitada, asegúrese de que el cable de red se encuentra insertado firmemente en ambos extremos, y que cualquiera que sea el dispositivo de red (hub, switch o router) al que la RasPi se conecta, esté encendido y trabajando. A continuación, vuelva a habilitar la interfaz con el siguiente comando:

```
sudo ifup eth0
```

Puede hacer una prueba de conexión de red utilizando el comando `ping`, el cual envía datos hacia una computadora remota y espera por una respuesta. Si todo funciona bien, deberá ver la misma respuesta como la que se muestra en la Figura 4-3. Si no es así, puede que necesite ajustar manualmente su configuración de red, lo cual aprenderá en el Capítulo 5, “Configuración de la Red”.

Figura 4-3

El resultado de una prueba exitosa de red, utilizando el comando ping

A screenshot of an LXTerminal window. The title bar says "LXTerminal". The menu bar has "File", "Edit", "Tabs", and "Help". The terminal content shows a user at the prompt "pi@raspberrypi:~\$ " typing "ping -c 1 www.raspberrypi.org". The output is: "PING www.raspberrypi.org (93.93.128.176) 56(84) bytes of data. 64 bytes from www.raspberrypi.org (93.93.128.176): icmp\_req=1 ttl=54 time=49.3 ms". This is followed by a separator line "--- www.raspberrypi.org ping statistics ---" and then the statistics: "1 packets transmitted, 1 received, 0% packet loss, time 0ms" and "rtt min/avg/max/mdev = 49.344/49.344/49.344/0.000 ms". The prompt "pi@raspberrypi:~\$ " is shown again at the bottom.

```
pi@raspberrypi:~$ ping -c 1 www.raspberrypi.org
PING www.raspberrypi.org (93.93.128.176) 56(84) bytes of data.
64 bytes from www.raspberrypi.org (93.93.128.176): icmp_req=1 ttl=54 time=49.3 ms

--- www.raspberrypi.org ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 49.344/49.344/49.344/0.000 ms
pi@raspberrypi:~$
```

## El Kernel de Emergencia

El núcleo Linux es el corazón del sistema operativo que controla a la RasPi. Es el responsable de todo, desde asegurar que usted pueda acceder a sus archivos hasta permitir a los programas comunicarse unos con otros.

Al encender su RasPi, se cargará el núcleo normal por defecto. Existe además de éste, un segundo núcleo incluido en la mayoría de las distribuciones, el cual permanece sin ser utilizado. Éste, es un *núcleo de emergencia* y como su nombre lo sugiere, se suele utilizar cuando el núcleo normal no está funcionando.

Es poco probable que necesite alguna vez arrancar su RasPi utilizando el núcleo de emergencia, pero vale la pena aprender cómo hacerlo por si acaso. Esto es especialmente importante si está actualizando su núcleo o está utilizando una nueva y poco probada distribución. En ocasiones, el software recién publicado tiene *errores* de programación que no se descubrieron antes de su lanzamiento. Cuando se encuentre con extraños errores después de la actualización, el núcleo de emergencia puede utilizarse para reducir el problema a la nueva versión del núcleo.

El núcleo Linux es simplemente un archivo ubicado en el directorio `/boot` llamado `kernel.img`. Cuando la RasPi es encendida por primera vez y empieza a cargar el sistema operativo, comienza por buscar este archivo y si no lo encuentra, la RasPi no funciona. El núcleo de emergencia es un segundo archivo ubicado en el mismo directorio `/boot`, llamado `kernel_emergency.img`.

En la mayoría de los casos el núcleo de emergencia es casi idéntico al núcleo estándar. Cuando se llevan a cabo modificaciones en el núcleo estándar, por ejemplo, para mejorar el rendimiento o añadir nuevas características, el núcleo de emergencia permanece intacto. De esta manera, si los cambios en el núcleo estándar ocasionan problemas de estabilidad, un usuario puede decirle simplemente a la RasPi que cargue el núcleo de emergencia en lugar del núcleo estándar.

Existen dos formas para arrancar con el núcleo de emergencia, y ambas requieren el uso de una PC y un lector de tarjetas SD si la RasPi por sí sola no pudiera arrancar. De otra forma, lo siguiente puede llevarse a cabo sobre la misma RasPi.

La forma más fácil de arrancar el núcleo de emergencia es renombrar el archivo existente `kernel.img` a `kernel.img.bak` y luego renombrar el archivo `kernel_emergency.img` a `kernel.img`. Cuando RasPi comience a cargar, ahora lo hará con el núcleo de emergencia por defecto. Para volver al núcleo estándar, simplemente revierta el proceso: renombre `kernel.img` a `kernel_emergency.img` y `kernel.img.bak` a `kernel.img`.

Un método alternativo para cargar el núcleo de emergencia es editando el archivo `cmdline.txt` (ubicado en el directorio `/boot`) y agregar la siguiente entrada al final de la línea de comandos existente:

```
kernel=kernel_emergency.img
```

Esto le indica a la RasPi que debe cargar el núcleo llamado `kernel_emergency.img` en lugar del habitual `kernel.img`. Revertir el proceso es tan simple como abrir de nuevo el archivo `cmdline.txt` y eliminar la entrada.

Aprenderá más sobre el archivo `cmdline.txt` y cómo éste afecta el funcionamiento de la Raspberry Pi en el Capítulo 7, “Configuración Avanzada de la Raspberry Pi”.

# Capítulo 5

## Configuración de la Red

**Para la mayoría de los usuarios**, configurar la red de la RasPi es tan fácil como conectar un cable al puerto Ethernet del Modelo B (o un adaptador USB Ethernet en el caso del Modelo A). Para otros usuarios, sin embargo, la red requerirá de configuración manual.

Si sabe que su red no cuenta con un servidor *DHCP* (*Dynamic Host Configuration Protocol*), un sistema que le dice a la RasPi y a otros dispositivos de la red cómo deben conectarse) o si desea utilizar un adaptador USB inalámbrico con la RasPi continúe leyendo.

## Red Cableada

En algunos casos, para que la red de la RasPi funcione correctamente es posible que tenga que configurarla manualmente. Normalmente, la red de un hogar, escuela u oficina cuenta con un servidor DHCP que le dice a la RasPi y a otros dispositivos sobre la red cómo deben conectarse. Algunas redes, sin embargo, no tienen un servidor DHCP y necesitan ser configuradas manualmente.

La lista de las interfaces de red junto con la información sobre cómo éstas deben ser configuradas, se encuentran almacenadas en un archivo llamado **interfaces** ubicado en la carpeta **/etc/network**. Es un archivo que sólo el usuario root puede editar, debido a que eliminar una interfaz de red de la lista provocaría que aquella deje de funcionar.

Desde la terminal, puede editar este archivo usando cualquier editor de texto. Por simplicidad, utilizaremos el editor **nano** para este proceso. Abra el archivo para su edición con el siguiente comando:

```
sudo nano /etc/network/interfaces
```

**Nano** es un editor de textos ligero y poderoso, con una interfaz de usuario sencilla (ver Figura 5-1). Puede mover el cursor a través del documento con las teclas de flecha, para guardar mantenga presionada la tecla CTRL y presione la tecla O, y para salir mantenga presionada la tecla CTRL y presione X.

La línea que necesita editar para la configuración manual comienza con **iface eth0 inet**. Borre **dhcp** del final de esta línea y reemplácela con **static**, presione Intro para comenzar una nueva línea, y a continuación, complete los detalles restantes en el siguiente formato colocando una tabulación al principio de cada línea:

```
[Tab] address xxx.xxx.xxx.xxx  
[Tab] netmask xxx.xxx.xxx.xxx  
[Tab] gateway xxx.xxx.xxx.xxx
```

Figure 5-1  
Editando /etc/network/interfaces con nano



```
File Edit Tabs Help
GNU nano 2.2.4 File: /etc/network/interfaces Modified

# Used by ifup(8) and ifdown(8). See the interfaces(5) manpage or
# /usr/share/doc/ifupdown/examples for more information.

auto lo wlan0

iface lo inet loopback
iface eth0 inet dhcp
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos  
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

Asegúrese de presionar la tecla Tab al principio de cada línea, y no escribir realmente *[Tab]*. Los caracteres *x* en las líneas de configuración representan las direcciones de red que necesitará introducir. Para **address**, debe introducir la dirección IP estática que desea asignar a la RasPi. Para **netmask**, debe introducir la máscara de red (la cual controla el tamaño de la red conectada) en el formato conocido como *notación decimal con puntos*. Si está utilizando una red doméstica, esta dirección suele ser 255.255.255.0. Para **gateway**, debe introducir la dirección IP de su router o cable módem.

A modo de ejemplo, la configuración para una red doméstica común se vería de la siguiente forma:

```
iface eth0 inet static
[Tab] address 192.168.0.10
[Tab] netmask 255.255.255.0
[Tab] gateway 192.168.0.254
```

Cuando haya terminado de editar el archivo, presione CTRL + O para guardarlo, y en seguida presione CTRL + X para salir de nano y volver a la terminal. Para utilizar su nueva configuración de red, reinicie el servicio de red escribiendo lo siguiente:

```
sudo /etc/init.d/networking restart
```



Si necesita regresar a la configuración automática vía DHCP, tiene que volver a editar el archivo `interfaces` y borrar las configuraciones `address`, `netmask` y `gateway`. Reemplazar `static` con `dhcp` al final de la línea `iface` y luego reiniciar el servicio de red de nueva cuenta.

Establecer una dirección IP de forma manual no es suficiente para conseguir que su RasPi se conecte al mundo exterior. Las computadoras sobre las redes modernas cuentan tanto un identificador numérico conocido como dirección IP como con un nombre de host o nombre de dominio. Es este último, un nombre amigable, lo que significa que simplemente puede escribir `www.raspberrypi.org` en su navegador, en vez de intentar recordar `93.93.128.176`.

Un sistema denominado servidor *DNS* (*Domain Name Service*) es el responsable de buscar los nombres amigables que usted le proporciona y convertirlos en los números necesarios para acceder al sistema. Éste funciona parecido a un directorio telefónico. Antes de que usted sea capaz de acceder a los sistemas conectados a la Internet mediante sus nombres de dominio, necesitará indicarle a la RasPi qué servidores DNS debe utilizar.

La lista de los servidores DNS, más conocida como *nameservers* en la jerga de Linux, se encuentra almacenada en `/etc/resolv.conf`. Cuando el sistema obtiene sus detalles de red a través de DHCP, este archivo se llena automáticamente. Cuando usted establece la dirección manualmente, necesitará proporcionar las direcciones de los *nameservers* (servidores de nombres) de su red. Normalmente, esta dirección debería ser la dirección de su router como la encontró en la línea `gateway` del archivo `interfaces` (descrito anteriormente en este capítulo).

Para establecer los servidores de nombres (*nameservers*), abra el archivo con `nano` escribiendo el siguiente comando en la terminal:

```
sudo nano /etc/resolv.conf
```

Agregue cada servidor de nombres en una línea por separado, precedida con `nameserver` y un espacio. A modo de ejemplo, la configuración `resolv.conf` para una red que utiliza los servidores de nombres de acceso público de Google para resolver nombres de dominio se vería semejante a ésta:

```
nameserver 8.8.8.8
nameserver 8.8.4.4
```

Se dará cuenta que las direcciones de los servidores de nombres necesitan ser proporcionadas como direcciones IP, en vez de nombres de dominio. Si los proporcionó como nombres de dominio, la RasPi entrará en un bucle infinito intentando encontrar un servidor de nombres para preguntarle a éste cómo puede encontrar los servidores de nombres.

Guarde el archivo presionando CTRL + O y a continuación salga de `nano` presionando CTRL + X. Reinicie la interfaz de red escribiendo lo siguiente:

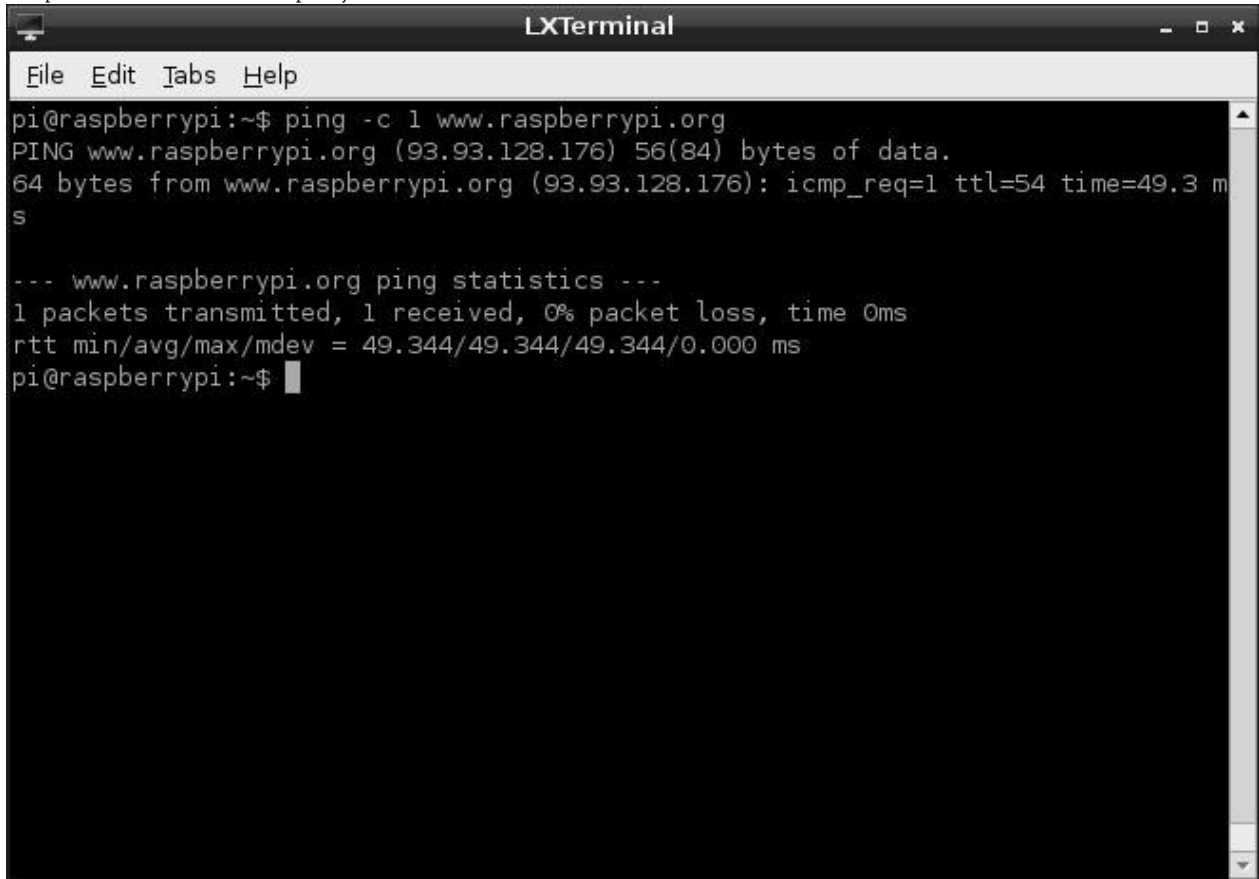
```
sudo /etc/init.d/networking restart
```

Luego, compruebe la configuración, ya sea abriendo un navegador web o utilizando el siguiente comando `ping` (ver Figura 5-2):

```
ping -c 1 www.raspberrypi.org
```

Figura 5-2

Una prueba exitosa de red en la Raspberry Pi Modelo B

The image shows a screenshot of an LXTerminal window. The title bar reads "LXTerminal". The menu bar includes "File", "Edit", "Tabs", and "Help". The terminal content shows a user at the prompt "pi@raspberrypi:~\$ " typing the command "ping -c 1 www.raspberrypi.org". The output shows a successful ping to www.raspberrypi.org (93.93.128.176) with 56(84) bytes of data, a response time of 49.3 ms, and 0% packet loss. The statistics show 1 packet transmitted and received. The prompt "pi@raspberrypi:~\$ " is visible at the bottom of the terminal window.

```
pi@raspberrypi:~$ ping -c 1 www.raspberrypi.org
PING www.raspberrypi.org (93.93.128.176) 56(84) bytes of data:
64 bytes from www.raspberrypi.org (93.93.128.176): icmp_req=1 ttl=54 time=49.3 ms

--- www.raspberrypi.org ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 49.344/49.344/49.344/0.000 ms
pi@raspberrypi:~$
```

## Red Inalámbrica

Aunque ninguno de los modelos actuales de la Raspberry Pi incluye hardware integrado para redes Wi-Fi, es posible agregar conectividad inalámbrica a través de un sencillo adaptador Wi-Fi USB. Sin embargo, tendrá que configurar el adaptador antes de que pueda utilizarlo para poner su Raspberry Pi en línea.

### CONSEJO

Los adaptadores Wi-Fi USB son demasiado hambrientos de energía. Si conecta uno directamente al puerto USB de la RasPi, lo más probable es que simplemente no funcione. En vez de hacer eso, conecte un hub USB a la RasPi y a continuación inserte el adaptador Wi-Fi en él.

Antes de empezar a configurar la interfaz inalámbrica, necesitará conocer el *SSID* (también conocido como *el nombre de la red*) del router inalámbrico al cual desea conectarse, el tipo de cifrado que utiliza y la contraseña que requiere. Además necesita saber de qué tipo de red inalámbrica se trata. Un adaptador USB diseñado para una Wi-Fi 802.11a no puede conectarse a una red 802.11g y viceversa.

## Instalación del Firmware

Para que el adaptador USB inalámbrico sea reconocido por el sistema se necesita de un paquete de software conocido como *firmware*. Mientras que algunas distribuciones tienen una selección de los firmwares Wi-Fi más comunes ya instalados por defecto, otras distribuciones no los tienen. Raspbian incluye los paquetes de firmware inalámbricos más comunes, si tiene instalado Raspbian como su sistema operativo puede avanzar más adelante a la sección “Utilizando wpa\_gui para Conectarse a una Red Inalámbrica”; de lo contrario, continúe leyendo las instrucciones en esta sección.

Para poder descargar los archivos del firmware, la RasPi debe estar conectada a la Internet. Si puede prescindir de un puerto cableado de su router o gateway por unos minutos, esto no es un problema. Sin embargo, si la conexión inalámbrica es la única forma de la que dispone para estar en línea, necesitará descargar manualmente el paquete de instalación del firmware desde otro equipo y luego transferirlo a la RasPi, ya sea copiando el firmware a la tarjeta SD de la RasPi o conectando un dispositivo de almacenamiento externo, como una memoria USB.

Para encontrar cuál es el archivo correcto del firmware que debe de descargar, necesitará saber qué tipo de adaptador inalámbrico tiene. Aunque son muchas las compañías que venden adaptadores USB inalámbricos de marca, el número de compañías que realmente fabrican los componentes son en realidad muy pocas. Diferentes fabricantes pueden utilizar el mismo tipo de chip dentro de sus adaptadores USB inalámbricos, haciendo que todos aquellos adaptadores sean compatibles con el mismo firmware. Por consiguiente, la etiqueta sobre el dispositivo o su embalaje no son suficientes para saber cuál firmware se debe de instalar. En lugar de eso, necesitará conectar el dispositivo a la RasPi y examinar el *kernel ring buffer* en busca de mensajes de error. Si ya ha conectado el adaptador inalámbrico como se indica en el Capítulo 2, “Comenzando con la Raspberry Pi”, puede continuar. En caso contrario, conecte el adaptador ahora.

El *kernel ring buffer* es una porción especial de la memoria utilizada por el núcleo Linux para almacenar sus salidas de información en un formato humanamente comprensible. Es una parte importante del sistema operativo Linux: mientras la RasPi arranca, el texto aparece tan rápidamente que es casi imposible leerlo, por lo que es importante que los usuarios sean capaces de ver los mensajes más adelante en otro momento para leer los errores y diagnosticar los problemas.

Con el adaptador conectado pero sin los paquetes del firmware instalados, el núcleo desplegará una serie de mensajes de error al ring buffer. Para leer estos mensajes, puede utilizar el comando `dmesg` para imprimir los contenidos del búfer en la pantalla. En la terminal o en la consola (si no ha cargado el entorno de escritorio), simplemente escriba el siguiente comando para visualizar el búfer:

```
dmesg
```

Esto imprimirá todo el kernel ring buffer, el cual contendrá todos los mensajes de salida que ha generado el núcleo desde que la RasPi fue encendida. Si la RasPi ha estado en funcionamiento desde hace mucho tiempo, puede ser mucho el texto que encuentre. Para localizar el mensaje de error referente al adaptador inalámbrico, puede ser de gran ayuda enviar la salida de `dmesg` a través de una herramienta llamada `grep`. Al utilizar `grep` puede buscar a través del búfer el texto relacionado con el firmware que falta. Pasando la salida de `dmesg` por una *tubería* hacia `grep` indicando un término de búsqueda, las cosas se vuelven mucho más claras. Escriba lo siguiente en la terminal:

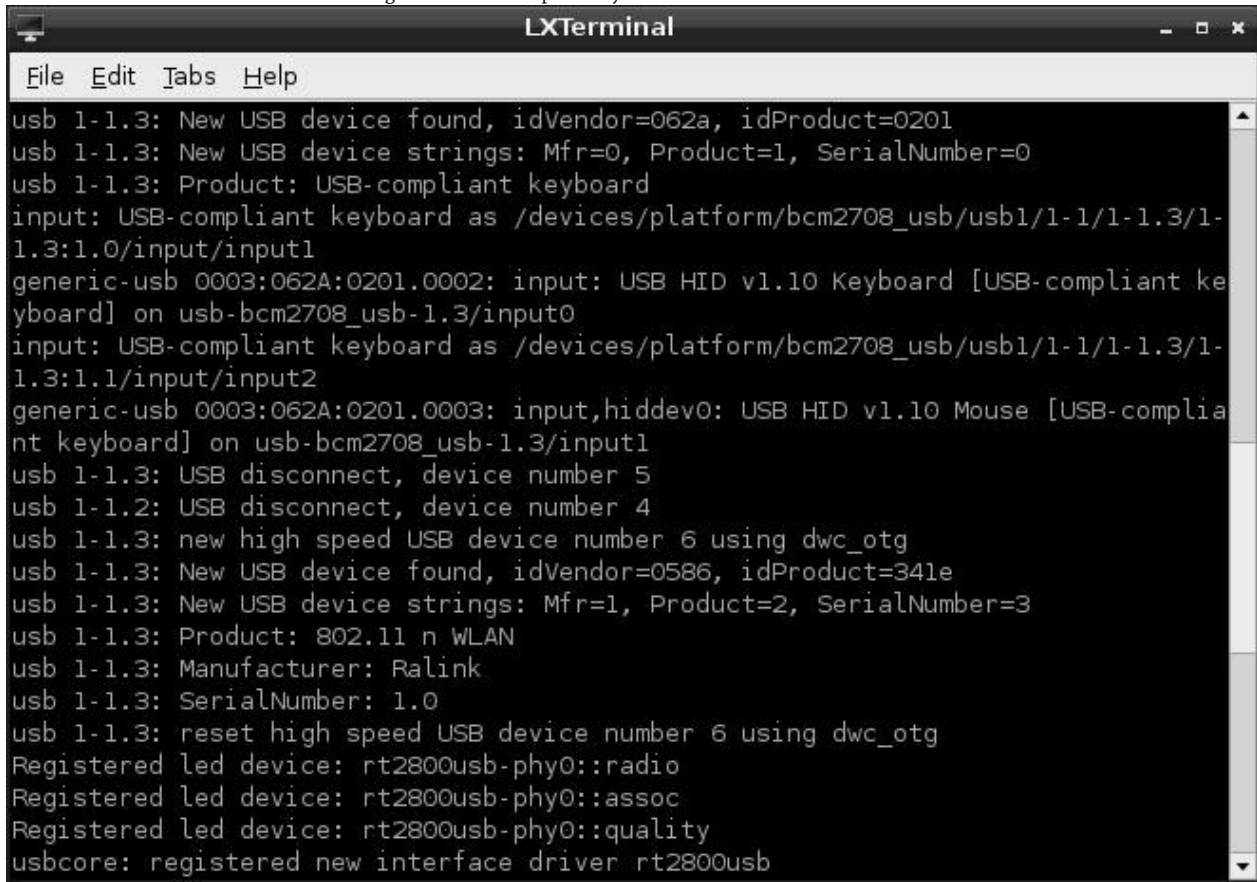
```
dmesg -t | grep ^usb
```

El símbolo `|` es conocido como *tubería* y le dice a Linux que envíe la salida de un programa (que iría normalmente hacia un archivo o a la pantalla) a la entrada de otro programa. Múltiples programas pueden encadenarse de esta forma. En este ejemplo, `grep` está diciendo que buscará a través de la salida de `dmesg` (las pantallas llenas de texto del comando anterior) por cualquier empleo del término `usb` al comienzo de la línea (indicado por el carácter `^`).

La salida exacta que produzca aquella búsqueda dependerá del fabricante de su adaptador USB inalámbrico. En la Figura 5-3, la salida que se muestra pertenece a la de un Adaptador USB inalámbrico Zyxel NWD2015 enchufado a la RasPi.

Figura 5-3

Buscando el término usb dentro del kernel ring buffer con un adaptador Zyxel inalámbrico conectado.



```
usb 1-1.3: New USB device found, idVendor=062a, idProduct=0201
usb 1-1.3: New USB device strings: Mfr=0, Product=1, SerialNumber=0
usb 1-1.3: Product: USB-compliant keyboard
input: USB-compliant keyboard as /devices/platform/bcm2708_usb/usb1/1-1/1-1.3/1-1.3:1.0/input/input1
generic-usb 0003:062A:0201.0002: input: USB HID v1.10 Keyboard [USB-compliant keyboard] on usb-bcm2708_usb-1.3/input0
input: USB-compliant keyboard as /devices/platform/bcm2708_usb/usb1/1-1/1-1.3/1-1.3:1.1/input/input2
generic-usb 0003:062A:0201.0003: input,hiddev0: USB HID v1.10 Mouse [USB-compliant keyboard] on usb-bcm2708_usb-1.3/input1
usb 1-1.3: USB disconnect, device number 5
usb 1-1.2: USB disconnect, device number 4
usb 1-1.3: new high speed USB device number 6 using dwc_otg
usb 1-1.3: New USB device found, idVendor=0586, idProduct=341e
usb 1-1.3: New USB device strings: Mfr=1, Product=2, SerialNumber=3
usb 1-1.3: Product: 802.11 n WLAN
usb 1-1.3: Manufacturer: Ralink
usb 1-1.3: SerialNumber: 1.0
usb 1-1.3: reset high speed USB device number 6 using dwc_otg
Registered led device: rt2800usb-phy0::radio
Registered led device: rt2800usb-phy0::assoc
Registered led device: rt2800usb-phy0::quality
usbcore: registered new interface driver rt2800usb
```

La parte importante de esta salida es la línea que dice **Manufacturer**. En el caso del ejemplo del Zyxel NWD2105, se lee **Ralink**, la cual es la empresa verdadera que fabrica el chip que se encuentra dentro del adaptador USB inalámbrico Zyxel. Éste es el firmware de la empresa que debe instalarse para que el adaptador inalámbrico funcione.

---

**CONSEJO**

Si no puede encontrar algo utilizando `usb` como término de búsqueda, puede probar el mismo comando empleando términos como `firmware`, `wlan` o `wireless`. Si todavía no puede ver algo útil, escriba `lsusb` para obtener una lista de todos los dispositivos USB conectados al sistema.

---

Utilizando el nombre del fabricante obtenido mediante `dmesg`, busque los archivos del firmware usando la herramienta de búsqueda `apt-cache` enseñada anteriormente. Para el ejemplo del adaptador Zyxel NWD2015, el comando `apt-cache` debería ser:

```
apt-cache search ralink
```

Si `apt-cache` no logra encontrar el firmware, puede que necesite llegar a una conclusión basada en los paquetes de firmware de la siguiente lista. No se preocupe si instala el equivocado, cualquier firmware puede desinstalarse rápidamente utilizando `apt-get remove`, además, tener varios paquetes de firmware no hace daño. Los siguientes paquetes de firmware se encuentran disponibles en la distribución Raspbian recomendada de la Raspberry Pi:

- **atmel-firmware**-Para los dispositivos basados en el chipset Atmel AT76C50X
- **firmware-atheros**-Para los dispositivos basados en chipsets Atheros
- **firmware-brcm80211**-Para los dispositivos basados en chipsets Broadcom
- **firmware-intelwimax**-Para los dispositivos basados en chipsets WiMAX de Intel
- **firmware-ipw2x00**-Para los adaptadores Intel ProWireless (incluyendo el 2100, 2200 y 2915)
- **firmware-iwlwifi**-Para otros adaptadores inalámbricos Intel (incluyendo el 3945, 4965 y los series 5000)
- **firmware-ralink**-Para los dispositivos basados en chipsets Ralink
- **firmware-realtek**-Para los dispositivos basados en chipsets Realtek
- **zd1211-firmware**-Para los dispositivos basados en el chipset ZyDAS 1211

EL firmware para el adaptador inalámbrico Zyxel de ejemplo es proporcionado por el paquete **firmware-ralink** de esta lista. Este paquete puede instalarse utilizando **apt-get**, pero únicamente mientras la RasPi se encuentre conectada a la Internet a través de su puerto Ethernet alámbrico o mediante un adaptador USB Ethernet. Una vez conectado, instale el firmware escribiendo lo siguiente:

```
sudo apt-get install paquetefirmware
```

Reemplace *paquetefirmware* en este comando con el nombre del paquete que encontró al utilizar **apt-cache**. Para el ejemplo Zyxel NWD2105, el comando completo sería **sudo apt-get install firmware-ralink**.

## Instalación Offline del Firmware Inalámbrico

Si no puede conectar la RasPi a Internet utilizando cualquier otro método que no sea empleando una conexión inalámbrica, necesitará descargar el firmware en una computadora diferente. En un navegador web, cargue un motor de búsqueda y escriba el nombre del paquete del firmware seguido del nombre de la distribución que está utilizando y su versión.

Si está usando la distribución Raspbian recomendada, el firmware para el chipset Ralink RT2x00 del ejemplo puede ser encontrado buscando el término **firmware-ralink debian wheezy**. La búsqueda lo conducirá hacia un *archivo de paquete* para descargar. En el caso de Debian y Raspbian, es un archivo **.deb**. Para Pidora, el mismo firmware se proporciona como un archivo **.rpm**.

Descargue este archivo y luego cópielo a la tarjeta SD de la RasPi en el directorio **/home/pi**, o en una memoria USB u otro dispositivo de almacenamiento externo. Cargue la RasPi, y en seguida, cuando llegue el momento de instalar el firmware, reemplace el nombre del paquete con el nombre del archivo que ha descargado. Para la tarjeta del ejemplo Zyxel NWD2105, el comando sería el siguiente:

```
sudo apt-get install firmware-ralink_0.35_all.deb
```

Con el firmware ya instalado, desconecte el adaptador USB inalámbrico y vuélvalo a conectar a la RasPi. Esto reiniciará la búsqueda del núcleo a fin de encontrar los archivos del firmware, que ahora sí será capaz de encontrar. Esos archivos permanecerán en su lugar y se cargan automáticamente cuando el adaptador USB inalámbrico es conectado. Sólo tendrá que llevar a cabo el proceso de instalación una sola vez.

## Utilizando **wpa\_gui** para Conectarse a una Red Inalámbrica

La forma más sencilla de conectarse a una red inalámbrica desde la Raspberry Pi es utilizando la herramienta **wpa\_gui**. Esta herramienta cuenta con una interfaz gráfica de usuario y puede ejecutarla desde el escritorio haciendo clic en el icono llamado Wifi Config.

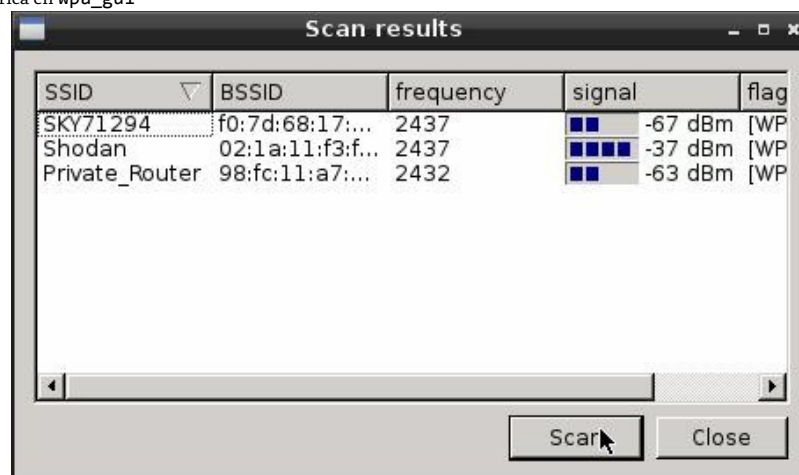
La ventana principal `wpa_gui` tiene dos listas desplegables, etiquetadas *Adapter (Adaptador)* y *Network (Red)* ver Figura 5-4). La primera debe estar completada ya con el identificador de su adaptador usb de red inalámbrica, `wlan0`. Si tiene más de un adaptador conectado, puede hacer clic en la flecha para elegir cuál desea configurar; si nada aparece en la lista, vuelva atrás unas cuantas páginas y asegúrese que el firmware de su adaptador esté correctamente instalado.

**Figura 5-4**  
La aplicación `wpa_gui`



La búsqueda de una red inalámbrica es un proceso conocido como *escaneo*, el cual puede ser activado en la herramienta `wpa_gui` haciendo clic en el botón `Scan` en la parte inferior derecha de la ventana. Esto mostrará una segunda ventana que desplegará los resultados del escaneo (ver Figura 5-5). Mire en la lista para localizar el nombre de su red inalámbrica (la cual, como es de esperarse, será la que muestra la señal más fuerte en la columna `signal` en la lista ) y haga doble clic en su entrada.

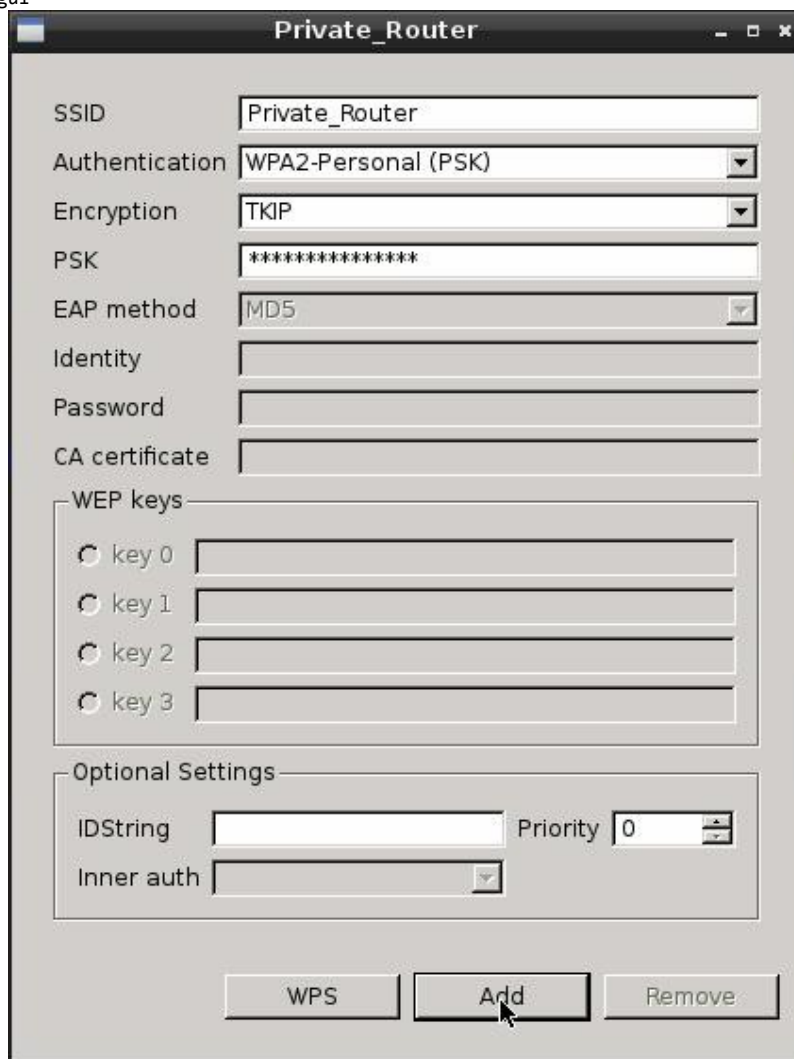
**Figura 5-5**  
Elegiendo una red inalámbrica en `wpa_gui`



La ventana que aparece cuando hace doble clic sobre una red le pedirá introducir diferentes configuraciones, dichas configuración dependerán del tipo de red a la cual trata de conectarse (ver Figura 5-6). Para una red sin cifrado (lo que no es una buena idea dado que cualquier persona dentro del rango podría utilizarla) `wpa_gui` no le pedirá ningún otro configuración adicional; simplemente haga clic en el botón Add en la parte inferior.

Si su red utiliza cifrado, necesitará elegir el tipo de cifrado, utilizando las listas desplegables Authentication (Autenticación) y Encryption (Cifrado). En muchos de los casos, `wpa_gui` habrá seleccionado estos valores automáticamente. Si es así, sólo escriba la contraseña en la casilla etiquetada con PSK y haga clic en Add. Si el tipo de cifrado de la red es *WEP (Wired Equivalent Privacy)* antes que *WAP (Wireless Protected Access)*, el cifrado más seguro), necesitará escribir al menos una clave en la sección WEP Keys. Si utiliza una red empresarial autenticada, puede agregar los detalles de su identidad, contraseña y certificado de cifrado, pero la mayoría de los usuarios caseros no necesitarán utilizar estos campos.

Figura 5-6  
Agregando una red a `wpa_gui`



Si su red incluye una opción *WPS (Wi-Fi Protected Setup)*, puede hacer clic en el botón WPS en la parte inferior izquierda de la ventana para realizar una instalación con un sólo clic. Basta con presionar el botón WPS sobre su router o punto de acceso y luego hacer clic en el botón WPS en `wps_gui` para conectarse automáticamente.



Cuando haya llenado los detalles de su red, haga clic en Add. Esto cerrará la ventana y lo regresará a la ventana principal de `wpa_gui` (pero ahora la lista desplegable Network deberá tener una entrada con el nombre de su punto de acceso inalámbrico). Llegados a este momento, `wpa_gui` automáticamente lo conectará a la red inalámbrica y una dirección IP deberá de aparecer en la parte inferior de la pantalla (ver Figura 5-7).

Figura 5-7

Conectado a una red inalámbrica a través de `wpa_gui`



Si necesita conectarse a una red diferente en el futuro, sólo tiene que iniciar el proceso de nuevo desde el principio. La conexión a una nueva red no ocasiona que `wpa_gui` olvide la anterior red; ambas aparecerán en la lista desplegable y podrá conectarse a cualquier red que se encuentre dentro del rango simplemente seleccionándola desde la lista, una vez ya configurada.

También puede utilizar `wpa_gui` para desconectarse de la red inalámbrica sin tener que retirar físicamente el adaptador inalámbrico. Sólo haga clic en el botón Disconnect en la parte inferior de la ventana y después de unos segundos su RasPi será desconectada de la red. Para re-conectarse, haga clic en el botón Connect.

## Utilizando la Terminal para Conectarse a una Red Inalámbrica

Si su Raspberry Pi está trabajando sin una interfaz gráfica de usuario, también puede conectarla a las redes inalámbricas desde la consola o terminal. Primero, compruebe que el adaptador USB inalámbrico esté funcionando como debería, utilizando el comando `iwlist` para escanear los puntos de acceso inalámbrico más cercanos. Probablemente la lista que obtenga sea muy larga como para ser mostrada en una sola pantalla, por lo tanto necesitará pasar la salida del comando por una tubería hacia el comando `less` para realizar una pausa cada vez que la pantalla se llene, de la siguiente forma:

```
sudo iwlist scan | less
```

Este comando devolverá una lista de todas las redes que se encuentran al alcance de la RasPi junto con sus detalles (ver Figura 5-8). Si recibe un mensaje de error en este punto (en particular, uno que afirma que la red o la interfaz

está caída) verifique que ha instalado el firmware correcto y que el adaptador USB inalámbrico está conectado al hub USB con alimentador propio.

Figura 5-8  
Escaneando redes inalámbricas con `iwlist`

```

LXTerminal
File Edit Tabs Help
wlan0 Scan completed :
      Cell 01 - Address: F0:7D:68:17:2D:5F
              Channel:6
              Frequency:2.437 GHz (Channel 6)
              Quality=41/70 Signal level=-69 dBm
              Encryption key:on
              ESSID:"SKY71294"
              Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s
                        24 Mb/s; 36 Mb/s; 54 Mb/s
              Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 48 Mb/s
              Mode:Master
              Extra:tsf=000000e2c9453f6a
              Extra: Last beacon: 960ms ago
              IE: Unknown: 0008534B593731323934
              IE: Unknown: 010882848B962430486C
              IE: Unknown: 030106
              IE: Unknown: 2A0104
              IE: Unknown: 2F0104
              IE: IEEE 802.11i/WPA2 Version 1
                  Group Cipher : TKIP
                  Pairwise Ciphers (2) : CCMP TKIP
                  Authentication Suites (1) : PSK
              IE: Unknown: 32040C121860
  
```

Puede comprobar el estado actual de la red con el comando `iwconfig`. Al igual que `ifconfig`, el comando `iwconfig` permite examinar el estado de una interfaz de red y emitir comandos de configuración. Sin embargo, a diferencia de `ifconfig`, `iwconfig` está especialmente diseñado para las redes inalámbricas e incluye características específicas para éstas. Escriba a continuación el nombre del comando en la terminal:

`iwconfig`

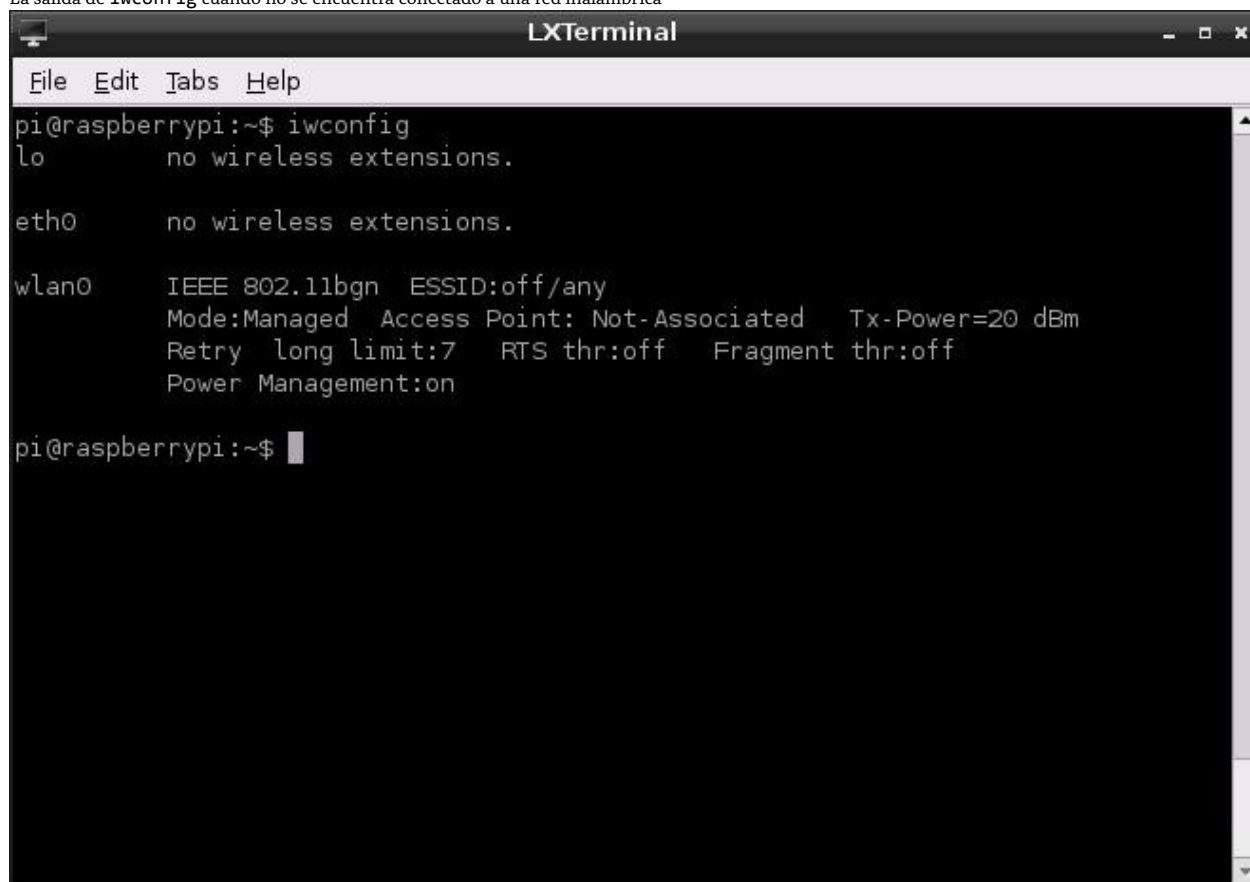
La salida de `iwconfig` se muestra en la Figura 5-9 y está dividida en las siguientes secciones:

- **Interface Name**-Cada dispositivo tiene su propio nombre de interfaz, como con las redes cableadas. Si la interfaz es una conexión inalámbrica, se mostrarán detalles adicionales. El nombre predeterminado para una conexión inalámbrica de la RasPi es `wlan0`.
- **Standard**-Los estándares inalámbricos IEEE 802.11 cuentan con una variedad de diferentes tipos, distinguidos por una letra sufijo. Esta sección enumera los estándares soportados por el adaptador USB inalámbrico. Para el adaptador del ejemplo, se leerá `IEEE 802.11bgn` para los tipos de redes con los cuales puede tratar.
- **ESSID**-El SSID de la red en el cual el adaptador está conectado. Si el adaptador actualmente no se encuentra conectado a una red, aquí se leerá `off/any`.

- **Mode**-El modo en el que actualmente se encuentra funcionando el adaptador, el cual puede ser uno de los siguientes:
  - **Managed**-Una red inalámbrica estándar, con clientes conectándose a los puntos de acceso. Este es el modo empleado por la mayoría de las redes domésticas y comerciales.
  - **Ad-Hoc**-Una red inalámbrica de dispositivo a dispositivo, sin puntos de acceso.
  - **Monitor**-Un modo especial en el que la tarjeta permanece escuchando fuera todo el tráfico sea o no éste el destinatario. Este modo se utiliza comúnmente para la resolución problemas de la red al capturar el tráfico de la red inalámbrica.
  - **Repeater**-Un modo especial que obliga a una tarjeta inalámbrica a reenviar el tráfico a otros clientes de la red para incrementar la fuerza de la señal.
  - **Secondary**-Un subconjunto del modo Repeater, el cual obliga a la tarjeta inalámbrica a actuar como un repetidor de reserva.
- **Access Point**-Las direcciones de los puntos de acceso en los que el adaptador se encuentra actualmente conectado. Si el adaptador no está conectado a un punto de acceso inalámbrico, aquí se leerá **Not-Associated**.
- **Tx-Power**-La potencia de transmisión del adaptador inalámbrico. El número mostrado aquí indica la intensidad de la señal que el adaptador está enviando: cuanto mayor sea el número, más fuerte será la señal.
- **Retry**-La configuración actual para el reintento de retransmisión del adaptador inalámbrico, empleado sobre las redes congestionadas. Normalmente no necesita ser modificada y algunas tarjetas no permiten que se altere.
- **RTS**-La configuración actual del adaptador para el “apretón de manos”, *Ready To Send and Clear To Send (RTS/CTS)*, empleado en las redes ocupadas para evitar colisiones. Normalmente es establecida por el punto de acceso sobre la conexión.
- **Fragment**-El tamaño máximo de fragmento, empleado en las redes ocupadas para dividir los paquetes en múltiples fragmentos. Normalmente es establecido por el punto de acceso en la conexión.
- **Power Management**-El estado actual de la funcionalidad de administración de energía del adaptador, la cual reduce la demanda de energía del dispositivo cuando la red inalámbrica se encuentra inactiva. Éste tiene poco efecto sobre la RasPi, pero comúnmente se encuentra habilitado en los dispositivos que son alimentados por baterías, como las laptops.

Figura 5-9

La salida de `iwconfig` cuando no se encuentra conectado a una red inalámbrica



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

eth0        no wireless extensions.

wlan0       IEEE 802.11bgn  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

pi@raspberrypi:~$
```

Para conectar la RasPi a una red inalámbrica, necesita añadir algunas líneas dentro del archivo `/etc/network/interfaces` (Para conocer todos los detalles sobre cómo se presenta este archivo, vea la sección “Red Cableada” al principio de este capítulo).

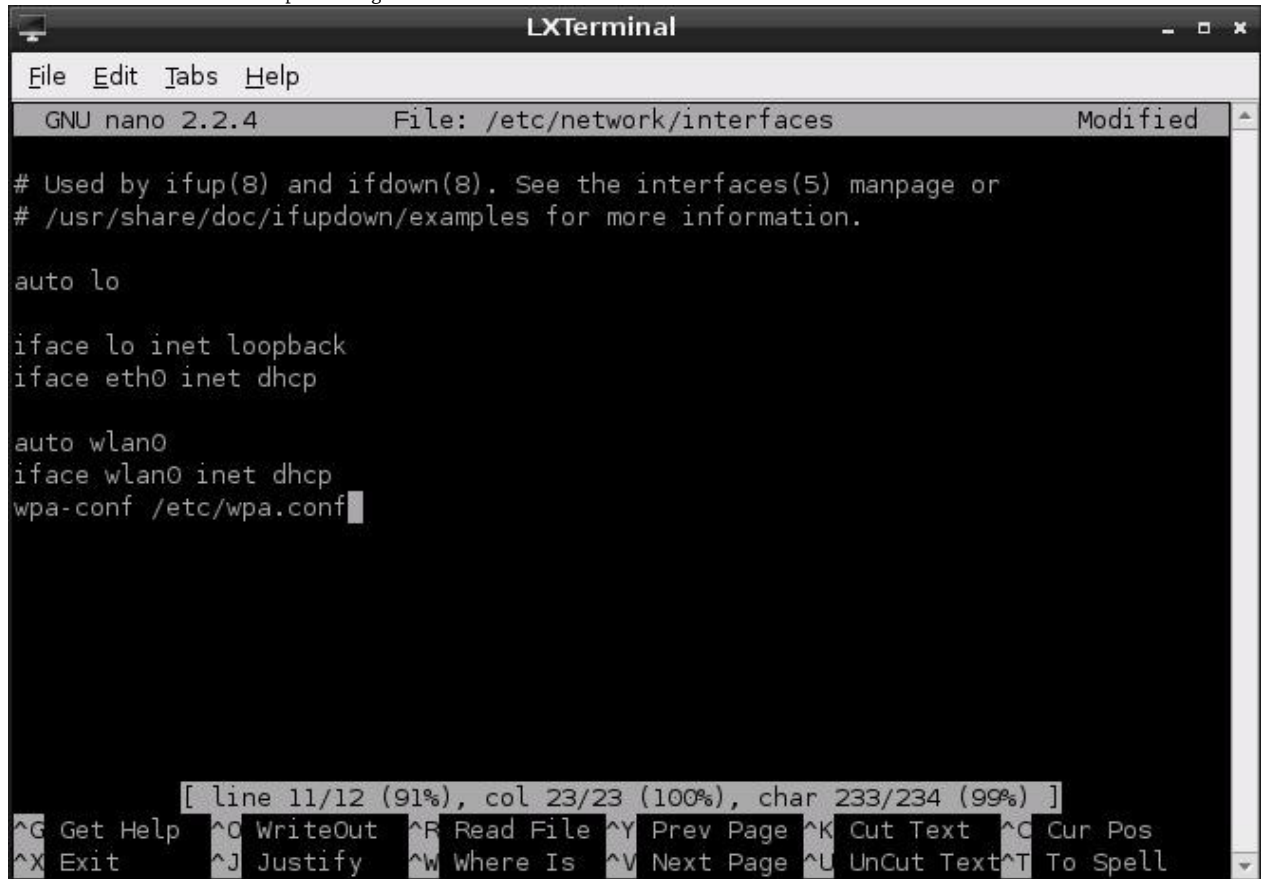
Primero, abra el archivo en el editor de textos `nano`:

```
sudo nano /etc/network/interfaces
```

Al final del archivo, cree una nueva entrada para el adaptador USB inalámbrico para que se vea de la siguiente manera (ver figura 5-10):

```
auto wlan0
iface wlan0 inet dhcp
wpa-conf /etc/wpa.conf
```

Figura 5-10

Editando el archivo `interfaces` para conseguir acceso a la red inalámbrica

```
LXTerminal
File Edit Tabs Help
GNU nano 2.2.4 File: /etc/network/interfaces Modified

# Used by ifup(8) and ifdown(8). See the interfaces(5) manpage or
# /usr/share/doc/ifupdown/examples for more information.

auto lo

iface lo inet loopback
iface eth0 inet dhcp

auto wlan0
iface wlan0 inet dhcp
wpa-conf /etc/wpa.conf

[ line 11/12 (91%), col 23/23 (100%), char 233/234 (99%) ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Una vez que la entrada está en su lugar, guarde el archivo presionando CTRL + O y luego salga de nano con CTRL + X.

**CONSEJO**

El ID del dispositivo de wlan0 es correcto si se trata del primer dispositivo inalámbrico que haya configurado en su RasPi. Si no lo es, el número del final será distinto. Escriba `iwconfig` para ver la lista actual de los dispositivos inalámbricos y modifique las líneas en el código del ejemplo anterior en la forma que corresponda.

La última línea del archivo `interfaces` hace referencia a un archivo de configuración, `wpa.conf`, que aún no existe. Este archivo es utilizado por una herramienta conocida como `wpa_supplicant`, diseñada para proporcionar a Linux una manera fácil de conectarse a las redes protegidas con el cifrado WPA (*Wireless Protected Access*).

Con la ayuda de `wpa_supplicant` puede conectar la RasPi a casi cualquier red inalámbrica (independientemente de si está protegida por WPA o por su reemplazo más reciente WPA2) tanto en el modo AES (*Advanced Encryption Standard*) como en el modo TKIP (*Temporal Key Integrity Protocol*). A pesar de su nombre, `wpa_supplicant` también le permite conectarse a las redes inalámbricas que utilizan el viejo estándar de cifrado WEP (*Wired Equivalent Privacy*).

El programa `wpa_supplicant` almacena su configuración en un archivo llamado `wpa.conf`, ubicado en el directorio `/etc`. Para comenzar a configurar el acceso inalámbrico en la RasPi, primero abra un nuevo archivo en blanco para su edición, escribiendo lo siguiente:

```
sudo nano /etc/wpa.conf
```

Introduzca las siguientes dos líneas, las cuales de nueva cuenta, son las mismas para cualquier tipo de red inalámbrica. Reemplace *Su\_SSID* con el SSID de la red inalámbrica a la cual desea conectarse y luego finalice el archivo con las líneas que correspondan al tipo de cifrado de su red.

```
Network={  
[Tab] ssid="Su_SSID"
```

Llegados a este punto del archivo de configuración, los detalles requeridos pueden variar dependiendo del tipo de red inalámbrica que está configurando. Las siguientes sub-secciones detallan las instrucciones para completar la configuración de las redes sin cifrado, WEP y WPA.

### Sin Cifrado

Si su red inalámbrica no tiene cifrado, finalice el archivo `wpa.conf` de la siguiente manera:

```
[Tab] key_mgmt=NONE  
}
```

Guarde el archivo con CTRL + O y luego salga de `nano` con CTRL + X.

### Cifrado WEP

Si su red inalámbrica utiliza el cifrado WEP, finalice el archivo `wpa.conf` como sigue:

```
[Tab] key_mgmt=NONE  
[Tab] wep_key0="Su_Clave_WEP"  
}
```

Reemplace *Su\_Clave\_WEP* con la clave ASCII para el cifrado WEP de su red inalámbrica. Guarde el archivo con CTRL + O y luego salga de `nano` con CTRL + X.

---

#### CONSEJO

El cifrado WEP es extremadamente inseguro. El software disponible hoy en día fácilmente puede romper el cifrado de una red protegido por WEP en pocos minutos, permitiendo que un tercero utilice su red. Si todavía utiliza WEP, considere cambiar a WPA o WPA2 para una mejor seguridad.

---

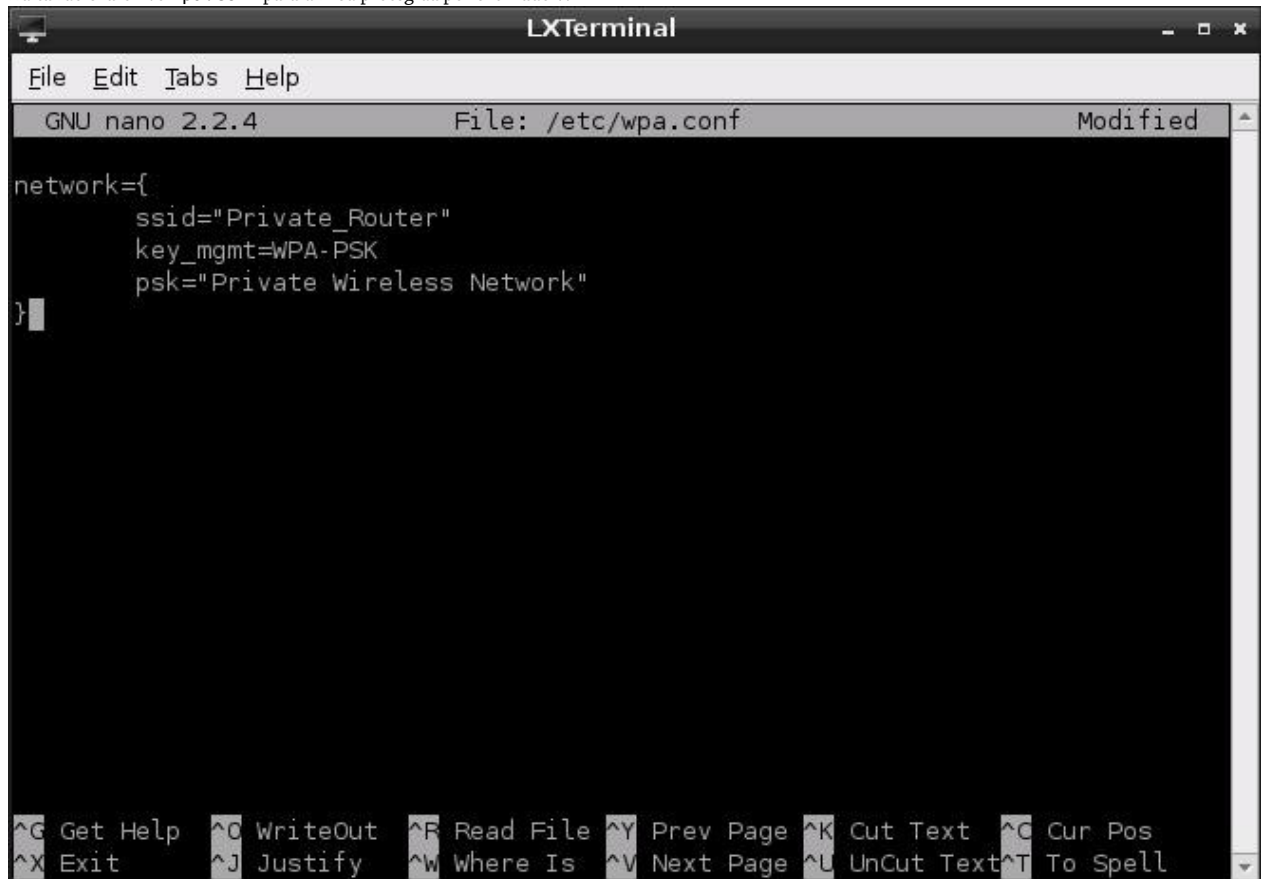
### Cifrado WPA/WPA2

Si su red inalámbrica utiliza el cifrado WPA y WPA2, finalice el archivo `wpa.conf` de la siguiente forma:

```
[Tab] key_mgmt=WPA-PSK  
[Tab] psk="Su_Clave_WPA"  
}
```

Reemplace *Su\_Clave\_WPA* con la frase contraseña para el cifrado de su red inalámbrica. La Figura 5-11 muestra un ejemplo de la configuración de una red inalámbrica con el SSID “*Private\_Router*” y la frase contraseña WPA “*Private Wireless Network*”. Guarde el archivo con CTRL + O y luego salga de `nano` con CTRL + X.

Figura 5-11

Editando el archivo `wpa.conf` para un red protegida por el cifrado WPA

```
GNU nano 2.2.4 File: /etc/wpa.conf Modified

network={
    ssid="Private_Router"
    key_mgmt=WPA-PSK
    psk="Private Wireless Network"
}
```

### Conexión a la Red Inalámbrica

La red inalámbrica de la RasPi se encuentra ahora configurada y entrará en funcionamiento la próxima vez que la RasPi sea reiniciada. Para iniciar la red inalámbrica sin tener que reiniciar, escriba lo siguiente:

```
sudo ifup wlan0
```

Para asegurarse de que la red se encuentra en funcionamiento, desconecte el cable Ethernet de la RasPi (si está conectado) y escriba lo siguiente:

```
ping -c 1 www.raspberrypi.org
```

---

#### CONSEJO

Si comienza a tener problemas con su RasPi tras la instalación de un adaptador USB inalámbrico, podría deberse a un conflicto con otros dispositivos USB. Algunos modelos de adaptadores son conocidos por causar problemas con algunos teclados USB. Para conocer una lista actualizada de los adaptadores que se conocen funcionan bien, así como aquellos que se sabe que provocan conflictos, visite <http://www.element14.com/community/docs/DOC-44703/1/raspberry-pi-wifi-adapter-testing> o la wiki eLinux en [http://elinux.org/RPi\\_USB\\_Wi-fi\\_Adapters](http://elinux.org/RPi_USB_Wi-fi_Adapters).

---

# Capítulo 6

## La Herramienta de Configuración del Software de la Raspberry Pi



**La Raspberry Pi** está diseñada para ser tan controlable como sea posible a través de la edición de archivos de configuración ubicados en el directorio `/boot` de la tarjeta SD. Para los principiantes, estos archivos pueden parecer muy complejos (aunque, con un poco de tiempo, pronto develará sus secretos), pero sin ellos algunas de las características más avanzadas de la RaSPi no estarían disponibles.

La Herramienta de Configuración del Software de la Raspberry Pi, **raspi-config**, está diseñada para resolver este problema. Ofrece acceso a la mayoría de las tareas comunes de configuración a través de una sencilla interfaz, basada en menús, la **raspi-config** hace que sea fácil para los recién llegados ajustar el rendimiento del sistema, modificar la división de la memoria, alterar la configuración de overscan o simplemente cambiar la distribución del teclado en la consola.

Al momento de escribir estas líneas, **raspi-config** es una herramienta exclusiva de la distribución Raspbian Linux. Se está trabajando para llevar la funcionalidad de la herramienta a otros sistemas operativos pero por ahora se recomienda que los principiantes se apeguen a la plataforma Raspbian que tiene un buen soporte a fin de sacar ventaja de las herramientas de configuración más útiles.

---

**ADVERTENCIA**

Aunque **raspi-config** está diseñada para ser una herramienta segura, algunos ajustes (en particular, la opción **overclock**) pueden provocar que la Raspberry Pi sea incapaz de arrancar. Asegúrese de leer cada sección de este capítulo cuidadosamente antes de utilizar **raspi-config** para llevar a cabo modificaciones en su Raspberry Pi.

---

## Ejecución de la Herramienta

La Herramienta de Configuración del Software de la Raspberry Pi edita una serie de archivos que son importantes para la ejecución de la Raspberry Pi. Por consiguiente, la herramienta sólo puede ejecutarse como el *usuario root*, con el comando **sudo**. Para cargar la herramienta, escriba el siguiente comando:

```
sudo raspi-config
```

Si recibe un mensaje de error indicando que el comando no se reconoce, significa que no tiene instalado **raspi-config**. Este suele pasar cuando utiliza una versión obsoleta de Raspbian. Para actualizar su RasPi, puede descargar la última versión desde el sitio web oficial de la Raspberry Pi o ejecutar el siguiente comando:

```
sudo apt-get update && sudo apt-get upgrade
```

El menú basado en texto de **raspi-config** se navega utilizando las teclas de cursor: las flechas Arriba y Abajo mueven la barra de selección roja a través de las opciones disponibles, mientras que las teclas Izquierda y Derecha mueven la barra entre la lista de opción y los botones Select and Finish en la parte de abajo.

La tecla Intro es utilizada para activar una opción cuando está iluminada por la barra roja. La acción por defecto es siempre Select (Seleccionar), de tal manera que si ha seleccionado una opción en el menú, no hay necesidad de presionar la flecha Derecha para elegir Select antes de presionar la tecla Intro.

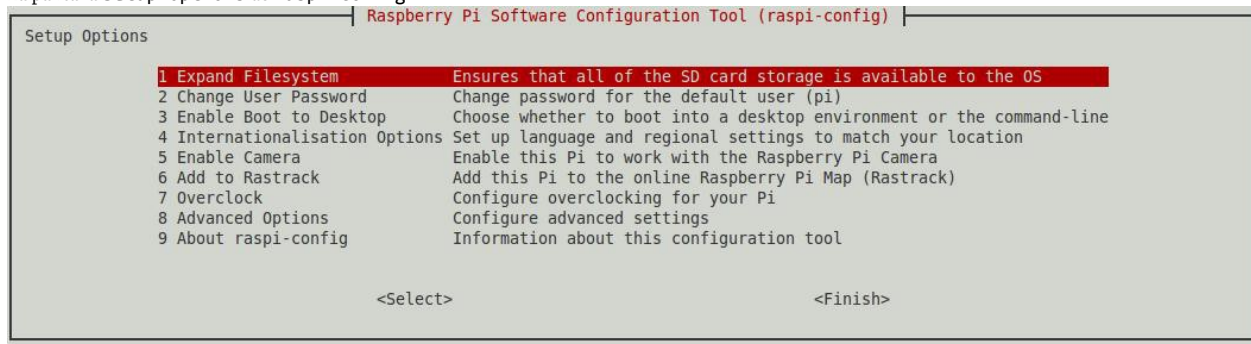
Algunas de las opciones en **raspi-config** tardan un poco en ejecutarse, particularmente las opciones Update y Expand Filesystem. Durante este tiempo, es importante asegurarse de no desenchufar la RasPi; hacerlo puede dejarlo con sistema de archivos dañado, lo que significa que tendrá que reinstalar su sistema operativo desde cero.

## La Pantalla de Opciones del Sistema

El menú inicial que presenta **raspi-config** es la pantalla Setup Options (ver Figura 6-1) y proporciona acceso a la mayor parte de la funcionalidad básica de la herramienta. Si ejecutó la herramienta por accidente, simplemente presione la tecla Derecha dos veces seguido de Intro para salir y volver a la consola o terminal. De otro modo, seleccione la opción correspondiente a la característica que está tratando de habilitar o modificar, de acuerdo a las siguientes descripciones.

Figura 6-1

La pantalla Setup Options de raspi-config



## 1 Expand Filesystem (Expandir el Sistema de Archivos)

La opción Expand Filesystem, permite aumentar el tamaño del sistema de archivos de Raspbian para que ocupe todo el espacio disponible que tiene la tarjeta SD. La primera vez que instale Raspbian, se le preguntará si desea o no expandir el sistema de archivos al momento de su primer arranque; por lo tanto, sólo necesitará elegir esta opción si ha clonado su instalación en una tarjeta SD de mayor tamaño.

Para proceder a expandir el sistema de archivos cerciórese de que la barra de selección roja se encuentra sobre la opción Expand Filesystem y presione Intro. El menú **raspi-config** desaparecerá y una serie de mensajes de la consola se mostrarán en pantalla. Una vez que la expansión inicial haya finalizado, presione Intro para quitar el mensaje que aparece sobre reiniciar la RasPi y luego, presione la tecla Derecha dos veces seguido de la tecla Intro para salir de la herramienta. Cuando se el software le pregunte, presione Intro para reiniciar la RasPi.

Cuando la RasPi se reinicie, se llevará a cabo el resto de la expansión del sistema de archivos. En las tarjetas de gran tamaño o de menor velocidad, esto puede tardar algo de tiempo. Es importante que la RasPi no sea interrumpida durante este proceso porque si pierde la energía, el sistema de archivos puede corromperse. Si esto ocurre, se perderán todos los archivos que tenga almacenados en la tarjeta SD de la RasPi y necesitará reinstalar su sistema operativo desde cero. No obstante, la tarjeta en sí, no resultará dañada.

## 2 Change User Password (Cambiar la Contraseña de Usuario)

Por defecto, Raspbian incluye sólo una sola cuenta de usuario que no es *root* llamada *pi*. Esta cuenta es utilizada para el trabajo cotidiano con la RasPi y viene por defecto con la contraseña: *raspberrypi*. Aunque esta contraseña está bien para el uso privado, sería una buena idea cambiar la contraseña (o también el nombre del usuario) para mejorar su seguridad cuando está trabajando en una red de acceso público (en otra conexión a Internet o punto de acceso Wi-Fi).

Aunque es posible cambiar la contraseña manualmente mediante el comando `passwd` (ver Capítulo 3, “Administración del Sistema Linux” para conocer los detalles), los usuarios principiantes encontrarán que es más fácil cambiar la contraseña utilizando **raspi-config**. Elija la opción Change User Password moviendo la barra de selección con las teclas de cursor y luego presione Intro. Cuando se le indique, introduzca una nueva contraseña y luego confirme el cambio volviendo a introducir la misma contraseña. Para salir de **raspi-config**, presione la tecla Derecha dos veces seguido de la tecla Intro.

## 3 Enable Boot to Desktop (Activar el Inicio del Escritorio)

Normalmente, Raspbian se carga a una línea de comandos que tiene una interfaz basada en texto, conocida como la consola. Se carga nada más a la consola para garantizar que la Raspberry Pi se encuentre lista lo más pronto como

sea posible y además ahorrar memoria. Muchos de los usos más comunes de la RasPi (incluyendo el servidor web o la grabación de video/imágenes con el Módulo de Cámara de la Raspberry Pi) trabajan bien desde la consola sin la necesidad de cargar la interfaz gráfica de usuario.

Sin embargo, si usted utiliza su RasPi como una computadora de propósito general, puede que le resulte molesto estar escribiendo `startx` cada vez que encienda su RasPi para cargar en entorno de escritorio. Al elegir la opción Enable Boot Desktop con las teclas del cursor y presionar Intro se le preguntará si la RasPi debería o no arrancar directamente al escritorio. Si elige Yes con las teclas de cursor y presiona Intro, la interfaz gráfica de usuario se cargará tan pronto como la RasPi haya terminado de arrancar (el equivalente a escribir `startx` cada vez que se carga). Si elige No, la consola basada en texto será cargada.

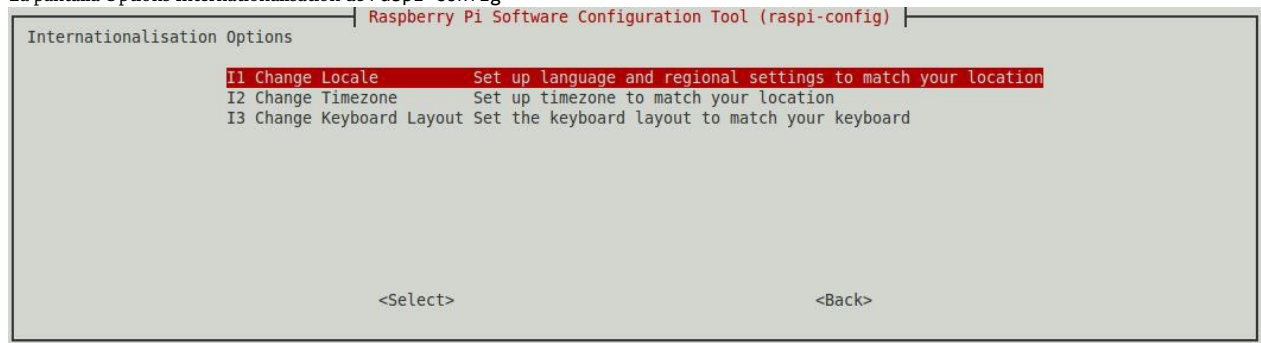
Cuando haya hecho su elección, presione la tecla de flecha Derecha dos veces seguida por la tecla Intro para salir de `raspi-config`.

## 4 Internationalisation Options (Opciones de Internacionalización)

Esta opción de menú permite que los usuarios de otros países distintos a Reino Unido configuren la RasPi a sus necesidades (ver Figura 6-2). Por defecto, Raspbian está listo para utilizar las configuraciones del idioma inglés, así como la zona horaria y distribución del teclado del Reino Unido. Los usuarios de otros países encontrarán que ciertas teclas de sus teclados no escriben los caracteres correctos, en particular, aquellos que utilizan diseños de teclados diferentes a QWERT como AZERTY o QWERTZ.

Figura 6-2

La pantalla Options Internationalisation de `raspi-config`



Use las teclas del cursor para seleccionar la opción Internationalisation Options y presione Intro. Esta opción carga un submenú que proporciona acceso a tres ajustes de localización diferentes: Change Locale, Change Timezone y Change Keyboard Layout.

### I1 Change Locale (Cambiar la Configuración Regional)

Esta opción (seleccionada con las teclas del cursor y la tecla Intro) proporciona una lista de los lenguajes disponibles para la RasPi. La lista es extensa e incluye los idiomas más comunes. Utilice las teclas para desplazarse a través de la lista que aparece y presione la barra espaciadora para poner un símbolo de asterisco(\*) enseguida del lenguaje o lenguajes que desea utilizar.

Cada lenguaje es nombrado de una forma en particular: los dos primeros caracteres representan el nombre del idioma basado en el identificador de dos letras de la Organización Internacional de Normalización (ISO). A continuación viene un guión bajo seguido por otros dos caracteres que representan el identificador específico del país para ese idioma. El prefijo `en_GB`, por ejemplo, especifica la variante del Inglés utilizado en Reino Unido; `en_US`, por contraste, es el Inglés como se usa en los Estados Unidos de América.

Enseguida del identificador de idioma y país se encuentra una cadena que muestra la *codificación de caracteres* utilizada para el idioma. La mayoría de los idiomas tendrán más de una opción aquí, pero la mayoría de los usuarios

sólo necesitarán seleccionar una: la codificación UTF-8. Ésta, especifica la codificación del Formato de Transformación Unicode de 8-bits (el estándar de codificación más comúnmente utilizado y compatible).

Cuando haya elegido su configuración regional, presione Intro para confirmar los cambios y seleccionar su idioma predeterminado. Esto le permite tener varias configuraciones regionales instaladas al mismo tiempo sin ningún problema, aunque sea consciente de que cada configuración ocupará un espacio en la tarjeta SD de la Raspberry Pi. Elija una configuración por defecto con las teclas de cursor y confirme la selección con la tecla Intro.

Si ha seleccionado varias configuraciones regionales, los cambios pueden tardar algo de tiempo en llevarse a cabo. Cuando el menú de `raspi-config` reaparezca, elija Finish con las teclas de cursor y presione Intro para salir.

---

**CONSEJO**

Cambiar la configuración regional del sistema no traduce todas las aplicaciones a su lenguaje nativo. Cuando una traducción se encuentre disponible, ésta será utilizada; de lo contrario el texto en Inglés se mostrará como el predeterminado. Cambiar la configuración local tampoco tendrá ningún efecto sobre el nombre de los comandos: por ejemplo, echo seguirá siendo llamado echo, independientemente de su configuración regional.

---

## **I2 Change Timezone (Cambiar el Huso Horario)**

El reloj del sistema de la Raspberry Pi está establecido por defecto a la hora del Tiempo Medio de Greenwich. Si vive en una zona horaria distinta la Raspberry Pi mostrará erróneamente el tiempo. Al elegir la opción de menú Change Timezone con las teclas de cursor y presionando Intro se le presentará un tercer menú, que contiene una enumeración de las áreas geográficas como el Océano Índico, los Estados Unidos y Europa. Utilice las teclas de cursor para seleccionar su área local y presione Intro.

La siguiente pantalla muestra una lista de las ciudades o regiones dentro de aquella área seleccionada. Desplácese por la lista con las teclas del cursor y presione Intro para seleccionar su ciudad. Si su ciudad no aparece en la lista, seleccione la ciudad más cercana geográficamente o la capital de su país si ninguna otra ciudad aparece listada. Cuando haya confirmado su cambio, presione Finish y presione Intro para salir de `raspi-config`.

## **I3 Change Keyboard Layout (Cambiar la Disposición del Teclado)**

Esta es una de las opciones más importantes del menú internationalisation, ya que le permite modificar la distribución de teclas de su teclado. Por defecto, la Raspberry Pi está configurada para esperar un teclado Británico en la distribución estándar QWERTY. Si utiliza un teclado diferente, elija esta opción con las teclas de cursor y presione Intro para cargar el menú de selección de distribuciones de teclados.

Utilizando las teclas de cursor, elija la distribución que corresponda a su teclado y actívela con la tecla Intro. Cuando haya regresado al menú Setup Options, presione la tecla de flecha Derecha dos veces seguida de la tecla Intro para salir de `raspi-config`.

## **5 Enable Camera (Activar Cámara)**

Esta opción debe utilizarse solamente si tiene instalado en su sistema el Módulo de Cámara de la Raspberry Pi (ver Capítulo 15, “El Módulo de Cámara de la Raspberry Pi”). Elija la opción Enable Camera con las teclas de cursor, presione Intro y luego presione nuevamente Intro para confirmar el cambio. Esto le permite a la Raspberry Pi estar preparada para utilizar con el Módulo de Cámara.

Si va a quitar el Módulo de Cámara de su Raspberry Pi, elija la opción de menú Enable Camera con las teclas de cursor, presione Intro y luego utilice la tecla de flecha Derecha para seleccionar No antes de presionar Intro nuevamente. Aunque esto no es estrictamente necesario ayuda a mantener las cosas ordenadas (aún así, tener una Raspberry Pi configurada para una cámara que no está conectada no hará ningún daño).

Es necesario asegurarse de que su Raspberry Pi cuenta con al menos 128MB disponibles para la unidad de procesamiento gráfico (GPU) a fin de poder utilizar el Módulo de Cámara para grabar video; vea la sección “A3 Memory Split” más adelante en este capítulo para conocer cómo configurar esta opción.

## 6 Add Rastrack (Añadir al Rastrack)

Rastrack es un mapa interactivo de la Raspberry Pi creado por Ryan Walmsley. Está diseñado para ayudar a los aficionados en las proximidades a localizarse unos con otros y demostrar las áreas del mundo que tienen una gran concentración de usuarios de Raspberry Pi. El servicio es voluntario y no está conectado con la Fundación Raspberry Pi.

Para que una Raspberry Pi aparezca en el mapa Rastrack, debe de estar registrada. Este es un proceso opcional y si a usted le preocupa su privacidad puede saltarse este registro sin que su RasPi pierda alguna característica o funcionalidad. Sin embargo, si desea que su RasPi aparezca en el mapa, elija la opción Add to Rastrack con las teclas de cursor y presione Intro. Tenga en cuenta que necesitará de una conexión a Internet para continuar.

Presione Intro para confirmar que ha leído el mensaje que aparece, en seguida, introduzca su nombre (o un apodo) seguido de la tecla Intro. Introduzca un correo electrónico a continuación y presione Intro nuevamente para añadir su Raspberry Pi al mapa. No necesita proporcionar su ubicación porque el software utiliza su conexión de Internet para colocar su RasPi en el mapa. Rastrack no proporcionará su ubicación exacta a nivel de calle, pero mostrará a los demás usuarios más o menos en qué lugar del mundo usted y su RasPi se encuentran.

Finalmente, presione Intro para volver a la pantalla principal de `raspi-config`, presione la tecla de flecha Derecha dos veces y luego Intro para salir de la herramienta y volver a la consola o terminal. Para ver el mapa, visite el sitio web [rastrack.co.uk](http://rastrack.co.uk).

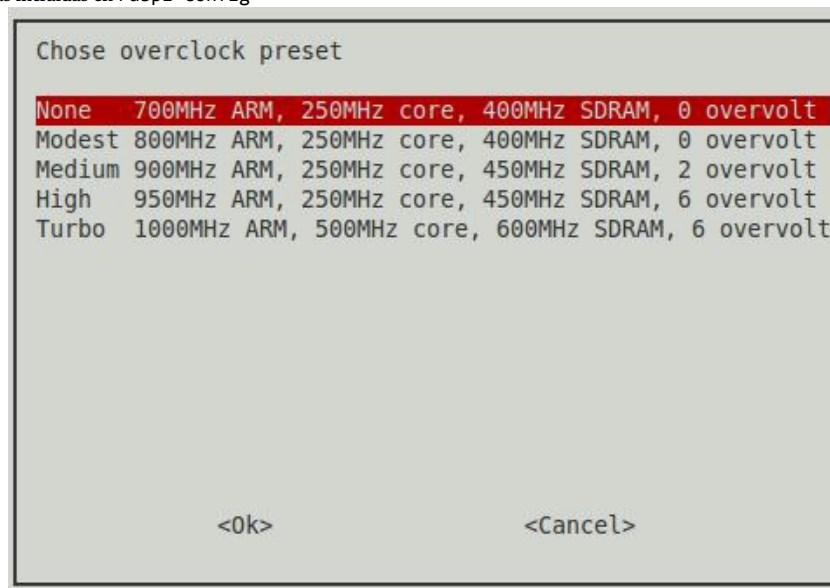
## 7 Overclock

Overclocking se refiere al proceso de hacer funcionar un dispositivo a una velocidad más alta de la que los fabricantes propusieron. El procesador BCM2835 de la Raspberry Pi puede funcionar por encima de su velocidad predeterminada de 700MHz para aumentar el rendimiento del sistema. No obstante, tal rendimiento tiene un costo, el chip funcionará a una mayor temperatura que antes, consumirá más energía y puede experimentar una vida útil más corta que una RasPi que está funcionando a su velocidad predeterminada.

Aunque es posible alterar el rendimiento del procesador BCM2835 manualmente (ver Capítulo 7, “Configuración Avanzada de la Raspberry Pi”) la forma más segura es utilizando `raspi-config`. Esto le limita a una selección de ajustes de overclock preconfigurados que son conocidos como seguros de utilizar con la mayoría de las Raspberrys (ver Figura 6-3).

Figura 6-3

Las opciones predefinidas incluidas en `raspi-config`



**ADVERTENCIA**

Aunque las configuraciones de overclock disponibles en raspi-config son razonablemente seguras y no dañan la garantía, no todas las Raspberry Pi pueden alcanzar las velocidades máximas. Si observa que su RasPi se comporta inestable (particularmente si experimenta que los archivos en la tarjeta SD se vuelven corruptos) disminuya la velocidad a una más baja de la que está actualmente utilizando o devuelva la RasPi a su velocidad predeterminada de 700MHz.

Para overclockear su Raspberry Pi, seleccione la opción Overclock con las teclas de cursor y presione Intro. Lea cuidadosamente el mensaje que aparece y luego presione Intro para confirmar que desea continuar. Si ha cambiado de opinión, puede simplemente presionar Intro tan pronto como aparezca la pantalla siguiente para mantener intacta la velocidad existente de la Raspberry Pi.

Si desea probar el overclocking de su RasPi, utilice las teclas de cursor para seleccionar una de las opciones de overclock preestablecidas que aparecen en la pantalla siguiente: None, Modest, Medium, High y Turbo. Casi todos los modelos de Raspberry Pi deben ser capaces de utilizar la configuración Modest, la cual incrementa la velocidad de reloj del procesador a 800MHz. Muchas otras Raspberry Pi serán capaces de utilizar la configuración Medium, que incrementa el voltaje aplicado al procesador para llegar a los 900MHz y adicionalmente incrementa la velocidad de la memoria de 400MHz a 450MHz.

Los dos configuraciones restantes pueden no funcionar en todos los modelos de Raspberry Pi. El procesador BCM2835 es un circuito complejo que difiere en su capacidad de ser overlockeado de unidad a unidad (incluso entre dos RasPis del mismo lote) La configuración High incrementa la velocidad a 950MHz pero requiere significativamente más voltaje que Medium o Modest para lograrlo, lo que significa más calentamiento y una reducción de la vida útil del procesador. La configuración Turbo incrementa el procesador a 1000MHz, además de doblar el reloj de gráficos a 500MHz e incrementar el reloj de memoria a 600MHz. Esta configuración ofrece el mejor de los rendimientos, pero está vinculado a la corrupción de la tarjeta SD (donde los archivos se vuelven ilegibles) en ciertos modelos de Raspberry Pi.

Si desea hacer uso del modo Turbo, asegúrese de contar con un respaldo de los archivos importantes almacenados en su RasPi antes de proceder. Si comienza a tener problemas, baje la configuración a High o una más abajo y su RasPi volverá a la normalidad.

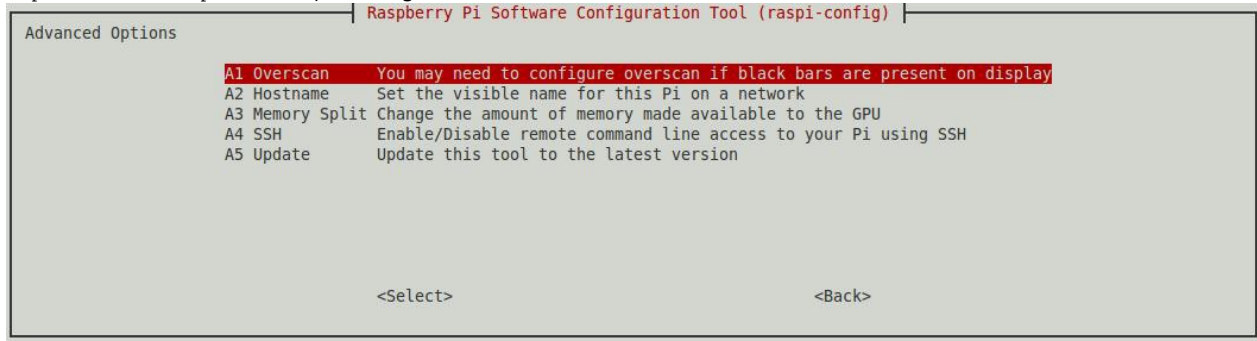
Cuando haya hecho su elección, presione la tecla Intro y luego presione Intro nuevamente para reiniciar la RasPi con la nueva velocidad.

Si selecciona una velocidad más alta que no es capaz la RasPi de aceptar, puede que no arranque correctamente. Si observa que su RasPi se carga a una pantalla negra o que constantemente se reinicia sin preguntarle si quiere iniciar la sesión, es porque ha elegido una velocidad demasiado alta. Mantenga presionada la tecla Shift en el teclado para pasar por alto temporalmente la configuración de overclock y así cargar la RasPi con su velocidad de fábrica. Una vez iniciada la RasPi, suelte la tecla Shift e inicie sesión, luego cargue **raspi-config** y ingrese al menú Overclock para elegir una opción más conservadora.

## 8 Advanced Options (Opciones Avanzadas)

Al elegir la opción Advanced Options con las teclas de cursor y presionando Intro se cargará el menú Advanced Options (ver Figura 6-4), que proporciona acceso a algunos ajustes adicionales que no se encuentran en el menú Setup Options. Si ingresa a este menú por equivocación, puede volver al menú principal Setup Options presionando la tecla de flecha Derecha dos veces seguido de la tecla Intro.

Figura 6-4

La pantalla Advanced Options de `raspi-config`

### A1 Overscan

Muchos televisores tiene la característica *overscan*, que significa que el área visible de imagen es ligeramente más pequeña que la imagen transmitida. En la emisión de televisión, el overscan se utiliza para ocultar datos adicionales tales como la información de código de tiempo, pero en la computación, esto produce que los bordes de la pantalla permanezcan ocultos. Por el contrario, utilizando una señal de televisión con un moderno monitor puede revelar los bordes previamente ocultos con sus datos adicionales.

Es posible que necesite ajustar el overscan por alguna de dos razones: porque la imagen de su RasPi se encuentra rodeada por barras negras, en cuyo caso el overscan necesita ser reducido o deshabilitado por completo; o porque la imagen de su RasPi se extiende más allá de los bordes visibles de su pantalla, en cuyo caso el overscan necesita ser incrementado. Esto último es lo más común cuando se utiliza la salida de video compuesto de la RasPi con un viejo televisor (ver Capítulo 2, “Comenzando con la Raspberry Pi”).

Para ajustar la configuración del overscan, seleccione la opción Overscan con las teclas de cursor y presione Intro. Un menú aparecerá preguntándole si el overscan debe ser activado o desactivado. Si está utilizando un monitor o televisor mediante el puerto HDMI, elija Disable presionando Intro; de lo contrario, presione la tecla de flecha Derecha para seleccionar Enable y confirme con la tecla Intro. Para salir de `raspi-config` presione la tecla de flecha Derecha dos veces hasta seleccionar Finish y presione Intro. Se le pedirá que reinicie para aplicar la nueva configuración; confirme con Intro.

El menú Overscan en `raspi-config` sólo le permite activar o desactivar la configuración; para controlar el tamaño del overscan y maximizar el área visible de la pantalla en un televisor viejo, vea el Capítulo 7, “Configuración Avanzada de la Raspberry Pi”.

### A2 Hostname (Nombre de Host)

Un *hostname* o *nombre de host* del sistema es el nombre que se utiliza para identificarlo en la red. Cuando usa la consola o terminal en la RasPi, verá el *nombre de host* como parte del prompt que acepta sus comandos. Los *nombres de host* debe ser únicos, lo que puede ocasionar un problema si cuenta con más de una Raspberry Pi en la misma red. Puede cambiar el *nombre de host* de la RasPi en cualquier momento utilizando la opción Hostname del menú Advanced Options de `raspi-config`, seleccionando tal opción con las flechas de cursor y luego presionando la tecla Intro.

Cuando accede a la configuración de Hostname, `raspi-config` le informará las normas a seguir para nombrar a un *nombre de host*. Debido a que los nombres de host se adhieren a una norma internacional (conocida como RFC o *petición de comentarios*), ciertos caracteres no están permitidos: un *nombre de host* sólo debe contener letras y números, y puede incluir guiones siempre y cuando no estén al principio o al final. Confirme que ha leído este mensaje presionando Intro.

Utilizando la tecla de Retroceso, borre el nombre de host actual que aparece en el cuadro y escriba el nuevo nombre que haya elegido. Puede optar por un nombre de host descriptivo, como “raspi-sala” o puede nombrar sus dispositivos basado en una temática como “bladerunner” para las películas de ciencia ficción. Cuando haya elegido su nombre, presione Intro para confirmar. Para salir de **raspi-config**, presione la tecla de flecha Derecha dos veces hasta seleccionar Finish y luego presione Intro. Se le pedirá reiniciar la RasPi; presione Intro para confirmar.

### A3 Memory Split (División de Memoria)

Dependiendo de qué modelo de Raspberry Pi tenga, contará con 512MB o 256MB de memoria disponible para su sistema. Esta memoria se encuentra dividida entre el procesador de propósito general (*CPU o unidad central de procesamiento*) y el procesador de gráficos (*GPU o unidad central de procesamiento gráfico*). Por defecto, la GPU tiene reservada 128MB de memoria mientras que el resto se dedica a la CPU.

Si su Raspberry Pi no utiliza el procesador gráfico, porque la está empleando como un servidor web sin una pantalla conectada, puede reducir la cantidad de memoria reservada para la GPU utilizando la opción Memory Split en **raspi-config**. Seleccione esta opción con las teclas de cursor y presione Intro para cargar el menú.

Con la tecla Retroceso, borre el valor actual que se muestra en la casilla e introduzca uno nuevo. El valor mínimo que puede darle a la GPU para garantizar su adecuado funcionamiento es de 16MB y puede introducirlo simplemente escribiendo el número 16. Si tiene un modelo de Raspberry Pi con 512MB de memoria, puede incrementar la división a 256MB, lo que permitirá obtener un mejor rendimiento cuando utilice la GPU para renderizar complejas escenas 3D en los videojuegos.

Puede elegir otros valores pero debe limitarse a duplicar el valor en cada paso: por ejemplo, 16MB puede ser incrementado a 32MB, 32MB a 64MB, 64MB a 128MB y así sucesivamente.

Seleccionar un valor desigual, como 17MB, no hace daño a la RasPi, pero puede resultar que un valor distinto al seleccionado sea utilizado.

Cuando haya elegido la división de memoria, presione la tecla Intro para confirmar. Para aplicar la configuración, salga de **raspi-config** presionando la tecla de flecha Derecha dos veces seguido de la tecla Intro y luego presione Intro nuevamente para confirmar que desea reiniciar.

### A4 SSH

El *intérprete de órdenes segura*, también conocido como *SSH (Secure Shell)*, es una forma de la que dispone usted para acceder a la terminal de la RasPi a través de una red. Con mucha frecuencia se utiliza cuando la Raspberry Pi funciona como un servidor independiente, ya que no necesita tener conectado un teclado o monitor, SSH le permite al usuario acceder a la terminal basada en texto y también transferir archivos hacia o desde la RasPi.

Algunos usuarios pueden querer mejorar la seguridad de sus Raspberrys desactivando esta característica. Seleccione la opción SSH utilizando las teclas de cursor y presione la tecla Intro. En la ventana que aparece, seleccione Activar o Desactivar para encender o apagar el SSH. Incluso si no necesita esta característica, es buena idea mantenerla activada para en un futuro poder solucionar algún problema que se presente; una mejor forma de incrementar la seguridad en su equipo es asegurándose de cambiar la contraseña del usuario *pi*.

### A5 Update (Actualización)

La opción Update es la que le garantiza que trabajará con la versión más reciente de **raspi-config**. Frecuentemente la herramienta **raspi-config** es actualizada con nuevas características o diversas configuraciones agregadas al menú, por lo tanto, siempre es una buena idea asegurarse de que su versión está actualizada.

Seleccione esta opción con las teclas de cursor y presione Intro para comenzar el proceso de actualización. Para completar este proceso requerirá de una conexión a Internet, por lo que si está utilizando una Raspberry Pi Modelo A necesitará conectar un adaptador de red USB para continuar.



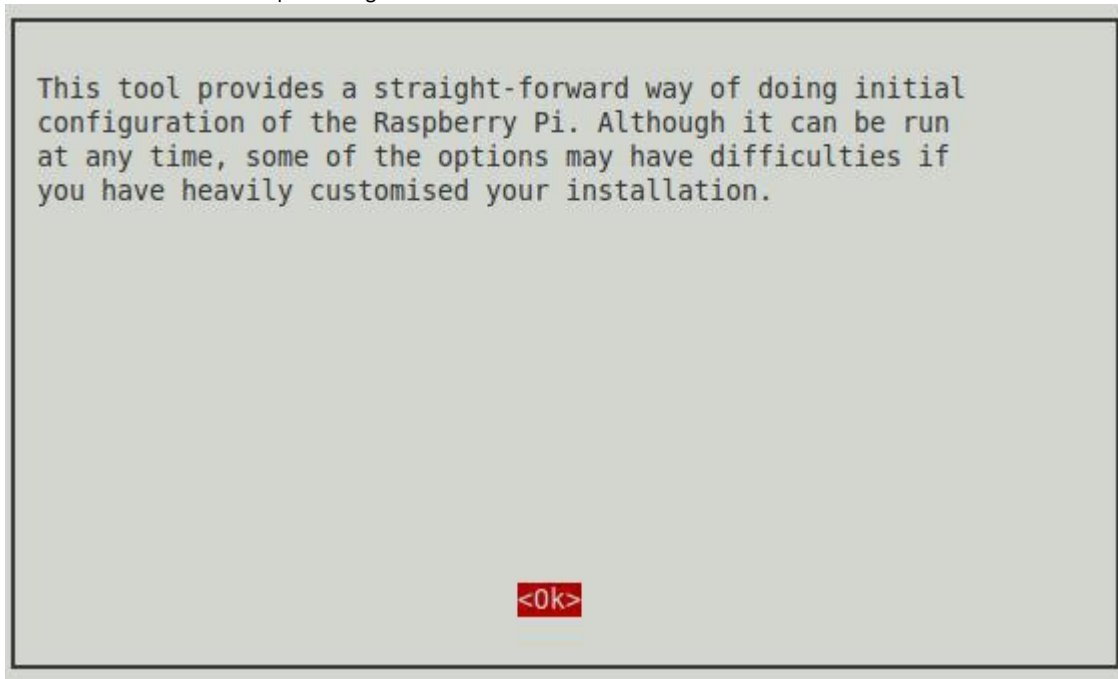
Considere que esta opción sólo actualizará a `raspi-config` y no otro aspecto del sistema. Para actualizar todo el sistema, incluyendo a `raspi-config`, vea las instrucciones en el Capítulo 3, “Administración del Sistema Linux”.

Cuando la actualización haya finalizado, salga de `raspi-config` presionando la tecla de flecha Derecha dos veces y en seguida la tecla Intro.

## 9 About raspi-config (Acerca de raspi-config)

Es la última opción disponible en el menú `raspi-config` y sólo muestra información sobre la herramienta. Al seleccionar esta opción con las teclas de cursor y presionando Intro se desplegará una ventana explicando el propósito de esta herramienta (ver Figura 6-5) y no se realizará ningún cambio en el sistema. El mensaje puede quitarse presionando Intro, para cerrar `raspi-config` presione la tecla de flecha Derecha dos veces y en seguida la tecla Intro.

**Figura 6-5**  
Acerca de la herramienta de software `raspi-config`



# Capítulo 7

## Configuración Avanzada de la Raspberry Pi

**Debido a sus** orígenes en la informática embebida, el chip BCM2835 en el interior de la Raspberry Pi no tiene nada que se le parezca al menú *BIOS* (*Basic Input-Output System*) de una PC en donde las variadas configuraciones de bajo nivel puede ser modificadas. En su lugar, la Raspberry Pi se apoya en archivos de texto que contienen cadenas de configuración que son cargadas por el chip cuando la RasPi es encendida.

Antes de echar un vistazo en las diversas opciones disponibles en estos archivos de texto (`config.txt` y `cmdline.txt`) una palabra de advertencia: modificar algunas de estas configuraciones con valores diferentes a los valores predeterminados puede resultar, en el mejor de los casos, que la RasPi no reinicie hasta que los archivos sean revertidos a su configuración anterior; y en el peor de los casos, que una mala configuración pueda dañar físicamente al sistema. Estas configuraciones potencialmente peligrosas serán señaladas con advertencias en este capítulo.

---

**CONSEJO**

Si está utilizando la distribución Raspbian, la manera más fácil de modificar la mayoría de las configuraciones es utilizando `raspi-config` (ver Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi”). Las instrucciones que se proporcionan aquí están dirigidas a los usuarios que utilizan otras distribuciones o que prefieran hacer las cosas manualmente.

---

## Edición de los Archivos de Configuración a través de NOOBS

Si ha editado sus archivos de configuración de tal manera que su sistema operativo ya no arranca, la manera más fácil de restaurar estos archivos es utilizando el software NOOBS (ver Capítulo 2, “Comenzando con la Raspberry Pi”). Si su sistema operativo fue instalado a través de NOOBS, puede utilizar la herramienta de edición de archivos de configuración; si instaló su sistema operativo manualmente, necesitará retirar la tarjeta SD y utilizar una segunda computadora para editar los archivos.

Para cargar NOOBS después de haberlo utilizado para instalar su sistema operativo, mantenga presionada la tecla Shift mientras enciende la RasPi. Esto evitará que se cargue el sistema operativo y en su lugar se cargue NOOBS, pero con una nueva opción: el botón Edit Config en la parte superior del menú (ver Figura 7-1).

Al hacer clic en el botón Edit Config (o presionando la tecla E) se cargará una ventana de edición de texto con dos archivos de configuración precargados: `config.txt` y `cmdline.txt` (ver figura 7-2). Puede hacer los cambios que se describen en este capítulo en el editor de textos con su teclado y ratón, y hacer clic en el botón OK que está en la parte inferior derecha para guardar sus cambios a la tarjeta SD. Cuando haya finalizado de realizar los cambios, haga clic en Exit sobre la pantalla principal para reiniciar la RasPi con sus nuevos ajustes.

---

**CONSEJO**

Si tiene una Raspberry Pi Modelo B que está conectada a Internet, puede también hacer clic en el botón Online Help o presionar la tecla H para cargar un navegador web que le llevará a los foros de Raspberry Pi. Este es un recurso invaluable si su RasPi no está funcionando adecuadamente, permitiéndole enviar una pregunta u obtener ayuda para volver a poner en marcha a su RasPi.

---

Figura 7-1  
El botón Edit Config en NOOBS

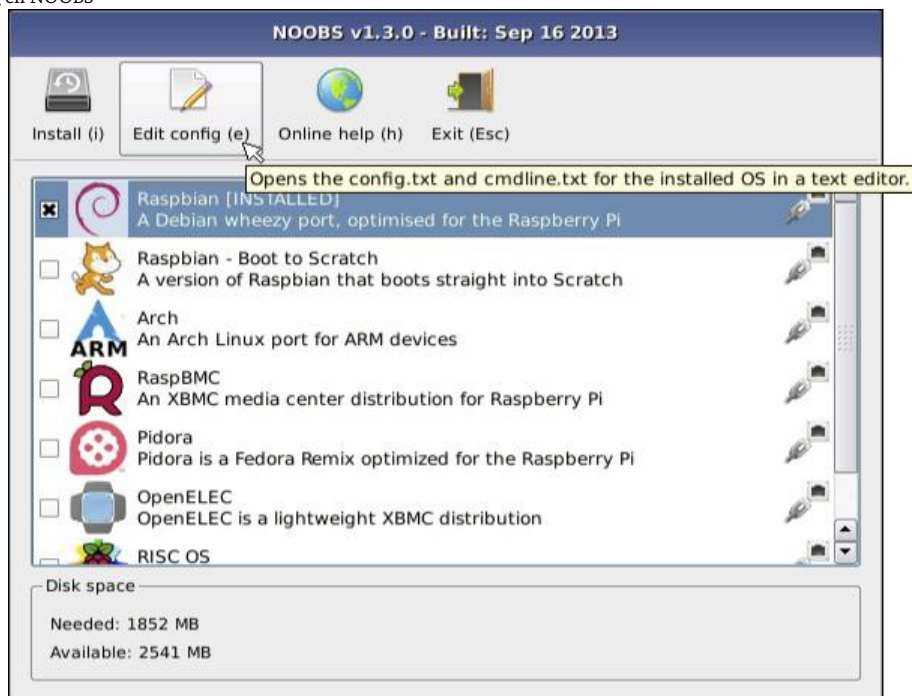
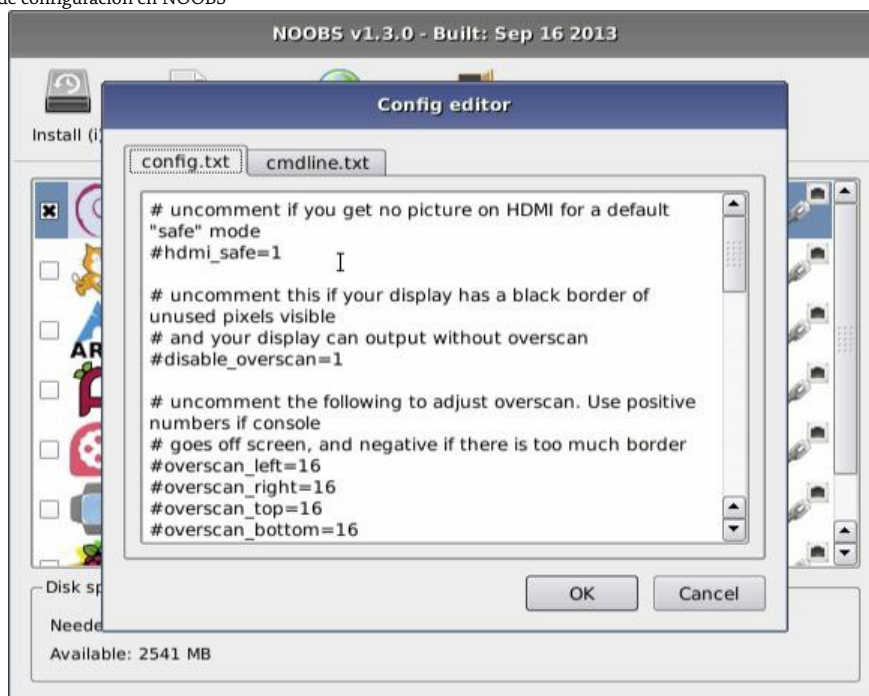


Figura 7-2  
Editando los archivos de configuración en NOOBS

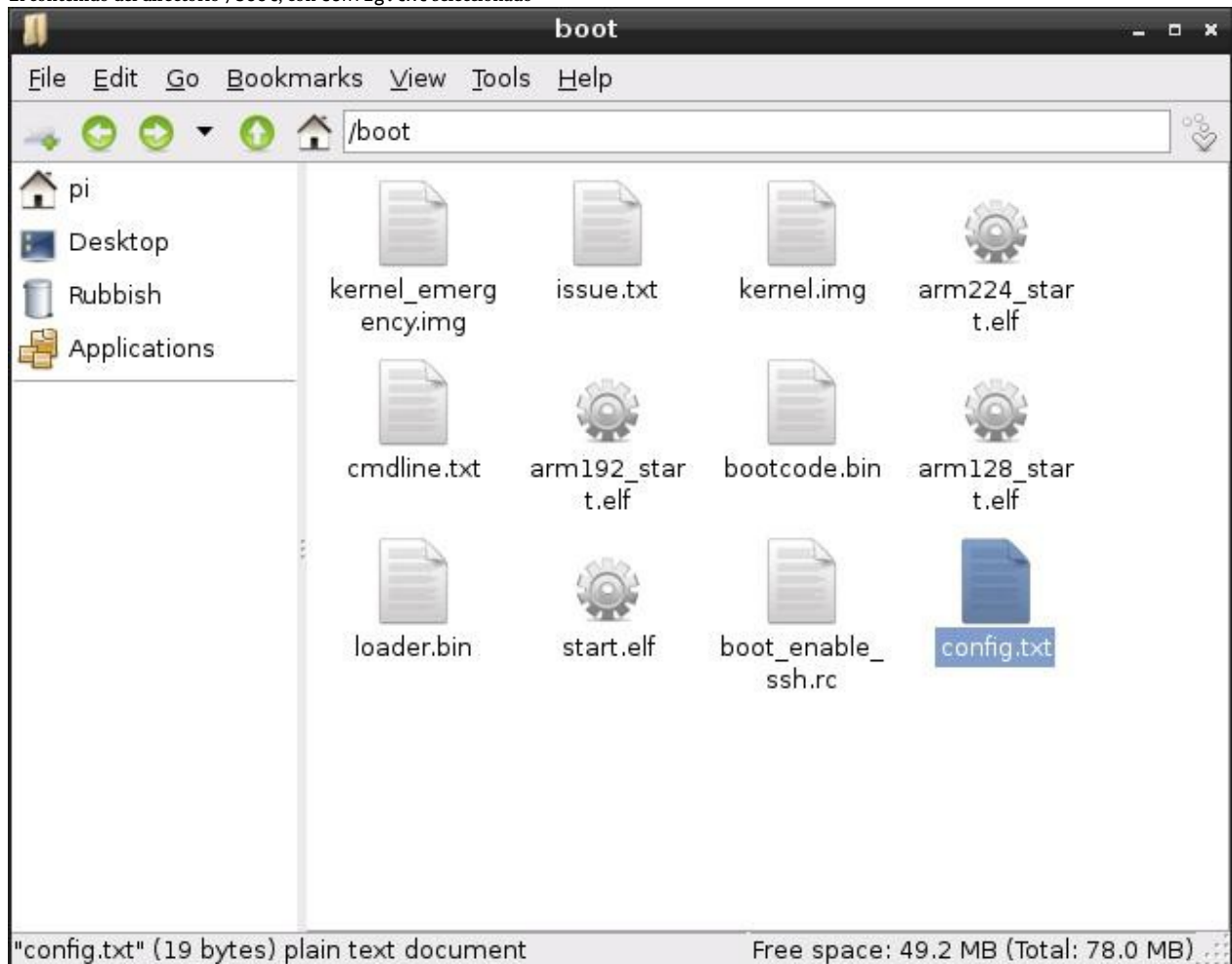


## Configuraciones Hardware - config.txt

El hardware de la RasPi está controlado por las configuraciones contenidas en un archivo llamado `config.txt`, ubicado en el directorio `/boot` (ver Figura 7-3). Este archivo le dice a la RasPi cómo configurar sus diferentes entradas y salidas, así como también, le indica a qué velocidad el chip BCM2835 y su módulo de memoria deben trabajar.

Figura 7-3

El contenido del directorio `/boot`, con `config.txt` seleccionado



El archivo `config.txt` es el lugar indicado para corregir problemas con la salida de gráficos, tales como que una imagen no llena la pantalla o que se sale del borde de la misma. Comúnmente, el archivo se encuentra vacío o (en algunas distribuciones) simplemente no está presente; esto solamente indica que la RasPi trabajará utilizando sus configuraciones predeterminadas. Si desea hacer cambios y el archivo no se encuentra, basta con crear un archivo de texto llamado `config.txt` y colocar las configuraciones que desea modificar.

El archivo `config.txt` puede controlar casi todos los aspectos del hardware de la RasPi, con la excepción de la forma en que las secciones CPU (unidad central de procesamiento) y GPU (unidad de procesamiento gráfico) del BCM2835 se reparten la memoria. Aprenderá más adelante en este capítulo cómo cambiar esa división en la sección “Particionamiento de la Memoria”.

El archivo `config.txt` sólo es leído cuando el sistema se inicia. Cualquier modificación hecha mientras la RasPi se encuentra en funcionamiento no tendrá ningún efecto hasta que el sistema sea reiniciado o apagado y nuevamente

encendido. En caso de que los cambios o las modificaciones no sean deseadas, bastará con borrar el archivo `config.txt` del directorio `/boot` para restaurar las configuraciones de vuelta a sus valores predeterminados. Si la RasPi no arranca con las nuevas configuraciones, sólo retire la tarjeta SD y desde otra PC borre el archivo `config.txt` de la partición `boot`, y posteriormente vuelva a insertar la tarjeta SD a la RasPi e intente de nuevo.

## Modificación de la Pantalla

Por lo general, la Raspberry Pi detectará el tipo de pantalla a la que está conectada y modificará sus configuraciones en consecuencia. En ocasiones, sin embargo, esta detección automática no funciona. Esto a menudo sucede cuando una Raspberry Pi de un país es conectada a una vieja TV de otro país. Si conecta su RasPi a la TV y no ve absolutamente nada, puede que tenga que hacer caso omiso a los valores predeterminados.

Varias configuraciones dentro del archivo `config.txt` pueden utilizarse para mejorar o cambiar la salida de video. Esas configuraciones y sus valores posibles son descritos en la siguiente lista.

---

### ADVERTENCIA

El ajuste manual de las configuraciones de salida de video HDMI (High Definition Multimedia Interface) o video compuesto puede dejar a su RasPi incapaz de comunicarse con su monitor. Por lo general es mejor usar las configuraciones detectadas automáticamente, a menos que sea incapaz de ver alguna imagen.

---

- **overscan\_left**-Este ajuste mueve la imagen hacia adentro un número determinado de píxeles para compensar el *overscan* de la TV. Si el texto en la RasPi desaparece fuera del borde de la pantalla, ajuste el *overscan* para solucionar esto. Se deben proporcionar valores que indican cuántos píxeles del lado izquierdo deben de saltarse.
- **overscan\_right**-Este ajuste hace el mismo trabajo que `overscan_left`, pero sobre el lado derecho de la pantalla.
- **overscan\_top**-De nuevo, este ajuste ignora cierto número de píxeles, pero esta vez sobre la parte superior de la pantalla.
- **overscan\_bottom**-Este ajuste puede utilizarse para saltarse un número de píxeles desde la parte inferior de la pantalla. Comúnmente, los valores para todos los ajustes `overscan_` deberían ser los mismos, creando así un borde rectangular alrededor de la pantalla.
- **disable\_overscan**- Si utiliza un monitor o televisión con interfaz HDMI, es posible que su imagen se muestre con un borde negro alrededor de la pantalla. Para deshacerse de ese borde, cualquiera de los ajustes predeterminados para *overscan* puede deshabilitarse estableciendo este valor a 1.
- **framebuffer\_width**-Este valor se mide en píxeles y su alteración cambiará el ancho de la consola. Si el texto aparece demasiado pequeño en su pantalla, intente cambiar este ajuste por un valor menor que el ancho predeterminado de la pantalla que tiene conectada.
- **framebuffer\_height**-Este ajuste cambia el tamaño de la consola de la misma forma que `framebuffer_width`, pero verticalmente.
- **framebuffer\_depth**-Controla la profundidad de color de la consola en bits por píxel. El valor predeterminado es de 16 bits por píxel, lo que ofrece 65,536 colores. Otros valores como 8 bits por píxel (256 colores), 24 bits por píxel (alrededor de 16.7 millones de colores) y 32 bits por píxel (alrededor de 1 billón de colores) son válidos, pero pueden provocar corrupción gráfica.
- **framebuffer\_ignore\_alpha**-Establecido a 1, desactiva el *canal alfa*, el cual controla la transparencia en la consola. La desactivación del canal alfa por lo general no es necesaria, pero puede corregir la corrupción gráfica producida cuando se establece `framebuffer_depth` a 32 bit por píxel.
- **sdtv\_mode**-Este valor afecta la salida de video compuesto de la RasPi, ajústelo para trabajar en varios países. Por defecto, la RasPi utiliza la versión norteamericana del estándar de video NTSC; los usuarios de otros países pueden necesitar cambiar este valor para obtener una imagen sobre una TV analógica. Los valores posibles son:

- **0-NTSC**, el estándar de video de Norteamérica
- **1-NTSC-J**, el estándar de video Japonés
- **2-PAL**, el estándar de video para el Reino Unido y otros países
- **3-PAL-M**, el estándar de video brasileño
- **sdtv\_aspect**-Controla la relación de aspecto de la salida compuesta analógica. Si la imagen se percibe alargada o aplastada, modifique este ajuste a la relación de aspecto de su TV. Los valores posibles son:
  - **1**-Relación de aspecto 4:3, común en los equipos viejos
  - **2**-Relación de aspecto 14:9, común para las televisiones de pantalla ancha más pequeñas
  - **3**-Relación de aspecto 16:9, común para las modernas televisiones de pantalla ancha
- **hdmi\_mode**-Además de establecer el modo de video para la salida compuesta analógica, es posible anular la detección automática de la resolución sobre el puerto HDMI. Esto es útil si desea ejecutar su RasPi a una resolución más baja que la resolución nativa de la pantalla a fin de hacer que las cosas sean más legibles a la distancia. El apéndice C, “Modos de Pantalla HDMI”, muestra una lista los posibles valores para esta configuración.
- **hdmi\_drive**-También es posible modificar el voltaje de salida para el puerto HDMI. Esto es útil cuando está utilizando un adaptador HDMI a DVI, porque los voltajes HDMI y DVI difieren ligeramente. Si encuentra que su imagen está cubierta de nieve o se ve “apagada” con una imagen muy brillante, intente modificar esta configuración. Los valores posibles son:
  - **1**-Voltajes de salida DVI. En este modo, nada de audio se incluye sobre el cable HDMI.
  - **2**-Voltajes de salida HDMI. En este modo, el audio va incluido sobre el cable HDMI.
- **hdmi\_force\_hotplug**-Obliga a la Raspberry Pi a utilizar el puerto HDMI, incluso si no detecta una pantalla conectada. Un valor de 0 permite a la RasPi intentar detectar la pantalla, mientras que un valor de 1 obliga a la RasPi a utilizar HDMI pese a todo.
- **hdmi\_group**-Establece el modo de grupo HDMI a CEA o DMT. Debe cambiar este ajuste de acuerdo al tipo de pantalla que está tratando de conectar antes de poder utilizar **hdmi\_mode** para controlar la resolución y frecuencia de salida. Los dos valores posibles son:
  - **1**-Establece el grupo HDMI al definido por la CEA (Consumer Electronics Association of America). Utilice esta configuración cuando la RasPi sea conectada una televisión de alta definición (HDTV) mediante el cable HDMI y utilice la primera lista de configuraciones listada en el Apéndice C, “Modos de Pantalla HDMI”.
  - **2**-Establece el grupo HDMI a la definida por la VESA (Video Electronics Standards Association) en la especificación DTM (Display Monitor Timings). Utilice esta configuración cuando la RasPi está conectada a un monitor de computadora mediante DVI y utilice la segunda lista de configuraciones del Apéndice C, “Modos de Pantalla HDMI”.
- **hdmi\_safe**-Obliga a la RasPi a utilizar un conjunto de configuraciones HDMI diseñadas para proporcionar la máxima compatibilidad con las pantallas conectadas al puerto HDMI. Establecer este valor a 1 equivale a establecer **hdmi\_force\_hotplug=1**, **config\_hdmi\_boost=4**, **hdmi\_group=1**, **hdmi\_mode=1** y **disable\_overscan=0**.
- **config\_hdmi\_boost**-Algunos monitores requieren de más potencia sobre la salida HDMI para funcionar. Si su imagen se ve cubierta de nieve, trate de incrementar este valor por etapas de 1 (para cables cortos) a 7 (para cables largos).

Cada opción dentro de **config.txt** debe estar en su propia línea, con el nombre de la opción seguido por un signo de igual (=) y entonces el valor requerido. Por ejemplo, para decirle a la RasPi que utilice un formato de TV analógica PAL con una relación de aspecto de 4:3 y un overscan de 20 píxeles por todos lados, coloque las siguientes líneas dentro **config.txt**:

```
sdtv_mode=2
sdtv_aspect=1
overscan_left=20
```

```
overscan_right=20  
overscan_top=20  
overscan_bottom=20
```

Para decirle a la RasPi que utilice una pantalla DVI a través del puerto HDMI en formato 720p60 sin ningún overscan, utilice los siguientes valores en su lugar:

```
hdmi_group=1  
hdmi_mode=4  
hdmi_drive=1  
disable_overscan=1
```

Para que los cambios tengan efecto, la RasPi debe reiniciarse. Si encuentra que los cambios que ha realizado han deshabilitado la salida de video de la RasPi en su monitor, simplemente inserte la tarjeta SD en otra computadora y modifique el archivo `config.txt` con las nuevas configuraciones, o simplemente bórralo por completo para restaurar los valores predeterminados.

## Opciones de Arranque

El archivo `config.txt` puede también ser utilizado para controlar la forma en que Linux se carga sobre la Raspberry Pi. Aunque el método más frecuente para controlar la carga del núcleo Linux es utilizando un archivo separado llamado `cmdline.txt` (el cual aprenderá más adelante en este capítulo), es también posible utilizar el archivo `config.txt`. Las siguientes opciones controlan el proceso de arranque:

- **disable\_commandline\_tags**-Este ajuste le dice al módulo `start.elf` que se cargue primero durante el proceso de arranque de la RasPi para saltar llenando las pasadas posiciones de memoria 0x100 antes de cargar el núcleo Linux. Esta opción no debe ser deshabilitada, hacerlo puede ocasionar que Linux se cargue incorrectamente y se cuelgue.
- **cmdline**-Los parámetros de línea de comandos que serán pasados al núcleo Linux. Este puede utilizarse en lugar del archivo `cmdline.txt`, comúnmente ubicado en el directorio `/boot`.
- **kernel**-El nombre del archivo del núcleo a ser cargado. Este puede utilizarse como una forma de cargar el núcleo de emergencia (ver Capítulo 4, “Solución de Problemas”).
- **ramfsfile**-El nombre del sistema de archivos RAM inicial (RAMFS) que será cargado. Este rara vez será modificado, a menos que haya creado un nuevo sistema de archivo inicial con el cual experimentar.
- **init\_uart\_baud**-La velocidad de la consola serie, en bits por segundo. El valor predeterminado es 115200, pero valores más bajos pueden mejorar la conexión si la RasPi es utilizada con un viejo hardware de terminal serie.

## Overclockeando la Raspberry Pi

El archivo `config.txt` no sólo controla las salidas de gráficos del procesador BCM2835 de la RasPi, sino que también le permite controlar el chip en otras formas posibles. En particular, le permite modificar la velocidad en la cual el chip trabaja, incrementando así su rendimiento a expensas de la vida útil del equipo (un proceso conocido como *overclocking*).

---

### ADVERTENCIA

El ajuste de cualquiera de las configuraciones listados en esta sección puede resultar en daños para su RasPi. En particular, los cambios en las configuraciones correspondientes a los voltajes de la memoria, GPU o CPU ya que establecerán un fusible en el chip, el cual inválida la garantía de la Raspberry Pi incluso si la configuración es regresada a la normalidad antes de que se produzca algún daño. Los daños ocasionados al utilizar estas configuraciones no serán reparadas por la Fundación Raspberry Pi o por el comerciante a quién adquirió su RasPi. En caso de tener alguna duda, *no modifique estas configuraciones*: las ganancias que se obtienen en el rendimiento a través del overclocking rara vez valen la pena el arriesgar a la RasPi.

---



El procesador multimedia BCM2835 en el interior de la RasPi es un system-on-chip (SoC) que cuenta con un diseño que está dividido en dos partes principales: el procesador gráfico (GPU) y el procesador central (CPU). En pocas palabras, la CPU se encarga de todas las tareas de procesamiento del día a día, mientras que la GPU se encarga de dibujar las cosas en la pantalla, tanto en 2D como en 3D.

Utilizando `config.txt`, puede conseguir hacer overclock en una o ambas partes del BCM2835. Puede además incrementar la velocidad en la cual trabaja el módulo de la memoria (localizada en la parte superior del chip en un formato de montaje paquete-sobre-paquete [PoP]).

El aumento de la frecuencia del funcionamiento de estos componentes da como resultado un pequeño incremento en el rendimiento de la RasPi: un incremento a la frecuencia de reloj de la GPU significa que los gráficos 3D (tales como los gráficos de los juegos) se renderizarán a un ritmo más rápido y que el video será decodificado más pronto para obtener una reproducción más suave; e incrementar la frecuencia de reloj de la CPU aumentará el rendimiento general del dispositivo, como también incrementará la frecuencia de la memoria.

La razón por la que la RasPi no es proporcionada con velocidades de funcionamiento mayores, está en primera instancia relacionada con el tiempo de vida útil de los chips. El BCM2835 está evaluado por su fabricante (Broadcom) para funcionar a una velocidad de 700MHz. Incrementar la velocidad por encima del nivel evaluado oficialmente puede funcionar, pero esto también traerá un efecto perjudicial en la vida útil del chip. A diferencia de un procesador de escritorio, los diseños SoC rara vez cuentan con mucho margen para el overclocking.

### Configuraciones de Overclocking

Si está dispuesto a correr el riesgo de estropear su RasPi (un proceso conocido como *bricking* o *hacer brick* entre los círculos de los dispositivos embebidos) en aras de una pequeña ganancia en el rendimiento, hay algunas configuraciones en `config.txt` que le pueden ayudar. Los siguientes ajustes controlan el rendimiento del SoC de la RasPi:

- **arm\_freq**-Establece la frecuencia principal de reloj de la parte de la CPU del BCM2835, para conseguir un aumento en el rendimiento de propósito general. La velocidad predeterminada es de 700 MHz.
- **gpu\_freq**-Establece la frecuencia de reloj de la parte de la GPU del BMC2835, para conseguir un aumento en el rendimiento gráfico a través de todas las operaciones. La velocidad predeterminada es de 250 MHz. Además, puede ajustar individualmente las partes del hardware de la GPU utilizando las siguientes opciones:
  - **core\_freq**-Establece la frecuencia principal de reloj de la GPU, dejando a las otras frecuencias intactas, para mejorar el rendimiento general de la GPU. El valor predeterminado es de 250 MHz.
  - **h264\_freq**-Establece la frecuencia de reloj del hardware decodificador de video de la GPU para mejorar la reproducción de datos de video H.264. El valor predeterminado es de 250MHz.
  - **isp\_freq**-Establece la frecuencia de reloj del ISP (Image Sensor Pipeline), para mejorar la velocidad de captura del hardware de video conectado (tales como una cámara). La velocidad predeterminada es de 250 MHz.
  - **v3d\_freq**-Establece la frecuencia de reloj del hardware de renderizado 3D de la GPU, para conseguir un aumento en el rendimiento de los juegos y la visualización. El valor predeterminado es de 250 MHz.
- **sdram\_freq**-Establece la velocidad de reloj del chip de la memoria de acceso aleatorio (RAM) sobre la RasPi para proporcionar a todo el sistema un pequeño aumento en el rendimiento. El valor predeterminado es de 400 MHz.
- **init\_uart\_clock**-Establece la velocidad de reloj predeterminada para el *Receptor/Transmisor Asíncrono Universal (UART)*, utilizado para controlar la consola serie. El valor predeterminado es de 3000000, el cual establece una velocidad de 3 MHz. La alteración de este valor probablemente tenga poco efecto más allá que el de corromper la salida de la consola serie.
- **init\_emmc\_clock**-Establece la velocidad de reloj predeterminada del controlador de la tarjeta SD. El valor predeterminado es de 80000000, el cual establece una velocidad de 80 MHz. Incrementar este valor puede

llevar a velocidades de lectura y escritura más rápidas de la tarjeta SD, pero puede conducir a la corrupción de datos.

Como un ejemplo, para overclockear la CPU a 800 MHz, la GPU a 280 MHz y la RAM a 420 MHz, introduzca las siguientes opciones dentro el archivo `config.txt`, una por línea:

```
arm_freq=800
gpu_freq=280
sdram_freq=420
```

Al igual que con la configuración de la pantalla, cualquier modificación realizada referente al overclocking no tendrá efecto hasta que la RasPi sea reiniciada. Para restablecer la configuración normal puede eliminar todo el archivo `config.txt` o (si además está utilizando este archivo para controlar las configuraciones de la pantalla) simplemente borre las líneas que tengan que ver con el overclocking y luego reinicie la RasPi.

Si ha overclockeado su RasPi y no puede arrancar, coloque la tarjeta SD en otra computadora para editar la configuración y luego pruebe de nuevo, o mantenga la tecla Shift presionada mientras RasPi reinicia para deshabilitar temporalmente sus nuevas configuraciones y la RasPi trabaje a su velocidad de reloj normal.

### Configuraciones de Sobretensión

Si está overclockeando su RasPi, es probable que golpee la pared después de que su dispositivo no se ponga en marcha. El punto exacto en la que la RasPi no será confiable para hacer overclock depende de cada dispositivo, debido a las variaciones naturales en el chip durante la fase de su fabricación. Para algunos usuarios, este límite puede ser tan bajo como 800 MHz; otros en cambio, podrán llevar a su RasPi tan alto como a 1 GHz (1,000 MHz) sin ningún problema.

Si desea sacarle un poco más de rendimiento a su RasPi, hay una manera para incrementar potencialmente este límite superior: un proceso conocido como *sobretensión* u *overvolting*. El procesador system-on-chip BCM2835 de la RasPi y el módulo de memoria asociado por lo general trabajan a 1.2 voltios. Es posible (aunque poco aconsejable) ignorar este valor predeterminado y forzar a los componentes a trabajar en voltajes más altos o más bajos. Aumentar el voltaje significa aumentar la señal dentro del chip, para tener así más probabilidades de llegar a velocidades más altas; esto también significa que el chip trabajará a más alta temperatura, reduciendo su vida útil drásticamente comparado a si se hubiera overclockeando solamente.

---

#### ADVERTENCIA

Ajustar cualquier de las opciones de voltaje en el archivo `config.txt` provoca que se funda un fusible en el BCM2835 de manera que no pueda restablecerse. Es un método infalible para saber si alguien ha estado intentando overclockear el chip fuera de sus especificaciones indicadas y *deja su garantía nula e inválida* (incluso si la causa del fallo no está relacionada con el overclock). Si regresa una RasPi para su reemplazo en garantía y el fusible ha cambiado, *no será reemplazado*. No trate de hacer sobretensión en una RasPi que no pueda darse el lujo de reemplazar.

---

A diferencia de las configuraciones descritas anteriormente, las cuales se han proporcionado dentro del archivo `config.txt` como valores absolutos, el ajuste del voltaje se lleva a cabo utilizando valores relativos al voltaje stock de 1.2 V de la RasPi. Para cada número entero superior a cero, el voltaje es incrementado en 0.025 V desde el voltaje stock. Para cada número entero por debajo de cero, el voltaje será sustraído en 0.025 V del voltaje stock.

Las configuraciones del ajuste del voltaje tienen límites superiores e inferiores de 8 y -16, equivalente a 0.2 V por encima del voltaje stock o 1.4 V absolutos y 0.4 por debajo del voltaje stock o 0.8 V absolutos. El voltaje debe ser ajustado en números enteros y no puede ser ajustado por debajo de 0.8 V (-16) o por encima de 1.4 V (8).

Los siguientes ajustes son accesibles desde `config.txt`:

- **over\_voltage**-Ajusta el voltaje principal del BCM2835. Los valores se proporcionan como un número

entero (entero sin decimal) correspondiendo a 0.025 V por encima o por debajo del valor por defecto (0,) con un límite inferior de -16 y un límite superior de 8.

- **over\_voltage\_sdram**-Ajusta el voltaje suministrado al chip de memoria de la RasPi. Al igual que **over\_voltage**, los valores se proporcionan como un número entero correspondiendo a 0.025 V por encima o por debajo del voltaje stock (0,) con un límite inferior de -16 y un límite superior de 8. Adicionalmente, puede ajustar los voltajes para cada uno de los componentes individuales de la memoria utilizando las siguientes opciones:
  - **over\_voltage\_sdram\_c**-Ajusta el voltaje suministrado al controlador de memoria. Los valores aceptados son los mismos que en **over\_voltage\_sdram**.
  - **over\_voltage\_sdram\_i**-Ajusta el voltaje suministrado al sistema de entrada/salida (E/S) de la memoria. Los valores aceptados son los mismos que en **over\_voltage\_sdram**.
  - **over\_voltage\_sdram\_p**-Ajusta el voltaje suministrado a los componentes de la capa física de la memoria (PHY). Los valores aceptados son los mismos que en **over\_voltage\_sdram**.

Como un ejemplo, las siguientes líneas introducidas en el archivo **config.txt** le darán al BCM2835 un pequeño aumento de 0.05 V a 1.25 V y al chip de memoria le darán un gran aumento de 0.1 V a 1.3 V:

```
over_voltage=2
over_voltage_sdram=4
```

Igual que con las otras configuraciones, si se eliminan las líneas del archivo **config.txt** o se borra el propio archivo regresará las cosas a la normalidad. Sin embargo, a diferencia de las otras configuraciones en esta sección, la evidencia permanecerá en la forma de un fusible fundido dentro del BCM2835, dejando inválida la garantía de la RasPi aun después de que los ajustes predeterminados sean restaurados.

## Desactivación de la Caché L2

El procesador SoC BCM2835 de la RasPi tiene 128 KB de *memoria caché nivel 2 (Level 2)* integrada. Aunque esta memoria es pequeña, es extremadamente rápida. Es utilizada para el almacenamiento temporal (o *caché*) de los datos y las instrucciones entre la lenta memoria principal y el propio procesador para mejorar el rendimiento.

Debido a los orígenes del BCM2835 como un procesador multimedia dirigido a los set-top boxes, esta caché L2 está diseñada para ser utilizado únicamente por la parte del GPU del chip. A diferencia de un procesador tradicional, la CPU no cuenta con algún caché L2 de su propiedad.

A través del archivo **config.txt** le puede decir al BCM2835 que permita a la parte del CPU acceder la memoria caché L2. En algunos casos, esto puede mejorar el rendimiento. En otros, puede perjudicar el rendimiento, debido a la ubicación física de la caché encontrándose a una distancia relativamente más lejos de la sección de la CPU del chip y más cercana a la del GPU.

El uso de la memoria caché L2 también requiere de una distribución Linux que haya sido compilada considerando el aprovechamiento de la memoria caché. Raspbian es una de tales distribuciones, y viene con la caché L2 habilitada para un mejor rendimiento. Debe mantenerla así y sólo deshabilitarla si ésta produce problemas con los sistemas operativos alternativos.

Para desactivar el acceso de la CPU a la caché L2, simplemente agregue la siguiente línea al archivo **config.txt**:

```
disable_l2cache=1
```

Al igual que con todas las configuraciones **config.txt**, el sistema debe ser reiniciado antes de que los cambios tengan efecto. Para activar el acceso de la CPU a la memoria caché, reemplace el 1 con 0.

## Activación del Modo de Prueba

Esta última opción en `config.txt` es una opción que la gran mayoría de los usuarios de la RasPi nunca tendrán que tocar, pero está incluida aquí para completar todo: el modo de prueba. Utilizado en la fábrica durante la producción de la Raspberry Pi, el modo de prueba (combinado con un hardware especial usado para verificar eléctricamente la placa) permite al personal de la fábrica asegurarse de que la RasPi está operando como debería.

---

### ADVERTENCIA

La activación del modo de prueba no provoca ningún daño permanente, pero la RasPi no podrá arrancar su sistema operativo hasta que el modo se desactive de nuevo y la fuente de alimentación sea desconectada y vuelta a conectar.

---

Si tiene curiosidad por ver lo que la RasPi muestra al personal de la fábrica, puede habilitar el modo de prueba introduciendo la siguiente opción en el archivo `config.txt`:

```
test_mode=1
```

Al igual que con otras configuraciones `config.txt`, el modo de prueba no se activará hasta que la RasPi sea reiniciada. El modo de prueba puede desactivarse de nuevo eliminando la línea en `config.txt`, borrando el archivo completamente o reemplazando el 1 con 0.

## Particionamiento de la Memoria

Aunque la Raspberry Pi tenga un chip de memoria de 256 MB o 512MB, esa memoria puede ser distribuida entre el hardware en varias formas. El BCM2835 está dividido en dos secciones principales: la sección de la CPU de propósito general y la sección de la GPU enfocada a los gráficos. Ambas secciones requieren de memoria para trabajar, lo que significa que los 256 MB o 512 MB de memoria en la Raspberry Pi necesitan compartirse entre las dos secciones.

La división típica es elegida por los responsables de la distribución Linux instalada sobre la RasPi. Algunos escogen proporcionar 128 MB a la GPU, asegurando así que el hardware de gráficos pueda funcionar a su máximo potencial. Otros permiten a la CPU obtener una parte más grande a fin de mejorar el rendimiento de propósito general.

Anteriormente, La división de la memoria era controlada utilizando diferentes versiones de un archivo de firmware llamado *start.elf*, en donde los distintos archivos se utilizaban para suministrar las cantidades de memoria diferentes a la CPU. Estos archivos ahora han sido reemplazados por una sola línea en el archivo `config.txt`, el cual puede ser editado para controlar la división de la memoria.

---

### ADVERTENCIA

Las aplicaciones que hacen trabajos gráficos pesados, tales como los juegos 3D y el software de reproducción de video en alta definición, por lo general necesitan 128MB de memoria para la GPU. Disminuir esta cantidad puede resultar en una caída dramática en el rendimiento. El Módulo de Cámara de la Raspberry Pi es incapaz de grabar video con menos de 128MB de memoria gráfica.

---

La división de la memoria es modificada editando el archivo `config.txt` como con las otras configuraciones hardware, editando o insertando la línea marcada `gpu_mem`. Esta línea le dice a la RasPi cuánto del total de la memoria (256 MB en el Modelo A, 512 MB en el Modelo B) debe proporcionar a la GPU, y el resto pasará a la CPU.

El valor puede ser establecido a 16 MB como mínimo, o 128 MB como máximo. La configuración debe ser ajustada en incrementos de 16 MB y escrito sin el sufijo MB (produciendo valores posibles de 16, 32, 48, 64, 80, 96, 112 y 128). Para dar a la GPU el mínimo de memoria, 16 MB por ejemplo, edite el archivo como sigue:

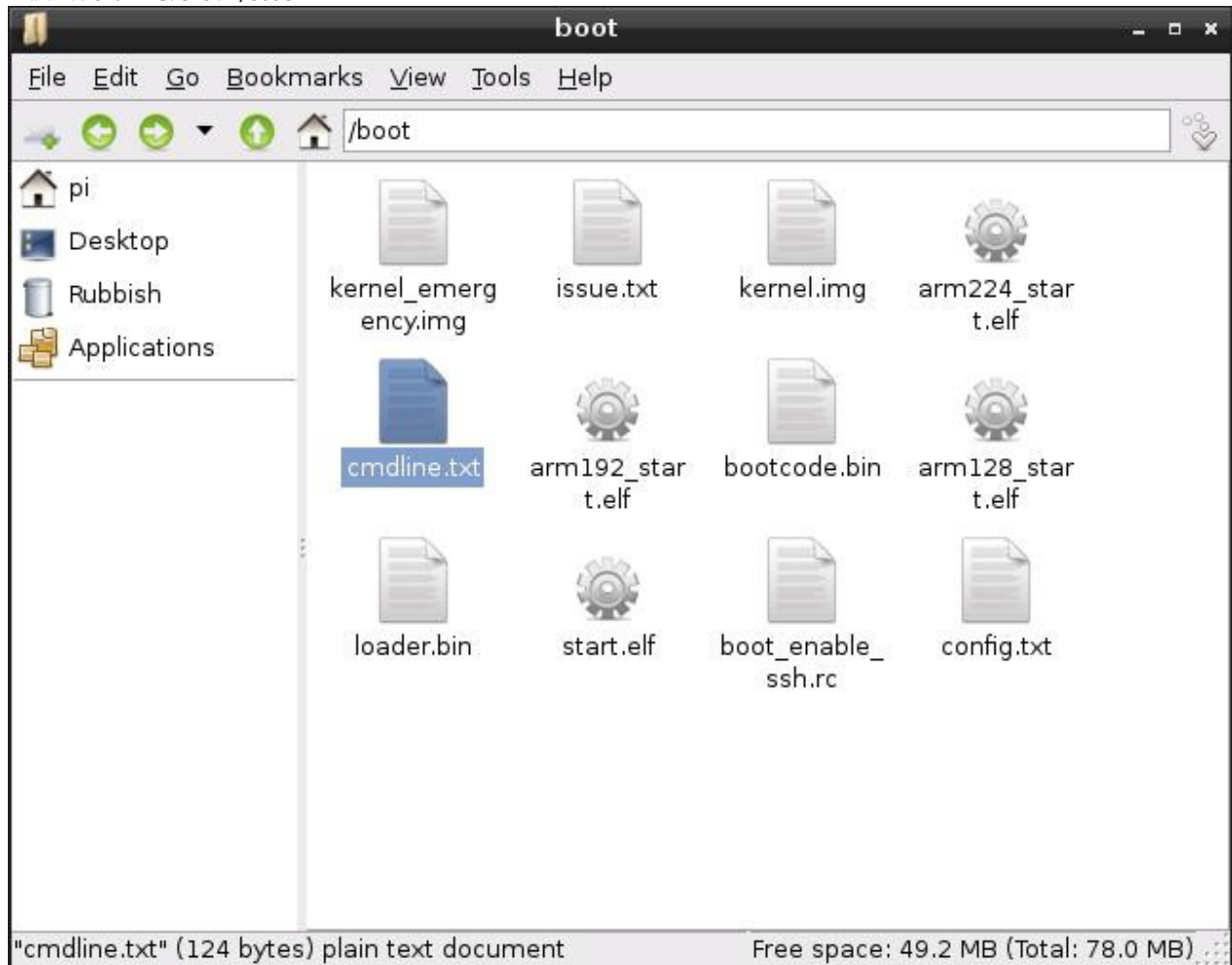
```
gpu_mem=16
```

## Configuraciones Software - cmdline.txt

Además del archivo `config.txt` que controla varias características del hardware de la RasPi, hay otro archivo de texto importante en el directorio `/boot`: `cmdline.txt` (ver Figura 7-4). Este archivo contiene lo que se conoce como la *línea del modo del núcleo* (las opciones pasadas al núcleo Linux con las que la RasPi se arrancará).

Figura 7-4

El archivo `cmdline.txt` en `/boot`



En una PC de escritorio o laptop basada en Linux, estas opciones se suelen pasar al núcleo por medio de una herramienta conocida como *gestor de arranque* (*bootloader*), el cual tiene su propio archivo de configuración. En la RasPi, estas opciones son simplemente introducidas directamente en el archivo `cmdline.txt` para que sean leídas por la RasPi durante el arranque.

Casi cualquier opción del núcleo soportada por Linux puede introducirse en el archivo `cmdline.txt` para modificar cosas tales como la apariencia de la consola o qué núcleo será cargado. Como un ejemplo, aquí está el archivo `cmdline.txt` de la distribución Debian, la cual debe ser escrita dentro del archivo en una sola línea continua:

```
dwc_otg.lpm_enable=0 console=ttyAMA0,115200 kgdboc=ttyAMA0,115200 console=tty1
root=/dev/mmcblk0p2 rootfstype=ext4 rootwait
```

La primera opción, `dwg_otc.lpm_enable`, le dice a la RasPi que deshabilite el modo *OTG (On-The-Go)* de su controlador USB, para prevenir problemas que pueden suceder cuando la funcionalidad está habilitada sin el soporte adecuado en el sistema operativo. La mayoría de las distribuciones para la RasPi deshabilitan este modo.

La opción `console` le dice a Linux que debe crear una consola serie (dispositivo `ttyAMA0`) y a qué velocidad debe de trabajar. En la mayoría de los casos, la velocidad debe ser dejada a su valor predeterminado de 115,200 bps (bits por segundo). Si la RasPi está siendo utilizada para comunicarse con viejos dispositivos, esta velocidad puede ser reducida en la forma que corresponda.

La opción del núcleo `kgdboc` habilita la depuración del núcleo Linux a través de la consola serie creada utilizando el parámetro `console`. Para la mayoría de los usuarios, esta opción es innecesaria. Para los desarrolladores, tener acceso a la depuración del núcleo por medio de una conexión serie resulta muy útil. Muchas distribuciones dejan esta opción habilitada por si acaso.

La segunda entrada `console` crea el dispositivo `tty1`, el cual es la pantalla que ve usted llena de texto al momento de arrancar por primera vez la RasPi. Sin esta entrada, no sería capaz de utilizar la RasPi sin tener que conectar algo a la consola serie creada por la primera opción `console`.

La opción `root` le dice al núcleo Linux en dónde puede encontrar su *sistema de archivos raíz* que contiene todos los archivos y directorios necesarios para que el sistema funcione. En la distribución Raspbian por defecto, el *sistema de archivos raíz* se encuentra en la segunda partición de la tarjeta SD (dispositivo `mmcblk0p2`). Esta opción puede modificarse para apuntar hacia un dispositivo de almacenamiento externo conectado al puerto USB, el cual puede acelerar considerablemente el funcionamiento de la RasPi en comparación a tener el sistema de archivos raíz almacenado en la tarjeta SD.

Además de indicarle al núcleo en dónde puede encontrar su sistema de archivos raíz, también requiere saber en qué formato fue creada la partición. Debido a que Linux soporta una gran variedad de sistemas de archivos, la opción `rootfstype` le dice expresamente a la distribución Raspbian que utilice un sistema de archivos EXT4.

Por último, el parámetro `rootwait` le dice al núcleo que no debe de tratar de ir más lejos con el arranque del sistema hasta que el dispositivo que contiene el sistema de archivos raíz se encuentre disponible. Sin esta opción, la RasPi puede quedar atorada ya que comenzaría a arrancar antes de que la tarjeta SD (relativamente lenta) se encuentre completamente lista para ser accedida.

Ninguno de estos parámetros del núcleo son únicos de la RasPi (con la excepción de `dwc_otg`). La configuración del gestor de arranque de cualquier distribución Linux incluiría una lista de opciones muy similar a la del archivo `cmdline.txt`.

Por lo general, no debería tocar el archivo `cmdline.txt`. Este archivo es creado por los mantenedores de la distribución específicamente para aquella distribución de Linux y puede ser diferente en la próxima versión. Las entradas que funcionan para Pidora podrían no funcionar con Raspbian y viceversa. Las opciones disponibles para `cmdline.txt` dependen sobre qué núcleo la distribución está utilizando y qué características fueron incluidas cuando el núcleo fue compilado.

Si es usted un desarrollador del núcleo, puede utilizar `cmdline.txt` para pasar parámetros que habiliten o deshabiliten la nueva funcionalidad que ha compilado dentro del núcleo. Igual que con el archivo `config.txt`, cualquier cambio requiere de un reinicio para que tome efecto.

Parte II

# **Contruyendo un Centro Multimedia, Máquina Productiva o Servidor Web**

- Capítulo 8**      La RasPi como una PC Teatro en Casa
- Capítulo 9**      La RasPi como una Máquina Productiva
- Capítulo 10**    La RasPi como un Servidor Web





# Capítulo 8

La RasPi como una PC Teatro en Casa

**Una de las** tareas más populares que una RasPi puede cumplir es la de una PC Teatro en Casa o HTPC. El Broadcom BCM2835 en el interior de la RasPi está específicamente diseñado como un poderoso centro multimedia, desarrollado originalmente para ser utilizado en las HTPCs.

La parte gráfica del diseño system-on-chip (SoC) BCM2835, un módulo Broadcom VideoCore IV, tiene la capacidad de reproducir video de alta definición a toda velocidad empleando el popular formato H.264. El chip también es capaz de reproducir archivos de audio en diferentes formatos, tanto por la salida analógica (3.5 mm) como por la salida digital (HDMI).

El pequeño tamaño de la RasPi, su bajo consumo de energía y su funcionamiento silencioso lo convierten en un dispositivo muy tentador para los aficionados de los teatros en casa. Desde su lanzamiento han aparecido una variedad de distribuciones y paquetes de software diseñados para convertir a la RasPi en una PC teatro en casa fácil de utilizar, pero esto no significa necesariamente que tenga que abandonar su sistema operativo actual para disfrutar de las capacidades multimedia de la RasPi.

## Reproduciendo Música desde la Consola

Si es usted un desarrollador, probablemente pase la mayor parte de su tiempo trabajando dentro de la consola de la RasPi. Con la mayoría del software de reproducción de música enfocados a funcionar dentro de un entorno de escritorio, la experiencia de trabajar dentro de la consola puede ser muy silenciosa (pero no tiene por que serlo).

La RasPi soporta un paquete de reproducción de música (con interfaz de texto) muy potente llamado **moc** (el cual significa *music on console* / *música en la consola*). A diferencia de otras herramientas (como **LXMusic**), **moc** puede instalarse y utilizarse incluso si no cuenta con una interfaz gráfica de usuario (GUI) instalada sobre su RasPi.

Para comenzar, instale el paquete **moc** desde los repositorios de su distribución. Para la distribución Raspbian, esto es tan sencillo como escribir el siguiente comando en la consola o en una ventana de terminal:

```
sudo apt-get install moc
```

---

### CONSEJO

Algunas distribuciones ya cuentan con otra herramienta llamado **moc**, que nada tiene que ver con la reproducción de audio. Si la instalación de **moc** no da el resultado que esperaba, sustituya **mocp** como el nombre del paquete.

---

Si pasa mucho tiempo fuera de la interfaz gráfica de usuario trabajando desde la consola, **moc** es una buena opción para escuchar música. A diferencia de otras herramientas, **moc** se ejecuta en un segundo plano, lo que significa que su música no será interrumpida si comienza a hacer cualquier otra cosa.

Para cargar **moc**, el comando será **mocp** en lugar de **moc**. La razón para esto es que ya existe otra herramienta que utiliza el comando **moc**, por eso se eligió un nombre distinto para evitar que el sistema operativo se confunda entre los dos paquetes.

Para iniciar **moc**, basta con introducir en la consola (o en una ventana de terminal si está usando un entorno de escritorio) lo siguiente:

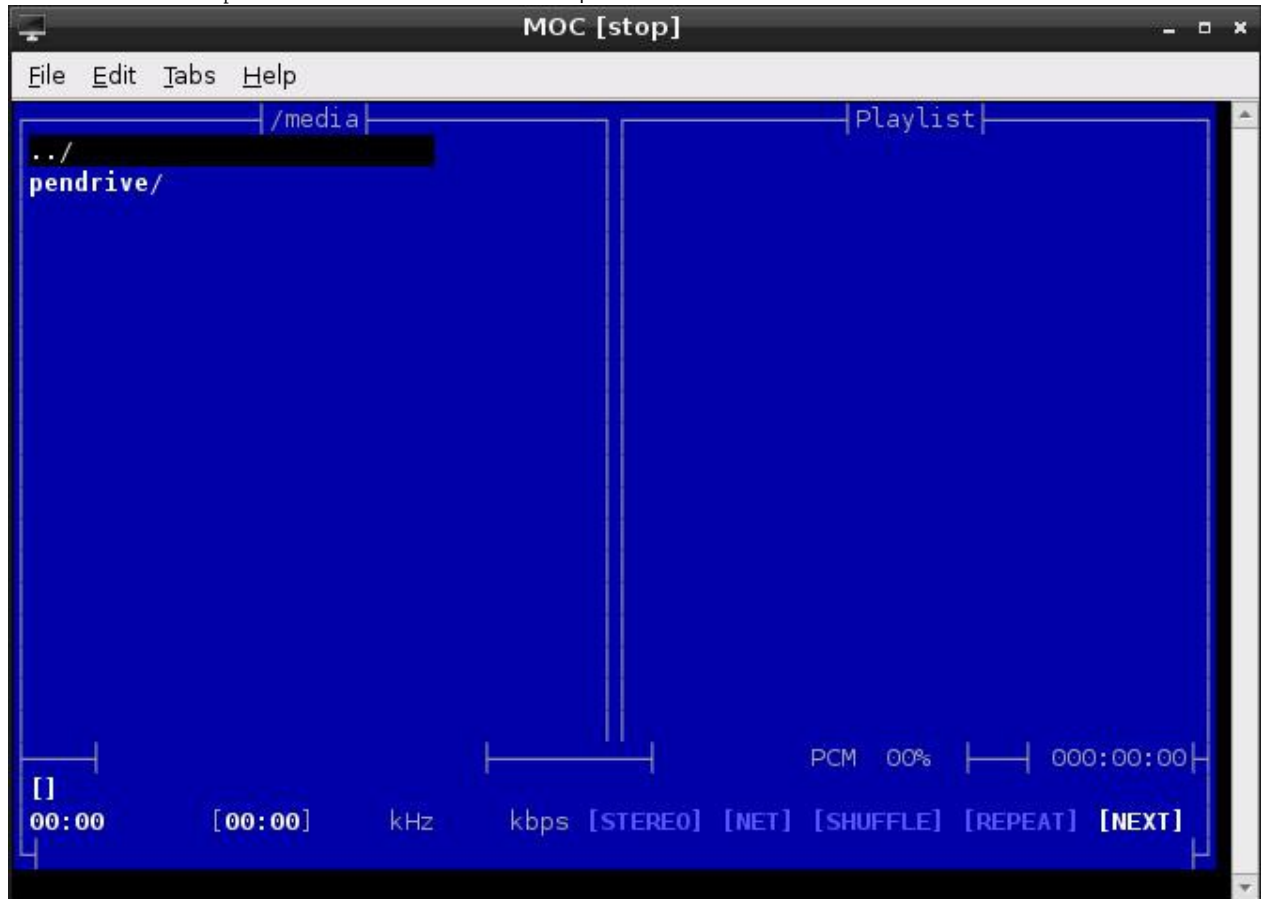
```
mocp
```

La interfaz estándar de **mocp** se encuentra dividida en dos paneles (ver Figura 8-1). El panel de la izquierda es un explorador de archivos, que le permite buscar la música a reproducir. Las teclas de cursor desplazan hacia arriba y hacia abajo la lista, mientras que la tecla Intro inicia la reproducción de la canción que está actualmente seleccionada. Si presiona Intro sobre el nombre de un directorio, el explorador entrará al mismo, mientras que

presionarlo sobre la entrada `../` de la parte superior de la lista lo llevará de regreso a un nivel superior de directorio. El panel de la derecha muestra la lista de reproducción actual.

Figura 8-1

La interfaz estándar del reproductor de música basado en consola `mocp`



Donde la potencia de `mocp` se hace evidente es cuando se sale de la aplicación al presionar la tecla `Q`. Si `mocp` se encuentra en medio de la reproducción de música, éste continuará haciéndolo incluso cuando utilice la consola o ventana de terminal para otras actividades. Al ejecutar de vuelta el comando `mocp` se restaurará su interfaz, permitiéndole cambiar canciones, pausar o detener la reproducción. También es posible controlar directamente a `mocp` desde la terminal, sin tener que utilizar su interfaz. Ejecutando el comando `mocp` con banderas (las opciones que siguen a un comando, con un carácter de guión como prefijo) le permitirán iniciar, detener, pausar, avanzar o cambiar la reproducción sin tener que entrar al software.

Las banderas más utilizadas para `mocp` son las siguientes:

- **-s**-Detiene la reproducción actual
- **-G**-Pausa la reproducción o la reanuda si se encuentra actualmente pausada
- **-f**-Salta a la siguiente canción en el directorio o la lista de reproducción
- **-r**-Vuelve a la canción anterior en el directorio o lista de reproducción
- **-i**-Imprime la información sobre la canción actual en la terminal o consola
- **-x**-Detiene la reproducción y sale por completo de `mocp`

Para obtener más información sobre cómo controlar `mocp`, escriba `man mocp`.

## HTPC Dedicado con Raspbmc

Ser capaz de reproducir música en la Raspberry Pi es algo bueno, salvo que la BCM2835 tiene la capacidad de hacer mucho más que eso. Empleando su GPU VideoCore IV, puede decodificar y reproducir video H.264 FULL HD 1080p, transformando a la RasPi en un poderoso centro multimedia de diminutas dimensiones y con una demanda de energía increíblemente baja.

Sin embargo, para sacar provecho de la RasPi como una PC teatro en casa, es necesario algo de software adicional. Este software puede ser instalado dentro de la distribución Raspbian, pero hay una forma más sencilla de empezar: cambiando a la distribución *Raspbmc*.

Raspbmc, es una distribución creada por Sam Nazarko dirigida específicamente a transformar la Raspberry Pi en un centro multimedia con todas sus características, con reproducción de audio y video, visualización de fotos y capacidades de streaming sobre Internet. Se basa en la popular distribución Xbmc, que ha sido seleccionada por varios fabricantes para potenciar sus sistemas set-top boxes comerciales.

Si está pensando hacer uso de la salida de video de alta definición y las capacidades de decodificación H.264 de la RasPi en su sistema de teatro en casa, la distribución Raspbmc es una excelente opción y es tan fácil ponerla en funcionamiento. Primero, descargue el instalador desde el sitio oficial en <http://www.raspbmc.com/download/>. Esta es la manera más fácil de conseguir y poner a Raspbmc en funcionamiento. Los instaladores para Linux, OS X y Windows están dispuestos para que automáticamente descarguen el archivo de imagen de Raspbmc y lo graben a una tarjeta SD conectada al equipo. Aunque también puede descargarse manualmente el archivo de imagen y utilizar las instrucciones del Capítulo 1, “Conozca la Raspberry Pi”, no es necesario hacerlo.

---

### ADVERTENCIA

Si ya cuenta con una tarjeta SD que utiliza con su RasPi, tenga cuidado, pues la instalación de Raspbmc eliminará el contenido de la tarjeta. Respalde los archivos que desee mantener o si desea poder cambiar entre las dos distribuciones, compre una segunda tarjeta SD específicamente para su uso con Raspbmc o instálelo junto con Raspbian utilizando la herramienta NOOBS descrita en el Capítulo 2, “Comenzando con la Raspberry Pi”.

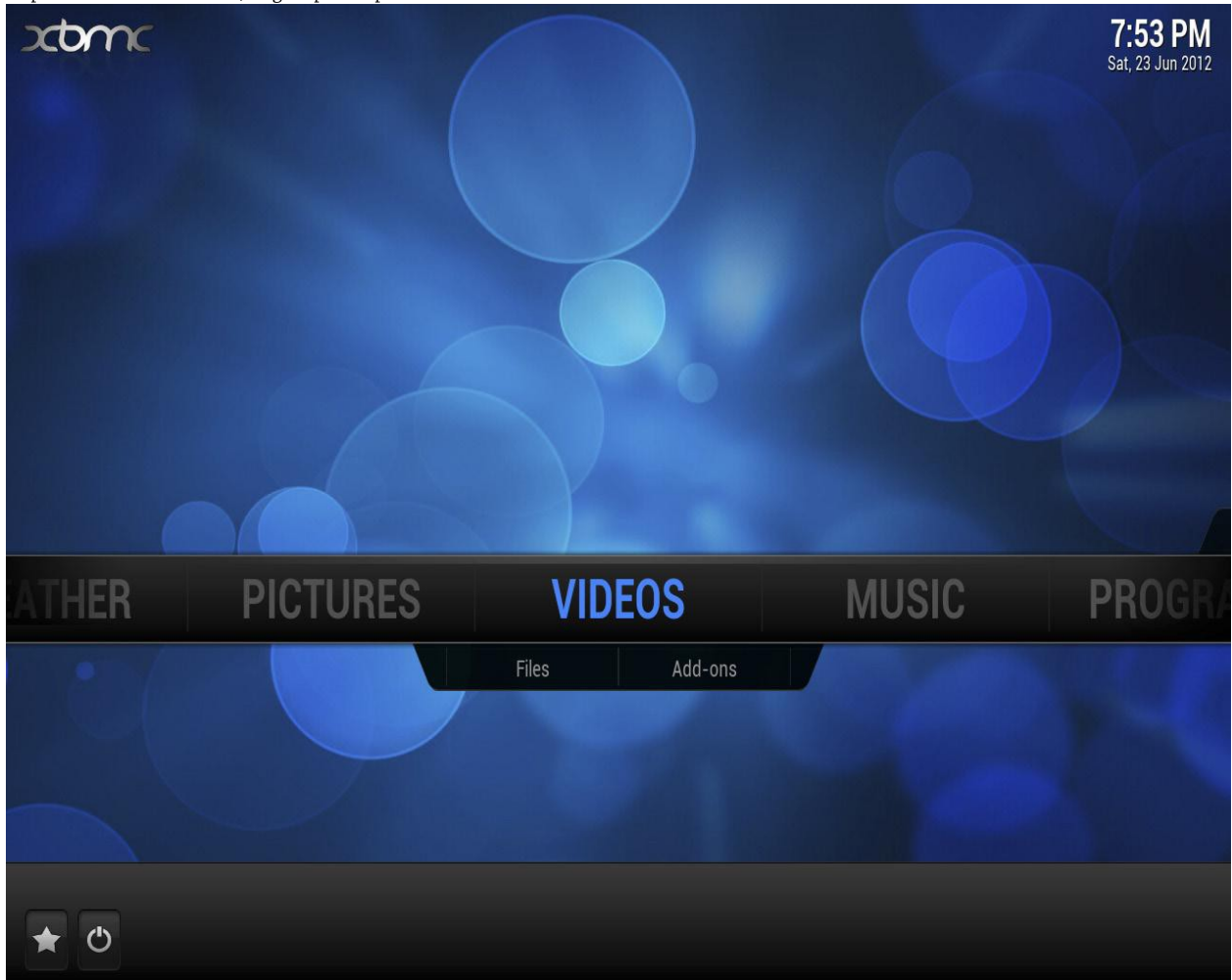
---

Cuando el instalador de Raspbmc haya finalizado, inserte la tarjeta SD en la RasPi y vuelva a conectar la fuente de alimentación, pero asegúrese de que el cable Ethernet esté también conectado, porque Raspbmc necesitará descargar algunos datos de la Internet cuando se carga por primera vez. Esta carga inicial de Raspbmc puede tomar entre 10 o 15 minutos en completarse, ya que se descargan actualizaciones y se crean particiones sobre la tarjeta SD, pero las cargas posteriores serán significativamente más rápidas. No se preocupe si en el primer arranque la RasPi parece haberse colgado en la etapa del formato de las particiones (una pausa prolongada aquí es normal).

Cuando se ha cargado, Raspbmc automáticamente inicia el servicio Xbmc (ver Figura 8-2). Este proporciona una interfaz de usuario diseñada específicamente para utilizarse en la sala de su casa. Todo es accesible a través del teclado o ratón, con grandes textos fáciles de leer y menús dispuestos en categorías para encontrar las cosas más fácilmente. Además, puede comprar controles remotos de infrarrojos, los cuales vienen con un receptor que se conecta al puerto USB de la RasPi y un transmisor, que le permiten disfrutar de una verdadera experiencia de tener un teatro en casa sin estorbosos teclados o largos cables.

Figura 8-2

La pantalla de inicio de Xbmc, cargada por Raspbmc

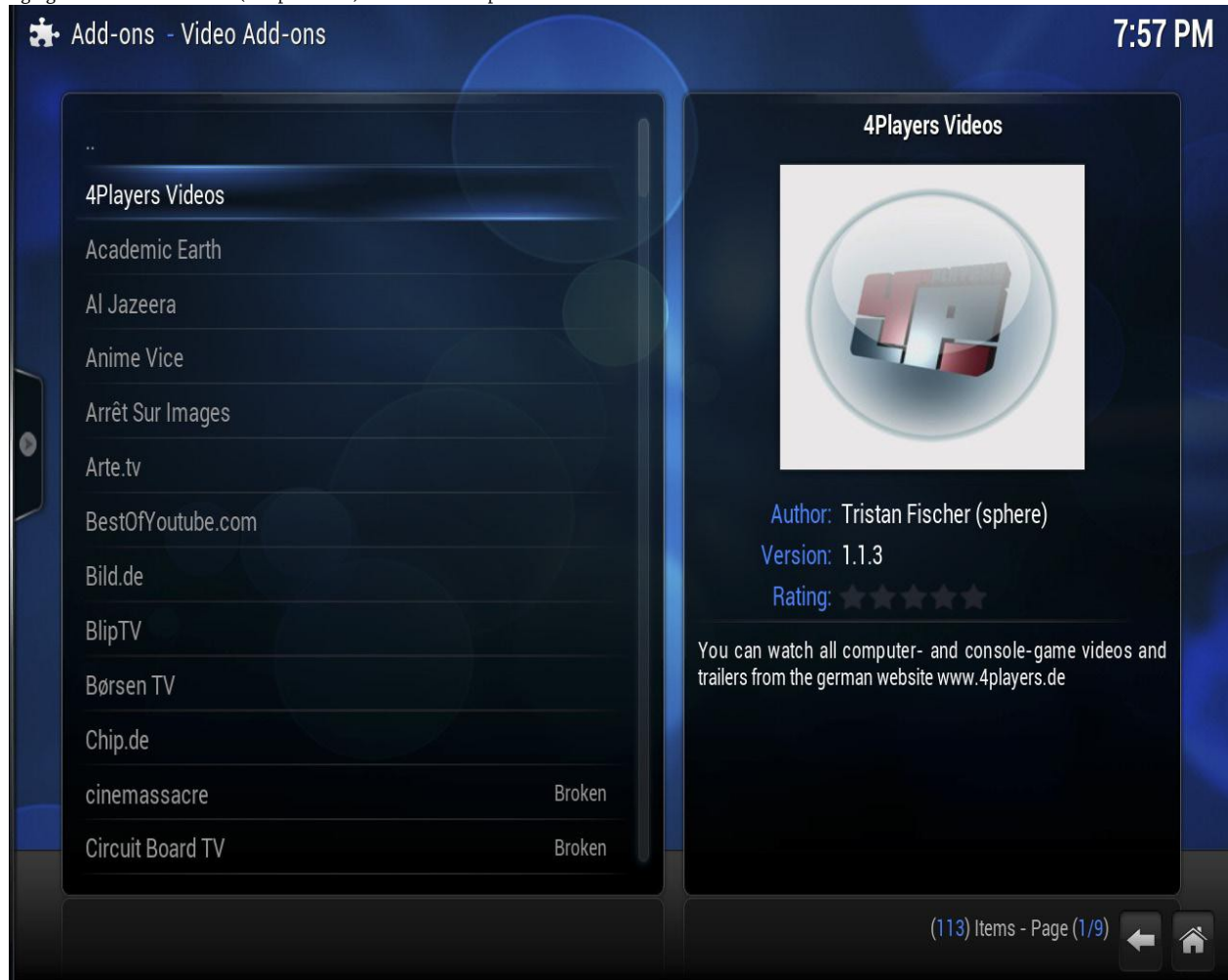


## Streaming Multimedia por Internet

Por defecto, Xbmc está configurado para reproducir sólo los archivos que se encuentran en la propia Raspberry Pi. Si selecciona la opción Add-ons que se encuentra debajo del menú Videos, podrá agregar al dispositivo algunas capacidades impresionantes de streaming de Internet, incluyendo el acceso a varios canales de televisión y servicios de streaming exclusivos de Internet. Después de hacer clic en Add-ons, seleccione Get More para acceder a la lista completa de los plug-ins compatibles (ver Figura 8-3).

Figura 8-3

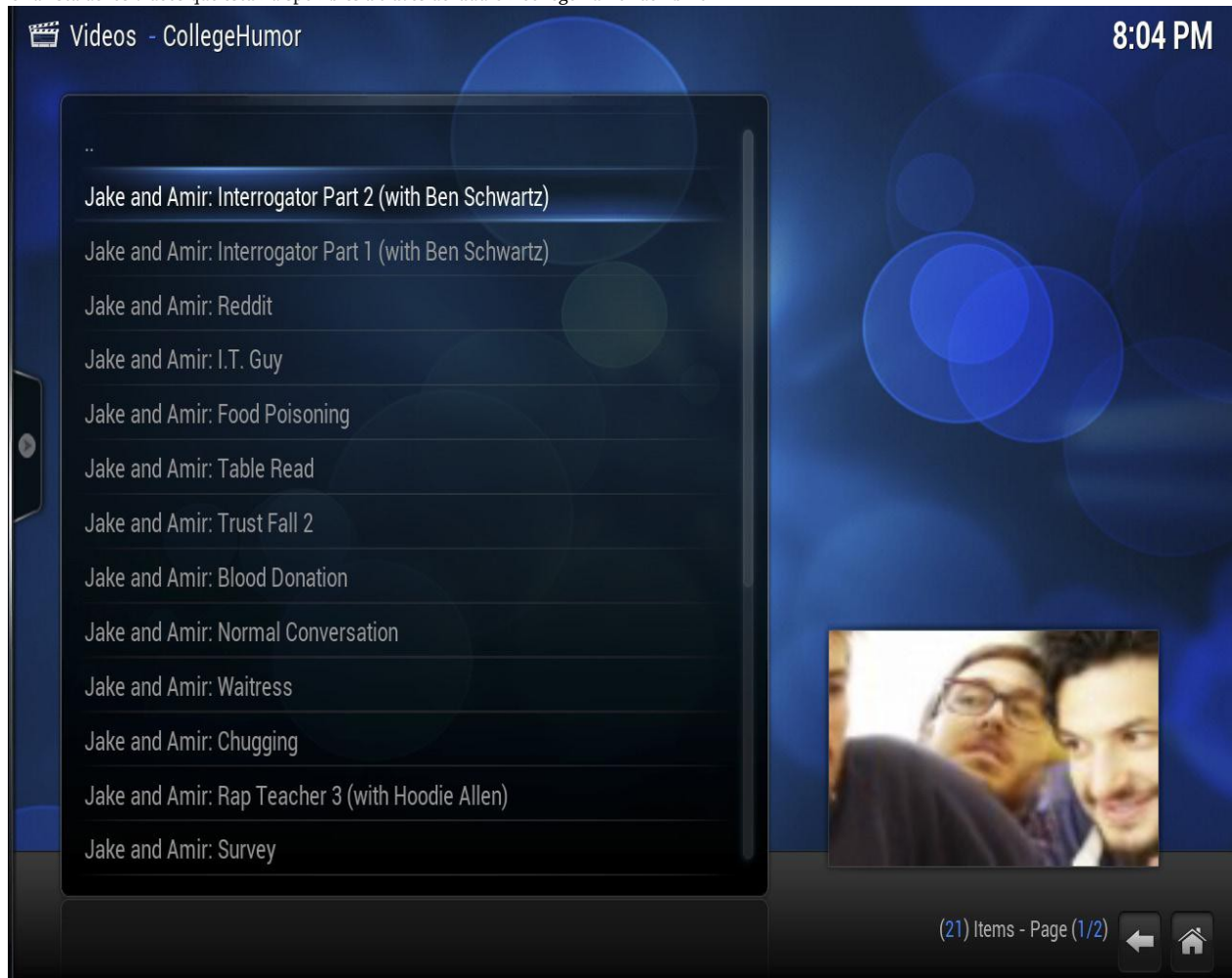
Agregando un nuevo add-on (complemento) de video en Raspbmc



Desplácese por la lista con el ratón o las teclas de cursor y haga clic en alguna entrada o presione la tecla Intro para acceder a más información. Para instalar un add-on, sólo haga clic en Install en el cuadro de información que apareció cuando hizo clic sobre alguna entrada de la lista. De esta manera se descargará el add-on desde Internet y automáticamente se instalará en Xbmc. Tenga cuidado con los add-ons que se encuentran listados como Broken ya que estos han sido reportados por no funcionar apropiadamente y no deben de instalarse hasta que sus problemas hayan sido corregidos por el desarrollador del add-on.

Después de que termine de seleccionar e instalar los add-ons de video, haga clic en el botón Home sobre la esquina inferior derecha de la pantalla para volver a la interfaz principal de Xbmc. Ahora haga clic sobre la palabra Video en el centro de la pantalla (o presione Intro) y luego elija Video Addons en las opciones que aparecen. Esto le permite acceder a sus add-ons instalados. Haga clic en alguno para descargar una lista de archivos que puede visualizar. Si el add-on tiene varias categorías, éstas se descargarán primero, en este caso, haga clic en una categoría individual para ver los archivos que contiene (ver Figura 8-4).

**Figura 8-4**  
Una lista de los videos que están disponibles a través del add-on CollegeHumor de Xbmc



Bajo los menús Music y Videos se encuentran disponibles otros Add-ons que funcionan de la misma forma. Con el uso de estos add-ons, puede ver los contenidos de fotos y escuchar el stream de audio de sitios como Flickr, Picasa, The Big Picture, Grooveshark, Sky.fm y SoundCloud.

## Streaming Multimedia por Red Local

El software Xbmc soporta el estándar de streaming multimedia *UPnP* (*Universal Plug and Play*), que le permite interconectar una amplia variedad de dispositivos en su red doméstica. El soporte UPnP se encuentra presente en los teléfonos móviles, consolas de videojuegos y dispositivos *NAS* (*Network-Attached Storage*) más modernos y hace posible que los dispositivos puedan compartir y hacer streaming de archivos videos, música y fotografías. Muchas laptops y PCs de escritorio también cuentan con el soporte para UPnP o vienen con el estándar *DLNA* (*Digital Living Network Alliance*) estrechamente vinculado a UPnP (revise su documentación para saber cómo habilitar esta característica en sus dispositivos).

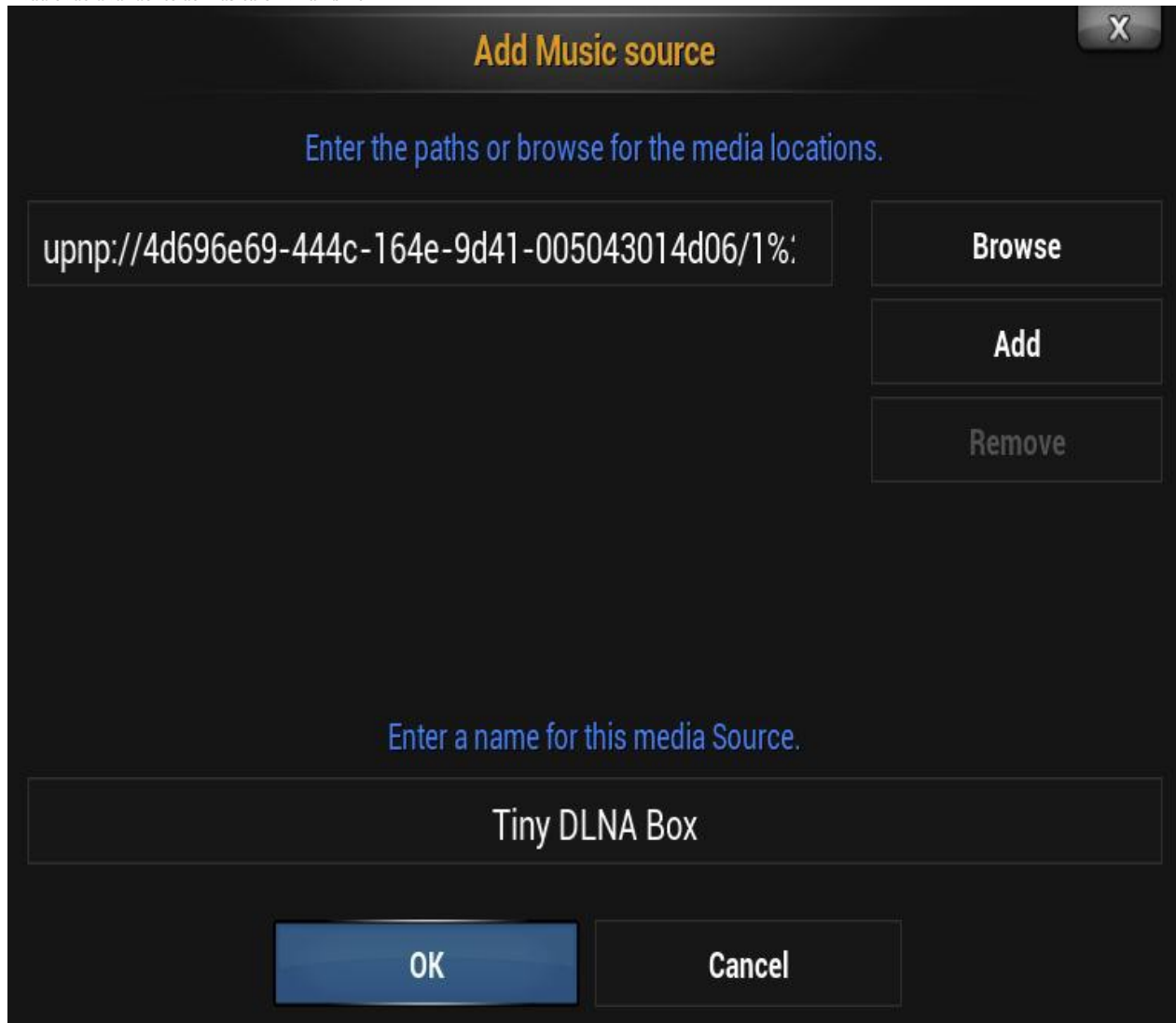
Xbmc no se limita únicamente a las conexiones UPnP, sino que también puede conectarse a los servidores de red que operan con el estándar *NFS* (*Network File System*) común en los sistemas basados en Unix, el estándar *SMB* (*Server Message Block*) común para los servidores Windows y el estándar *Zeroconf* presente en los dispositivos OS X. Sin importar cuál dispositivo conectado a la red está usted utilizando para almacenar su contenido multimedia, es muy probable que Xbmc soportará al menos un forma de poderse conectar a su dispositivo.







Figura 8-6  
Añadiendo una fuente de música UPnP a Xbmc



También es posible utilizar el mismo menú para añadir un disco duro externo como una fuente para Xbmc, simplemente seleccione su entrada desde la lista inicial. La mayoría de las unidades externas aparecerán automáticamente y no necesitan ser añadidas explícitamente (sólo es necesario añadir una unidad como una fuente si sus contenidos no aparecen en los menús de Xbmc).

## Configurando de Raspbmc

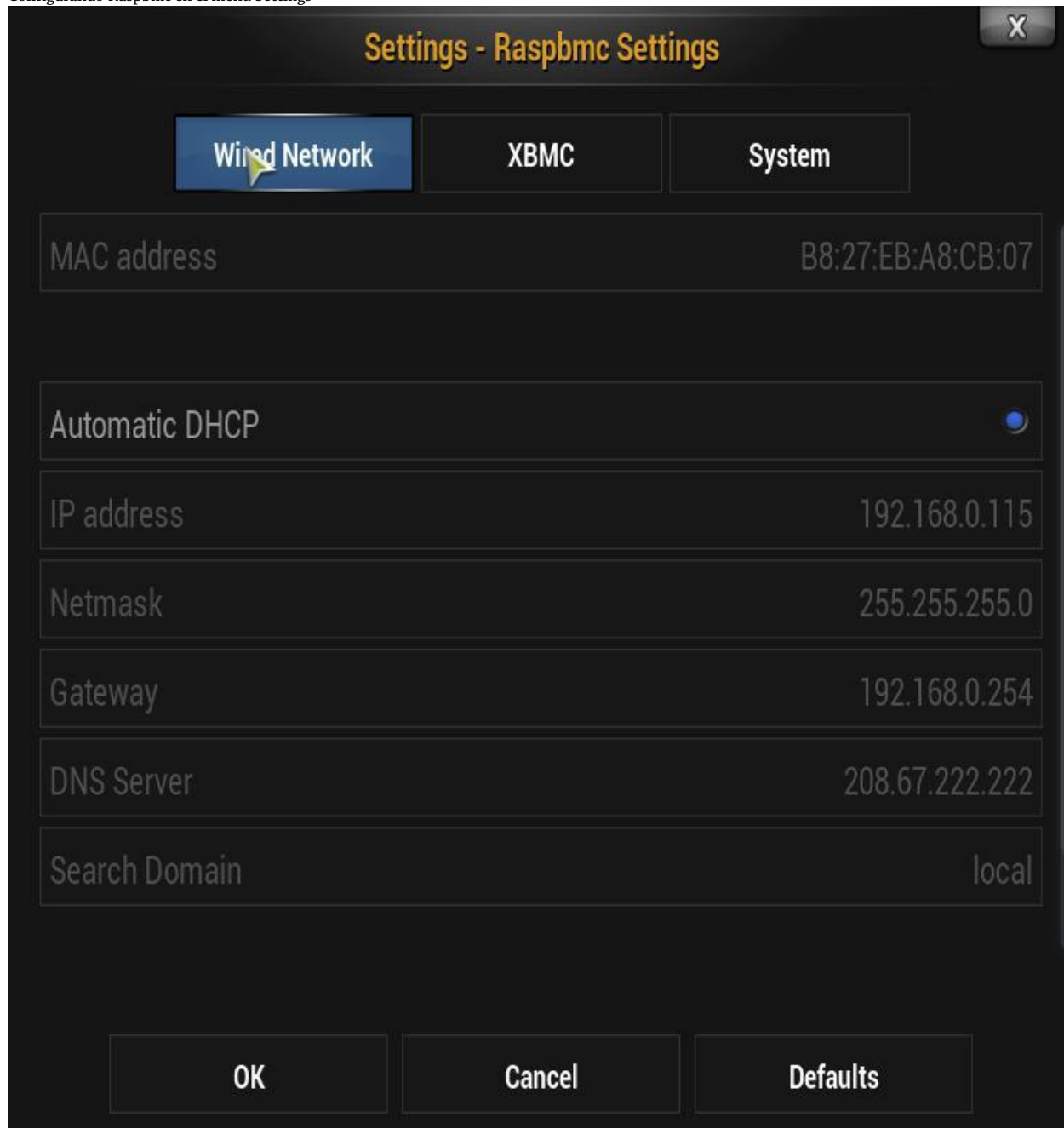
El menú Programs de la pantalla de inicio le permite acceder al menú Raspbmc Settings (Figura 8-7) donde las configuraciones para la red, las actualizaciones de software y las configuraciones diversas del sistema se pueden modificar. Normalmente, este menú puede ignorarse, sin embargo, si está experimentando problemas para conectarse a Internet o su sistema Raspbmc está fallando con frecuencia, el menú Programs puede ofrecerle herramientas importantes para resolver estos problemas.

Al hacer clic en la opción Raspbmc Settings que está debajo del menú Programs se abre una ventana con tres pestañas. La primera pestaña le permite ajustar la configuración de la red, incluyendo la desactivación del DHCP y el ajuste manual de la configuración de red (ver Capítulo 5, "Configuración de la Red", para más detalles). Si requirió

establecer la configuración de red de su Raspberry Pi de forma manual en otras distribuciones, este es el lugar indicado para hacer lo mismo pero en Raspbmc.

Figura 8-7

Configurando Raspbmc en el menú Settings



La segunda pestaña, etiquetada con XBMC, le permite instalar la *nightly build* (la compilación de todas las noches) del software XBMC. Es llamada *nightly build* porque es creada automáticamente desde el código fuente del programa de cada noche y contiene las últimas modificaciones hechas por los desarrolladores del software. Por consiguiente, este software puede tener corregidos muchos errores con los que se pudo haberse encontrado (pero también puede tener modificaciones no probadas que pueden hacer que las cosas empeoren). En esta misma pantalla se le permite cambiar entre las distintas versiones de XBMC, por lo que resulta fácil probar la “nightly built” y volver a la versión por defecto en caso de que llegará a encontrarse con algún problema.

Por último, la pestaña System le permite establecer una nueva contraseña para la cuenta de usuario *pi* y así pueda proteger el acceso a Raspbmc sobre la red. De igual forma que con la distribución Raspbian de propósito general, esta contraseña se encuentra establecida a *raspberry* por defecto. La pestaña System también le permite deshabilitar la actualización automática del software Raspbmc. Debe dejarla habilitada siempre que sea posible, ya que esto le garantizará poder recibir de forma automática las últimas correcciones de errores y de seguridad liberadas, pero si se encuentra trabajando con una conexión de Internet que se mide por megabytes consumidos, puede deshabilitar esta característica.

# Capítulo 9

## La RasPi como una Máquina Productiva

**La flexibilidad que** tiene la Raspberry Pi la convierte en una buena opción para ser utilizada como una PC de escritorio de propósito general de bajo consumo energético. Aunque nunca alcanzará los mismos niveles de rendimiento que una PC de escritorio o laptop estándar, su bajo costo y consumo de energía amigable con el medio ambiente ayudan a resolver problemas que en veces son posibles empleando un modesto rendimiento.

Aunque la distribución Raspbian de la Raspberry Pi que se proporciona en el sitio oficial no incluye algunos de los acostumbrados programas de productividad que se podrían esperar en una PC de propósito general (como el procesador de textos, la hoja de cálculo o el editor de imágenes), estos programas pueden ser instalados a través del comando `apt-get`. También es posible evitar la instalación del software de manera local y emplear los paquetes de software basado en la nube mediante el navegador web, el cual puede ofrecer un mejor rendimiento que aquellos paquetes que son instalados de manera local a costa de perder flexibilidad y algunas funcionalidades avanzadas.

Empleando cualquiera de los métodos aquí descritos en este capítulo (aplicaciones instaladas localmente o servicios basados en la nube) podrá conseguir que la RasPi pueda utilizarse como una máquina de uso cotidiano para la oficina y el trabajo escolar, sin perjudicar su utilidad como una plataforma para la programación y la experimentación.

---

**CONSEJO**

Si está planeando utilizar la RasPi como una máquina de productividad pura sería una buena idea reservar más memoria para el uso de propósito general y menos para el procesador gráfico. Para aprender cómo modificar esta división de memoria, vea el Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi”.

---

## Utilizando Aplicaciones Basadas en la Nube

Si tiene su RasPi conectada a Internet la mayor parte del tiempo, ya sea a través del puerto Ethernet integrado del Modelo B o con algún adaptador USB inalámbrico conectado al Modelo A, el software *basado en la nube* le ofrece una poderosa y aún ligera forma de utilizar el software de ofimática sobre la RasPi.

El software basado en la nube es llamado así porque éste no vive localmente en su computadora como una pieza normal de software. En vez de eso, el software se almacena sobre poderosos servidores ubicados en centros de datos por todas partes del mundo y son accedidos a través de la Internet utilizando un navegador web. Aprovechando las capacidades de procesamiento y almacenamiento de un servidor remoto más poderoso, es posible conseguir que la RasPi trabaje con documentos y tareas más complicadas sin disminuir su velocidad.

El software basado en la nube también tiene algunas otras ventajas sobre las aplicaciones instaladas de forma local. Cualquier aplicación basada en la nube se verá igual sobre cualquier dispositivo y muchas de estas aplicaciones incluyen versiones orientadas hacia los móviles diseñadas para ser accedidas por los smartphones o las tablets. Además, los archivos se almacenan sobre los servidores remotos, volviéndolos accesibles desde cualquier dispositivo y sin ocupar ningún espacio sobre la tarjeta SD de la RasPi.

No obstante, las aplicaciones basadas en la nube no son perfectas. Por lo general se quedan unos pasos atrás en cuanto a funcionalidad se refiere frente a sus contrapartes instaladas localmente y a menudo carecen de características avanzadas o soportan pocos formatos de archivo. Además son inaccesibles cuando no hay una conexión disponible a Internet, convirtiendo a estas aplicaciones en una mala elección para los usuarios con conexiones poco seguras.

Si piensa que la mejora de rendimiento y el ahorro de espacio en su tarjeta SD de la RasPi valen la pena, continúe leyendo. Si no, salte a la siguiente sección de este capítulo para aprender cómo instalar LibreOffice.org, una suite de oficina de código abierto equivalente a Microsoft Office.

Las suites más populares de oficina basadas en la nube son las siguientes:

- **Google Drive**-Puesta en funcionamiento por el gigante de la búsqueda y la publicidad Google, Google Drive (anteriormente conocido como Google Docs) incluye un procesador de textos, una hoja de cálculo y

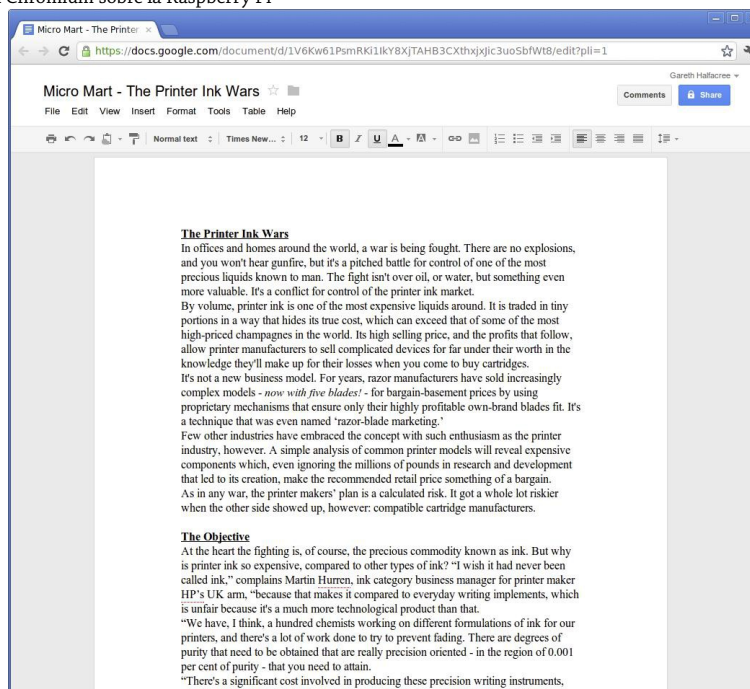
una herramienta de presentaciones (ver Figura 9-1). Los usuarios corporativos pueden registrarse con una cuenta Google Apps, la cual proporciona funcionalidad mejorada. Si dispone de una cuenta de correo electrónico basado en la web de Gmail, su cuenta automáticamente funcionará con Google Drive. Puede acceder al servicio en <http://docs.google.com>.

- **Zoho**-Con cinco millones de usuarios registrados, Zoho es otra opción muy popular. Al igual que Google Drive incluye un procesador de textos, hoja de cálculo y el paquete de presentaciones, pero Zoho además ofrece características mejoradas enfocadas a los negocios tales como una base de conocimiento basada en la wiki, gestión financiera e incluso la gestión de relaciones con los clientes. Muchas de las características avanzadas, no obstante, requieren de una cuenta de pago. Puede acceder al servicio en <http://www.zoho.com>.
- **Office 365**-Si es un usuario de Microsoft Office, Office 365 es una gran opción. Basada sobre la misma interfaz de usuario de las actuales ediciones de la suite de Microsoft Office para los equipos de escritorio, Office 365 es poderoso y flexible. A diferencia de Zoho y Google Drive, Office 365 no cuenta con un nivel de usuario gratis y requiere de una suscripción mensual. Además, algunas características no funcionan cuando el software es accedido desde una computadora Linux. Puede suscribirse al servicio en <http://office365.microsoft.com>.
- **ThinkFree Online**-Una interfaz basada en la web para el software Hancom ThinkFree Office, ThinkFree Online ofrece el software de procesador de textos, hoja de cálculo y presentaciones de manera gratuita con 1 GB de almacenamiento. El sistema también se vincula a ThinkFree Mobile para las tablets y los smartphones, así como con el software empresarial ThinkFree Server. Puede suscribirse al servicio en <http://online.thinkfree.com>.

Desafortunadamente, muchos de estos servicios basados en la Web requieren de un navegador más capaz que Midori, el cual viene en la distribución Raspbian recomendada como el navegador web predeterminado. Por consiguiente, tendrá que instalar un navegador distinto a fin de poder utilizar alguno de los anteriores paquetes. Las siguientes instrucciones son para instalar el navegador Chromium, el cual es un proyecto de código abierto que se basa en el navegador Chrome de Google. La relativamente poca memoria que utiliza convierte a Chromium en una buen opción para la RasPi. Si está utilizando la distribución Pidora, la cual viene con Mozilla Firefox como su navegador predeterminado, puede saltarse estas instrucciones y proceder a la siguiente sección.

Figura 9-1

Google Drive ejecutándose en Chromium sobre la Raspberry Pi



Para instalar el navegador Chromium bajo Raspbian, abra una terminal y escriba lo siguiente:

```
sudo apt-get install chromium-browser
```

---

**CONSEJO**

Asegúrese de instalar el paquete chromium-browser, y no el paquete chromium (este último es un juego arcade shoot-'em-up, y aunque es divertido, ¡no le ayudará en su búsqueda para usar las suites de oficina basadas en la nube sobre la RasPi!)

---

Con Chromium instalado, utilizar una suite de oficina basada en la nube es tan sencillo como visitar el sitio, suscribirse para obtener una cuenta (proporcionando los detalles de su tarjeta de crédito en caso de los servicios premium como los de Microsoft Office 365) e iniciar sesión. Si observa un bajo rendimiento, resulta útil cambiar el particionamiento de la memoria para otorgar al procesador ARM una mayor parte. El Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi” tiene instrucciones completas sobre cómo hacer esto.

## Utilizando LibreOffice

Si prefiere no utilizar un servicio basado en la nube, la alternativa es instalar LibreOffice.org. Diseñado como una alternativa multiplataforma de código abierto de la popular suite Microsoft Office y basado en el proyecto OpenOffice.org, LibreOffice es poderoso y ofrece casi tanta funcionalidad como su inspiración de código cerrado.

Tanta funcionalidad, sin embargo, tiene un costo. El paquete LibreOffice.org es de gran tamaño, ocupa cerca de 400 MB de espacio en la tarjeta SD una vez que todas las dependencias son incluidas. Esto puede ser un problema, en el sistema por defecto, la distribución Raspbian recomendada tiene menos espacio libre disponible en la tarjeta SD que el que LibreOffice.org requiere. (Para obtener más información sobre la disposición de la partición sobre la distribución Raspbian, vea la sección “Sistema de Archivos” en el Capítulo 3, “Administración del Sistema Linux”).

Si desea instalar LibreOffice.org, necesitará una tarjeta SD de 4 GB o más grande. También necesitará redimensionar el sistema de archivos raíz para hacer uso del espacio libre sobre la tarjeta SD. Para obtener instrucciones sobre cómo hacerlo, consulte el Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi”. Asegúrese de realizar esta tarea antes de continuar con este capítulo o verá que su tarjeta SD se quedará sin espacio antes de poder utilizar LibreOffice.

Con el suficiente espacio libre en la tarjeta SD, instalar LibreOffice no es más complejo que instalar cualquier otro paquete. A pesar de que viene con una gran cantidad de paquetes adicionales, un *metapquete* se encarga de todo e instala el software con un solo comando. Abra una ventana de terminal y escriba lo siguiente:

```
sudo apt-get install libreoffice
```

---

**CONSEJO**

Si al tratar de instalar el software recibe mensajes de error indicando que los archivos del Libre no se encuentran, es probable que la caché de paquetes no esté actualizada. Ejecute el comando `sudo apt-get update` para actualizar la caché y luego inténtelo de nuevo. (Ver Capítulo 3, “Administración del Sistema Linux” para más detalles).

---

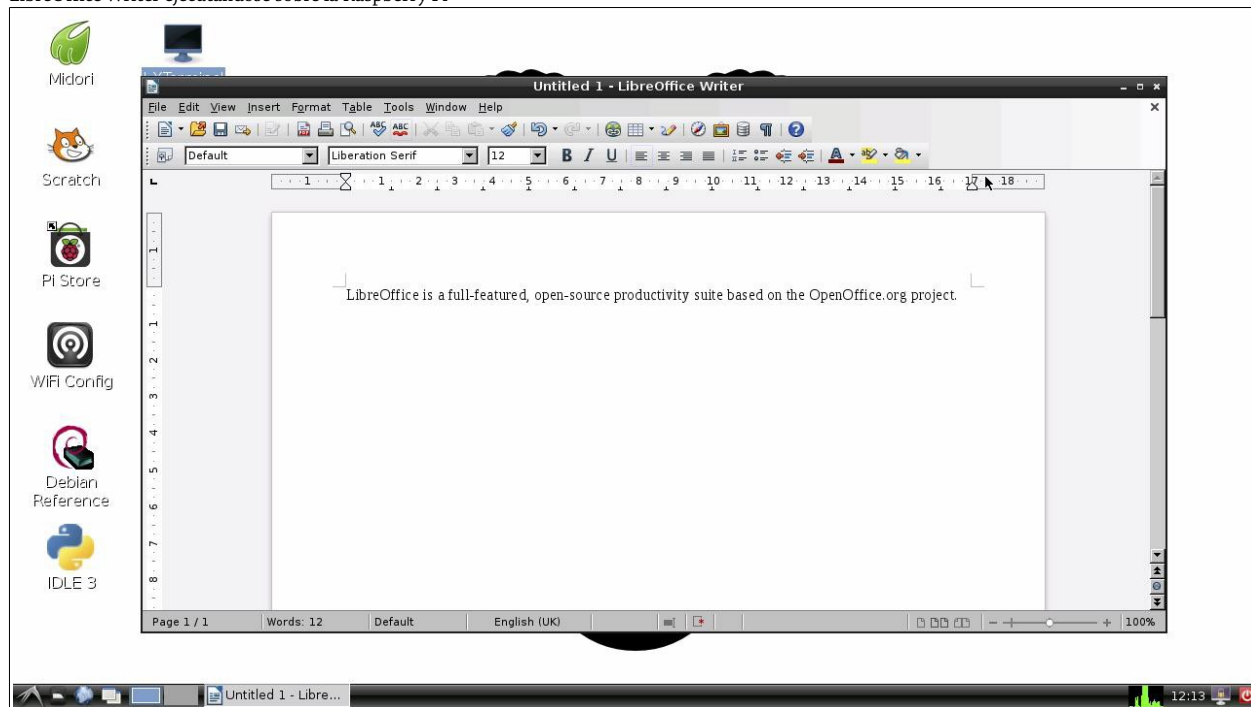
Una vez instalado, LibreOffice aparecerá como una serie de entradas en el menú Aplicaciones dentro del entorno de escritorio de la RasPi. Estas entradas son las siguientes:

- **LibreOffice**-La aplicación principal, la cual ofrece enlaces a las secciones individuales de la suite
- **LibreOffice Calc**-La aplicación de hoja de cálculo, equivalente a Microsoft Excel
- **LibreOffice Draw**-Una aplicación de ilustración vectorial, diseñada para dibujar imágenes escalables de alta calidad utilizables como clipart en otros programas LibreOffice
- **LibreOffice Impress**-La aplicación de presentaciones, equivalente a Microsoft PowerPoint

- **LibreOffice Math**-Un pequeño pero potente paquete diseñado para crear y editar fácilmente fórmulas científicas y ecuaciones que pueden incrustarse en otros programas LibreOffice
- **LibreOffice Writer**-La aplicación de procesador de textos, equivalente a Microsoft Word (ver Figura 9-2)

**Figura 9-2**

LibreOffice Writer ejecutándose sobre la Raspberry Pi



De forma predeterminada, LibreOffice guarda y carga los archivos en un formato conocido como *ODF* (*Open Document Format*). Este es un formato de archivo basado en estándares y libre de regalías, soportado por la mayoría de las suites de oficina (incluyendo las nuevas versiones de Microsoft Office).

Al guardar un archivo en LibreOffice, puede cambiar el formato utilizando un menú desplegable en el cuadro de diálogo Save As. Bajo File Type, puede seleccionar una variedad de formatos incluyendo algunos que son totalmente compatibles con las versiones anteriores de Microsoft Office. Cuando comparta sus archivos LibreOffice creados en la RasPi con los usuarios que cuentan con software anticuado, recuerde cambiarlos de formato para garantizar que todas las personas pueda abrir sus archivos. Como alternativa, puede convencerlos de que instalen LibreOffice, el cual se encuentra disponible de forma gratuita para Linux, Windows y OS X.

## Edición de Imágenes con The Gimp

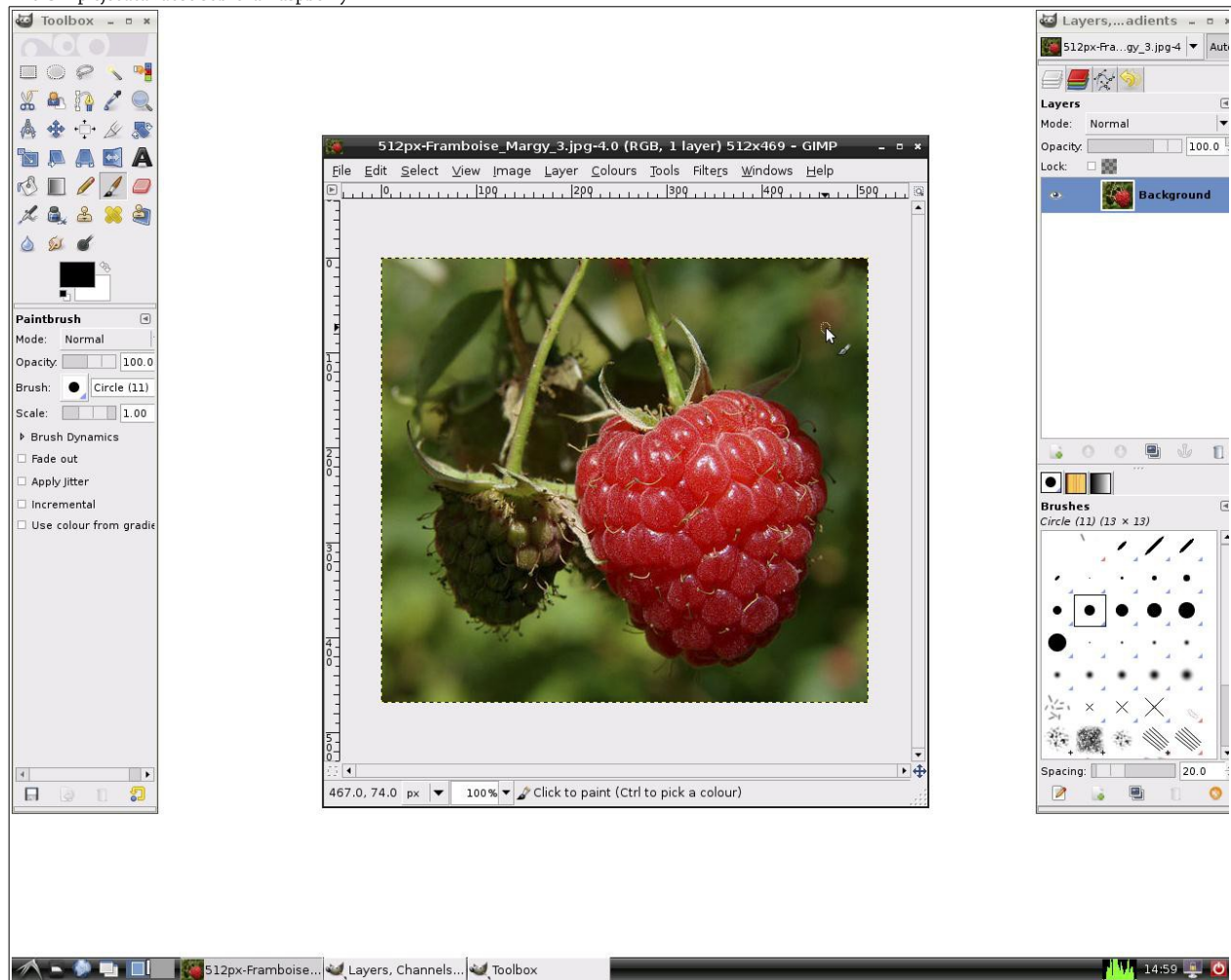
LibreOffice es una poderosa pieza de software, pero un área en donde tiene carencias es en la edición de imágenes. Aunque LibreOffice Draw es una poderosa herramienta para el trabajo ilustrativo, desafortunadamente, no se le puede utilizar para retocar fotografías digitales. Las imágenes de *mapas de bits* son muy diferentes a las imágenes vectoriales para las que Draw fue diseñado poder editar.

Para la edición de imágenes, una de las herramientas más poderosas que existen para Linux es el GNU Image Manipulation Program, comúnmente conocido como The Gimp. The Gimp es uno de los proyectos más populares de código abierto porque ofrece poderosas características para la edición de imágenes de mapas de bits con una interfaz de usuario similar a la interfaz que emplea el paquete comercial Adobe Photoshop (ver Figura 9-3).



Figura 9-3

The Gimp ejecutándose sobre la Raspberry Pi



The Gimp no es instalado por defecto en la mayoría de las distribuciones de la Raspberry Pi, por lo que tendrá que conectar su RasPi a la Internet e instalarlo a través del sistema de gestión de paquetes (ver Capítulo 3, “Administración del Sistema Linux”, para más detalles). The Gimp ocupa mucho espacio sobre la tarjeta SD (aunque no tanto como LibreOffice) así que asegúrese de contar con el suficiente espacio libre antes de instalarlo.

Para instalar The Gimp, abra una ventana de terminal y escriba lo siguiente:

```
sudo apt-get install gimp
```

Puede tomarle un tiempo acostumbrarse a usar The Gimp, porque su interfaz de usuario utiliza tres diferentes ventanas en lugar de sólo una. De forma predeterminada, la ventana de la izquierda contiene la caja de herramientas (Toolbox); la ventana a la derecha muestra las Capas, Canales, y opciones de Degradados; y la ventana del centro (principal) muestra la imagen que está actualmente editando. Si abre más de una imagen, obtendrá otra ventana principal adicional, pero no obstante, sólo una ventana para la Toolbox y una ventana para las Capas, Canales y Degradados.

Por defecto, el *Manual de Usuario de The Gimp* no se encuentra instalado. Para la RasPi, esto es algo bueno: The Gimp tiene un gran alcance y su manual de usuario necesita una considerable cantidad de espacio sobre la tarjeta SD. Si trata de acceder al manual presionando la tecla F1 o seleccionando Help desde el menú Help, se le propondrá

leer una versión en línea, basada en la web en su lugar. Haga clic en el botón Read Online para abrir el manual de usuario en el navegador.

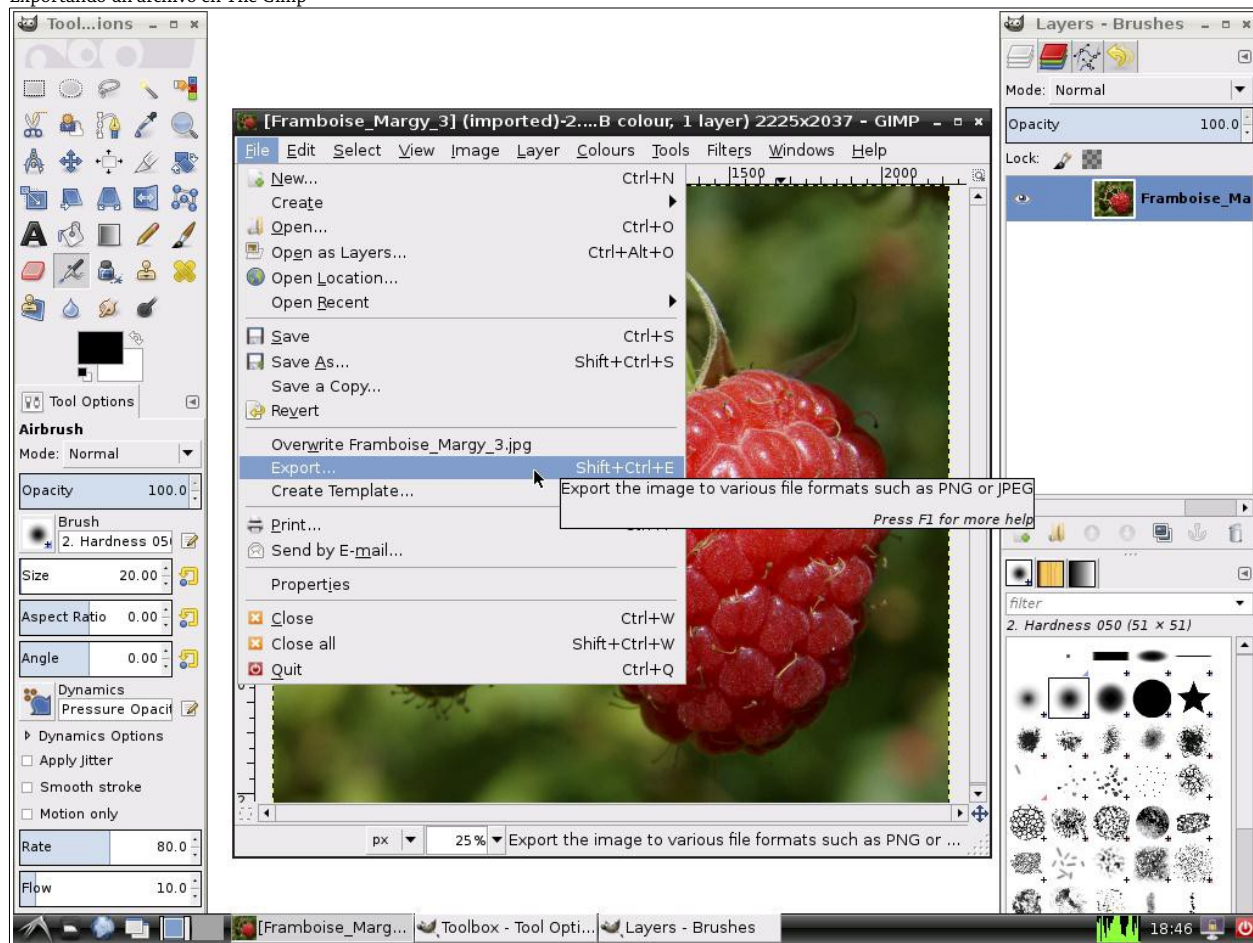
The Gimp es una utilidad muy poderosa y usa una gran cantidad de memoria. Por consecuencia, se ejecuta de forma relativamente lenta sobre la Raspberry Pi (aunque es definitivamente usable). Sea paciente con él, especialmente si está abriendo grandes fotografías desde una cámara digital. Puede ayudarlo aumentando la cantidad de memoria disponible para el sistema (ver Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi”, para obtener más detalles sobre cómo hacer esto).

Al momento de guardar sus archivos en The Gimp puede utilizar una amplia variedad de formatos de archivo. Si está planeando volver a su archivo y continuar con la edición, debe utilizar el formato de archivo predeterminado XCF. Este archivo mantiene intacto los *metadatos*, además utiliza la compresión de datos sin pérdida para mantener al máximo la calidad de la imagen y también soporta imágenes compuestas por varias capas.

Si planea subir su imagen a la Internet o compartirla con las demás personas, se recomienda utilizar un formato más portátil como el JPG o el PNG. Para cambiar el formato del archivo, seleccione la opción “Export” del menú File (ver Figura 9-4). Esto le permitirá elegir el formato del archivo de entre una amplia variedad de tipos de archivos.

Figura 9-4

Exportando un archivo en The Gimp





# Capítulo 10

La RasPi como un Servidor Web

**Aunque la RasPi** es notablemente menos poderosa que la mayoría de los sistemas que se pueden encontrar dentro de un centro de datos, esto no significa que la RasPi no pueda actuar como un útil servidor dentro de un ambiente casero o de negocios. A pesar de contar con poca memoria y relativamente poco poder de procesamiento, su bajo consumo de energía y su ejecución silenciosa lo convierten en una gran opción para servir páginas sencillas de bajo tráfico a una red local o incluso afuera en la Internet.

Una gran parte de los modernos servidores web funcionan ejecutando una combinación de Linux, Apache, MySQL y PHP (comúnmente conocidos como la *pila LAMP*). Linux es el encargado de proveer sistema operativo subyacente; MySQL el soporte para la base de datos; Apache el servidor web; y PHP el lenguaje de scripting para las páginas dinámicas. Con un servidor basado en LAMP, es posible la ejecución de paquetes extremadamente complejos que van desde sistemas de gestión de contenidos como WordPress hasta foros interactivos tales como phpBB. Todo esto es posible con la Raspberry Pi, siempre y cuando no espere un rendimiento similar al de un poderoso servidor comercial.

---

**CONSEJO**

Los servidores web trabajan mejor cuando disponen de mucha memoria. Para garantizar el máximo rendimiento, cambie el particionamiento de la memoria de la RasPi para reservar sólo 16 MB o 32 MB para la GPU (ver Capítulo 6, “La Herramienta de Configurando del Software de la Raspberry Pi”) y no ejecute al mismo tiempo la *interfaz gráfica de usuario (GUI)*.

---

## Instalación de la Pila LAMP

Si está ejecutando la distribución Raspbian recomendada para la Raspberry, entonces ya se encuentra a un cuarto del camino para tener completamente funcionando la pila LAMP, es concreto, ya tiene la parte Linux instalada. El siguiente paso es instalar los componentes restantes: Apache, MySQL y PHP. En la terminal o consola, escriba los siguientes comandos para instalar esos paquetes:

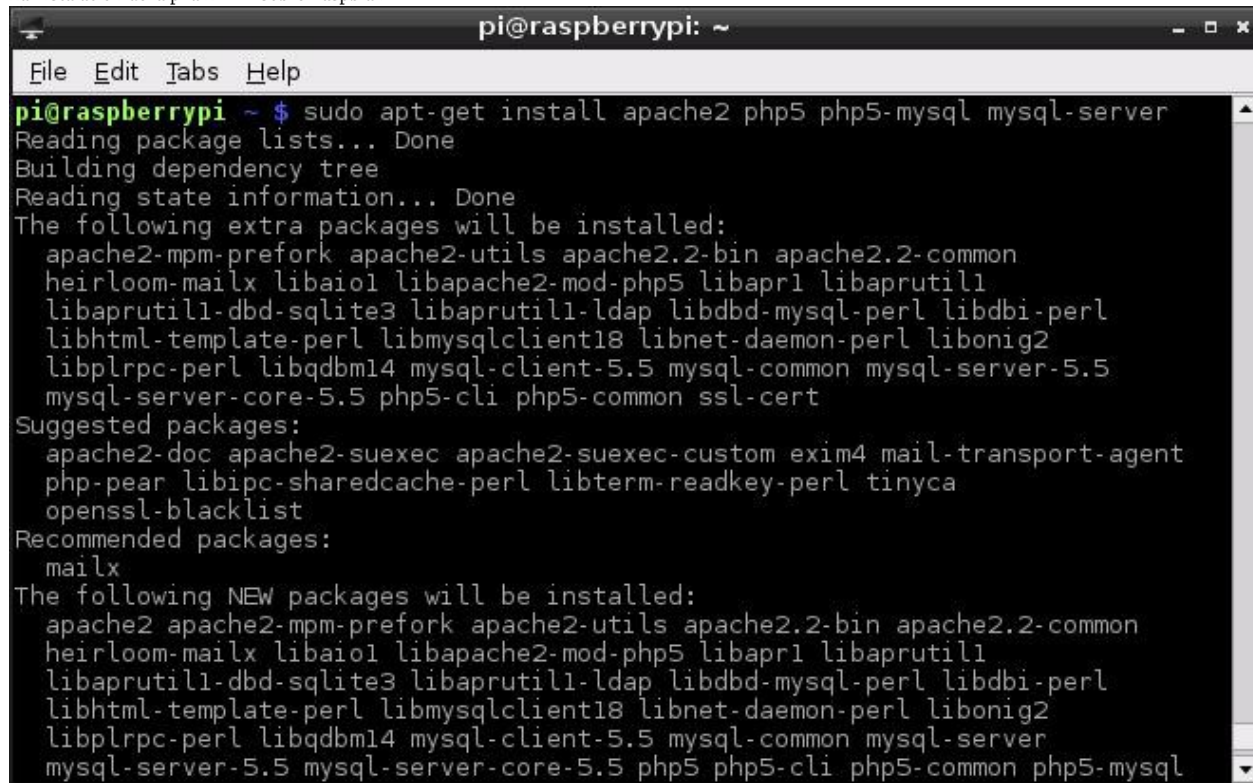
```
sudo apt-get update
sudo apt-get install apache2 php5 php5-mysql mysql-server
```

Esto hará que el gestor de paquetes `apt` (ver Capítulo 3, “Administración del Sistema Linux”) obtenga una serie de dependencias necesarias para conseguir la ejecución totalmente funcional de la pila LAMP (ver Figura 10-1). Estos paquetes y sus dependencias ocupan mucho espacio sobre la tarjeta SD (alrededor de 113 MB en total) de tal forma que si no ha redimensionado la partición raíz sobre la tarjeta SD, vuelva al Capítulo 6, “La Herramienta de Configuración de Software de la Raspberry Pi”, para ver las instrucciones sobre cómo expandir la partición raíz.

La instalación completa de la pila LAMP puede tardar bastante tiempo sobre la RasPi. No se preocupe si el sistema parece bloquearse por uno o dos minutos; después, la instalación debería continuar normalmente. Durante el proceso de instalación, MySQL le solicitará escribir una contraseña (ver Figura 10-2). Asegúrese de elegir una contraseña segura, ya que ésta protegerá a la base de datos MySQL, la cual, de acuerdo a lo que su servidor web fue diseñado a hacer, podrá almacenar nombres de usuario, detalles de pago y otra clase de información de identificación personal. ¡También asegúrese de elegir una contraseña que pueda recordar! Se le pedirá que confirme la contraseña (para verificar si hubiera un error de escritura) y luego la instalación continuará.

Figura 10-1

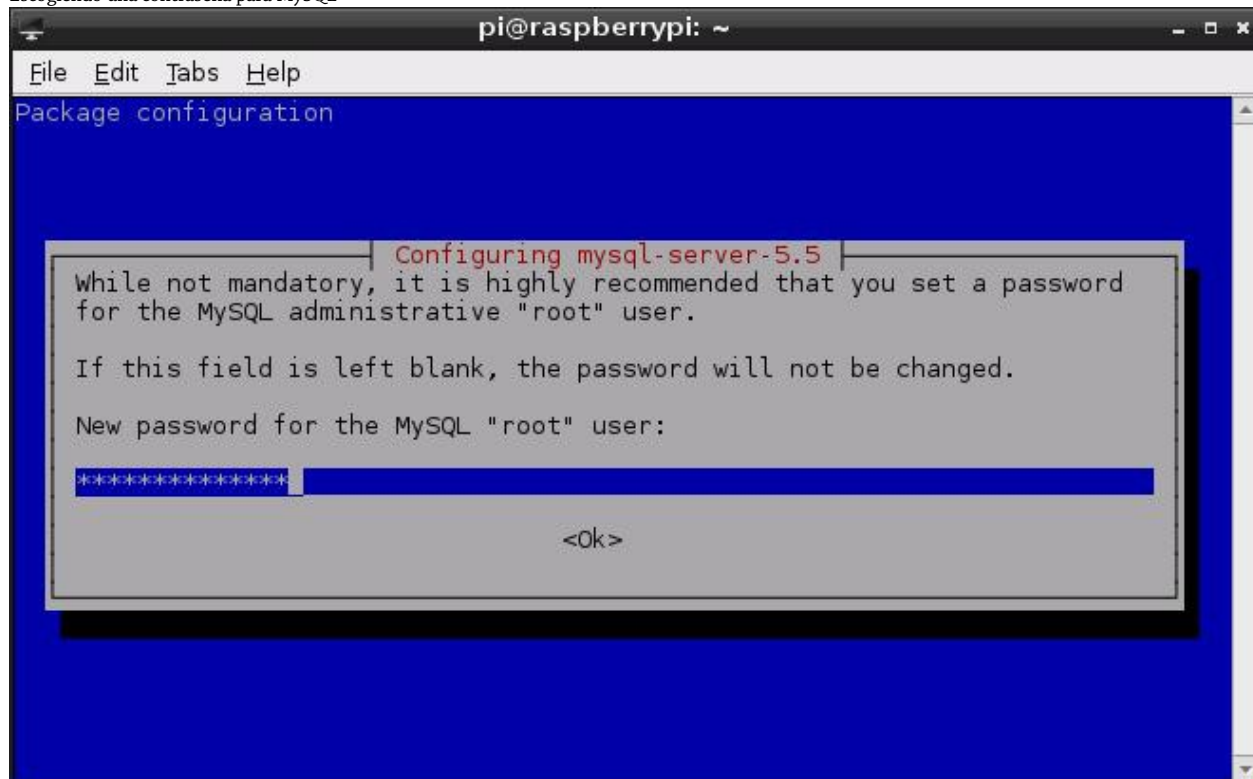
La instalación de la pila LAMP sobre Raspbian



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi ~$ sudo apt-get install apache2 php5 php5-mysql mysql-server  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following extra packages will be installed:  
  apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common  
  heirloom-mailx libaio1 libapache2-mod-php5 libapr1 libaprutil1  
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libdbd-mysql-perl libdbi-perl  
  libhtml-template-perl libmysqlclient18 libnet-daemon-perl libonig2  
  libplrpc-perl libqdbm14 mysql-client-5.5 mysql-common mysql-server-5.5  
  mysql-server-core-5.5 php5-cli php5-common ssl-cert  
Suggested packages:  
  apache2-doc apache2-suexec apache2-suexec-custom exim4 mail-transport-agent  
  php-pear libipc-sharedcache-perl libterm-readkey-perl tinycat  
  openssl-blacklist  
Recommended packages:  
  mailx  
The following NEW packages will be installed:  
  apache2 apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common  
  heirloom-mailx libaio1 libapache2-mod-php5 libapr1 libaprutil1  
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libdbd-mysql-perl libdbi-perl  
  libhtml-template-perl libmysqlclient18 libnet-daemon-perl libonig2  
  libplrpc-perl libqdbm14 mysql-client-5.5 mysql-common mysql-server  
  mysql-server-5.5 mysql-server-core-5.5 php5 php5-cli php5-common php5-mysql
```

Figura 10-2

Escogiendo una contraseña para MySQL

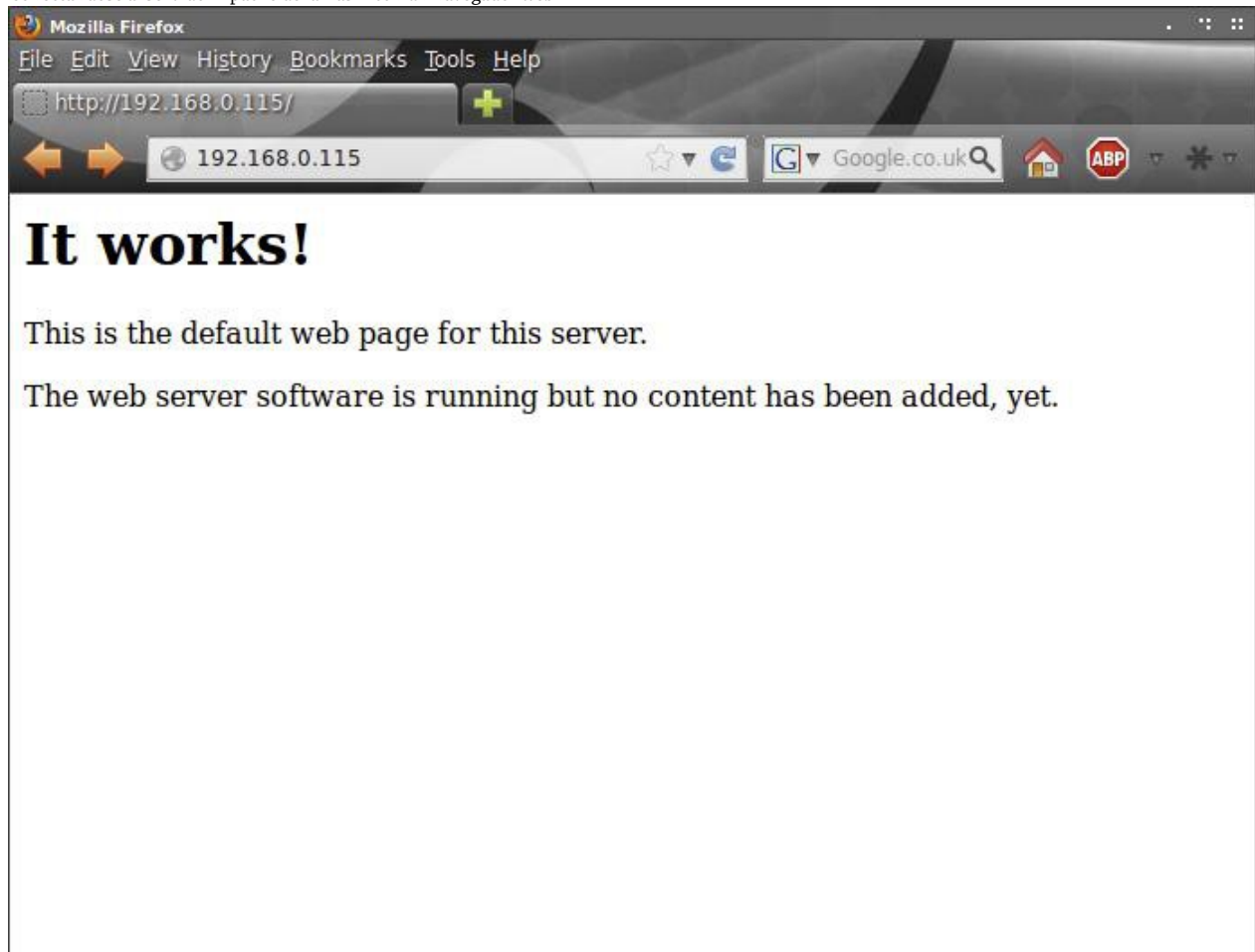


```
pi@raspberrypi: ~  
File Edit Tabs Help  
Package configuration  
  
Configuring mysql-server-5.5  
While not mandatory, it is highly recommended that you set a password  
for the MySQL administrative "root" user.  
  
If this field is left blank, the password will not be changed.  
  
New password for the MySQL "root" user:  
*****  
  
<Ok>
```

Cuando la instalación del software ha finalizado, tanto el servidor MySQL como el Apache (conocidos en la jerga Linux como *daemons* o *demonios*) se ejecutarán en segundo plano. Para comprobar que los servidores estén funcionando correctamente, utilice otra computadora sobre la red para conectarse a la Raspberry Pi utilizando un navegador Web. En la barra de direcciones, escriba la dirección IP de la RasPi para visualizar la página predeterminada de la Instalación de Apache (ver Figura 10-3). Si no está seguro de cuál es la dirección IP que la RasPi está utilizando, escriba `ifconfig` en la terminal y busque la dirección IP que aparece en la sección `eth0`, o si no está utilizando el puerto Ethernet integrado del Modelo B busque en la sección correspondiente al adaptador de red que esté utilizando sobre la RasPi. Para obtener más información sobre la comprobación y el ajuste de las configuraciones de red sobre la RasPi, vea el Capítulo 5, “Configuración de la Red”

Figura 10-3

Conectándose al servidor Apache de la RasPi con un navegador web



El paso final es confirmar que el módulo de scripting PHP ha sido cargado correctamente en Apache. Este módulo es importante pues le permite al servidor web Apache ejecutar scripts en PHP para servir contenidos dinámicos. Sin un módulo PHP trabajando (o un módulo alternativo para otros lenguaje de scripting como Python) Apache sólo sería capaz de servir páginas estáticas. Para probar el módulo Apache PHP, cree un nuevo archivo de script PHP utilizando el siguiente comando, escrito todo en una sola línea:

```
sudo sh -c 'echo "<?php phpinfo(); ?>" > /var/www/phptest.php'
```

Este comando crea un nuevo archivo llamado `phptest.php` en el directorio `/var/www`. Este archivo le dice a PHP que cree una página de información con fines de diagnóstico. Visite este archivo utilizando un navegador web desde otra computadora escribiendo `http://direccionip/phptest.php` (reemplazando *direccionip* con la dirección



IP de la Raspberry Pi) o sobre la misma RasPi escribiendo dentro de la barra de direcciones `http://localhost/phptest.php` (ver Figura 10-4).

**Figura 10-4**

Comprobación del módulo PHP de Apache sobre la Raspberry Pi



Cuando haya terminado de comprobar, elimine el archivo `phptest.php` con el siguiente comando:

```
sudo rm /var/www/phptest.php
```

#### CONSEJO

Aunque Apache es el servidor web más común, existen otros. Si encuentra que el rendimiento de Apache es demasiado lento, experimente con `lighttpd` (un servidor web ligero diseñado para utilizar menos memoria que Apache). Lo puede instalar en Raspbian con el comando `sudo apt-get install lighttpd`.

Con la pila LAMP instalada y trabajando, ahora puede crear su propio sitio web que será servido por la RasPi. Mientras los sitios no sean demasiado complejos y no reciban muchísimos usuarios al mismo tiempo, la RasPi podrá hacer frente a la tarea de manera admirable haciendo que su pequeño tamaño y su diminuto consumo de energía más que compensen cualquier desaceleración debida a que su sitio se esté volviendo popular.

Por defecto, los archivos para el servidor web se almacenan en la carpeta `/var/www`, en la cual sólo el usuario `root` puede escribir. Para modificar en dónde Apache encontrará sus archivos (por ejemplo, para mover el sitio web a una unidad de almacenamiento externa con mayor capacidad) edite el archivo de texto `000-default` que se encuentra ubicado en la carpeta `/etc/apache2/sites-enabled`. Para obtener mas información sobre la configuración de Apache, PHP y MySQL, escriba los siguientes comando en la terminal o consola:

```
man apache2
man php5
man mysql
```



## Instalación de WordPress

WordPress es una de las plataformas más populares de blogs, es proyecto de código abierto que tiene como objetivo proporcionar a los usuarios una plataforma sencilla y poderosa para crear sitios webs atractivos e interactivos. WordPress está construido sobre una base de PHP y Javascript, y ofrece una atractiva interfaz basada en la web para crear sitios webs enriquecidos. Por ejemplo, algunos de los sitios de noticias más populares en el mundo están contruidos sobre una plataforma personalizada de WordPress.

Para instalar WordPress sobre la Raspberry Pi escriba el siguiente comando en la terminal o consola:

```
sudo apt-get install wordpress
```

Al igual que la pila LAMP, WordPress viene con una selección de dependencias (ver Figura 10-5). Necesitará asegurarse de contar con aproximadamente 37 MB de espacio libre sobre la tarjeta SD de la RasPi para su instalación completa, además de los 113 MB requeridos para la pila LAMP. Si tiene suficiente espacio libre, escriba Y para continuar el proceso de instalación.

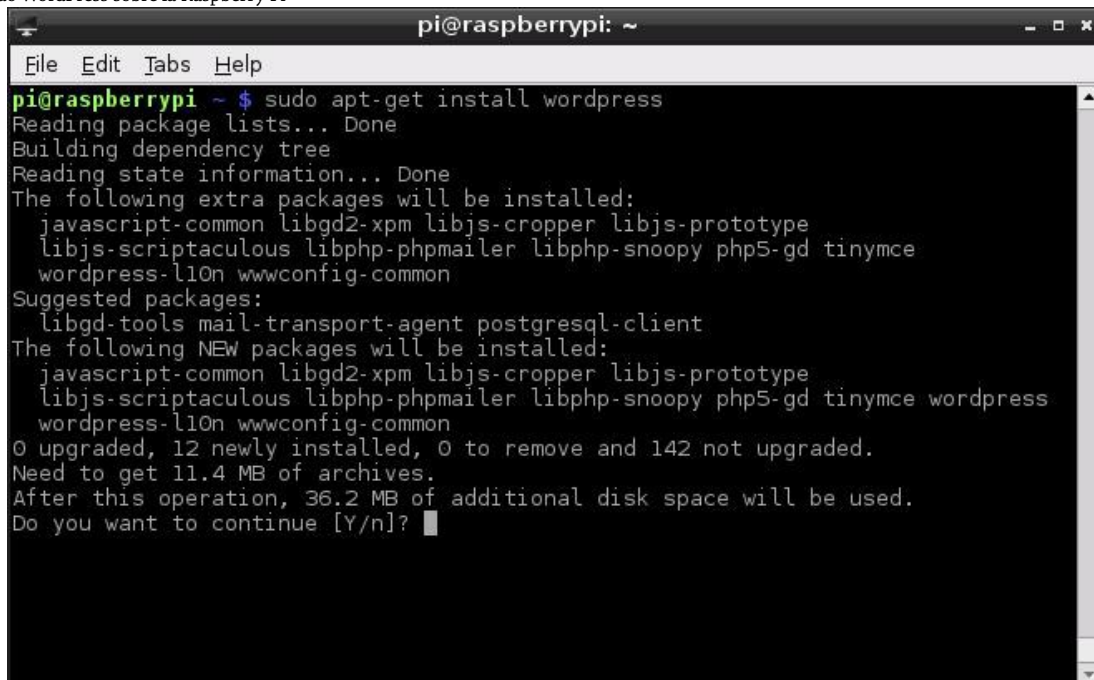
Cuando WordPress haya terminado de instalarse, su directorio de instalación predeterminado (/usr/share/wordpress) necesita ser *vinculado* al directorio /var/www a fin de que Apache pueda ver los archivos. Escriba el siguiente comando en la terminal:

```
sudo ln -s /usr/share/wordpress /var/www/wordpress
```

La vinculación de un archivo o directorio es un concepto distinto a la de su movimiento: los archivos de WordPress ahora existen simultáneamente tanto en /usr/share como en /var/www, sin ocupar ningún espacio adicional sobre la tarjeta SD de la RasPi. Si le ha indicado a Apache que utilice un directorio diferente para el sitio web por defecto, modifique el comando de vinculación en la forma que corresponda. A continuación, ejecute el script de configuración de MySQL de WordPress utilizando el siguiente comando, escrito todo en una sola línea:

```
sudo bash /usr/share/doc/wordpress/examples/setup-mysql -n wordpress localhost
```

Figura 10-5  
Instalando WordPress sobre la Raspberry Pi



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi ~$ sudo apt-get install wordpress  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following extra packages will be installed:  
  javascript-common libgd2-xpm libjs-cropper libjs-prototype  
  libjs-scriptaculous libphp-phpmailer libphp-snoopy php5-gd tinymce  
  wordpress-l10n wwwconfig-common  
Suggested packages:  
  libgd-tools mail-transport-agent postgresql-client  
The following NEW packages will be installed:  
  javascript-common libgd2-xpm libjs-cropper libjs-prototype  
  libjs-scriptaculous libphp-phpmailer libphp-snoopy php5-gd tinymce wordpress  
  wordpress-l10n wwwconfig-common  
0 upgraded, 12 newly installed, 0 to remove and 142 not upgraded.  
Need to get 11.4 MB of archives.  
After this operation, 36.2 MB of additional disk space will be used.  
Do you want to continue [Y/n]?
```

Esto agregará una nueva base de datos a MySQL, instalada como parte de la pila LAMP, para que WordPress la utilice. Esta base de datos almacenará sus cuentas de usuario, mensajes, comentarios y otros detalles. Una vez que el script se haya completado, se le pedirá que visite `http://localhost` en algún navegador sobre la Raspberry Pi para que continúe la instalación. Esta instrucción es un tanto incorrecta: la dirección que necesita visitar para finalizar la instalación de WordPress es `http://localhost/wordpress` (ver Figura 10-6).

Figura 10-6

Configurando WordPress en el navegador web NetSurf

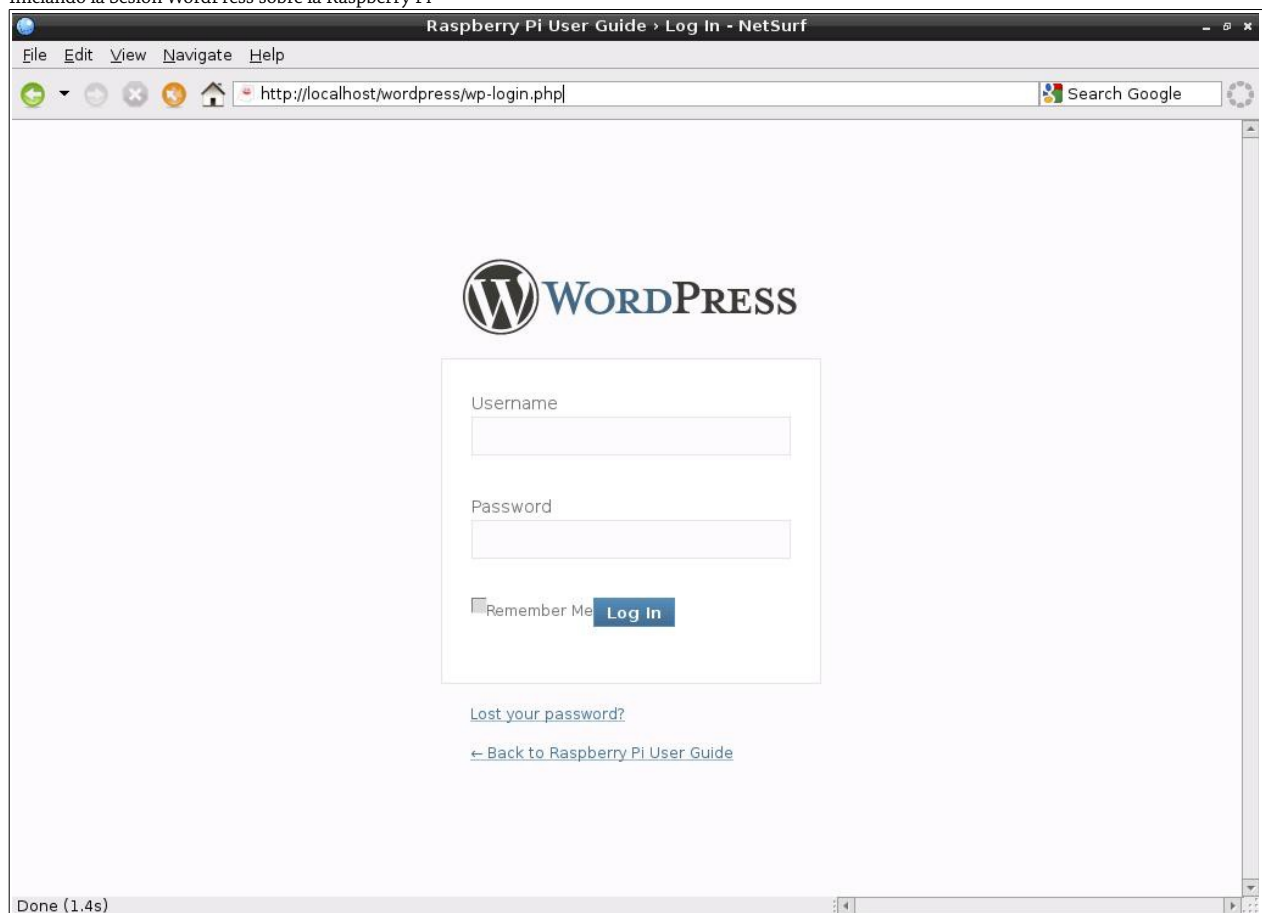


Llene el formulario que se carga dentro del navegador web, elija un nombre descriptivo para su sitio y establezca una contraseña segura (pero recordable) para su usuario. Asegúrese de cambiar el campo Nombre de Usuario de `admin` a cualquier otra cosa, a fin de volver más seguro su sitio. Cuando haya llenado todos los campos, haga clic en el botón Instalar WordPress en la parte inferior de la página.

El proceso de instalación tardará uno o dos minutos en completarse y en seguida se cargará en el navegador una nueva página confirmando la instalación exitosa de WordPress. Para comenzar a utilizar WordPress, haga clic en el botón Iniciar Sesión en la parte inferior de la página actual para iniciar sesión dentro de WordPress con el nombre de usuario y la contraseña que eligió en la pantalla anterior (ver Figura 10-7).

Figura 10-7

Iniciando la Sesión WordPress sobre la Raspberry Pi



Antes de que pueda acceder a WordPress desde otra computadora, necesitará crear un archivo adicional de configuración. Este se crea mediante la vinculación del archivo de configuración existente (configurado para el acceso local) utilizando el siguiente comando, escrito todo en una misma línea:

```
sudo ln -s /etc/wordpress/config-localhost.php /etc/wordpress/config-direccionip.php
```

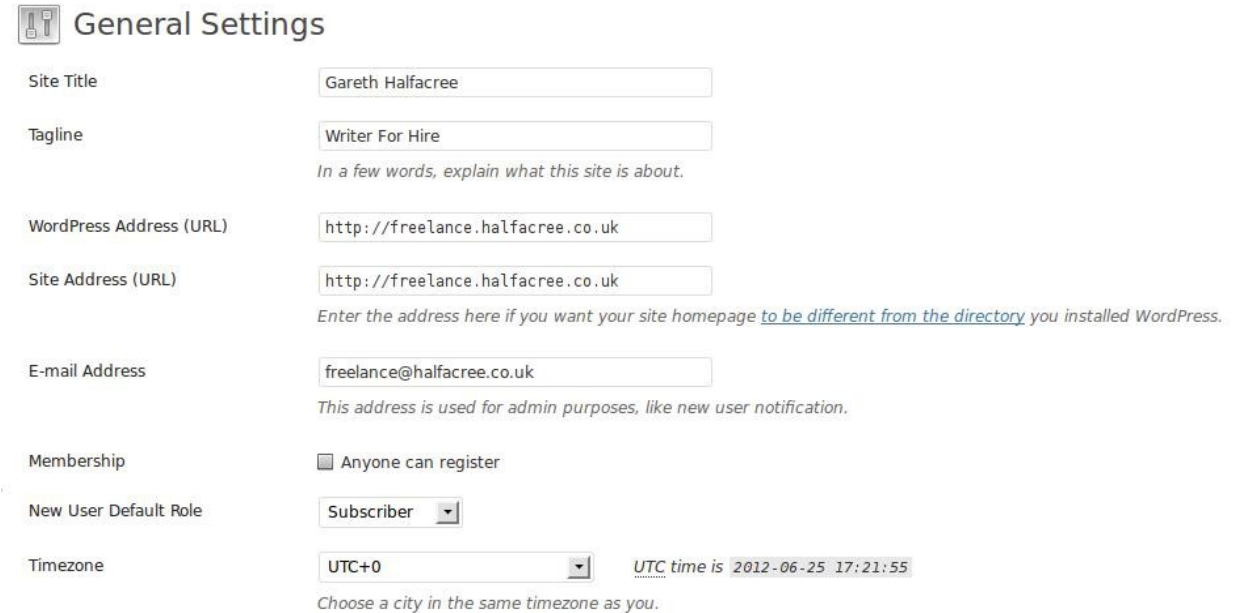
Reemplace en este código *direccionip* con la dirección IP de su Raspberry Pi. Si le dio a la Raspberry Pi un nombre de host en el DNS, también puede crear un archivo de configuración para ese nombre de host utilizando el mismo comando, pero reemplazando *direccionip* con el nombre del host elegido. Si no tiene un nombre de host, simplemente utilice la dirección IP de la RasPi. Por ejemplo, el comando para una RasPi sobre la dirección IP 192.168.0.115 debería ser la siguiente:

```
sudo ln -s /etc/wordpress/config-localhost.php /etc/wordpress/config-192.168.0.115.php
```

Para terminar de configurar el acceso externo, seleccione General desde el menú Configuraciones que se localiza al lado izquierdo del Dashboard de WordPress y cambie la URL para que coincida con la dirección IP de la RasPi o con el nombre de host elegido (ver Figura 10-8).

Figura 10-8

Modificando la URL en el menú Configuraciones Generales de WordPress



**General Settings**

Site Title: Gareth Halfacree

Tagline: Writer For Hire  
*In a few words, explain what this site is about.*

WordPress Address (URL): http://freelance.halfacree.co.uk

Site Address (URL): http://freelance.halfacree.co.uk  
*Enter the address here if you want your site homepage [to be different from the directory](#) you installed WordPress.*

E-mail Address: freelance@halfacree.co.uk  
*This address is used for admin purposes, like new user notification.*

Membership: ☒ Anyone can register

New User Default Role: Subscriber

Timezone: UTC+0  
*Choose a city in the same timezone as you.* UTC time is 2012-06-25 17:21:55

**CONSEJO**

WordPress hace uso intenso de la memoria. Puede conseguir un mejor rendimiento, utilizando la RasPi como un *servidor sin cabeza (headless)*. Un *servidor sin cabeza* es un sistema que se ejecuta sin tener una interfaz gráfica de usuario cargada. Entonces, puede utilizar un navegador web sobre otra computadora conectada a la red para acceder al Dashboard de WordPress en <http://direccionip/wordpress/wp-login.php>. Intentar acceder al Dashboard de WordPress directamente sobre la RasPi puede tardar mucho tiempo.

El software WordPress incluye una función de actualización automática, que le asegura que su instalación esté ejecutando la última versión. Debido a su popularidad, WordPress es a menudo blanco de los ataques del malware y con frecuencia son liberadas actualizaciones para parchar agujeros de seguridad o añadir nuevas funcionalidades. Sin embargo, cuando WordPress se instala a través de APT no cuenta con los permisos necesarios para mantenerse actualizado. Para corregir esto, escriba el siguiente comando en la terminal:

```
sudo chown -R www-data /usr/share/wordpress
```

Esto le otorga al usuario **www-data** (la cuenta utilizada por el servidor web Apache) los derechos para cambiar los archivos ubicados en la carpeta **/usr/share/wordpress**. Esto permitirá que la actualización automática funcione cuando es seleccionada desde el Dashboard de WordPress.

Para obtener más información sobre cómo utilizar WordPress visite el sitio oficial en <http://www.wordpress.org>.

Parte III

# Programando con la Raspberry Pi

**Capítulo 11**    Una introducción a Scratch

**Capítulo 12**    Una introducción a Python



# Capítulo 1 1

## Una Introducción a Scratch

**Hasta ahora en** este libro, ha aprendido mucho acerca de cómo utilizar los programas que otras personas han escrito sobre la Raspberry Pi. El objetivo principal del proyecto Raspberry Pi es hacer que cualquier persona escriba sus propios programas. La fundación Raspberry Pi está trabajando para conseguir que el dispositivo sea adoptado como una herramienta educativa para todas las edades.

Un requisito fundamental para alcanzar ese objetivo es garantizar que los niños puedan experimentar lo divertido que es crear sus propios programas, en lugar de limitarse sólo a ser consumidores de programas hechos por otros. El secreto para hacer esto se llama Scratch.

## Presentando a Scratch

Creado por el grupo Lifelong Kindergarten en el Technology Media Lab del Instituto de Massachusetts en 2006 como un ramificación de los lenguajes Squeak y Smalltalk, Scratch toma los conceptos fundamentales de la programación y los hace accesible a todos. La larga mecanografía (cansada y aburrida para los niños más pequeños) es reemplazada con un sencillo ambiente de arrastrar y soltar parecido a un rompecabezas, el cual sin embargo estimula el pensamiento programático e introduce los conceptos fundamentales utilizados por todos los lenguajes de programación.

Considerado oficialmente un programa para niños de ocho años en adelante, pero accesible incluso por los programadores más jóvenes con un poco de ayuda y orientación, Scratch es engañosamente poderoso. Detrás de su interfaz de usuario colorida y conducida por el ratón se encuentra un lenguaje de programación que incluye una impresionante funcionalidad multimedia. No debería ser ninguna sorpresa que, de los más de dos millones y medio de proyectos Scratch compartidos por los usuarios del software sobre el sitio oficial, la mayoría sean juegos.

Alentar a los niños a aprender cómo hacer sus propios juegos puede ser una gran manera de sacar un poco de aprendizaje en sus tiempos de recreo. La interfaz amigable de Scratch y el excelente manejo de los conceptos básicos significa que sea poco probable que los niños se sientan frustrados por una súbita curva de aprendizaje escarpada. Mejor aún, los conceptos aprendidos en Scratch ofrecen una excelente base para la progresión a un lenguaje más flexible como Python (ver Capítulo 12, “Una Introducción a Python”).

Incluso antes de pasar a otros lenguaje, Scratch ofrece más que un solo framework para juegos. Puede ser utilizado para crear presentaciones y caricaturas interactivas, así como interfaces con sensores externos y motores a través del uso de un complemento de hardware, como el PicoBoard y el kit de robótica LEGO WeDo.

La distribución recomendada Raspbian para la Raspberry Pi viene con la última versión de Scratch precargada, de manera que si usted ha seguido las recomendaciones en este libro está ya listo para comenzar. Si está utilizando un sistema operativo distinto, visite el sitio oficial de Scratch en <http://scratch.mit.edu> para descargar el software (también podrá encontrar versiones para Windows y OS X de Apple en el mismo sitio).

## Ejemplo 1: Hola Mundo

Al aprender un nuevo lenguaje de programación, es tradicional comenzar con un programa muy básico: uno que muestra una sola línea de texto. Este es comúnmente conocido como un programa Hola Mundo, y es el primer paso para aprender cómo crear sus propios programas.

A diferencia de un lenguaje de programación tradicional, Scratch no espera que el usuario memorice los nombres de las instrucciones tales como `print` o `inkey$`. En vez de eso, casi todo se hace arrastrando y colocando bloques de código y arreglándolos de un modo lógico.

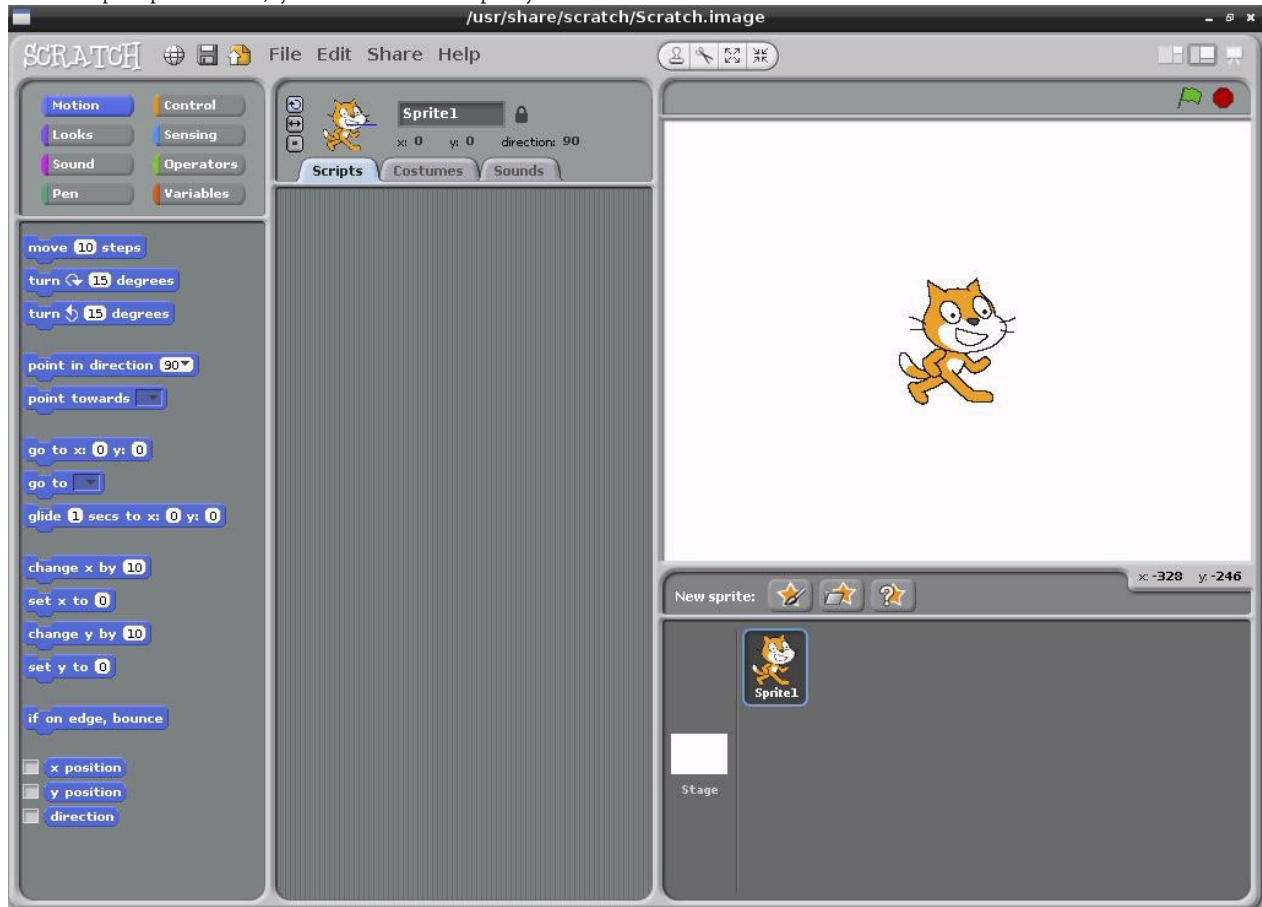
Si acaba de instalar Raspbian en su tarjeta SD, el idioma de la interfaz gráfica de usuario de Scratch se mostrará en inglés. Para cambiar al español el idioma de la interfaz gráfica vaya al Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi” y lea la sección “4 Internationalisation Options” para conocer como cambiar el idioma.



Para empezar, cargue Scratch haciendo doble clic en su icono en el escritorio o haciendo clic en su entrada en el menú Programación. Después de unos segundos, la interfaz principal de Scratch aparecerá (ver Figura 11-1). Si la ventana se muestra descentrada o pequeña, presione el botón Maximizar (el botón de en medio que está entre los tres botones de control de la ventana que se muestran sobre la esquina superior derecha de la barra de título) para llenar la pantalla.

Figura 11-1

La interfaz principal de Scratch, ejecutándose sobre la Raspberry Pi



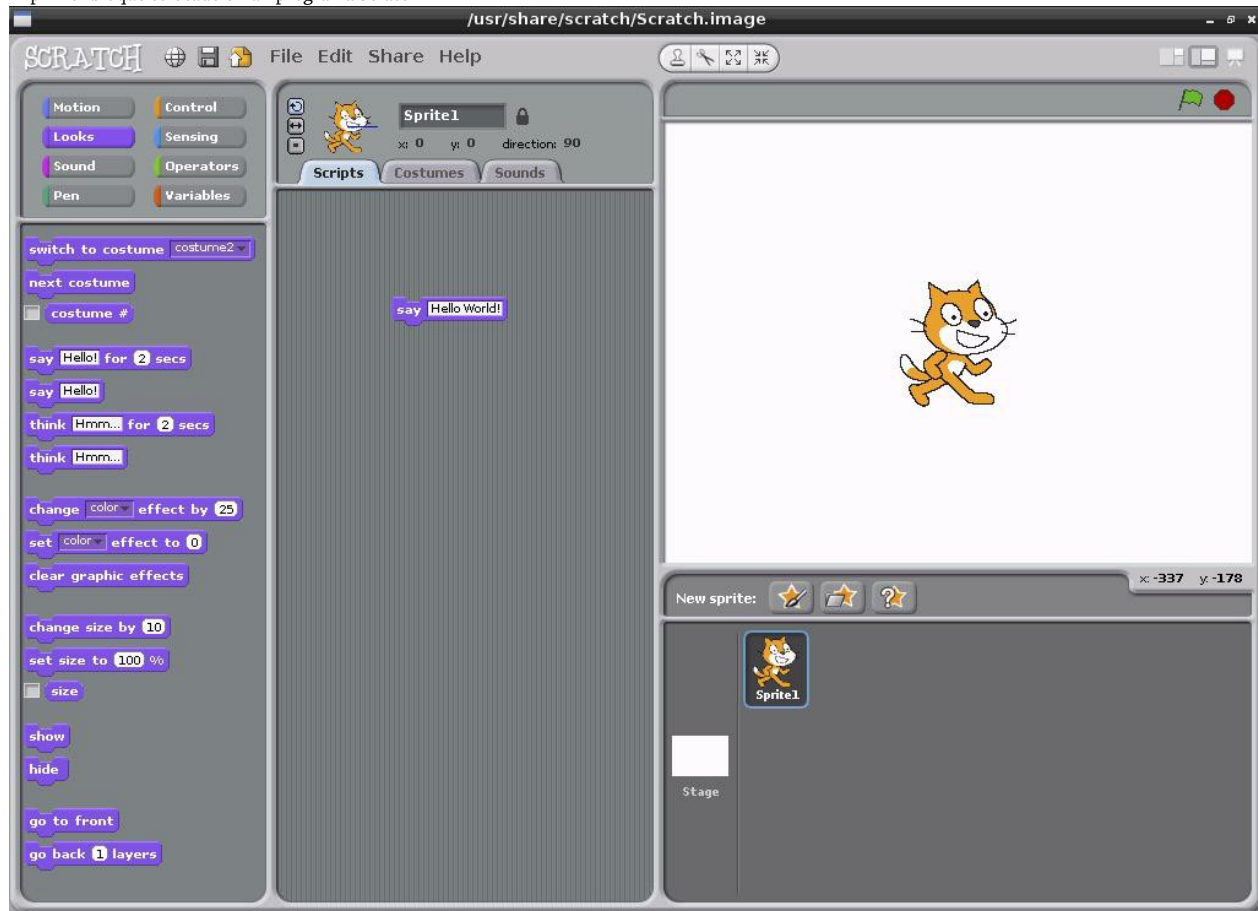
La interfaz de Scratch está dividida en varios paneles. A la izquierda está la *paleta de bloques*, que contiene todos los diferentes bloques de código que se pueden usar para crear un programa. Una lista de los objetos dentro del programa, conocidos como *sprites* u *objetos*, aparece en la parte inferior derecha junto con un mando para controlar la *escena* sobre la cual los objetos aparecen. La escena en sí está en la parte superior derecha de la ventana, que es donde verá el programa en ejecución. Por último, el centro de la ventana es donde el propio programa es construido.

Al comenzar la aplicación, un nuevo proyecto Scratch incluye ya una escena en blanco y un solo objeto o sprite. Lo que le falta es un programa, por lo que al hacer clic en el icono de la bandera verde en la parte superior derecha de la ventana ninguna acción se llevará a cabo, esto es porque Scratch aún no sabe qué es lo que usted desea que él haga.

Para el programa Hola Mundo, necesitará cambiar la paleta de bloques en la parte izquierda de la pantalla al modo Apariencia haciendo clic en su botón. En la lista de los bloques de Apariencia está uno que se lee **decir ¡Hola!**, Haga clic en este bloque y arrástrelo hacia el espacio vacío en el centro de la ventana etiquetada Programas. Para seguir las décadas de tradición detrás de este tipo de programas, usted también puede hacer clic sobre el bloque en donde dice **decir ¡Hola!** Y personalízelo para leerse **¡Hola Mundo!** si así lo desea (ver Figura 11-2).

Figura 11-2

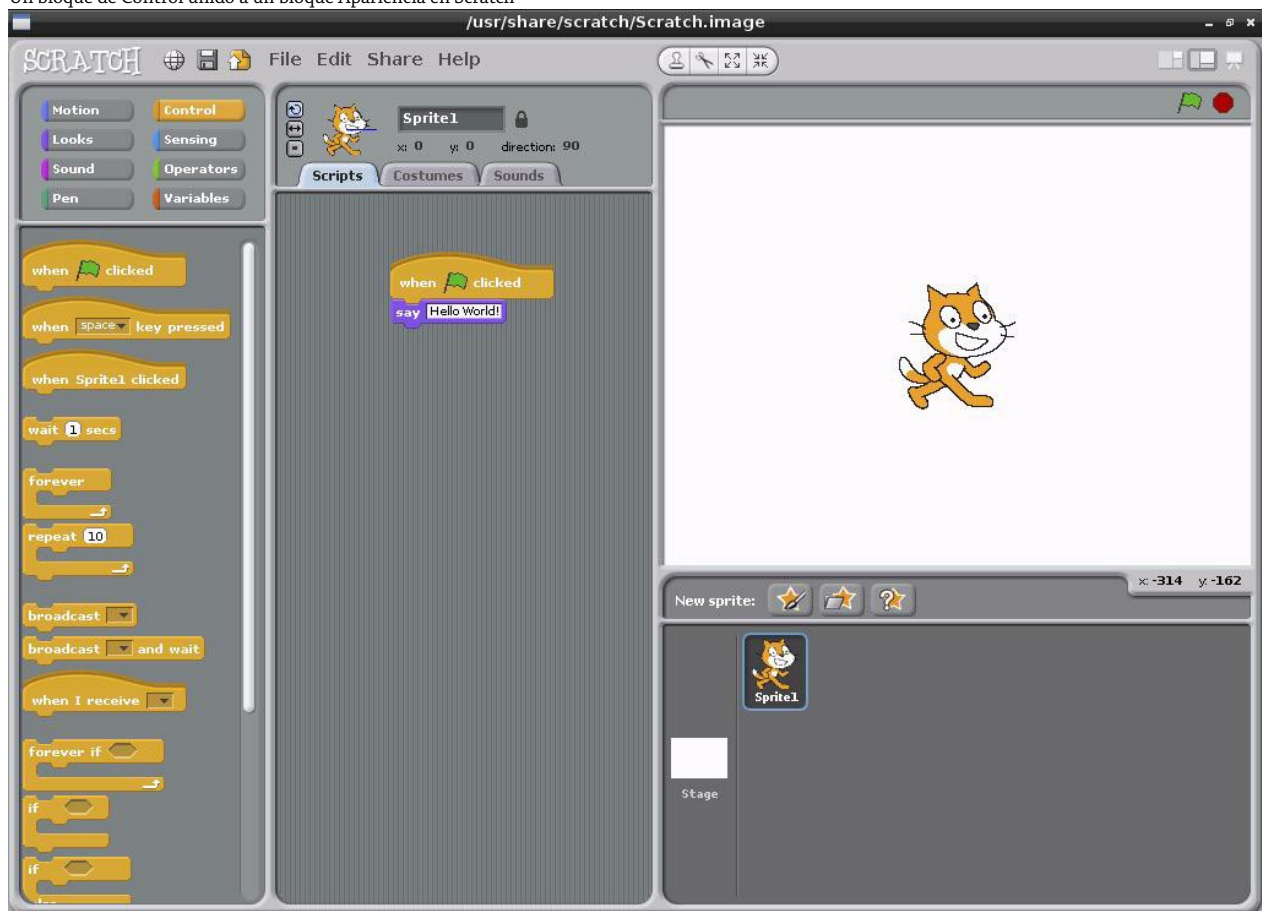
El primer bloque colocado en un programa Scratch



Si vuelve a hacer clic en la bandera verde, el programa aún no hará nada. Esto es porque a pesar de que Scratch sepa que debe de hacer que el objeto del gato diga algo, Scratch no sabe cuándo debe de hacerlo. El evento requiere un *bloque disparador*, el cual se puede encontrar en la sección Control de la paleta de bloques.

Entre a esa sección ahora haciendo clic en Control, y luego arrastre el bloque que está hasta arriba en la parte superior (etiquetado **al presionar [icono bandera]**) y colóquelo justo encima del ladrillo púrpura (ver Figura 11-3). Si lo arrastra lo suficientemente cerca, éste automáticamente se unirá al ladrillo existente como una pieza de rompecabezas.

Figura 11-3  
Un bloque de Control unido a un bloque Apariencia en Scratch



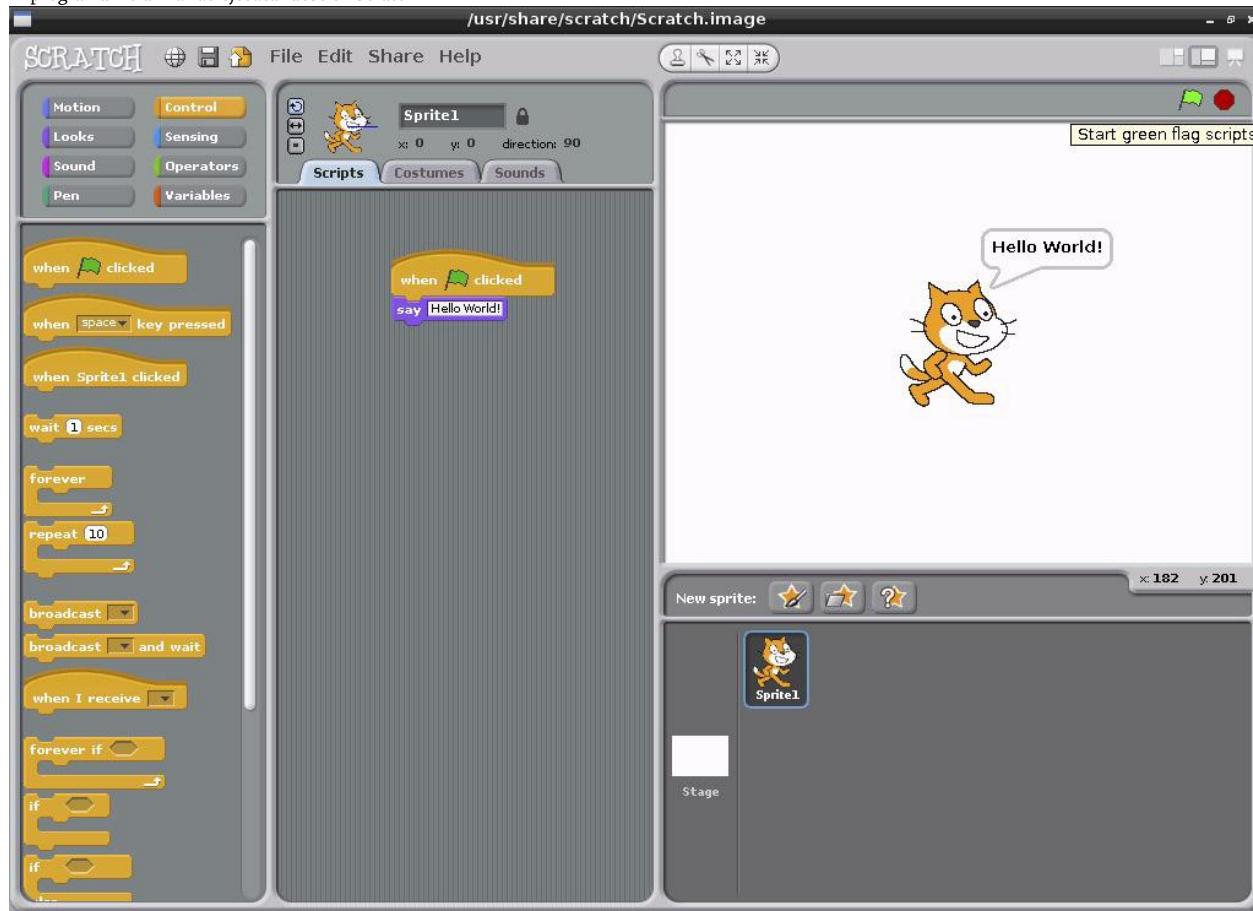
Este concepto de conectar varios ladrillo juntos es el corazón de Scratch. Si mira en el ladrillo de Control que ha colocado, verá que no hay huecos de conexión en la parte superior. Esto quiere decir que no pueden apilar otros ladrillos encima, porque este ladrillo de Control está diseñado para disparar o activar una serie de otros ladrillos directamente y debe de existir al principio de la pila. La parte inferior del ladrillo **decir**, por su parte, tiene un conector que encaja en la parte superior de otros ladrillos, lo que indica que más ladrillos pueden ser colocados por debajo suyo.

Con los dos ladrillos en su lugar, haga clic de nuevo en el icono de la bandera verde en la parte superior derecha de la ventana. Esta vez, un bocadillo de diálogo aparecerá de la boca del gato (ver Figura 11-4), y el programa Hola Mundo esta completo.

Antes de pasar al siguiente ejemplo, tómese el tiempo de guardar su trabajo utilizando el menú Archivo. Scratch está diseñado para tener un solo proyecto abierto a la vez, por lo que si crea un nuevo archivo en blanco, el archivo actual se cerrará para hacer espacio. No se preocupe si se olvida de esto cuando vaya a crear un nuevo archivo, si existen cambios que aún no han sido guardados en el archivo, Scratch le preguntará si desea guardarlos antes de cerrarlo.

Figura 11-4

El programa Hola Mundo ejecutándose en Scratch



## Ejemplo 2: Animación y Sonido

Mientras que el programa Hola Mundo es un ejemplo muy tradicional, no es particularmente interesante. Además, falla en presumir el verdadero poder de Scratch, el cual reside en sus impresionantes capacidades multimedia y sistema de manejo de sprites u objetos. Este sistema es particularmente adecuado para crear animaciones sencillas que pueden formar la base de un juego interactivo.

Para empezar, inicie un nuevo proyecto en Scratch cargando el programa nuevamente o seleccionando Nuevo desde el menú Archivo. Al igual que con cualquier otro proyecto que inicie, Scratch proporcionará un objeto por defecto (este es el objeto que usted estará controlando).

Para controlar una sencilla animación en Scratch, utilice la sección Movimiento de la paleta de bloques. Cuando inicia un nuevo proyecto, la paleta Movimiento es la paleta activa por defecto. Arrastre el bloque etiquetado con **mover 10 pasos** al área de Programas. Como su nombre lo indica, este bloque le dice al objeto seleccionado que se mueva 10 pasos en la dirección en la que mira. Un objeto en Scratch de manera predeterminada siempre se inicia mirando directamente a la derecha, por eso, el bloque **mover 10 pasos** moverá el objeto 10 pasos a la derecha).

Diez pasos no es un valor muy grande, así que haga clic sobre el valor 10 y cámbielo a 30. El bloque deberá leerse entonces como **mover 30 pasos**. Una animación de un gato moviéndose hacia la derecha de la escena no resulta muy interesante, así que cambie a la paleta de bloques de Sonido y arrastre el bloque **tocar sonido miau** al área de programas y vincule el bloque por debajo del bloque existente **mover**. Para mantener al gato en esta posición por

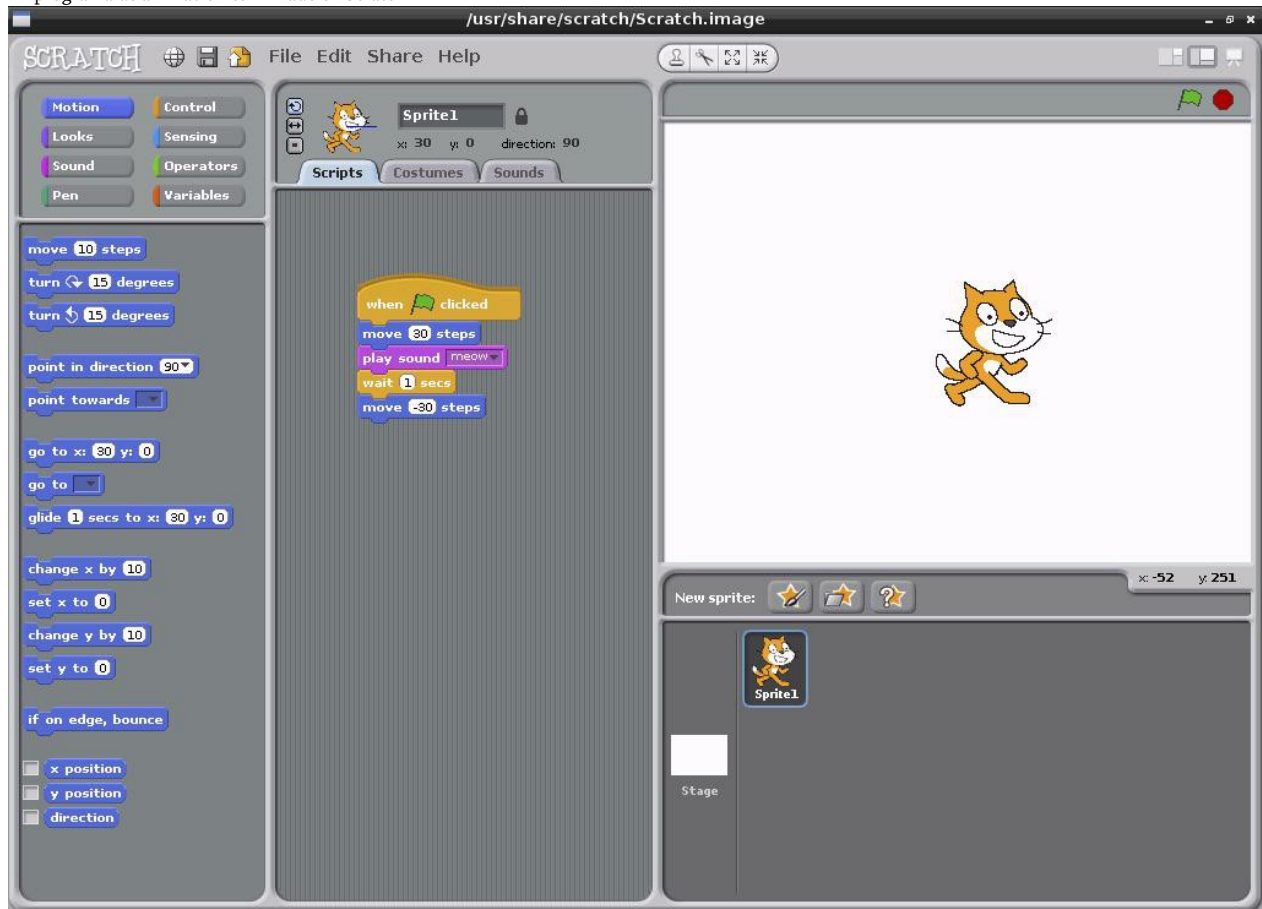
un rato, añade un bloque **esperar 1 segundos** desde la paleta de bloques Control. Sin este bloque, el objeto se movería tan rápidamente entre su posición inicial y la posición final.

Para hacer que el programa se ejecute varias veces sin que el gato desaparezca fuera del borde de la escena, añade otro bloque **mover 10 pasos** por debajo del bloque **tocar sonido** y modifícelo para que se lea **mover -30 pasos**. Scratch afortunadamente le permitirá utilizar cifras negativas como ésta: si un valor de 30 hace que el objeto se mueva una cierta distancia a la derecha, -30 hará que se mueva exactamente la misma distancia pero en la dirección opuesta.

Por último, añade un bloque **al presionar [icono bandera]** desde la paleta Control a la parte superior de la pila de bloques del área de Programas para completar el programa como se muestra en la Figura 11-5. Haciendo clic sobre el icono de la bandera verde en la parte superior derecha de la pantalla provocará que el programa se ejecute, asegúrese de que tiene los altavoces o audífonos conectados a la RasPi para escuchar el efecto completo!

Figura 11-5

El programa de animación terminado en Scratch



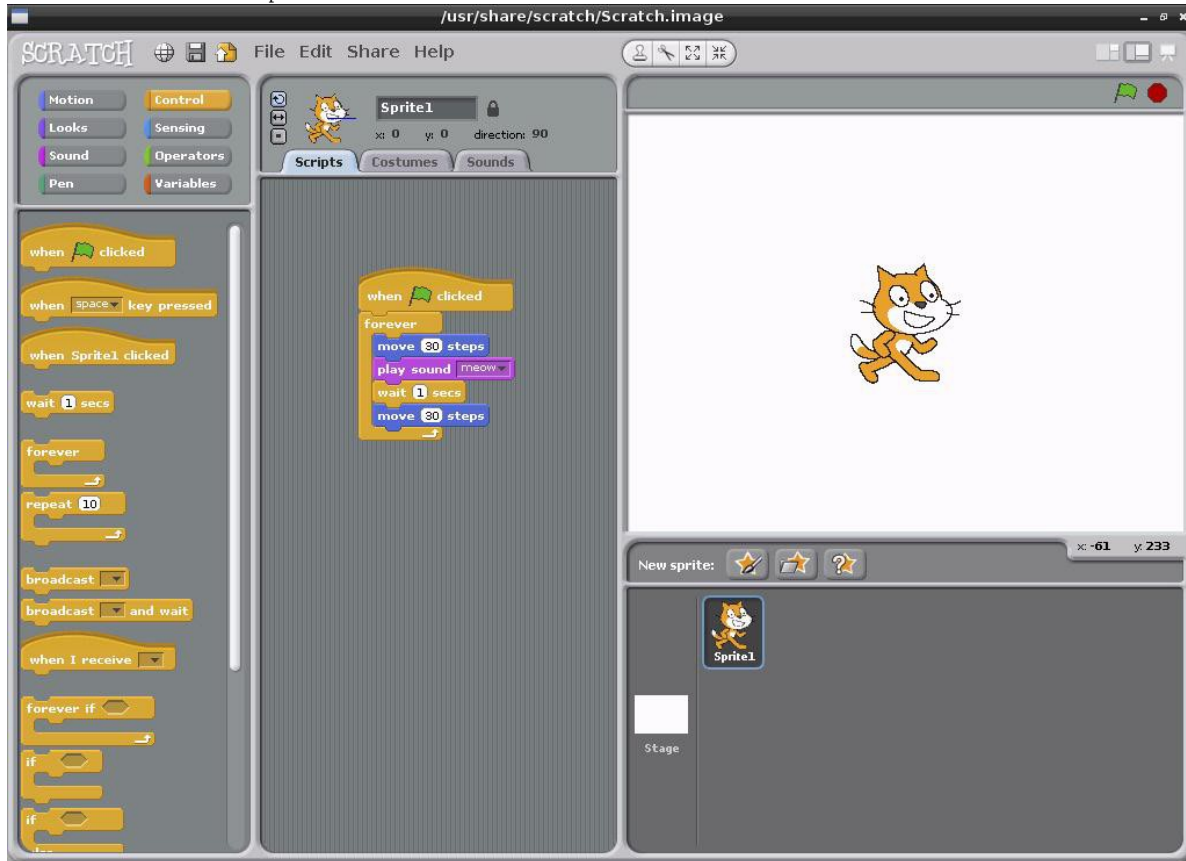
Este sencillo programa de animación puede ser extendido de varias maneras. Utilizando la opción **Nuevo Objeto** que se encuentra justo debajo de la escena en la parte derecha de la ventana de Scratch le permitirá al programador agregar más objetos que podrán moverse y reproducir sonidos de manera independiente. Agregándoles el bloque **decir** del primer ejemplo (o el bloque **pensar**, que sirve para crear una burbuja de pensamiento en vez de una burbuja de discurso) le permitirán crear un historietita animada.

Este simple ejemplo nos enseña conceptos importantes de programación: a pesar de tener no más de cinco bloques de largo, éste cubre el movimiento de objetos en distancias positivas o negativas, la reproducción de sonido y el concepto de los *retrasos* en un programa. Para introducir aún otro concepto (con el riesgo de volverse loco con los

ruidos repetitivos) trate de añadir un bloque **por siempre** desde la paleta de bloques Control (ver Figura 11-6). Esto agrega un *bucle* al programa, provocando que se ejecute su lista de instrucciones por siempre (o al menos hasta que el ruido sea demasiado para usted y decida hacer clic sobre el botón rojo de Detener que está en la parte superior de la escena).

**Figura 11-6**

Añadiendo un bucle infinito a la simple animación de Scratch



### Ejemplo 3: Un juego sencillo

Utilizar Scratch para crear una sencilla animación es interesante, pero el software también permite a los usuarios leer entradas desde el teclado para introducir algo de interactividad. Agregando algunos controles de animación sencillos al programa anteriormente descrito se puede crear un sencillo juego y al mismo tiempo, introducir los conceptos de *colisión de objetos*, *sentencias si (condicionales)* y *entradas*.

Para este ejemplo, inicie un nuevo proyecto Scratch (no olvide guardar el ejemplo anterior, si no lo ha hecho) y comience arrastrando un bloque **mover 10 pasos** al área de Programas. Esta vez, en lugar de decirle a los bloques de código que se ejecuten cuando se hace clic en el icono de la bandera, vaya a la paleta de bloques Control y arrastre un bloque **al presionar tecla espacio** encima del bloque mover.

Como su nombre lo sugiere, el bloque **al presionar tecla espacio** se pone en estado de alerta o mira por la entrada del usuario (en este caso, en espera de que la barra espaciadora sea presionada) y la utiliza como la detonante para la ejecución de una lista de bloques. El bloque trabaja en cualquier momento, si presiona la barra espaciadora ahora, el objeto obedecerá sus instrucciones y se moverá 10 pasos a la derecha.

El bloque **al presionar tecla espacio** se diferencia además del bloque **al presionar [icono de bandera]** en otra forma importante: este bloque puede ser personalizado. Haga clic en el botón de flecha hacia abajo que está junto a la palabra **espacio** para ver una lista de todas las teclas que el bloque puede ver, y a



continuación, seleccione **flecha derecha** en la lista para cambiar el bloque en un bloque al presionar teclado flecha derecha.

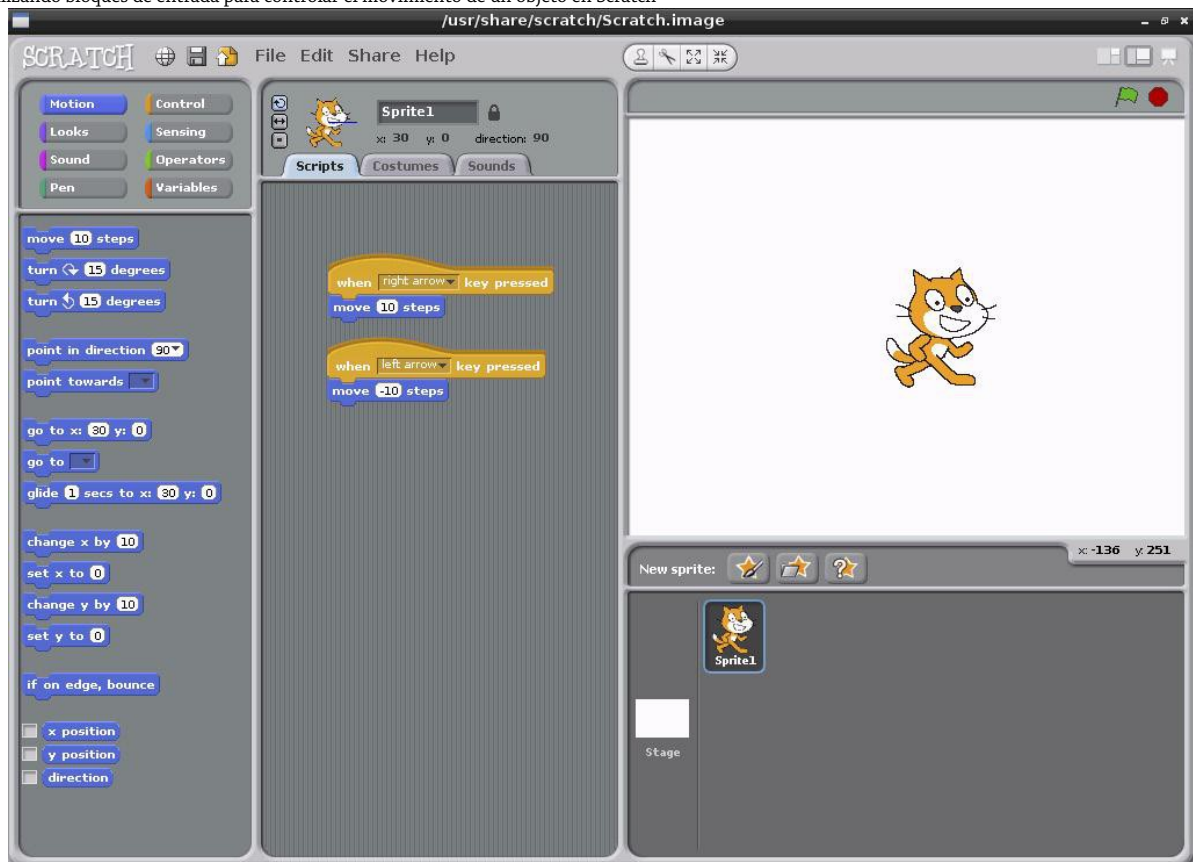
Un juego en donde jugador puede moverse sólo en una dirección no resulta muy divertido, así que arrastre un nuevo bloque al **presionar tecla** dentro del área de programas. Este bloque no puede conectarse a la lista de bloques existente (sólo se puede tener un simple bloque disparador o detonante), por lo tanto, inicie una nueva lista en algún sitio más abajo. Como lo hizo anteriormente, utilice el botón flecha abajo junto a la palabra **espacio** para personalizar el bloque, cambiándolo a un bloque al **presionar tecla flecha izquierda**. Por último, cambie la paleta de bloques de vuelta al modo **Movimiento** y conecte un bloque **mover 10 pasos** por debajo del nuevo bloque al **presionar tecla izquierda** y luego modifíquelo para que se lea: **mover -10 pasos**.

Si presiona las teclas izquierda y derecha ahora, verá que el gato se mueve de acuerdo a su entrada (ver Figura 11-7): presionando la flecha izquierda moverá el gato 10 pasos a la izquierda (aunque, por lo que a Scratch se refiere, se está moviéndose 10 pasos menos a la derecha), y presionando la flecha derecha moverá el gato 10 pasos a la derecha.

Ahora que el objeto puede ser movido por el jugador, es tiempo de darle al objeto algo que hacer. Como éste es sólo un juego muy muy sencillo, la meta debe ser también muy sencilla: recoger algo de comida. Haga clic en el botón **Escoger un Nuevo Objeto Desde Archivo**, ubicado al centro de los tres botones encima de la paleta de Objetos en la parte inferior derecha de la ventana de Scratch.

Figura 11-7

Utilizando bloques de entrada para controlar el movimiento de un objeto en Scratch



Un cuadro de diálogo aparecerá pidiéndole que elija un objeto: haga doble clic en la carpeta **Things** y en seguida haga doble clic en el objeto **CheesyPuffs** (que se muestra en la Figura 11-8). Esto colocará el objeto como una nueva entrada en la paleta de Objetos, dándole un nuevo objeto que podrá en el juego.

**CONSEJO**

El lenguaje Scratch es *multi-hilo* y parcialmente *orientado a objetos*. Esto significa que cada elemento en el programa, incluyendo los objetos o sprites, pueden tener su propio código adjunto y que cada sección de código se ejecutará de forma simultánea e independiente de cualquier otro bloque. Utilizadas apropiadamente, estas características permiten crear programas con bastante complejidad.

Por defecto, cualquier objeto que se añade a un proyecto Scratch aparecerá en el mero centro del escenario. Por consiguiente, esto puede ocultar el objeto del gato que ya existe. Haga clic sobre el nuevo objeto y arrástrelo a la derecha del gato para solucionar el problema.

Con el plato de churros hecho a un lado, aún sigue siendo demasiado grande para que nuestro gato antropomórfico de dos patas pueda comerlos. Haga clic en el botón Achicar Objeto, que está ubicado en la parte superior derecha del área del escenario y que se tiene la figura de cuatro flechas apuntando hacia adentro. Si no está seguro de qué botón se trata, pase el puntero del ratón por encima de cada icono para que vea una breve descripción de lo cada uno hace.

**Figura 11-8**

Añadiendo el objeto Cheesy-Puffs al proyecto del juego en Scratch



Haciendo clic en el botón Achicar Objeto (o en el botón Agrandar Objeto, el cual hace exactamente lo opuesto) se cambiará el cursor del ratón en un duplicado del icono del botón. Utilizando este nuevo cursor, haga clic en el objeto Cheesy-Puffs para reducirlo. Siga haciendo clic y el objeto seguirá reduciéndose. Una vez que el tamaño se ve más razonable, haga clic en cualquier lugar fuera del área del escenario para devolver el cursor del ratón a la normalidad. A continuación puede arrastrar el plato lo más cerca del borde derecho del escenario, si así lo desea.

Trate de utilizar las teclas de su teclado para mover el objeto del gato hacia el objeto Cheesy-Puffs. Como puede ver, cuando los objetos se encuentran, no pasa nada. Esto es porque el programa no contiene instrucciones que realizar



cuando los dos objetos se traslapen (lo que se conoce como una *colisión de objetos*) por lo tanto no hace nada. Esto puede ser rectificado con un nuevo tipo de bloque: un bloque *Sensorial*.

Con el objeto Cheesy-Puffs actualmente activo (su imagen debe mostrarse en la parte superior del panel Programas, pero si no es así, simplemente haga doble clic en el objeto sobre el escenario), cambie a la paleta de bloques al modo Sensorial haciendo clic sobre el botón Sensores. Desde la paleta Sensores, arrastre el bloque ¿tocando ? dentro del panel Programas.

Al igual que cuando el bloque **al presionar espacio** es utilizado para controlar el movimiento del objeto gato, el bloque ¿tocando ? puede ser personalizado. Haga clic en la flecha hacia abajo junto al signo de interrogación, y seleccione Objeto1 (el objeto del gato) desde la lista. Este bloque será ahora activado cuando los dos objetos colisionen.

---

#### CONSEJO

Puede darle un nombre un objeto haciendo clic en la casilla que está junto a su imagen en el panel de Programas y escribiendo dentro un nombre. Proporcione a los objetos nombres prácticos (por ejemplo, Gato, Plato, etc.) ayuda a recordar más fácilmente que es lo que el objeto hace en el programa.

---

Mire la forma del bloque ¿tocando Objeto1?. Como puede ver, su figura no tiene conectores parecidos a los de un rompecabezas ni arriba ni abajo, su forma es más parecida a un diamante (la misma figura utilizada para representar una sentencia condicional en un diagrama de flujo). Esto no es casualidad: la mayoría de los bloques de Sensores necesitan ser incrustados dentro de un bloque de Control a fin de operar.

Cambie la paleta de bloques a el modo Control y busque el bloque **si** (su figura es semejante a la de una letra C aplastada y llena de hoyos). Observe que el bloque **si** tiene un hueco con forma de diamante (la misma forma que el bloque ¿tocando Objeto1?) Arrastre el bloque **si** dentro del panel de Programas y luego arrastre el bloque ¿tocando Objeto1? dentro de su hueco con forma de diamante. El resultado es un bloque de dos colores que ahora se lee **si ¿tocando Objeto1?**.

Este bloque representa un **si condicional** en el programa: cuando se llega a este punto, cualquier código ubicado dentro de sus límites será ejecutado si y sólo si la *condición* se cumple. En este caso, la condición es que Objeto2 está siendo tocado por Objeto1. Utilizando los bloques lógicos **y**, **o** y **no** de la paleta de bloques Operadores, es posible construir complejas sentencias condicionales.

## Lógica Booleana

Llamada así por George Boole, la lógica Booleana o álgebra de Boole es un concepto clave para comprender cómo las computadoras funcionan. En Scratch, la lógica booleana está implementada en tres ladrillos de Operadores: **y**, **o** y **no**.

El operador **y** requiere que dos entradas (en el caso de Scratch, 2 bloques de Sensores) sean ambas verdaderas antes de que su propia salida sea verdadera. Si ninguna de las entradas es verdadera, o si ambas entradas son falsas, su propia salida será falsa. Como un ejemplo, puede utilizar este operador para comprobar si un objeto está tocando otros dos objetos.

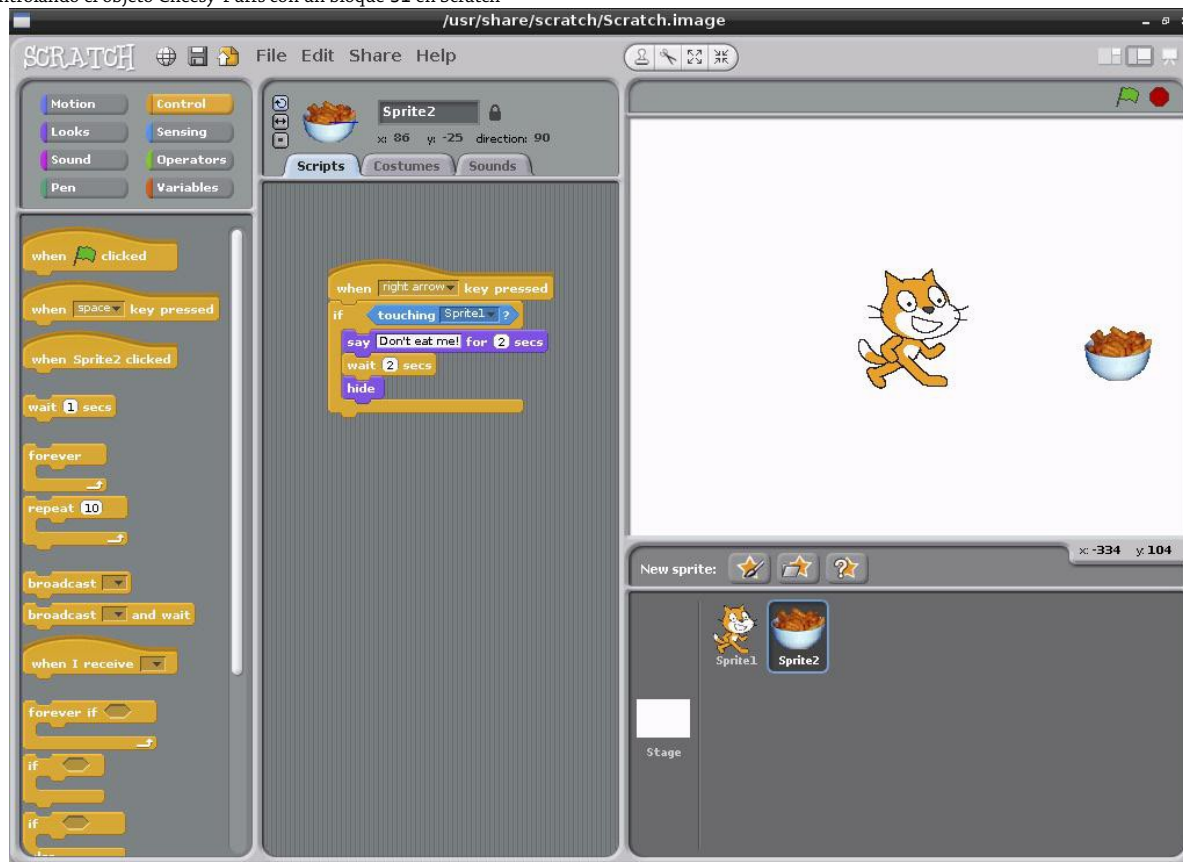
El operador **o** requiere que una o alguna de sus dos entradas sean verdaderas. Si cualquiera de sus entradas es verdadera, la salida del operador será también verdadera. Este es un modo práctico de reutilizar código: si usted tiene varios objetos que son letales para el objeto jugador, un solo bloque de código puede ser utilizado con el operador **o** para detonar cuando alguno de los objetos enemigos ha sido tocado.

El operador **no**, por último, es conocido como un inversor: cualquiera que sea su única entrada, dará como salida la opuesta. Si su entrada es falsa, entonces la salida será verdadera; si la entrada es verdadera, entonces la salida será falsa.

Desde la paleta de bloques Apariencia, arrastre el bloque **decir ¡Hola! por 2 segundos** en el centro del bloque condicional **si ¿tocando Objeto1?**. Cambie el texto para que se lea **!No me comas!** y luego agregue un bloque de Control **esperar 1 segundos**, cambiando su valor a 2. Agregue un bloque al **presionar tecla espacio** en la parte superior, cambiando su valor para que se lea **al presionar tecla flecha derecha**. Por último, arrastre el bloque **esconder** desde la paleta Apariencia a la parte inferior del bucle para acabar con una lista de bloques como la que se muestra en la Figura 11-9.

**Figura 11-9**

Controlando el objeto Cheesy-Puffs con un bloque si en Scratch

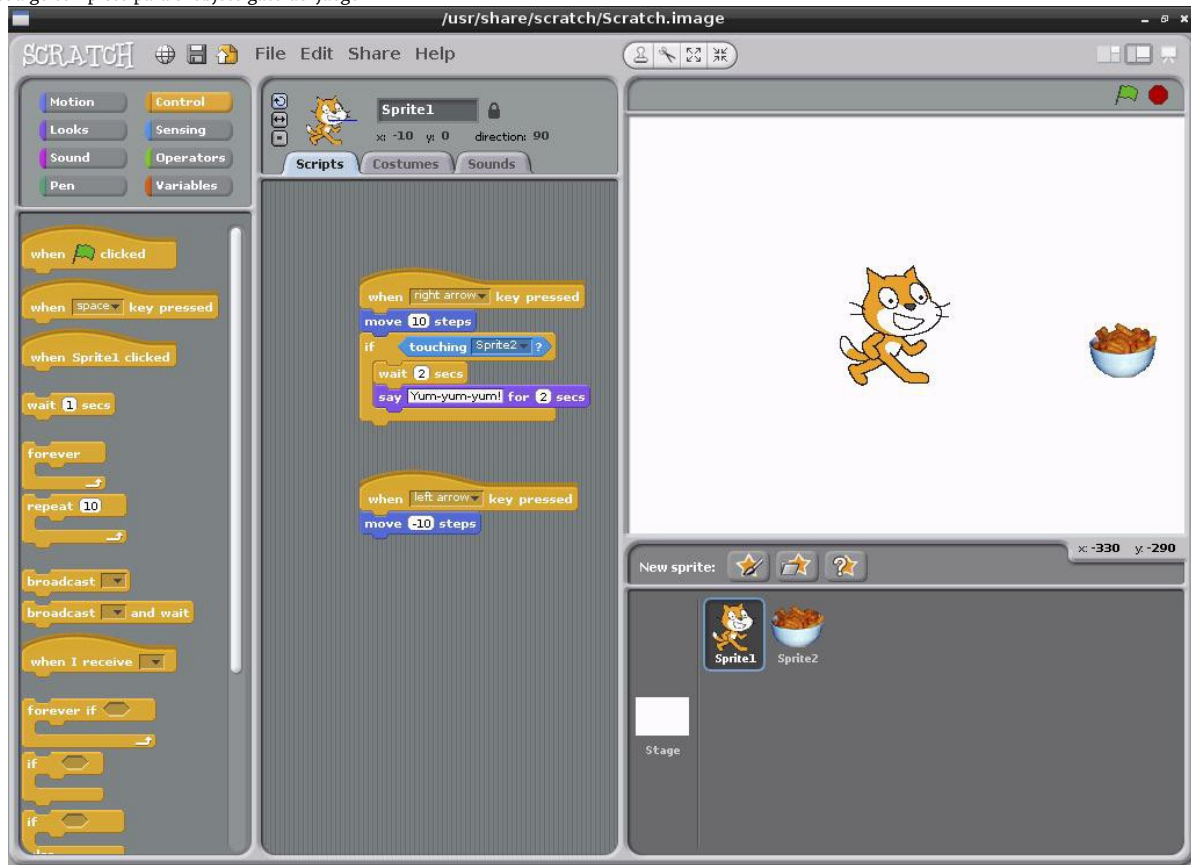


Haga doble clic en el objeto del gato sobre el escenario para volver a editar sus programas. El programa que creó para el objeto Cheesy-Puffs desaparecerá, pero no se preocupe (su programa también será guardado pero sólo será visible cuando se está editando aquel objeto particular).

Arrastre otro bloque **si** desde la paleta Control para que quede junto al otro bloque sensorial **¿tocando?**, y esta vez, cambie el bloque Sensorial para que ambos bloques juntos se lean **¿si tocando Objeto2?**. Dentro de este bloque, inserte un bloque de Control **esperar 1 segundos** con su valor cambiado a 2 y un bloque de Apariencia **decir ¡Hola! por 2 segundos** con el mensaje cambiado para leerse **¡Yum-yum-yum!**. Por último, arrastre todo el bloque apilado para que se conecte en la parte inferior del bloque **al presionar tecla flecha derecha** existente, justo abajo del bloque **mover 10 pasos**. El código final para el objeto del gato debe verse parecido a los bloques en la Figura 11-10.

Figura 11-10

El código completo para el objeto gato del juego



Si mueve el gato hacia el objeto del plato Cheesy-Puffs con la tecla de flecha derecha, el juego comienza. Cuando el gato alcanza los Chesy-Puffs, el cambio del diálogo ocurre y el plato debe desaparecer.

Aunque este ejemplo es eficaz para introducir algunos conceptos importantes de programación, no es la mejor forma que el juego pudo haber sido codificado. Scratch incluye un sistema de difusión de mensajes que permite al código anexo a un objeto comunicarse con el código anexo del otro, lo que le permite crear colisiones más claras en consecuencia.

Para experimentar con la difusión de mensajes, trate de utilizar los bloques **enviar a todos** y **al recibir** de la paleta Control. Un mensaje creado por un bloque de difusión en algún objeto puede detonar la ejecución del código en cualquier otro objeto utilizando el bloque **al recibir** con una señal o bandera determinada, lo que significa que puede utilizar este bloque para enlazar varios objetos y sus códigos juntos.

## Robótica y Sensores

Además de las entradas y salidas puramente basadas en software, también es posible vincular Scratch con el hardware externo utilizando placas de circuito con sensores y sistemas de robótica. Gracias a su naturaleza multiproceso y al potente sistema de mensajería entre procesos, Scratch puede ser utilizado para crear un motor robótico sorprendentemente avanzado y es por mucho la forma más fácil de colocar una Raspberry Pi como el corazón de un robot sencillo.

A diferencia de Python, que es un lenguaje de programación más potente pero significativamente más complejo, Scratch no puede hacer uso del puerto GPIO de la Raspberry para comunicarse con el hardware externo. Por consiguiente, es necesario contar con un poco de equipo adicional antes de poder interactuar con el mundo físico.

## Sensibilidad con la PicoBoard

Diseñada por la compañía SparkFun electronics, la PicoBoard es un módulo de expansión que se conecta a una computadora que ejecuta Scratch y dispone de un deslizador variable de entrada, un sensor de luz, un botón, un sensor de sonido y cuatro pares de pinzas caimanes que pueden ser utilizadas para monitorear dispositivos eléctricos externos.

Actualmente hay dos tipos de PicoBoard: el modelo original, que utiliza una conexión serie, y su reemplazo, que utiliza una conexión USB. El último modelo es soportado por la Raspberry Pi. Basta con conectarlo a un puerto USB libre o en algún puerto del concentrador USB que tenga conectado a su RasPi, reiniciar Scratch y entonces será capaz de utilizar sus diversas funciones dentro de la interfaz de Scratch.

Los detalles sobre la PicoBoard se pueden encontrar en <http://www.sparkfun.com/products/10311>.

## Robótica con LEGO

Además de añadir módulos de sensor, es posible controlar un sistema robótico externo a través de Scratch. El kit de robótica LEGO Education WeDo está diseñado para conectar motores, sensores de distancia y sensores de velocidad a una computadora (todos los cuales puede ser controlados utilizando bloques especializados encontrados en Scratch).

Para los niños, el kit WeDo es una excelente manera de empezar con la robótica, ya que los componentes están acoplados en forma de piezas de LEGO regular y no se tiene el riesgo de recibir alguna descarga eléctrica cuando algo se conecte de manera equivocada. El kit WeDo se jacta de compatibilidad con los kits de LEGO regular y LEGO Technik también, lo que significa que es posible construir grandes proyectos rápida y fácilmente.

Más información sobre el uso del kit LEGO Education WeDo con Scratch puede encontrarse en <http://info.scratch.mit.edu/WeDo>.

## Lectura Sugerida

Aunque este capítulo sirve como una breve introducción a Scratch, está lejos de ser una revisión exhaustiva. Es también un poco pesada para los lectores más jóvenes, que tienden a aprender más rápidamente cuando sus lecciones son acompañadas por muchas coloridas ilustraciones.

La sección Soporte del sitio oficial de Scratch, hospedado por el MIT en <http://info.scratch.mit.edu/es/Support>, incluye un enlace a la Guía de Inicio de Scratch en formato PDF. La guía describe cómo trabajar con Scratch de una manera colorida y amable para los niños, es una gran herramienta educativa. Esta experiencia de aprendizaje puede acrecentarse todavía más si se combina con las Tarjetas de Scratch, que son tarjetas flash descargables que contiene explicaciones sobre cada uno de los tipos de bloques que se encuentran en Scratch.

El MIT también tiene un foro de usuarios de Scratch, lo que permite a los aficionados del software aprender juntos y compartir soluciones para sus problemas comunes. La membresía es gratuita y el sitio funciona bien utilizando el navegador web Midori incluido en la distribución Raspbian para la Raspberry Pi. Se puede acceder al foro en <http://scratch.mit.edu/discuss/>.

Sin embargo, la forma más fácil de mejorar sus habilidades con Scratch, es jugando. El nombre Scratch tiene sus raíces en el turntablism: cuando un DJ hace girar un disco, la aguja crea un sonido de rayado. Así como los DJs mezclan canciones existentes en algo nuevo, se anima a los aficionados de Scratch a subir sus creaciones al sitio oficial para que otros los descarguen, examinen, modifiquen y los mezclen. El sitio oficial de Scratch actualmente hospeda más de 2.5 millones de programas Scratch, convirtiéndolo en una fuente perfecta para aprender cómo están utilizando Scratch para crear proyectos y para compartir sus ideas propias con los demás usuarios. Una lista con los proyectos compartidos más recientes se puede encontrar en <http://scratch.mit.edu/explore/>.

# **Capítulo 12**

Una Introducción a Python

La primera parte del nombre de la Raspberry Pi proviene de una antigua tradición que utiliza el nombre de una fruta para denominar a un nuevo sistema informático (desde las microcomputadoras clásicas como el Acorn, Apricot y Tangerine hasta las marcas más modernas reconocidas hoy en día Apple y Blackberry) pero la segunda mitad de su nombre tiene su origen en el lenguaje de programación *Python*.

## Presentando a Python

Flexible y poderoso, Python fue originalmente desarrollado a finales del año 1980 en el Instituto Nacional de Investigación de Matemáticas y Ciencias de la Computación por Guido van Rossum como un sucesor del lenguaje ABC. Desde su introducción, Python ha crecido en popularidad gracias a que cuenta con una clara y expresiva sintaxis de desarrollado con un enfoque que garantiza que el código es legible.

Python es un *lenguaje de alto nivel*. Esto significa que el código Python es escrito en un inglés ampliamente reconocido, dotando a la Raspberry Pi de comandos rápidos de aprender y fáciles de comprender. Esto se encuentra en un contraste marcado con los lenguajes de *bajo nivel*, tales como el ensamblador, que están más apegados a cómo la computadora “piensa” pero son casi imposibles de comprender por un humano sin experiencia. La naturaleza del alto nivel y la clara sintaxis de Python lo convierten en una herramienta valiosa para cualquier persona que desea aprender a programar. Es también el lenguaje recomendado por la Fundación Raspberry Pi para aquellas personas que buscan progresar desde el sencillo Scratch (descrito en el Capítulo 11, “Una Introducción a Scratch”) a algo más “práctico” de programación.

Python está publicado bajo una licencia de código abierto y se encuentra disponible de manera gratuita para todos los sistemas informáticos con Linux, OS X y Windows. Python soporta multiplataforma, lo que significa que el software escrito en Python de la RasPi puede ser utilizado también en las computadoras que funcionan con casi cualquier otro sistema operativo (excepto si se tratan de programas hagan uso del hardware específico de la RasPi, como el puerto GPIO). Para aprender cómo Python puede utilizarse para controlar este puerto, consulte el Capítulo 13, “Aprendiendo a Hackear Hardware”.

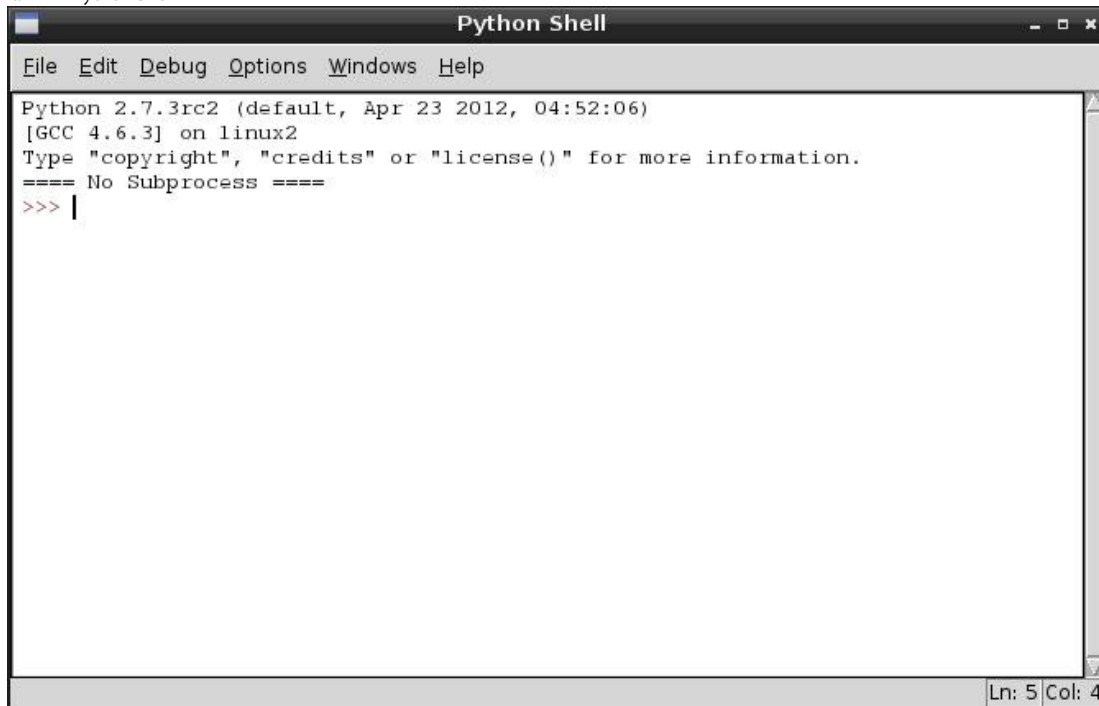
## Ejemplo 1: Hola Mundo

Como aprendió en el Capítulo 11, “Una Introducción a Scratch”, la manera más fácil de aprender un nuevo lenguaje de programación es creando un proyecto que imprima “¡Hola Mundo!” en la pantalla. En Scratch, sólo tenía que arrastrar y colocar bloques de código escrito previamente, pero en Python, necesita escribir este programa totalmente a mano.

Un proyecto Python no es más que un simple archivo de texto que contiene las instrucciones escritas para que la computadora las reproduzca. Este archivo puede ser creado utilizando cualquier editor de texto. Por ejemplo, si le gusta trabajar en la consola o en una ventana de terminal, puede utilizar nano; o si prefiere una interfaz gráfica de usuario (GUI), puede utilizar Leafpad. Otra alternativa es utilizar un entorno de desarrollo integrado (IDE) como IDLE, que proporciona funcionalidad específica de Python que no se puede encontrar en algún editor de texto estándar, como pueden ser: la comprobación de la sintaxis, facilidades de depuración y la habilidad para ejecutar su programa sin tener que salir del editor. Este capítulo le dará instrucciones sobre cómo crear archivos Python utilizando el IDLE, pero claro, el programa IDE que elija para programar dependerá de usted. Este capítulo también incluye instrucciones para saber cómo ejecutar los archivos creados directamente desde la terminal, los cuales pueden ser utilizados en conjunto con cualquier editor de texto u otro IDE.

Para comenzar el proyecto Hola Mundo, abra IDLE desde el menú Programación en el entorno de escritorio de la distribución Raspbian. Si no está utilizando IDLE, cree un documento en blanco en su editor de textos favorito y salte el resto de este párrafo. Por defecto, IDLE se abre en modo Python shell (ver Figura 12-1), en este modo cualquier instrucción que escriba en la ventana inicial será inmediatamente ejecutada. Para crear un nuevo proyecto Python que será ejecutado más adelante, haga clic en el menú File y seleccione New Window para abrir un nuevo archivo en blanco.

Figura 12-1  
La ventana IDLE Python Shell



---

**CONSEJO**

Seleccionar IDLE 3 desde el menú Programación en lugar de IDLE cargará la versión de Python 3.0. Este utiliza una forma ligeramente diferente del lenguaje o *sintaxis* que puede ocasionar problemas con los programas escritos en Python 2. Todos los ejemplos en este capítulo fueron escritos pensando en Python 2, por lo tanto asegúrese de cargar IDLE y no IDLE 3.

---

Es buena práctica iniciar todos los programas Python con una línea conocida como *shebang*, la cual obtiene su nombre de los caracteres # y ! al comienzo de la línea. Esta línea le dice al sistema operativo en dónde debe buscar los archivos de Python. Aunque esto no es completamente necesario para los programas que serán ejecutados dentro del IDLE o que invoquen a Python explícitamente en la terminal, es requerido para los programas que serán ejecutados directamente cuando sean llamados por sus nombres de archivo de programa.

Para asegurarnos que los programas se ejecutarán sin importar en donde está instalado el ejecutable de Python, la primera línea de su programa debería decir lo siguiente:

```
#!/usr/bin/env python
```

Esta línea le dice al sistema operativo que busque en la variable de entorno *\$PATH* (que es donde Linux almacena la ubicación de los archivos que pueden ser ejecutados como programas) la ubicación de Python, la línea *shebang* debería funcionar en cualquier distribución que utilice la RasPi. La variable *\$PATH* contiene una lista de directorios en donde los archivos ejecutables se encuentran almacenados y es utilizada para localizar los programas cuando usted escribe sus nombres en la consola o en una ventana de terminal.

Para conseguir el objetivo de imprimir un mensaje, debe de utilizar el comando `print` de Python. Como su nombre lo sugiere, este comando imprime texto en el dispositivo de salida (por defecto, en la consola o ventana de terminal desde donde el programa está siendo ejecutado). Su uso es muy simple: cualquier texto siguiente a la palabra `print` y colocado entre comillas será imprimido en el dispositivo de salida estándar. Introduzca la siguiente línea en su nuevo proyecto:

```
print "¡Hola, Mundo!"
```

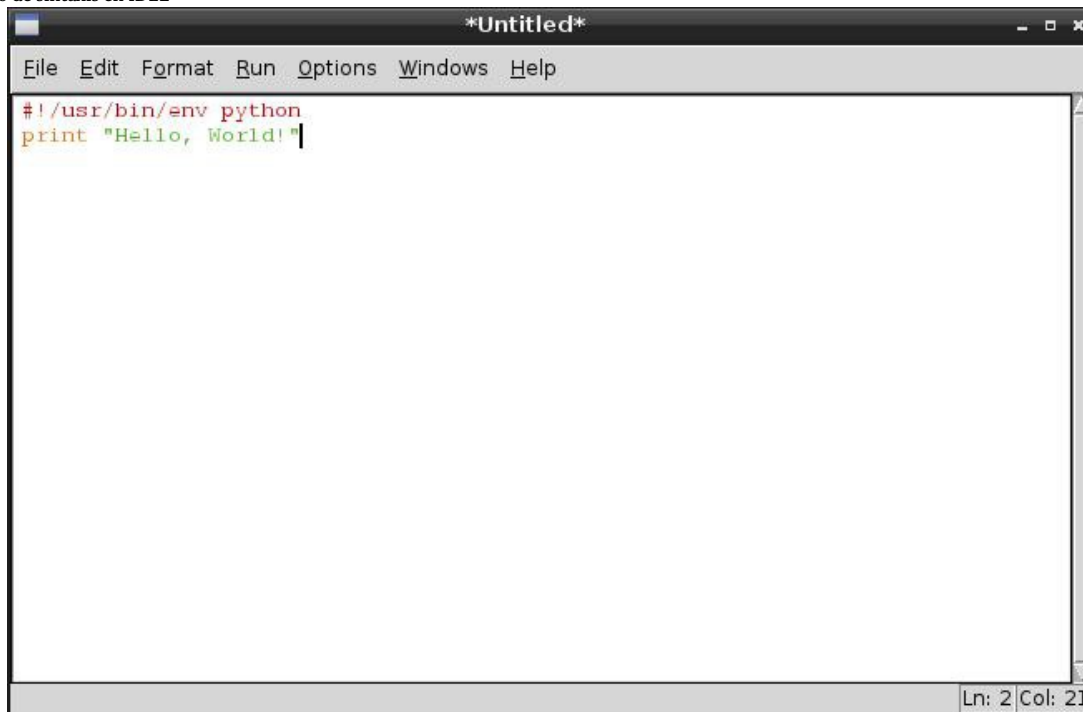
El programa final debe verse similar a este:

```
#!/usr/bin/env python
print "¡Hola Mundo!"
```

Si está creando el programa de este ejemplo en IDLE en lugar de crearlo en un editor de texto plano, se dará cuenta que el texto es multicolor (vea la Figura 12-2, donde los colores están representados con diferentes tonos gris en la edición impresa). Esta es una característica conocida como *coloreado de sintaxis*, es una característica que tienen los IDEs y las herramientas de edición de texto más avanzadas. El coloreado de sintaxis cambia el color de las secciones del texto de acuerdo a su función, con el fin de hacer que el programa sea más fácil de comprender de un sólo vistazo. También facilita detectar los llamados *errores de sintaxis* que ocurren al olvidar colocar una comilla final en algún comando `print`. Para este breve ejemplo, el coloreado de sintaxis no es necesario, pero para programas más grandes, será una herramienta invaluable que nos ayudará a localizar errores.

Figura 12-2

Coloreado de sintaxis en IDLE



Antes de ejecutar su programa, guárdelo como `holamundo.py` utilizando el menú File. Si está utilizando IDLE, al archivo será agregada la extensión `.py` de manera automática. Si está usando un editor de textos, asegúrese de escribir `.py` al final del nombre del archivo (no `.txt`) al momento de guardarlo. Esta extensión indica que el archivo contiene código Python (aunque Python es lo suficientemente inteligente para ejecutarlo, incluso si es guardado con una extensión de archivo diferente).

El cómo hacer para ejecutar el archivo python dependerá de si está utilizando IDLE o un editor de texto. En IDLE, simplemente elija Run Module desde el menú Run, o presione la tecla F5 en su teclado. Esto cambia la ventana activa a la consola Python y se ejecuta el programa. A continuación, debería ver el mensaje `¡Hola Mundo!` Mostrarse en la pantalla en azul (ver Figura 12-3). Si no es así, verifique su sintaxis (en particular, compruebe que tiene las comillas al comienzo y al final del mensaje en la línea `print`).

Si ha creado el programa `holamundo.py` en un editor de textos, necesitará abrir una ventana de terminal desde el menú Accesorios del escritorio. Si ha guardado el archivo en cualquier lugar excepto en su directorio home, tendrá además que utilizar el comando `cd` para cambiarse a ese directorio (ver Capítulo 3, "Administración del Sistema



Linux”). Una vez que esté en el directorio correcto, pude ejecutar su programa escribiendo lo siguiente:

```
python holamundo.py
```

Figura 12-3

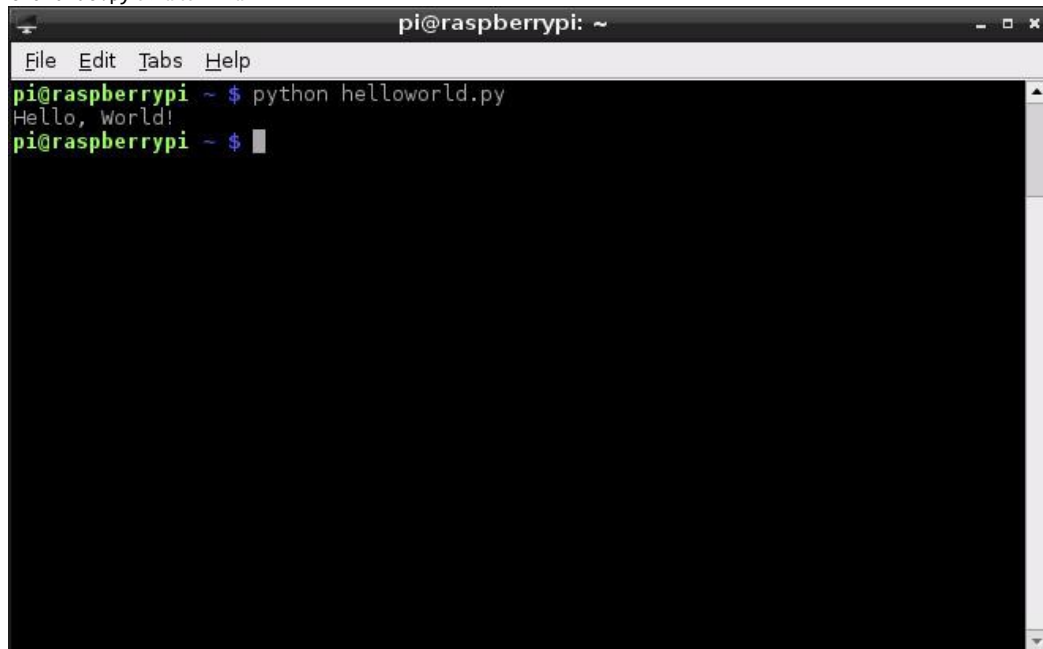
Ejecutando holamundo.py en IDLE



Esto le dice al sistema operativo que ejecute Python y luego cargue el archivo `holamundo.py` para su ejecución. A diferencia de la consola Python en IDLE, Python se cerrará cuando llegue al final del archivo y lo pondrá devuelta en la terminal. El resultado, sin embargo, será el mismo: el mensaje ¡Hola Mundo! Impreso en la salida estándar (ver Figura 12-4).

Figura 12-4

Ejecutando holamundo.py en la terminal



## Creando Programas Python Ejecutables

Normalmente, la única manera de ejecutar un programa Python es diciéndole al software Python que abra el archivo. Sin embargo, con la línea shebang al principio del archivo, es posible ejecutar el archivo directamente sin tener que llamar a Python primero. Esta manera de ejecutar los archivos resulta útil para crear sus propias herramientas que podrán ser ejecutadas desde la terminal: una vez copiado dentro de una ubicación en la variable de entorno \$PATH del sistema, el programa Python puede ser llamado con tan sólo escribir su nombre.

Primero, necesita decir a Linux que el archivo Python debe ser marcado como ejecutable (un atributo que indica que el archivo se trata de un programa). Para proteger al sistema del malware que se descarga de la Internet este atributo no se establece de manera automática, sólo los archivos que son marcados como ejecutables podrán ser cargados. Para convertir el archivo holamundo.py en ejecutable, utilice el comando chmod (descrito en el Capítulo 3, “Administración del Sistema Linux”) escribiendo lo siguiente:

```
chmod +x holamundo.py
```

Ahora trate de ejecutar el programa directamente escribiendo lo siguiente:

```
./holamundo.py
```

A pesar de que usted no llama al programa Python, el programa holamundo.py debe de ejecutarse del mismo modo que si hubiera escrito python holamundo.py. El programa sólo puede ser ejecutado si se invoca con su ubicación completa (/home/pi/holamundo.py) o desde el directorio actual utilizando ./ como la ubicación. Para hacer que el archivo sea accesible en la misma forma que lo son los otros comandos de terminal, nuestro programa necesitará ser copiado a /usr/local/bin con el siguiente comando:

```
sudo cp holamundo.py /usr/local/bin/
```

El prefijo sudo es necesario, ya que, por razones de seguridad, los usuarios sin privilegios no pueden escribir en el directorio /usr/local/bin. Con el archivo holamundo.py ubicado en el directorio /usr/local/bin (que está incluido en la variable \$PATH) ya puede ser ejecutado desde cualquier directorio con tan sólo escribir su nombre. Intente cambiarse a un directorio distinto, y enseguida, ejecute el programa escribiendo lo siguiente:

```
holamundo.py
```

Para crear sus programas hechos a la medida de modo que parezcan utilidades nativas, puede cambiarles el nombre quitando la extensión .py. Para cambiar el nombre del programa holamundo.py de esta manera, sólo escriba la siguiente línea en la terminal:

```
sudo mv /usr/local/bin/holamundo.py /usr/local/bin/holamundo
```

Una vez renombrado, el programa puede ejecutarse simplemente escribiendo holamundo en la terminal o consola.

## Ejemplo 2: Comentarios, Entradas, Variables y Bucles

Aunque el programa Hola Mundo es una introducción suave al lenguaje, no es muy interesante. Por su naturaleza, ese programa sólo cubre lo básico y no introduce algunos de los conceptos necesarios para poder crear programas útiles o interesantes. El siguiente ejemplo, no obstante, utiliza algunas de las herramientas básicas necesarias para crear programas interactivos en Python.

Como en el Ejemplo 1, comience abriendo un documento nuevo en el IDLE o en su editor de textos y luego inicie el programa con la siguiente línea shebang:

```
#!/usr/bin/env python
```

Como se mencionó anteriormente, esta línea no es estrictamente necesaria a menos que el programa vaya a ser convertido en ejecutable, pero no hace ningún daño y es un buen hábito poner esta línea.

A continuación, agregue un comentario al programa para proporcionar un contexto en caso de que necesitará abrir el archivo en una fecha posterior. Tome en cuenta que el comentario necesita ser introducido en una sola línea:

#### # Ejemplo 2: Un programa Python de la Guía de Usuario de Raspberry Pi

En Python, cualquier cosa después de un símbolo hash (#) (con la excepción de la línea shebang) es tratada como un comentario. Cuando Python se encuentra con un comentario lo ignora y salta a la siguiente línea. Comentar su código es una muy buena práctica: aunque usted pueda saber lo que una sección de código en particular hace ahora, las cosas podrían no ser tan claras cuando abra el archivo nuevamente seis meses más tarde de haberlo escrito. Los comentarios ayudan a hacer que el código sea más fácil de mantener, si usted decide compartir su código con otras personas, los comentarios les ayudarán a comprender que es lo que se supone hace cada sección. Para programas sencillos, no es estrictamente necesario incluir comentarios (pero es un buen hábito colocarlos). Los comentarios pueden estar en su propia línea, como el comentario anterior, o al final de una línea, en cuyo caso Python ejecutará la línea de código hasta que llegue al símbolo de hash.

A continuación, se pregunta al usuario su nombre mediante la siguiente línea:

```
nombreUsuario = raw_input("¿Cual es su nombre? ")
```

Esta pequeña línea realmente hace mucho. La primera parte, `nombreUsuario =`, le dice a Python que cree una nueva *variable* (un lugar para el almacenamiento de una pieza de información) llamada `nombreUsuario`. El símbolo de igual le dice a Python que a la variable debe asignarse lo que le sigue. Sin embargo, en este caso lo que le sigue no es una pieza de información, sino otro comando: `raw_input`. Esta es una herramienta diseñada para aceptar entradas de *cadena* (texto) desde el teclado y permite que un mensaje sea impreso en la salida predeterminada para que el usuario sepa que escribe. Este comando ayuda a mantener el programa sencillo (sin la capacidad de imprimir un mensaje que le diga al usuario que escribe, una segunda línea con un comando `print` sería necesaria). Recuerde dejar un espacio al final del mensaje, de lo contrario, la entrada del usuario iniciará inmediatamente después del signo de interrogación.

---

#### ADVERTENCIA

Cuando le pida al usuario escribir en texto, siempre utilice `raw_input`. Esta instrucción es más segura que el comando `input`, si sólo utiliza `input`, un usuario malintencionado puede *inyectar* su propio código dentro de su programa y hacer que falle o que trabaje contrario a sus intenciones.

---

Con el nombre del usuario ya almacenado de forma segura en la variable `nombreUsuario`, el programa puede comenzar a mostrarse inteligente. Dé la bienvenida al usuario utilizando la siguiente línea:

```
print "Bienvenido al programa,", nombreUsuario
```

Esta línea demuestra una función secundaria del comando `print` introducido en el Ejemplo 1: la capacidad de imprimir los contenidos de las variables. Este comando `print` está dividido en dos secciones: la primer sección imprime todo lo que se encuentra entre los dos símbolos de comillas y la coma le indica a `print` que existen más cosas que deberán ser imprimidas en la misma línea. Escribir el nombre de la variable `nombreUsuario` será más que suficiente para que Python sepa que debe imprimir el contenido de dicha variable, produciendo así un mensaje personalizado a nombre propio del usuario.

Este programa de ejemplo va a adquirir la forma de una sencilla y agradable calculadora. A diferencia del Ejemplo 1, este programa continuará ejecutándose hasta que el usuario exprese lo contrario. Esto se consigue utilizando un bucle o ciclo, al igual que en Scratch. Comience el bucle escribiendo las siguientes dos líneas:

```
otraVez = 1
while otraVez == 1:
```

La primera línea crea una nueva variable llamada `otraVez` y establece su valor a 1. La segunda línea comienza el bucle y le dice a Python que mientras la variable `otraVez` sea igual a 1 debe continuar la repetición cíclica a través del siguiente código. En cuanto a las pocas líneas que están escritas enseguida, necesitarán contar con *sangría*, para colocar ésta se insertan cuatro espacio al comienzo de cada línea. Estos espacios le dicen a Python qué líneas forman parte del bucle y cuáles otras están fuera del él. Si está utilizando IDLE, los espacios son automáticamente insertados; si está utilizando un editor de texto, recuerde insertar los espacios manualmente.

## ¿Por qué == ?

Anteriormente, ha estado utilizando un sólo símbolo de igual para establecer o asignar el valor de las variables. El bucle `while`, sin embargo, utiliza dos símbolos. Al utilizar dos símbolos de igual consecutivos se lleva a cabo una *evaluación*, la cual compara el valor de una variable con lo que le sigue. En cambio, un sólo símbolo de igual establece a la variable el valor que le sigue.

La evaluación doble igual sólo es verdadera si la variable coincide exactamente con el valor dado. Así como esta evaluación existen otros símbolos para realizar otras evaluaciones: `>` significa mayor que, `<` significa menor que, `>=` significa mayor o igual que, `<=` significa menor o igual que, y `!=` significa no igual a.

Utilizando estos símbolos de evaluación, se puede controlar el flujo de un programa de acuerdo a las reglas de la lógica Booleana. Para más información sobre la lógica Booleana, consulte el Capítulo 11, “Una Introducción a Scratch”.

En su forma más simple, una calculadora toma dos número como entrada y realiza una operación matemática con ellos. Para hacer que su calculadora funcione, primero pregunte los dos números al usuario con las siguientes líneas:

```
primerNumero = int(raw_input("Escriba el primer numero: "))
segundoNumero = int(raw_input("Escriba el segundo numero: "))
```

Estas líneas no sólo utilizan la instrucción `raw_input` para pedir los dos números, sino que también utilizan `int`. La instrucción `int` (abreviación de *integer*) le dice a Python que trate la entrada como un número en lugar de una cadena. Obviamente, esto es importante para un programa de calculadora, ya que ésta no es para calcular palabras.

Con los dos números almacenados como variables, el programa puede realizar sus cálculos. Escriba las siguientes líneas que tienen el cometido de sumar, restar y multiplicar los dos números y enviar el resultado al usuario:

```
print primerNumero, "sumado a",segundoNumero, "es igual a",
primerNumero + segundoNumero
print primerNumero, "menos", segundoNumero, "es igual a",
primerNumero - segundoNumero
print primerNumero, "multiplicado por", segundoNumero, "es igual a",
primerNumero * segundoNumero
```

Observe que mientras los operadores de suma y resta utilizan los símbolos de más y menos, la multiplicación utiliza el símbolo `*` (asterisco). Note también que no hay espacios de formato entre las comillas. Esto es porque Python automáticamente añade espacios en donde es requerido cuando imprime enteros y cadenas juntos. Finalmente, observe que no hay operación de división (la cual será indicada utilizando el símbolo `/`). Esto es porque el programa calculadora de ejemplo sólo utiliza números *enteros*, sin decimales o partes fracciones.

Aunque la parte de cálculo del programa se ha completado, el programa se ejecutará por siempre ya que actualmente no existe nada que le indique a Python cuando es el momento de salir del bucle. Para permitir al

usuario poder salir del programa, agregue la siguiente línea:

```
otraVez = int(raw_input("Escriba 1 para introducir mas numeros, o cualquier otro numero para salir: "))
```

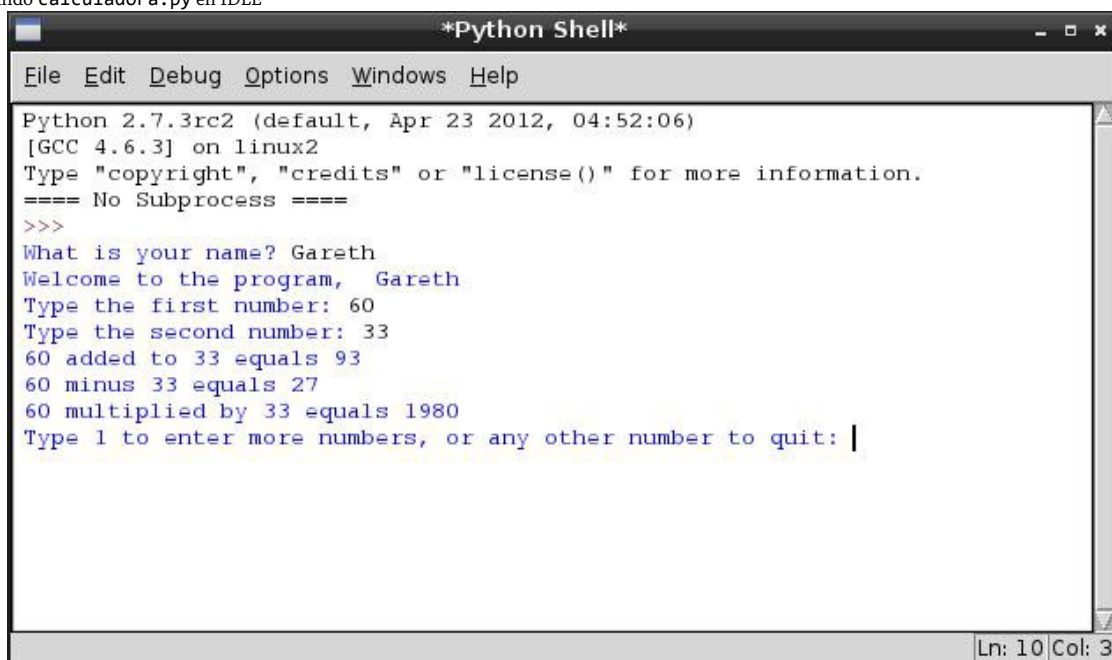
Esta instrucción permite al usuario cambiar la variable `otraVez` que controla el bucle `while`. Si el usuario introduce el número 1, la variable `otraVez` sigue siendo igual a 1 y el bucle se ejecutará de nuevo. Sin embargo, si el usuario introduce cualquier otro número, la evaluación ya no será verdadera (`otraVez` ya no es igual a 1) y el bucle terminará

El programa terminado debe verse parecido al siguiente:

```
#!/usr/bin/env python
# Ejemplo 2: Un programa Python de la Guía de Usuario de Raspberry Pi
nombreUsuario = raw_input("¿Cual es su nombre? ")
print "Bienvenido al programa,", nombreUsuario
otraVez = 1
while otraVez == 1:
    primerNumero = int(raw_input("Escriba el primer numero: "))
    segundoNumero = int(raw_input("Escriba el segundo numero: "))
    print primerNumero, "sumado a", segundoNumero, "es igual a",
    primerNumero + segundoNumero
    print primerNumero, "menos", segundoNumero, "es igual a",
    primerNumero - segundoNumero
    print primerNumero, "multiplicado por", segundoNumero, "es igual a",
    primerNumero * segundoNumero
    otraVez = int(raw_input("Escriba 1 para introducir mas numeros, o cualquier otro numero para salir: "))
```

Guarde el programa como `calculadora.py`, y ejecútelo eligiendo Run Module desde el menú Run en el IDLE o escribiendo `python calculadora.py` en la terminal. Introduzca su nombre de usuario cuando se le pida y luego proporcione los números que desea calcular (ver Figura 12-5) hasta que se aburra; escriba cualquier otro número distinto de 1 para salir del programa.

Figura 12-5  
Ejecutando `calculadora.py` en IDLE



```
*Python Shell*
File Edit Debug Options Windows Help
Python 2.7.3rc2 (default, Apr 23 2012, 04:52:06)
[GCC 4.6.3] on linux2
Type "copyright", "credits" or "license()" for more information.
==== No Subprocess ====
>>>
What is your name? Gareth
Welcome to the program, Gareth
Type the first number: 60
Type the second number: 33
60 added to 33 equals 93
60 minus 33 equals 27
60 multiplied by 33 equals 1980
Type 1 to enter more numbers, or any other number to quit: |
Ln: 10 Col: 3
```

Para ver más programas cortos que introducen conceptos importantes de Python, visite la página wiki oficial de Python Simple Programs en <http://wiki.python.org/moin/SimplePrograms>.

### Ejemplo 3: Juego con pygame

Para ilustrar el poder que tiene Python, el siguiente ejemplo crea un juego arcade completamente funcional basado en el clásico juego de la Serpiente. Para crear este juego, utilizaremos una librería externa de Python llamada `pygame`.

Originalmente escrita por Pete Shinnars, `pygame` es una colección de *módulos* python diseñados para añadir nuevas funcionalidades al lenguaje (funcionalidades específicamente diseñadas para hacer más fácil escribir juegos en Python). Cada módulo de `pygame` tiene el propósito de cubrir un aspecto necesario para desarrollar un juego, hay módulos para el sonido, los gráficos e incluso para el soporte de red. Aunque es posible escribir un juego en Python sin utilizar `pygame`, es mucho más fácil si saca ventaja del código ya escrito en la librería `pygame`.

Antes de que pueda escribir un programa `pygame`, necesita instalar la librería. Si está utilizando la distribución recomendada Raspbian, esto es tan simple como escribir lo siguiente en la consola o terminal:

```
sudo apt-get install python-pygame
```

Para las otras distribuciones, los archivos fuentes de `pygame` pueden descargarse desde el sitio oficial de `pygame` en <http://www.pygame.org/download.shtml>. Las instrucciones para su instalación se proporcionan en la misma página.

El comienzo de un programa `pygame` es el mismo que el de cualquier otro proyecto Python. Abra un documento nuevo en blanco (sea en el IDLE o en un editor de textos) y agregue la siguiente línea shebang en la parte superior:

```
#!/usr/bin/env python
```

Enseguida, necesita decir a Python que este programa utiliza los módulos de `pygame`. Para hacer esto, utilice la instrucción `import`, la cual le dice a Python que cargue un módulo externo (otro archivo Python) y lo haga accesible desde el programa actual. Escriba las siguientes dos líneas para importar los módulos necesarios dentro de su nuevo proyecto:

```
import pygame, sys, time, random
from pygame.locals import *
```

La primera línea importa el módulo principal `pygame` junto con los algunos otros módulos de Python: `sys`, `time` y `random`. Tales módulos también serán utilizados en este programa. Por lo general, una instrucción de un módulo debería ser invocada escribiendo su nombre, un punto y luego el nombre de tal instrucción, pero la segunda línea en el código anterior le dice a Python que cargue todas las instrucciones del módulo `pygame.locals` para que sean tratadas como si fueran instrucciones nativas. Por lo tanto, tendrá que mecanografiar menos cuando utilice estas instrucciones. Otros nombres de módulo, como `pygame.clock`, el cual es independiente de `pygame.locals` tendrá que ser escrito de manera completa.

Para configurar a `pygame` de tal manera que se encuentre listo para ser utilizando dentro de nuestro programa, escriba lo siguiente:

```
pygame.init()
fpsClock = pygame.time.Clock()
```

La primera línea le dice a `pygame` que se inicialice así mismo y la segunda línea, establece una nueva variable

llamada `fpsClock`, la cual será utilizada para controlar la velocidad del juego. A continuación, configuraremos una nueva superficie de pantalla para `pygame` (un `display` o superficie es el lienzo sobre el cual los objetos dentro del juego serán dibujados) con las dos líneas siguientes:

```
playSurface = pygame.display.set_mode((640, 480))
pygame.display.set_caption('Raspberry Snake')
```

En seguida, se definirán algunos colores para que el programa los utilice. Aunque este paso no es estrictamente necesario, nos ahorrará nuevamente algo de mecanografía: si deseará establecer que un objeto en particular sea de color rojo, simplemente utilice la variable `redColour` en lugar de llamar a la instrucción `pygame.Color` y memorizar los tres valores de color para el rojo, verde y azul. Para definir los colores en este programa de ejemplo, escriba las siguientes líneas:

```
redColour = pygame.Color(255, 0, 0)
blackColour = pygame.Color(0, 0, 0)
whiteColour = pygame.Color(255, 255, 255)
greyColour = pygame.Color(150, 150, 150)
```

Las siguientes líneas inicializan algunas de las variables del juego para que se encuentren listas y puedan ser utilizadas. Este es un paso importante, porque si estas variables son dejadas en blanco cuando el juego comience, Python no sabrá que hacer. No se preocupe en este momento de lo que cada variable hace, sólo escriba las siguientes líneas:

```
snakePosition = [100,100]
snakeSegments = [[100,100],[80,100],[60,100]]
raspberryPosition = [300,300]
raspberrySpawned = 1
direction = 'right'
changeDirection = direction
```

Observe que tres de las variables (`snakePosition`, `snakeSegments` y `raspberryPosition`) están inicializadas con una lista de valores separados por comas. Esto le dice a Python que cree las variables como *listas* (un número de diferentes valores almacenados dentro de un único nombre de variable). Más adelante, verá como puede acceder a los valores individuales almacenados dentro de una lista.

A continuación, necesita definir una nueva función (un fragmento de código Python al que se puede recurrir más adelante en el programa). Las funciones son útiles para evitar repetir código y hacer que el programa sea más fácil de comprender. Si tiene un conjunto específico de instrucciones que son requeridos en varios puntos del mismo programa, el utilizar `def` para crear una función significará que usted sólo tendrá que escribir este código una sola vez (y si más adelante necesita hacer algún cambio, dispondrá de un único lugar en donde hacerlos). Escriba las siguientes líneas para definir la función `gameOver`:

```
def gameOver():
    gameOverFont = pygame.font.Font('freesansbold.ttf', 72)
    gameOverSurf = gameOverFont.render('Game Over', True, greyColour)
    gameOverRect = gameOverSurf.get_rect()
    gameOverRect.midtop = (320, 10)
    playSurface.blit(gameOverSurf, gameOverRect)
    pygame.display.flip()
    time.sleep(5)
    pygame.quit()
    sys.exit()
```

Del mismo modo que con los bucles, el código dentro de una función debe tener un sangría. Cada línea siguiente a

la instrucción `def` debe de contar con cuatro espacios al principio; si está utilizando IDLE, estos espacios serán insertados automáticamente, pero si está utilizando un editor de texto, necesitará insertarlos a mano. Después de la última línea que tiene la función (`sys.exit`) puede eliminar la sangría.

La función `gameOver` utiliza una selección de comandos de `pygame` para realizar una tarea simple: escribir las palabras `Game Over` en la pantalla con una fuente de letra grande, hacer una pausa durante 5 segundos y luego, salir tanto de `pygame` como del propio Python. Puede verse extraño disponer de las instrucciones para salir del juego antes de que el juego haya comenzado, pero las funciones deben siempre ser definidas antes de ser llamadas. Python no ejecutará estas instrucciones hasta que se le indique que lo haga a través de la instrucción recién creada `gameOver`.

Con el principio del programa completo, es hora de comenzar con la sección principal. Esta tendrá lugar en un bucle infinito (un bucle `while` que nunca termina o sale). Esto será así para que el juego pueda continuar ejecutándose hasta que el jugador muera al chocar con una pared o al comerse su propia cola. Comience el bucle infinito con la siguiente línea:

```
while True:
```

Sin nada que evaluar, Python se asegurará de mirar si `True` es verdad. Como éste es siempre el caso, el bucle continuará su ejecución por siempre o al menos, hasta que usted le diga a Python que salga fuera del bucle llamando a la función `gameOver`.

Continúe el programa con las siguientes líneas, prestando atención a los niveles de sangría:

```
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
        elif event.type == KEYDOWN:
```

La primer línea que aparece justo después del inicio del bucle `while`, deberá de tener una sangría de cuatro espacios, esa línea a su vez es un bucle, que utiliza una instrucción `for` para comprobar los eventos de `pygame` tales como las teclas que son presionadas. Por consiguiente, la línea debajo de aquella instrucción necesita tener una sangría de cuatro espacios adicionales para un total de ocho, pero línea que sigue, también es un bucle, que utiliza una instrucción `if` para comprobar si el usuario ha presionado alguna tecla. Por lo tanto, la siguiente línea (`pygame.quit()`) tendrá una sangría adicional de cuatro espacios para tener un total de 12 espacios. Esta progresión lógica de la sangría le dice a Python en dónde comienza y termina cada bucle, lo cual es importante: si un número equivocado de espacios se utiliza, el programa no funcionará correctamente. Esta es la razón por la que se recomienda utilizar un entorno de desarrollo como IDLE, que trata de darle sangría automáticamente al código cuando es necesario, puede llegar a ser más fácil que utilizar un editor de texto plano.

Un bucle `if` le dice a Python que examine si una evaluación particular es verdadera. La primer evaluación, `if event.type == QUIT`, le dice a Python que ejecute el código con sangría debajo suyo si `pygame` reporta un mensaje `QUIT` (el cual ocurre cuando el usuario presiona la tecla `Escape`). Las dos líneas de abajo deben ser familiares desde la función `gameOver`: estas le dicen a `pygame` y a Python que cierre la ventana y salga.

La línea que comienza con `elif` es utilizada para extender bucles `if`. Una instrucción `elif` (abreviación par `else if`) es evaluada cuando una instrucción `if` anterior fue encontrada como falsa. En este caso, la instrucción `elif` es utilizada para ver si `pygame` está reportando un evento `KEYDOWN`, que se manifiesta cuando el usuario está presionando alguna tecla. Al igual que con la instrucción `if`, el código a ejecutarse cuando un `elif` sea verdadero debe de tener una sangría adicional de cuatro espacios más a cualquier sangría que la instrucción `elif` tenga. Escriba las siguientes líneas para darle a la instrucción `elif` algo que hacer cuando el usuario presiona alguna tecla:



```

if event.key == K_RIGHT or event.key == ord('d'):
    changeDirection = 'right'
if event.key == K_LEFT or event.key == ord('a'):
    changeDirection = 'left'
if event.key == K_UP or event.key == ord('w'):
    changeDirection = 'up'
if event.key == K_DOWN or event.key == ord('s'):
    changeDirection = 'down'
if event.key == K_ESCAPE:
    pygame.event.post(pygame.event.Event(QUIT))

```

Estas instrucciones modifican el valor de la variable `changeDirection`, que es utilizada para controlar la dirección de la serpiente que se desplaza durante el juego. Utilizando `or` con una sentencia `if` nos permite llevar a cabo más de una evaluación. En este caso, esto nos permite disponer de dos formas para controlar la serpiente: el jugador puede utilizar las teclas de cursor, o las teclas W, A, S y D para hacer que la serpiente se desplace hacia arriba, a la derecha, abajo o hacia la izquierda. Mientras no sea presionada una tecla, la serpiente continuará su desplazamiento hacia la derecha de acuerdo al valor establecido por `changeDirection` al comienzo del programa.

Si vuelve atrás en el código y mira las variables que inicializó al comienzo del programa, verá que hay otra variable llamada `direction`. Esta variable es utilizada junto con la variable `changeDirection` para averiguar si la instrucción que el usuario ha dado es válida. A la serpiente no debe permitírsele dar la vuelta inmediatamente hacia atrás de sí misma (pues si lo hace, la serpiente morirá y se termina el juego). Para prevenir que esto ocurra, la dirección solicitada por el jugador (almacenada en `changeDirection`) es comparada con la dirección actual en la cual la serpiente se desplaza (almacenada en `direction`). Si estas direcciones van en direcciones opuestas, la solicitud del usuario es ignorada y la serpiente continua en la misma dirección que antes. Escriba las siguientes líneas para establecer las comparaciones:

```

if changeDirection == 'right' and not direction == 'left':
    direction = changeDirection
if changeDirection == 'left' and not direction == 'right':
    direction = changeDirection
if changeDirection == 'up' and not direction == 'down':
    direction = changeDirection
if changeDirection == 'down' and not direction == 'up':
    direction = changeDirection

```

Con la entrada del usuario ya comprobada para garantizar que tienen sentido sus movimientos, la serpiente (que aparece en la pantalla como una serie de bloques) puede ser movida. En cada vuelta o revolución, la serpiente se mueve una distancia igual al tamaño de uno de estos segmentos en forma de bloque. Con cada segmento midiendo 20 píxeles, podrá decirle a `pygame` que desplace la serpiente un sólo segmento en cualquier dirección. Escriba el siguiente código:

```

if direction == 'right':
    snakePosition[0] += 20
if direction == 'left':
    snakePosition[0] -= 20
if direction == 'up':
    snakePosition[1] -= 20
if direction == 'down':
    snakePosition[1] += 20

```

Los operadores `+=` y `-=` son utilizados para modificar el valor de una variable en una cierta cantidad: `+=` establece la variable a su valor anterior más el nuevo valor, mientras que `-=` establece la variable a su valor anterior menos el nuevo valor. A manera de ejemplo, `snakePosition[0] += 20` es una forma abreviada de escribir

`snakePosition[0] = snakePosition[0] + 20`. El número entre los corchetes al lado del nombre de la variable `snakePosition` es la posición dentro de la lista que está siendo afectada: el primer valor en la lista `snakePosition` almacena la posición de la serpiente a lo largo del eje X, mientras que el segundo valor almacena la posición a lo largo del eje Y. Python comienza a contar desde cero, por lo que el eje X es controlado con `snakePosition[0]` y el eje Y con `snakePosition[1]`. Si la lista fuera más larga, entradas adicionales podrían ser afectadas incrementando el número: [2], [3] y así sucesivamente.

Aunque la lista `snakePosition` es siempre de una longitud de dos valores, hay otra lista creada al comienzo del programa que no lo es: `snakeSegments`. Esta lista almacena la ubicación del cuerpo de la serpiente (lo que está detrás de la cabeza). A medida que la serpiente come frambuesas y se hace más larga, la lista incrementa de tamaño y le proporciona la dificultad al juego: a medida que el juego progresa, se hace más difícil evitar golpear el cuerpo de la serpiente con la cabeza. Si la cabeza golpea el cuerpo, la serpiente muere y el juego termina. Escriba la línea siguiente para crear el crecimiento del cuerpo de la serpiente.

```
snakeSegments.insert(0,list(snakePosition))
```

La línea anterior utiliza la instrucción `insert` para insertar un nuevo valor dentro de la lista `snakeSegments`, ese nuevo valor a insertar será la posición actual de la serpiente. Cada vez que Python llega a esta línea, incrementará la longitud del cuerpo de la serpiente en un segmento y ubica ese segmento en la posición actual de la cabeza de la serpiente. Para el jugador, parecerá como si la serpiente creciera. No obstante, si sólo quiere que esto ocurra cuando la serpiente come una frambuesa (de lo contrario la serpiente sólo crecería y crecería) escriba las siguientes líneas:

```
if snakePosition[0] == raspberryPosition[0] and snakePosition[1] == raspberryPosition[1]:
    raspberrySpawned = 0
else:
    snakeSegments.pop()
```

La primera instrucción examina las coordenadas X y Y de la cabeza de la serpiente para ver si coinciden con las coordenadas X y Y de la frambuesa (el objetivo que el jugador está persiguiendo). Si los valores coinciden, se considera que la frambuesa fue comida por la serpiente y la variable `raspberrySpawned` es establecida a 0. La instrucción `else` le dice a Python que hacer si la frambuesa no ha sido comida: sacar (`pop`) al valor más primitivo de la lista `snakeSegments`.

La instrucción `pop` es simple pero inteligente, esta devuelve el valor más antiguo de la lista pero también lo elimina, haciendo que la lista sea un ítem más corto. En el caso de la lista `snakeSegment`, esta le dice a Python que elimine la porción del cuerpo de la serpiente más alejada de la cabeza. Para el jugador, parecerá como si la serpiente entera se hubiera movido sin crecer (en realidad, creció en el extremo final y se encogió en el otro). Es debido a la sentencia `else`, que la instrucción `pop` sólo se ejecuta cuando una frambuesa no ha sido comida. Si una frambuesa ha sido comida, la última entrada en la lista no es eliminada (por lo que la serpiente crece un segmento en tamaño).

Llegados a este punto del programa, es posible que el jugador haya comido una frambuesa. Un juego en donde sólo exista una frambuesa será aburrido, así que escriba las siguientes líneas para añadir una nueva frambuesa de vuelta en la superficie del juego en caso de que el jugador se haya comido la frambuesa que había:

```
if raspberrySpawned == 0:
    x = random.randrange(1,32)
    y = random.randrange(1,24)
    raspberryPosition = [int(x*20),int(y*20)]
    raspberrySpawned = 1
```

Esta sección de código ve si la frambuesa ha sido comida, se encarga de examinar si la variable `raspberrySpawned` está establecida a 0, y si es el caso, el código escoge una ubicación aleatoria en la superficie del juego utilizando el módulo `random` que se ha importado al inicio del programa. Esta ubicación es luego

multiplicada por el tamaño de un segmento de la serpiente (20 píxeles de ancho y 20 píxeles de alto) para proporcionar a Python un lugar en el campo de juego en donde colocar la nueva frambuesa. Es importante que la ubicación de la frambuesa sea establecida de manera aleatoria, esto evita que el jugador aprenda en donde aparecerá después la frambuesa. Por último, la variable `raspberryspawned` es establecida devuelta a 1, para asegurar que sólo habrá una sola frambuesa a la vez en la superficie del juego.

Ahora ya tiene el código necesario para hacer que la serpiente se mueva y crezca, así como también para hacer que las frambuesas sean comidas y creadas (un proceso conocido en los juegos como *respawning*). Sin embargo, nada de esto está siendo dibujado a la pantalla. Escriba las siguientes líneas:

```
playSurface.fill(blackColour)
for position in snakeSegments:
    pygame.draw.rect(playSurface,whiteColour,Rect(position[0], position[1], 20, 20))
    pygame.draw.rect(playSurface,redColour,Rect(raspberryPosition[0], raspberryPosition[1], 20, 20))
pygame.display.flip()
```

Estas líneas le dicen a Python que rellene el fondo de la superficie de juego en color negro, que dibuje la cabeza y los segmentos de la serpiente en color blanco, y que finalmente, dibuje una frambuesa en color rojo. La última línea, `pygame.display.flip()`, le dice a pygame que actualice la pantalla (sin esta instrucción, los elementos serían invisibles para el jugador). Cada vez que termine de dibujar los objetos sobre la pantalla, recuerde utilizar `pygame.display.flip()` para que el usuario pueda ver los cambios.

Actualmente, es imposible que la serpiente se muera. Un juego en donde el jugador nunca puede morir rápidamente se hace aburrido, así que introduzca las siguientes líneas para establecer algunos escenarios para la muerte de la serpiente:

```
if snakePosition[0] > 620 or snakePosition[0] < 0:
    gameOver()
if snakePosition[1] > 460 or snakePosition[1] < 0:
    gameOver()
```

La primera sentencia `if` comprueba para ver si la serpiente se ha ido fuera de la superficie de juego verticalmente, mientras que la segunda sentencia `if` comprueba si la serpiente ha ido fuera de la superficie de juego horizontalmente. En cualquier caso, son malas noticias para la serpiente porque la función `gameOver` (definida anteriormente en el programa) es llamada para imprimir un mensaje en la pantalla y salir del juego. La serpiente también debería de morir si su cabeza golpea cualquier parte de su cuerpo, así que agregue las siguientes líneas:

```
for snakeBody in snakeSegments[1:]:
    if snakePosition[0] == snakeBody[0] and snakePosition[1] == snakeBody[1]:
        gameOver()
```

La sentencia `for` hace un recorrido a través de cada una de las ubicaciones del segmento de la serpiente, desde la segunda entrada hasta última de la lista y las compara con la posición de la cabeza de la serpiente. Es importante comenzar la comparación en la segunda entrada utilizando `snakeSegments[1:]` y no en la primera. La primera entrada siempre contiene la posición de la cabeza y comenzar la comparación desde aquí resultaría en la muerte instantánea de la serpiente, tan pronto el juego comience.

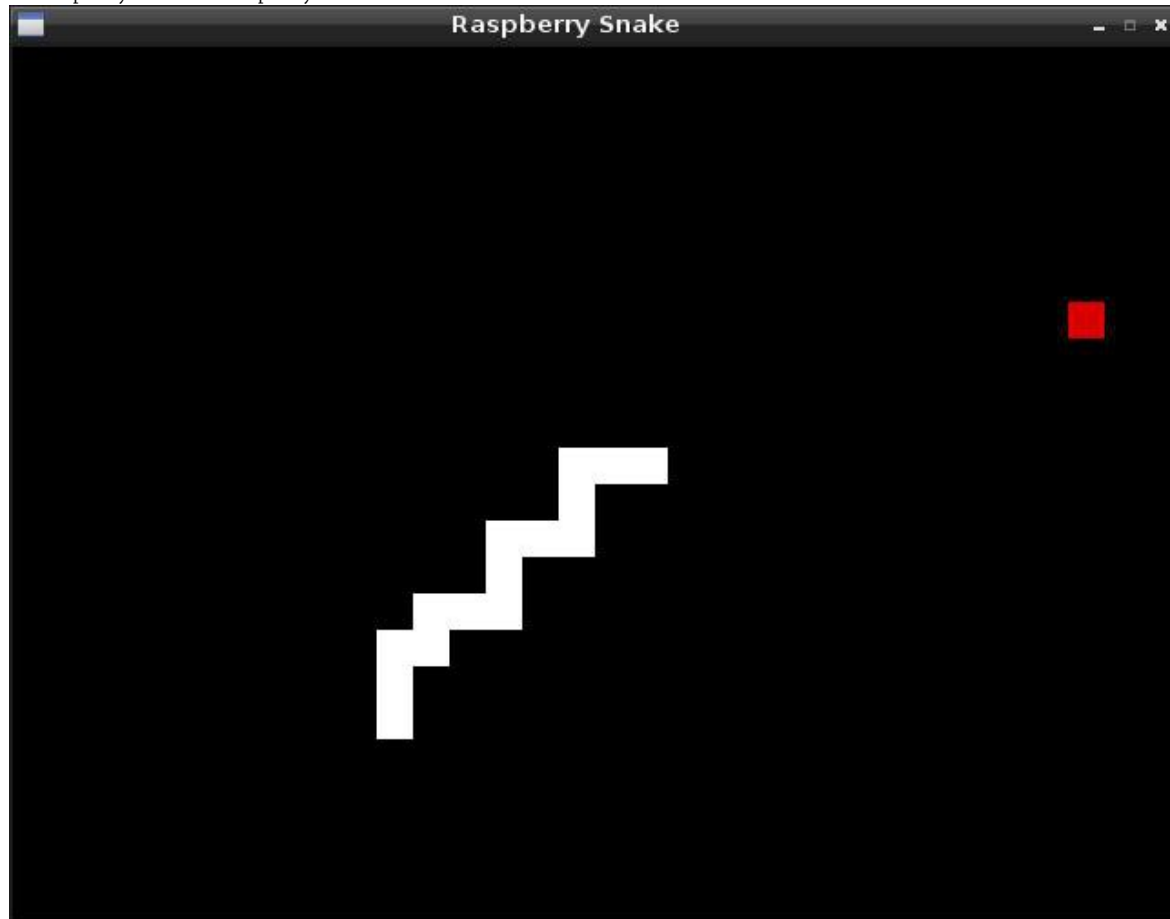
Por último, todo lo que se necesita para que nuestro juego este completo es controlar su velocidad utilizando la variable `fpsClock`. Sin esta variable (creada al inicio del programa), el juego correría demasiado rápido como para poder jugarse. Escriba la línea siguiente para terminar el programa:

```
fpsClock.tick(20)
```

Si piensa que el juego es demasiado fácil o demasiado lento, incremente este número; o si piensa que es demasiado difícil o demasiado rápido, disminuya el número. Guarde el programa como `raspberrysnake.py`, y ejecútelo utilizando la opción Run Module en el menú Run del IDLE o desde una terminal escribiendo `python raspberrysnake.py`. El juego comenzará tan pronto se cargue (ver Figura 12-6). ¡Así que asegúrese de estar listo!

Figura 12-6

Jugando Raspberry Snake en la Raspberry Pi



Una copia completa del listado del programa Raspberry Snake se incluye en el Apéndice A, “Python Recetas” y también en el sitio web de Raspberry Pi Guía del Usuario en <http://www.wiley.com/go/raspberrypiuserguide2e>. Descargar el código fuente desde el sitio web le ahorra algo de mecanografía, pero introducir el código a mano es una buena manera de asegurarse que está comprendiendo lo que cada sección hace. Además de las funciones utilizadas en Raspberry Snake, pygame ofrece un montón de características no utilizadas en este programa, como por ejemplo, la reproducción de audio, el manejo de sprites para mejorar las gráficas y el control del ratón. El mejor lugar para aprender acerca de las funciones más avanzadas es en el sitio web oficial, <http://www.pygame.org/wiki/tutorials>, en donde podrá descargar tutoriales y programas de ejemplo para hacerse una idea de cómo las cosas funcionan.

## Ejemplo 4: Python y la Gestión de Redes

Hasta ahora, ha aprendido cómo Python puede ser utilizado para crear programas independientes, pero el lenguaje también puede ser utilizado para crear programas que se comuniquen con el mundo exterior a través de una conexión de red de una computadora. El siguiente ejemplo, escrito por Tom Hudson, es una herramienta para supervisar a los usuarios conectados a un canal de IRC (Internet Relay Chat) y ofrece un breve vislumbre de las

posibilidades de la gestión de redes.

Como de costumbre, cree un nuevo proyecto en IDLE o en el editor de textos e introduzca la línea shebang junto con un comentario que describa el propósito del programa:

```
#!/usr/bin/env python
# IRC Channel Checker, escrito por Tom Hudson para la Raspberry Pi Guía del Usuario
```

A continuación, importe los módulos requeridos para el programa (`sys`, `socket` y `time`) con la siguiente línea:

```
import sys, socket, time
```

Usted ha utilizado anteriormente los módulos `sys` y `time` en el programa Raspberry Snake, pero aún no ha utilizado `socket`. El módulo `socket` otorga a Python la habilidad de abrir, cerrar, leer (desde) y escribir (hacia) a los sockets de la red (proporcionando capacidades rudimentarias de red a los programas Python). Este módulo `socket` es el que proporciona en este ejemplo la capacidad de conectarse a un servidor IRC remoto.

Hay algunas *constantes* que son necesarias para que este programa funcione. Las constantes se parecen a las variables en que pueden tener valores asignados en ellos, pero a diferencia de las variables, el valor en una constante no debe de cambiar. Para ayudar a diferenciar visualmente una constante de una variable, es una buena práctica escribir en mayúsculas cada una de las letras que componen su nombre (así es más fácil reconocer de un vistazo si en una determinada sección del código se está utilizando una constante o una variable). Escriba las siguientes dos líneas en el programa:

```
RPL_NAMREPLY = '353'
RPL_ENDOFNAMES = '366'
```

Estos números son códigos de estado IRC, proporcionados por el servidor para indicar cuándo determinadas operaciones se han completado. Tales códigos son utilizadas por el programa para saber cuándo ha recibido desde el servidor IRC la lista de nombres requerida. A continuación, configure las variables para la conexión del servidor introduciendo las siguientes líneas:

```
irc = {
    'host' : 'chat.freenode.net',
    'port' : 6667,
    'channel' : '#raspiuserguide',
    'namesinterval' : 5
}
```

La primera línea le dice a Python que cree un tipo de datos *dict* (abreviación para dictionary o diccionario), este tipo permite que múltiples variables sean almacenadas en una sola variable maestra (en este caso, `irc`). Estas variables individuales entonces pueden ser recuperadas más adelante en el programa. Aunque se podría escribir este programa sin utilizar diccionarios para almacenar las variables, el programa sería volvería más difícil de leer. El diccionario comienza con una llave de apertura y termina con la llave de cierre al final de la línea.

En la variable `host` se debe establecer el nombre de dominio totalmente calificado (FQDN, fully-qualified domain name) del servidor IRC al cual el programa se conectará. En este ejemplo, `chat.freenode.net` será utilizado, pero si desea personalizar el programa para que utilice un servidor distinto, cambie el nombre de dominio aquí. La variable `port` le dice al programa en qué puerto de red IRC se está ejecutando (normalmente este puerto será 6667). La variable `channel` le dice a Python a cuál canal unirse a fin de supervisar a los usuarios, mientras que `namesinterval` controla el tiempo que el programa espera para actualizar la lista de los usuarios (medido en segundos).

Se configura un segundo diccionario para almacenar las variables específicas del usuario escribiendo las siguientes

líneas:

```
user = {
    'nick' : 'botnick',
    'username' : 'botuser',
    'hostname' : 'localhost',
    'servername' : 'localhost',
    'realname' : 'Raspberry Pi Names Bot'
}
```

Al igual que con el diccionario `irc`, estas variables son almacenadas dentro de un diccionario llamado `user` para dejar en claro qué variables pertenecen a qué sección. La variable `nick` debe ser establecida con el sobrenombre de IRC (IRC nickname) que el programa utilizará. No utilice su sobrenombre habitual si está planeando conectarse al servidor IRC al mismo tiempo; en lugar de eso, intente añadir `-bot` al final de su nombre para dejar claro que el usuario es un programa en lugar de una persona real. Haga lo mismo con el nombre de usuario (`username`) y escriba en la variable `realname` un mensaje descriptivo que indique a quién pertenece este bot. Las variables `hostname` y `servername` pueden dejarse establecidas a `localhost`, o modificarse para que correspondan con su dirección de internet.

El módulo `socket` requiere que el usuario cree un objeto `socket`. Este objeto proporciona la conectividad de red al resto del programa. Cree el objeto `socket` escribiendo la siguiente línea:

```
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

En seguida, necesita decirle al programa que trate de conectarse al servidor IRC especificado en las variables al comienzo del programa. Escriba las siguientes líneas:

```
print 'Connecting to %(host)s:%(port)s...' % irc
try:
    s.connect((irc['host'], irc['port']))
except socket.error:
    print 'Error connecting to IRC server %(host)s:%(port)s' % irc
    sys.exit(1)
```

Los comandos `try` y `except` están incluidos en este código para el *manejo de errores*. Si el sistema falla en conectarse al servidor (por ejemplo, porque la RasPi no está conectada a Internet o porque el servidor se encuentra fuera de servicio por mantenimiento) el programa imprimirá un mensaje de error y elegantemente finalizará. La línea `s.connect` le dice al módulo `socket` que trate de conectarse al servidor IRC utilizando las variables `host` y `port` mantenidas en el diccionario `irc`.

Si el programa no se sale a causa de la excepción, se habrá conectado al servidor IRC con éxito. Sin embargo, antes de que pueda obtener la lista de nombres en un canal, necesita identificarse con el servidor y emitir algunos comandos utilizando la función `send` del módulo `socket`. Escriba las siguientes líneas en el programa:

```
s.send('NICK %(nick)s\r\n' % user)
s.send('USER %(username)s %(hostname)s %(servername)s :%(realname)s\r\n' % user)
s.send('JOIN %(channel)s\r\n' % irc)
s.send('NAMES %(channel)s\r\n' % irc)
```

La función `send` trabaja casi exactamente de la misma forma que la función `print`, excepto que en lugar de imprimir a la salida estándar (comúnmente la ventana de terminal o consola) la salida se envía a través de la conexión de red. En este caso el programa está enviando cadenas de texto al servidor IRC y le está diciendo que registre el programa utilizando el sobrenombre almacenado en la variable `nick` y los detalles del usuario almacenados en las variables `username`, `hostname`, `servername` y `realname`. A continuación, el programa envía

el comando para unirse al canal especificado en la variable `channel`, y finalmente, se envía el comando para recibir la lista de usuarios de ese canal. Aunque este ejemplo está adaptado a IRC, el mismo principio básico puede ser utilizado para emitir comandos a cualquier servicio de red (con algunas modificaciones), este programa puede utilizarse para listar los archivos sobre un servidor FTP o incluso los correos electrónicos no leídos en un servidor POP3.

La recepción de datos desde el socket es un poco más complicada. En primer lugar, necesita crear una variable de cadena vacía que actúe como un *búfer de recepción* para mantener los datos del servidor tal cual son recibidos hasta que puedan ser procesados. Inicialice el búfer escribiendo la siguiente línea:

```
read_buffer = ''
```

Note que hay dos comillas simples después del signo de igual, no se trata de una doble comilla.

Luego, cree una lista vacía, que será usada para almacenar los nombres de los usuarios escribiendo la siguiente línea:

```
names = []
```

El tipo de datos *lista* es el mismo que utilizó para almacenar las posiciones en el juego Raspberry Snake. A diferencia de una variable normal, la lista puede almacenar múltiples valores (en este caso, los nombres de los usuarios presentes en el canal IRC).

El siguiente paso es crear un bucle infinito, en donde el programa continuamente consultará al servidor por los nombres de usuario y los imprimirá en la pantalla. Comience el bucle infinito escribiendo:

```
while True:
    read_buffer += s.recv(1024)
```

La primera línea del bucle, después de `while True:`, le dice al módulo `socket` que reciba 1,024 bytes (1KB) de datos desde el servidor IRC y los coloque dentro de la variable `read_buffer`. Debido a que el operador `+=` es utilizado (en vez de sólo `=`) los datos recibidos serán agregados a los que ya se encuentren en el búfer. El valor de 1024 bytes es más o menos arbitrario.

El siguiente paso es dividir el búfer en líneas individuales de texto, utilizando lo siguiente:

```
lines = read_buffer.split('\r\n')
read_buffer = lines.pop();
```

La primera línea inicializa la variable `lines` con una línea completa de texto desde el búfer recibido utilizando la función `split` para encontrar los caracteres de *fin de línea* (representados por `\r\n`). Estos caracteres sólo aparecen al final de la línea de texto, de manera que cuando el búfer ha sido dividido de esta forma, usted conoce que `lines` contendrá sólo respuestas de línea completa desde el servidor. La instrucción `pop` en la segunda línea garantiza que sólo las líneas completas sean eliminadas desde el `read_buffer`: esto es debido a que las respuestas obtenidas desde el servidor son leídas en trozos de 1 KB, lo que es probable que en algún momento dado el búfer contendrá sólo fracciones de una línea. Cuando ese sea el caso, la fracción es dejada en el búfer lista para recibir el resto de la línea la próxima vez que el bucle se ejecuta y el siguiente trozo de 1 KB es recibido desde el servidor.

En este punto, la variable `lines` contiene una lista de respuestas completas (líneas completas) recibidas desde el servidor. Escriba lo siguiente para procesar estas líneas y encontrar los nombres de los participantes del canal:

```

for line in lines:
    response = line.rstrip().split(' ', 3)
    response_code = response[1]
    if response_code == RPL_NAMREPLY:
        names_list = response[3].split(':')[1]
        names += names_list.split(' ')

```

Este código es ejecutado para cada línea que se encuentra dentro de la variable `lines` y localiza el código de respuesta numérico del IRC proporcionado por el servidor. Aunque existen muchísimos códigos de respuesta diferentes, nuestro programa sólo está interesado en los dos códigos definidos como constantes al comienzo del programa: 353, el cual indica que una lista de nombre le sigue, y el 366, el cual indica que la lista ha terminado. La sentencia `if` busca la primera de estas respuestas, y luego, utiliza la función `split` para recibir estos nombres y agregarlos a la lista `names`.

Ahora, la lista `names` contiene todos los nombres recibidos desde el servidor en respuesta a la solicitud del programa. Sin embargo, pueden no ser todos los nombres; hasta que la respuesta 366 (que señala el final de los nombres de los miembros), no sea recibida, la lista está incompleta. Es por esto que la última `names += names_list.split(' ')` está añadiendo los nombres recién recibidos a la lista existente, en vez de borrarla completamente; cada vez que la sección del código se ejecuta, es probable que el programa sólo haya recibido una sub-sección de toda la lista de miembros. Para decirle a Python qué hacer cuando la lista completa sea recibida, introduzca las siguientes líneas:

```

if response_code == RPL_ENDOFNAMES:
    # Display the names
    print '\r\nUsers in %(channel)s:' % irc
    for name in names:
        print name
    names = []

```

Esto le dice a Python que cuando la respuesta 366 sea recibida, se imprima la lista de nombres (ahora ya completa) a la salida estándar antes de borrar la lista `names` nuevamente. Esta última línea (`name = []`) es importante, pues sin ella, cada vez que el bucle se ejecute, se añadirán los nombres de los usuarios a la lista a pesar de que éstos ya existan por una ejecución anterior.

Por último, finalice el programa introduciendo las siguientes líneas:

```

time.sleep(irc['namesinterval'])
s.send('NAMES %(channel)s\r\n' % irc)

```

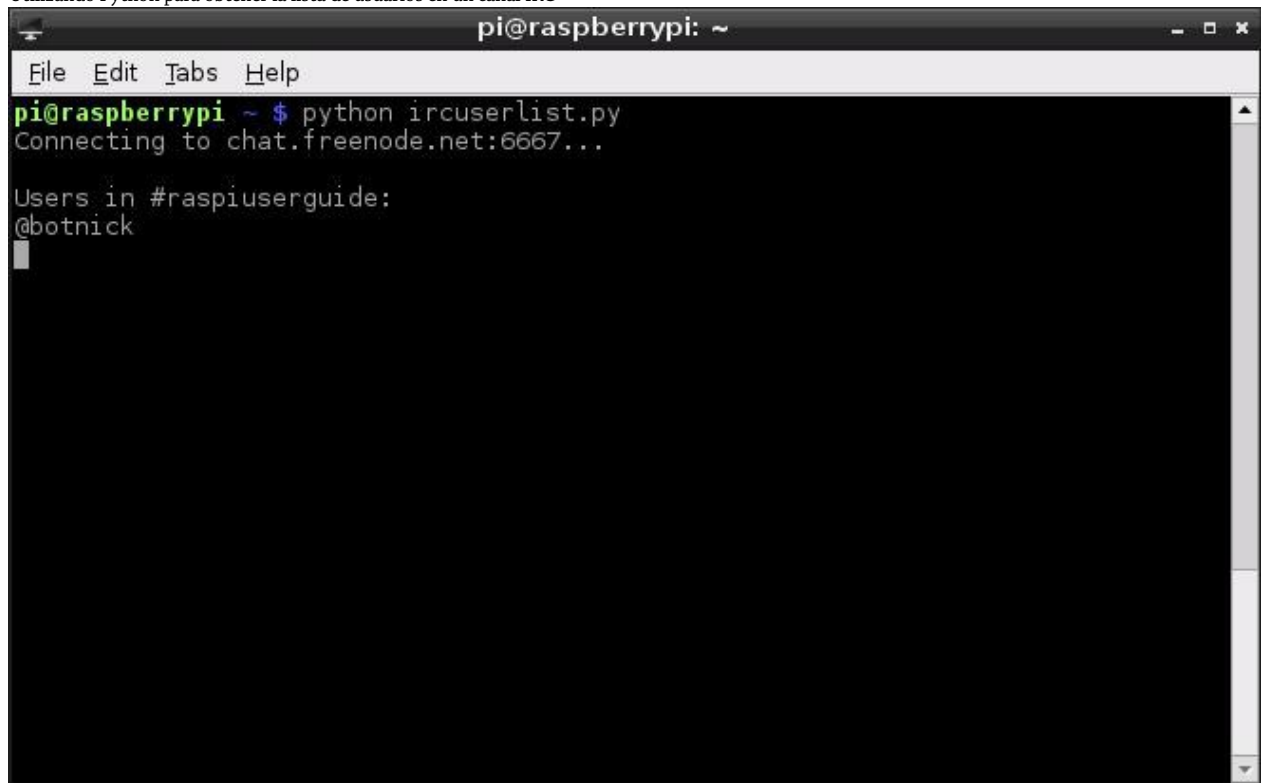
Esto le dice a Python que espere `namesinterval` número de segundos antes de enviar la siguiente solicitud para obtener los otros nombres de usuario y comienza el bucle de nuevo. Procure colocar `namesinterval` con un valor razonable, si el servidor IRC recibe demasiadas solicitudes en un lapso de tiempo muy corto, puede desconectarlo a la fuerza por *inundar* de peticiones al servidor.

Guarde el programa como `ircuserlist.py` y ejecútelo utilizando la opción Run Module en el menú Run del IDLE o desde la terminal escribiendo `python ircuserlist.py`. Cuando el programa se ejecuta por primera vez, le tomará un tiempo conectarse al servidor; sin embargo, una vez conectado, la lista de nombres (ver Figura 12-7) debería refrescar rápidamente. Para salir del programa, presione CTRL + C.



Figura 12-7

Utilizando Python para obtener la lista de usuarios en un canal IRC

A screenshot of a terminal window titled "pi@raspberrypi: ~". The window has a menu bar with "File", "Edit", "Tabs", and "Help". The terminal shows the command "python ircuserlist.py" being executed. The output indicates a connection to "chat.freenode.net:6667..." and lists the users in the "#raspiuserguide" channel as "@botnick".

```
pi@raspberrypi ~ $ python ircuserlist.py
Connecting to chat.freenode.net:6667...

Users in #raspiuserguide:
@botnick
```

Una copia completa de este programa se incluye en el Apéndice A, “Python Recetas” y también en el sitio web de la Raspberry Pi Guía del Usuario en <http://www.wiley.com/go/raspberrypiuserguide2e>. Descargar el código fuente desde el sitio web le ahorrará mecanografía, pero introducirlo a mano es una muy buena manera de comprender qué es lo que hace cada sección.

## Lectura Sugerida

Esperamos que esta lectura le haya dado una idea de lo que puede hacer con Python. Aunque este capítulo está lejos de ser una revisión exhaustiva (para hacerle plena justicia al lenguaje nos tomaría un libro considerablemente más grande). Sin embargo, existen muchísimos recursos allá afuera para poder aprender más sobre el lenguaje Python, algunas lecturas se recomienda a continuación:

- La *Guía de Python Oficial para Principiantes*, se puede acceder a ella en <http://wiki.python.org/moin/BeginnersGuide>.
- Una guía interactiva y gratuita que se ejecuta completamente desde su navegador se encuentra disponible para su descarga en <http://www.learnpython.org/>.
- Zed A. Shaw's *Learn Python The Hard Way* (Shavian Publicaciones, 2012) ofrece una gran comprensión sobre las mejores prácticas de codificación para Python, y a pesar de su nombre, es apto para los principiantes. Este libro se puede encontrar disponible de manera comercial o también puede leerse de forma gratuita en <http://learnpythonthehardway.org/>.

- Aunque un tanto anticuado y al ser sustituido por *Dive Into Python 3* (APRESS, 2009), *Dive Into Python* de Mark Pilgrim (APRESS, 2004) hace un buen trabajo cubriendo los fundamentos de Python a la hora de escribir programas. Está disponible gratuitamente en <http://www.diveintopython.net/> o para su compra en formato impreso en toda buena librería.
- Si prefiere del aprendizaje práctico con otras personas con sus mismos intereses, en la lista de Grupos de Usuarios Python locales (algunas veces llamados PIGgies) puede encontrar un gran recurso: <http://wiki.python.org/moin/LocalUserGroups>.
- Para aprender pygame, el libro de *Al Sweigart Making Games with Python & Pygame* (CreateSpace, 2012) resulta una gran introducción con ejemplos prácticos. Puede adquirirlo de forma impresa o descargarlo gratuitamente en <http://inventwithpython.com/>.



Parte IV

# Hardware Hacking

- Capítulo 13**    Aprendiendo a Hackear Hardware
- Capítulo 14**    El Puerto GPIO
- Capítulo 15**    El Módulo de Cámara de la Raspberry Pi
- Capítulo 16**    Placas de Expansión



# Capítulo 13

Aprendiendo a Hackear Hardware

**En los capítulos anteriores**, aprendió cómo la Raspberry Pi puede convertirse en una plataforma flexible para ejecutar una amplia variedad de software. La RasPi no es la única que hace eso, cualquier PC de escritorio o laptop pueden ejecutar el mismo software y en muchos de los casos ejecutarlos más rápido que el procesador de bajo coste energético de la Raspberry Pi.

Sin embargo, la RasPi tiene un haz bajo la manga, una particularidad que lo sitúa por encima y más allá de las capacidades de una PC promedio: su puerto con *26 pines de entradas-salidas de propósito general (GPIO)*, ubicado en la parte superior izquierda de la placa de circuito impreso de la RasPi.

La puerto GPIO permite a la RasPi comunicarse con otros componentes y circuitos, y le permite actuar como un controlador en un circuito electrónico más grande. A través del puerto GPIO, es posible hacer que la RasPi perciba temperaturas, mueva servomecanismos y se comuniquen con otros dispositivos informáticos utilizando una variedad de diferentes protocolos incluyendo el *Interfaz Periférico Serie (SPI)* y el *Inter-Circuitos Integrados(I<sup>2</sup>C)*. El Capítulo 14, “El Puerto GPIO”, proporciona detalles sobre cómo trabajar con los pines GPIO.

Sin embargo, antes de que puede construir sus propios circuitos que utilicen el puerto GPIO de la RasPi, necesitará disponer de algunas herramientas adicionales y entender un poco del lenguaje que rodea el mundo de la electrónica.

## Equipo Electrónico

Para comenzar a construir circuitos que puedan ser controlados por el puerto GPIO de la RasPi, necesitará de varios componentes y herramientas. El siguiente listado ofrece una muestra de algunos artículos que comprar para iniciarse en el mundo de la electrónica:

- **Protoboard** – o *Placa de Pruebas*, es tablero electrónico que cuenta con una rejilla de orificios espaciados a intervalos de 2.54mm en los que se pueden insertar y retirar componentes. Debajo de cada rejilla existen una serie de contactos eléctricos, que permiten a los componentes en la misma fila ser conectados entre sí sin la necesidad de cables. Una protoboard o breadboard es una valiosa herramienta para el trabajo electrónico, porque le permite rápidamente crear montajes de circuitos que pueden ser modificados sin la necesidad de ser soldados o desoldados.
- **Cables** – Mientras que la protoboard le permite a algunos componentes ensamblarse sin la necesidad de cables, aun así necesitará de cables para conectar una fila con otra. Estos cables son conocidos como *cables jumper* y si trabaja en una protoboard, es una buena idea conseguir cables de núcleo macizo en vez de cables con núcleo trenzado de hilos metálicos. El cable de núcleo macizo es más fácil de insertar dentro de los orificios de la protoboard que un cable trenzado. Además, es de mucha ayuda conseguirlos en varios colores, para que pueda emplear el código de colores para cada conexión según sea su propósito.
- **Resistencias** – La gran mayoría de los circuitos electrónicos emplean los componentes conocidos como *resistencias* y los proyectos de ejemplo en este capítulo no serán la excepción. Las resistencias son medidas en *ohmios*, representadas con el símbolo  $\Omega$ . Siempre procure tener un montón de cada variedad de valores comunes: 2.2 K $\Omega$ , 10 K $\Omega$ , y 68 $\Omega$  son buenos valores para empezar. Algunas tiendas minoristas cuentan con kits de resistencias, que incluyen una amplia gama de valores útiles.
- **Botones** – Un componente de entrada muy común, un *botón pulsador* completa un circuito eléctrico cuando se le presiona. En el nivel más básico, un teclado es poco más que una colección de pulsadores. Si está diseñando un circuito que proporciona una simple entrada a la RasPi, elija el botón denominado *interruptor momentáneo*.
- **LEDs** – El *Diodo Emisor de Luz* es el dispositivo de salida más común que existe. Un LED se ilumina cuando se le aplica voltaje, proporcionándole una retroalimentación visual cuando algún pin del GPIO de la RasPi

se encuentra en estado alto o bajo. Cuando compre sus LEDs para usar con la RasPi, opte por los de bajo consumo. El puerto GPIO no es tan poderoso y los LEDs de alto consumo (como el modelo blanco o azul luminoso) requerirán una fuente de alimentación externa y un componente extra conocido como *transistor*.

Adicionalmente, si está planeando hacer algo más permanente, una vez que haya terminado su prototipo en la protoboard (la cual conocerá más adelante en este capítulo), necesitará también lo siguiente:

- **Stripboard** – o *Placa perforada*. Puede considerarse como una protoboard de un sólo uso. A semejanza de las protoboard, los orificios están dispuestos en una rejilla de 2.54 mm. A diferencia de las protoboard, los componentes necesitan ser soldados en su lugar (obteniéndose un circuito electrónico permanente).
- **Soldador** – Cuando necesita conectar un componente permanentemente sobre un circuito, es necesario soldarlo. No tiene que gastar una fortuna en un soldador, pero si está al alcance de su presupuesto, un modelo con control de temperatura, es una sabia inversión. Asegúrese de que cualquier soldador que compre cuente con una pequeña punta puntiaguda (los de punta tipo cincel no son adecuados para el trabajo electrónico delicado).
- **Soldadura** – Su soldador necesita de soldadura. La soldadura es una mezcla de metales conductores con una sustancia limpiadora conocida como flux. Asegúrese de que la soldadura que adquiera sea la adecuada para trabajos de electrónica (la soldadura gruesa para tuberías es barata, pero puede dañar los delicados circuitos porque requiere demasiado calor para derretirse).
- **Soporte y Esponja** - Es importante disponer de algún lugar para colocar el soldador caliente cuando no está en uso, así como de una manera de limpiar la punta del cautín mientras lo utiliza. Algunos soldadores o cautines vienen con un soporte que incorpora una esponja de limpieza (si el suyo no lo tiene, adquiera un juego de soporte y esponja por separado).
- **Pinzas de Corte Lateral** – Los componentes “through-hole” con largas patas, una vez que son soldados en su lugar a través de los orificios de la placa, dejan partes excedentes. Las *pinzas de corte lateral* sirven para recortar los excedentes de las patas de estos componentes en una forma fina y rápida, sin dañar la unión de la soldadura.
- **Pinzas Tweezer** – Los componentes electrónicos pueden ser muy pequeños y difíciles de sujetar, un buen par de pinzas son muy importantes. Si está pensando utilizar componentes de montaje por superficie, en lugar de los componentes “through-hole” fáciles de soldar, las pinzas son absolutamente necesarias (¡sin las pinzas, se quemaría los dedos mientras trata de sujetar el componente y soldarlo al mismo tiempo!).
- **Soporte de trabajo** – Comúnmente conocido como *tercera mano*, estos son soportes pesados con pinzas o clips que sujetan en su lugar el elemento a soldar. Algunos soportes de trabajo incluyen una lupa integrada para el trabajo delicado, mientras que los soportes de trabajo más caros cuentan con una lámpara para ayudar a iluminar el área de trabajo.
- **Multímetro** - Los multímetros son medidores de prueba que tienen varias funciones, entre ellas, la de medir el voltaje, la resistencia y la capacitancia, así como también tienen la funcionalidad de hacer pruebas de continuidad para encontrar rupturas en los circuitos. Aunque el multímetro no es una necesidad absoluta, puede ser de gran utilidad para el diagnóstico de problemas con los circuitos. Los multímetros profesionales pueden ser costosos, pero un modelo sencillo es bastante económico y es una buena inversión para cualquier persona que se inicia en la electrónica.
- **Trenza para desoldar** - Los errores ocurren, pero no tienen porque ser permanentes. Una trenza para desoldar es una cinta metálica trenzada que puede colocarse sobre una junta y aplicarle calor, absorbiendo



la soldadura del componente en la trenza. Con la práctica, es posible utilizar la trenza de desoldar para rescatar componentes de equipos electrónicos de desecho (una manera práctica de obtener componentes comunes a un bajo costo).

## Interpretación de los Códigos de Colores de las Resistencias

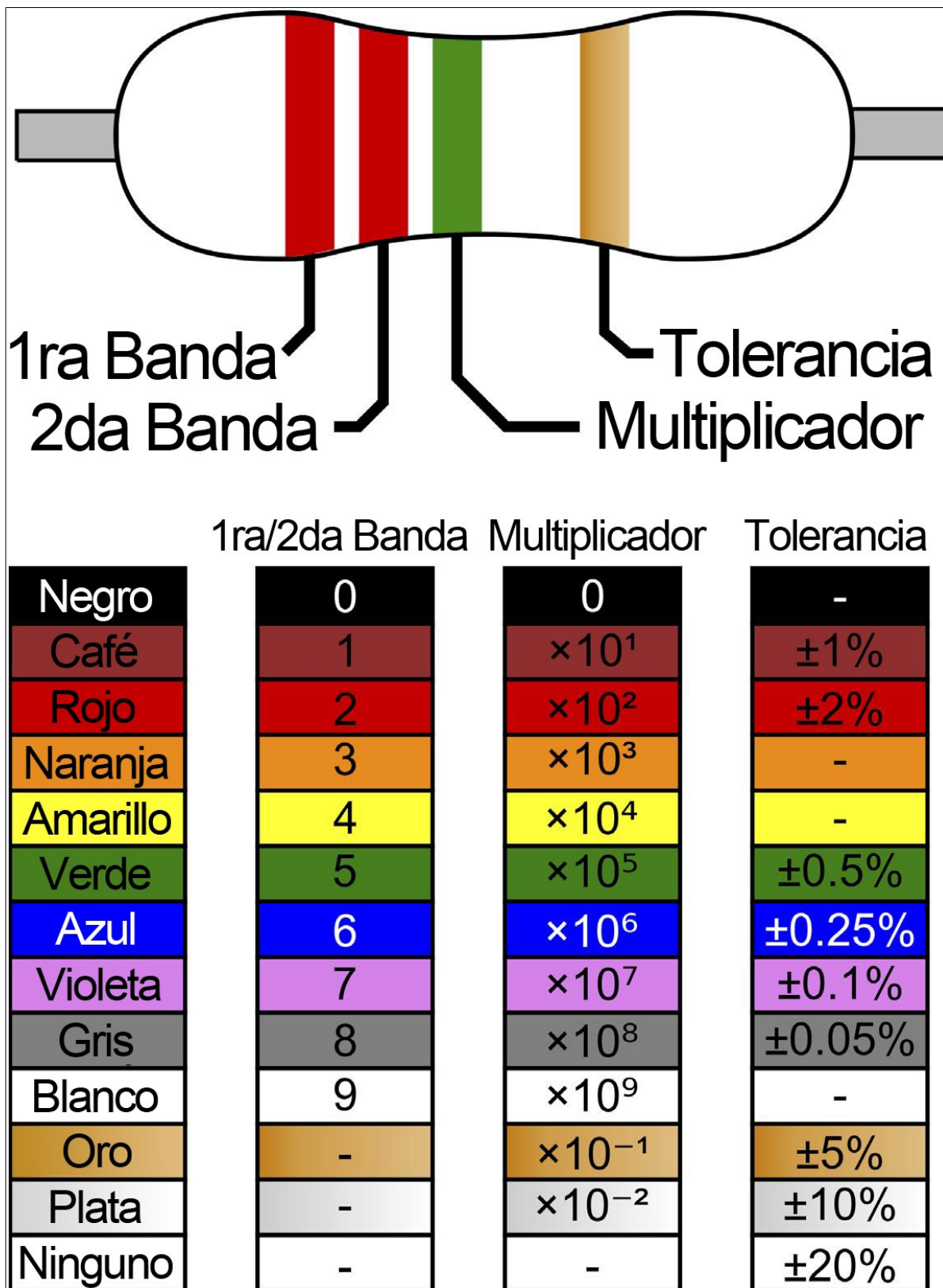
La mayoría de los componentes electrónicos son claramente identificados. Por ejemplo, los *capacitores* tienen sus *capacitancias* (medidas en *faradios*), impresas directamente en el componente, mientras que los *osciladores de cristal* tienen sus *frecuencias* impresas de la misma forma.

La excepción principal son las resistencias, que comúnmente no tienen ninguna escritura sobre sus superficies. En cambio, el valor de la resistencia en ohmios se calcula a partir de las bandas de color que adornan la superficie de la resistencia. Aprender a descifrar estas bandas es una habilidad importante para un hacker de hardware, porque una vez que una resistencia es retirada de su empaque, la única forma de averiguar su valor es utilizando un multímetro, el cual es una herramienta torpe y lenta para realizar este trabajo en particular.

Afortunadamente, los códigos de color de resistencias siguen un patrón lógico. La Figura 13-1 muestra una resistencia típica de cuatro bandas. Una versión en alta resolución de este diagrama está disponible en el sitio web de la Raspberry Pi Guía del Usuario en [www.wiley.com/go/raspberrypiuserguide2e](http://www.wiley.com/go/raspberrypiuserguide2e). Las dos primeras bandas están asignadas a un color que equivale a un valor de resistencia en ohmios. La tercera banda es el *multiplicador*, por el cual los dos primeros números son multiplicados para llegar al valor de resistencia real. La banda final indica la *tolerancia* de la resistencia, o cuán exacta es. Una resistencia con una baja tolerancia estará más cerca a su valor etiquetado que una resistencia con una alta tolerancia, pero pagará más por el componente.

Figura 13-1

Una resistencia de cuatro bandas y la tabla de decodificación para su código de color



Para leer la resistencia de ejemplo, primero se toman las dos bandas de la resistencia comenzando por la izquierda: esas bandas están coloreadas en rojo y rojo. El Rojo, que se observa en la tabla de la Figura 13-1, equivale al valor de 2, así que la lectura inicial es 22. La siguiente banda es verde, la cual es el multiplicador y equivale a  $10^5$  o 100,000 (10 seguido de cinco ceros). Multiplicando 22 por 100,000 da como resultado 2,200,000, que es el valor de la resistencia en ohmios: 2,200,000  $\Omega$ .

Hay 1,000 ohmios en un *kiloohmio* y 1,000 kiloohmios en un *megaohmio*. Por lo tanto, 2,200,000  $\Omega$  comúnmente se escribirá como 2,2 M $\Omega$ . La banda final dorada, situada al lado derecho de la resistencia, indica la tolerancia, o la precisión de la resistencia que es más o menos del 5 por ciento en el ejemplo que se muestra en la Figura 13-1.

Existen también resistencias de cinco bandas. Puede leerlas de la misma forma que una resistencia de cuatro bandas, excepto que las primeras tres bandas ofrecen cifras de resistencia, la cuarta el multiplicador y la quinta la tolerancia.

## Fuente de Componentes

Si no ha incursionado en la electrónica antes, puede ser difícil encontrar componentes y herramientas. Afortunadamente, existen un montón de fuentes en línea y fuera de línea que se especializan en proveer aquellos componentes difíciles de encontrar que pueda necesitar para completar su proyecto.

### Fuentes en línea

Dos de los mayores comerciantes al por menor de componentes electrónicos y herramientas en el mundo son RS Components y Farnell. Ambos operan oficinas y almacenes en todo el mundo y cuentan una selección importante de hardware entre los cuales elegir. Es probable que esté familiarizado con al menos uno de los dos comerciantes. Al momento de escribir estas líneas, RS Components y Farnell fueron las dos únicas empresas autorizadas por la Fundación Raspberry Pi para producir el hardware de la Raspberry Pi, así que a menos que esté leyendo este libro mientras considera la compra de un Raspberry Pi, tendrá que ordenar el dispositivo desde alguno de estos dos proveedores.

Tanto RS Components como Farnell se enfocan en las transacciones *negocio a negocio*, lo que significa que su principal fuente de ingresos es de compañías electrónicas que compran grandes cantidades de componentes. Sin embargo, ambos permitirán felizmente que los consumidores se registren en sus respectivas tiendas web y realicen pedidos tan pequeños como componentes individuales.

Cuando realice pequeños pedidos, tenga en cuenta que puede contraer algunos cargos adicionales. RS Components cobra por su servicio de entrega de día siguiente, y aunque el cargo es bastante razonable para los pedidos más grandes, estos cargos pueden costar mucho más que los componentes que están siendo comprados cuando se tratan de pequeños pedidos. Por otro lado, Farnell no cobra ningún cargo por la entrega de día siguiente pero requiere de una valor total mínimo para los pedidos realizados a través de su sitio web.

Dependiendo de donde viva usted, podrá ser capaz de reservar artículos para su recolección en los locales de RS Components o Farnell. Esto le ahorrará el costo del envío y le permitirá conseguir las mercancías más rápidamente, pero puede que esta opción no esté disponible en su ciudad. Los sitios web de ambos minoristas se pueden encontrar aquí:

RS Components: <http://www.rs-online.com>

Farnell: <http://www.farnell.com>

## Fuentes Fuera de Línea

Puede que alguna vez necesite un componente de manera inmediata, y que el envío de día siguiente no será lo suficientemente pronto como lo necesita. O quizás sólo necesita una sola resistencia o un pequeño trozo de cable y no se justifica pagar un alto costo de envío o un valor mínimo de pedido a alguno de los minoristas. Afortunadamente, existen tiendas físicas que se especializan en componentes electrónicos. Aunque ya no hay muchas tiendas de este tipo como en décadas pasadas, la mayoría de las ciudades cuenta con al menos una tienda física que tiene entre sus existencias las herramientas y componentes más comunes.

En el Reino Unido, la cadena de tiendas más popular de componentes electrónicos es Maplin Electronics. Establecida en Essex en 1972, la compañía ha crecido de ser una pequeña tienda que enviaba pedidos por correos a contar con 187 tiendas esparcidas por todo Reino Unido. Los habitantes de la mayoría de las ciudades en Reino Unido encontrarán a su alcance una tienda Maplin Electronics, que podrá ser una fuente útil para obtener hardware si se le están agotando los componentes electrónicos más comunes.

Maplin Electronics también ofrece pedidos por correo y reservas en línea a través de su sitio web, pero tenga cuidado con su enfoque de *negocio a consumidor*. Muchos componentes son considerablemente más costosos cuando se los ordena desde Maplin que desde RS Components o Premier Farnell, esto es porque la compañía confía más en el margen de ganancia por artículo que en obtener beneficios por las ventas en gran volumen de los artículos.

En los Estados Unidos y en otros países, Radio Shack sigue siendo la cadena más popular de tiendas de electrónica. Fundada en 1921, se jactan de tener 7,150 tiendas en todo el mundo, la compañía tiene en existencia una amplia selección de componentes y herramientas electrónicas más comunes que pueden comprarse en persona u ordenarlas a través de su sitio web.

Al igual que Maplin Electronics en el Reino Unido, Radio Shack opera con un enfoque de *negocio a consumidor*. Por consiguiente los compradores pueden llegar a pagar más por los componentes comprados en Radio Shack que si hubieran comprado las mismas piezas desde RS Components o Farnell. No obstante, el gran número de tiendas Radio Shack los hace una opción conveniente para aquellos compradores que tienen prisa o que están buscando artículos por única ocasión.

Tanto Maplin como Radio Shack tienen la ventaja de ser atendidas por personas con las que se puedes hablar. Algunos de los empleados en ambas compañías cuentan con conocimientos importantes en electrónica y están dispuestos a ofrecer su asesoramiento y asistencia si usted no está seguro de qué piezas necesita para su proyecto. Los sitios web de ambos minoristas se pueden encontrar aquí.

Maplin Electronics: <http://www.maplin.co.uk>

Radio Shack: <http://www.radioshack.com>

## Hobby Especialistas

Además de las principales cadenas, existen empresas más pequeñas que se especializan en trabajar con los aficionados. Aunque la variedad de artículos con la que cuentan no se compare con la que tienen las grandes cadenas, es comúnmente elegida y viene con la recomendación personal de la gente que está detrás de la empresa.

Algunas de estas tiendas de aficionados surgieron a raíz de Arduino, un proyecto de código abierto para crear una plataforma educativa de prototipado fácil de utilizar. Con la Raspberry Pi apelando a la misma audiencia que Arduino (aunque para tareas distintas) la mayoría de estas tiendas están incorporando a la RasPi entre sus líneas de productos.

Comprar en una tienda de aficionados tiene sus ventajas. Dado que el catálogo de productos vendidos en esas tiendas funcionan con la RasPi, nos ahorra mucho trabajo de investigación. Varias empresas diseñan sus propios hardware de expansión para varias plataformas, incluyendo a la RasPi. Estos dispositivos, están diseñados para hacer frente a una necesidad de la comunidad, pueden incluir puertos adicionales y/o hardware extra, o extender de

alguna otra manera la funcionalidad del dispositivo original.

En el Reino Unido, una de las tiendas de aficionados más populares es oomlout. Fundada por entusiastas del open-source que trabajaban con el hardware Arduino, son una fuente excelente para encontrar kits de expansión así como otros componentes comunes incluyendo botones, displays y transistores. A diferencia de los grandes minoristas, oomlout equipa sus componentes con todos los accesorios extras necesarios (como resistencias pull-up para los botones) y esquemas de los circuitos para su fácil montaje. Siempre que es posible, también se proporciona el código fuente de ejemplo para ver al componente los más pronto en funcionamiento.

En los Estados Unidos, Adafruit ofrece un servicio similar. Fundada con la intención de crear placas de expansión de código abierto para las tarjetas Arduino, Adafruit ofrece una amplia selección de componentes y kits (incluyendo una de las primeras placas de expansión diseñadas específicamente para la Raspberry Pi, ver Capítulo 16, “Placas de Expansión”, para más detalles).

Los sitios web de ambos minoristas pueden encontrarse aquí:

oomlout: <http://oomlout.co.uk/>

Adafruit: <http://www.adafruit.com>

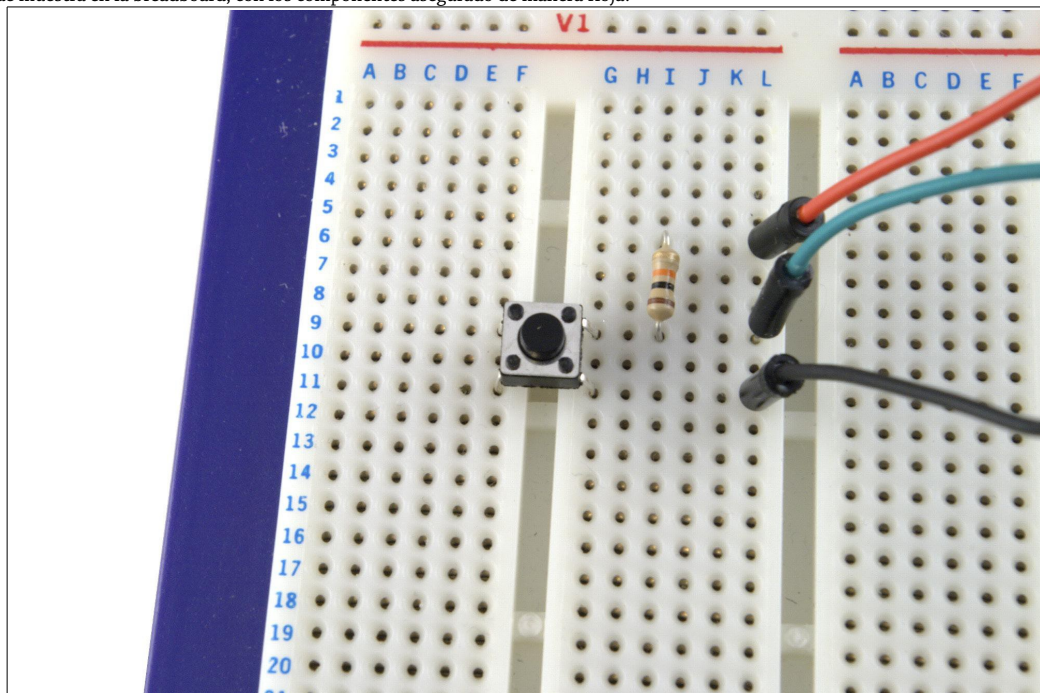
## Subiéndose a la Breadboard

Como se menciona anteriormente en este capítulo, las breadboard o protoboard son excelentes herramientas para crear circuitos prototipo para la experimentación de manera rápida. Estos circuitos son rápidos y reutilizables, y no dañan los componentes.

Sin embargo, las breadboards también tienen algunas desventajas. Son voluminosas, costosas y propensas a soltar las conexiones, lo que puede dar lugar a que los componentes críticos se caigan, especialmente cuando la breadboard está siendo transportada de un lugar a otro. La Figura 13-2 demuestra esto perfectamente: a pesar de los esfuerzos, el componente de botón se asegura de manera floja en la breadboard y es probable que se caiga si el circuito es manejado de manera brusca.

**Figura 13-2**

Un circuito de muestra en la breadboard, con los componentes asegurado de manera floja.





Esta razón, entre otras, es por la que la Raspberry Pi está construida sobre una *placa de circuito impreso (PCB)* en lugar de estar sobre una breadboard (aunque el método de la breadboard fue ciertamente utilizado en los primeros días de la creación del prototipo de la Raspberry Pi). Es posible imprimir y fabricar su propias PCB en casa, pero hay un paso intermedio más simple que puede tomar: utilizar una *stripboard* para crear circuitos independientes de forma permanente.

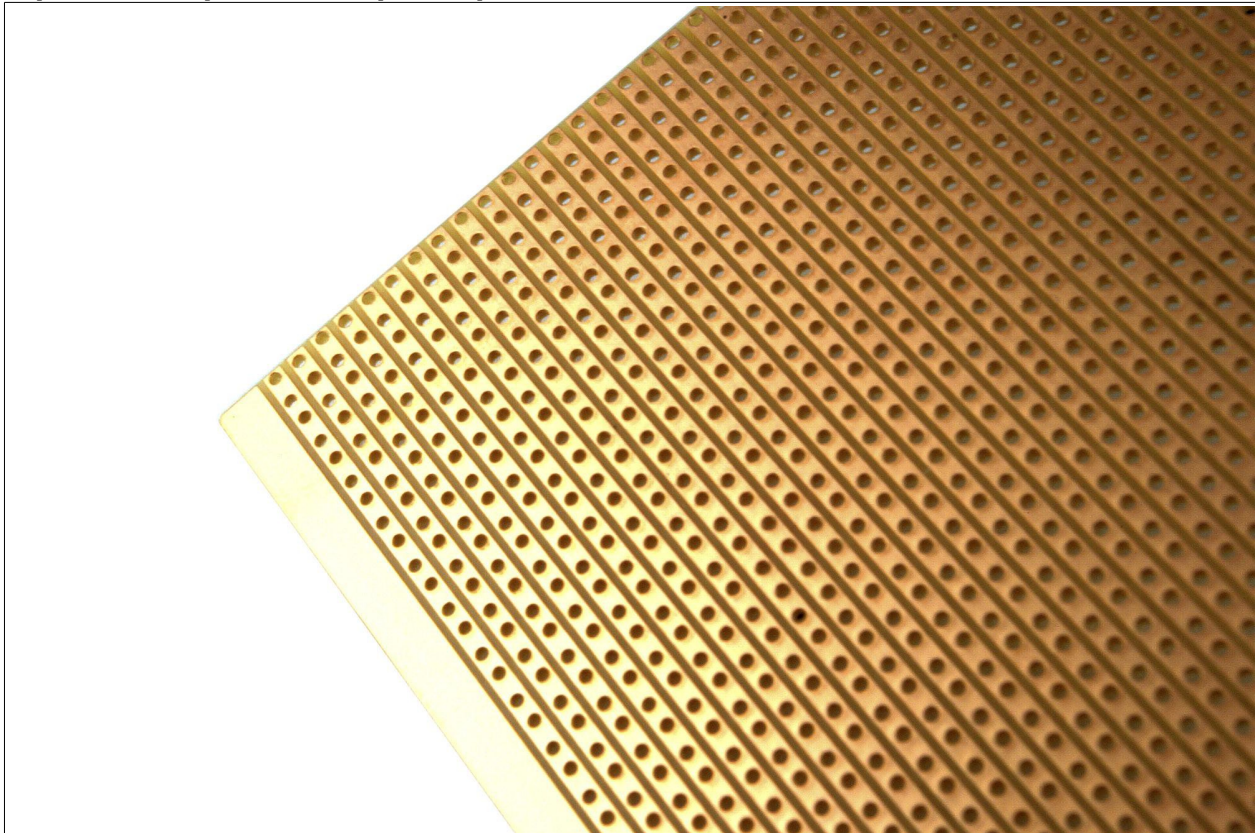
A primera vista, la stripboard se ve parecida a la breadboard, porque su superficie está cubierta de pequeños agujeros espaciados en 2.54mm. Sin embargo, a diferencia de la breadboard, no hay ningún mecanismo para asegurar que los componentes electrónicos colocados en estos agujeros queden fijos en su lugar, en vez de eso, usted tendrá que soldarlos en la stripboard. Con mucha frecuencia la stripboard es conocida por el nombre de su marca comercial que es *Veroboard*, la cual es una marca comercial de Vero Technologies en el Reino Unido y Pixel Print en Canadá.

La creación de un circuito stripboard tiene muchas más ventajas sobre la utilización de una breadboard. Una hoja de stripboard es significativamente más barata que su equivalente breadboard en tamaño y puede ser cortada para hacer los circuitos más pequeños. Además, esto permite que una sola pieza grande de stripboard sea utilizada en la creación de varios pequeños circuitos independientes.

Debido a que los componentes son soldados sobre la stripboard, los prototipos son también significativamente más robustos que los realizados en una breadboard. Un circuito stripboard puede ser transportado de un lugar a otro con poco riesgo de que sus componentes se desprendan y se pierdan. En la Figura 13-3 una pieza de stripboard volteada muestra las pistas de cobre que existen en su parte inferior.

**Figura 13-3**

Las pistas de cobre en la parte inferior de una pieza de stripboard



La stripboard es extremadamente fácil de utilizar, lo que la convierte en un gran trampolín para diseñar y fabricar placas de circuitos a la medida. Sin embargo, debe tener en cuenta lo siguiente antes de comprar una stripboard:

- Hay diferentes tipos de stripboard. Algunas stripboard tiene pistas de cobre en la parte inferior, que van todo el camino a través de una fila o bajan una columna, mientras que otras stripboards están divididas en dos filas separadas con un hueco en medio como una breadboard. Incluso hay otro tipo de stripboard, a menudo conocida como *placa proyecto* o *project board*, que no tienen pistas de cobre en absoluto y necesitan utilizar cables para unir sus componentes y estén eléctricamente juntos.
- Una Stripboard puede ser hecha en diferentes espesores y con diferentes materiales, y un tipo de stripboard podrá ser más adecuado para un determinado proyecto que para otro. Por ejemplo, un stripboard resistente al calor es buena para un circuito que va a estar en un ambiente con altas temperaturas, mientras que una stripboard más gruesa debe ser utilizada para los circuitos que serán expuestos al manejo rudo.
- Para hacer más ordenado el diseño de los componentes en una stripboard, es posible romper las pistas en la parte inferior para separar los componentes eléctricamente. Esto evita espacio desaprovechado en la placa y es una necesidad absoluta en los circuitos más complejos, pero para conseguir mejores resultados, necesitará una pequeña herramienta portátil llamada *cortador de pista*. Recuerde que debe añadirla a su lista de compras si está planeando utilizar una stripboard, aunque también una pequeña broca puede ser utilizada.

Existen también algunos trucos para manipular una stripboard que si son ignorados pueden hacerlas cosas más difíciles de lo realmente son:

- Las pistas de cobre en la parte inferior de la stripboard no están por lo general recubiertas en ninguna forma. Cuando toque las pistas, empañará el cobre y hará que sea más difícil de soldar. Evite tocar la parte inferior de la stripboard a menos que vaya a utilizarla inmediatamente; si es demasiado tarde, consiga un poco de estropajo de aluminio y dale al cobre un suave cepillado para eliminar la corrosión antes de intentar soldar.
- A diferencia de una placa de circuito impreso, la stripboard no tiene *máscara de soldadura* (una sustancia que impide a la soldadura que vaya a alguna otra parte que no sea a la que se supone debe ir). Como consecuencia, la soldadura es algo más complicada de lo que es en una PCB (es demasiado fácil unir accidentalmente las pistas con gotas demasiados grandes de soldadura). Si esto ocurre, utilice la trenza desoldadora para quitar el exceso de soldadura y vuelva a intentarlo.
- Los agujeros en la stripboard hacen que sea fácil quebrar la placa en tamaños personalizados, pero dejan los bordes irregulares. Después de quebrar una stripboard, tómese el tiempo para limar los bordes antes de montar su circuito. Asegúrese de usar una mascarilla cuando haga esto, porque el polvo de la stripboard no es particularmente saludable de respirar.

## Una Breve Guía de Soldadura

Tener un soldador es una cosa importante, pero más importante es saber cómo utilizarlo. La soldadura, como cualquier otra habilidad, requiere de práctica para alcanzar la maestría. Si sigue los consejos en esta sección y se mantiene practicando, será capaz de crear en poco tiempo limpias y ordenadas junturas de soldadura.

---

### ADVERTENCIA

Esto puede parecer obvio, pero vale la pena señalar que los soldadores se calientan mucho durante su uso. Asegúrese de no tocar la superficie metálica expuesta, incluso aléjese de la punta y tenga cuidado de dónde pone en reposo al cautín. Si es posible, compre uno con un soporte o consiga de manera separada un soporte resistente al calor. ¡Nunca deje un soldador caliente sin supervisión, y si se le cae, no tenga la tentación de tratar de atraparlo!

---

La soldadura trabaja fusionando una pequeña cantidad de metal para formar una unión entre dos componentes. Si voltea su Raspberry Pi, podrá ver mucha evidencia de esto. Todos los componentes electrónicos de mayor tamaño están conectados utilizando lo que se denomina como *soldadura through-hole*, en donde las patitas de los componentes son atravesados por los agujeros en la placa de circuito impreso y luego son soldados en su lugar. Los componentes electrónicos más pequeños están sujetos mediante *soldadura surface-mount* (*montaje superficial*).

La soldadura no está hecha de puro metal, también contiene una sustancia llamada *flux*, que está diseñada para arrasar cualquier deslustre sobre la superficie a ser soldada con el fin de garantizar que una unión sea lo más limpia posible. La mayoría de las soldaduras electrónicas incluyen de tres a cinco *núcleos* de flux. También puede comprar el flux en forma separada como una pasta o líquido, aunque para muchos de los aficionados en la soldadura, comprarla no será necesario.

Cuando está comenzando un proyecto de soldadura, asegúrese de contar con un espacio de trabajo limpio y bien iluminado. También asegúrese de que el área está bien ventilada. Los humos de soldadura no son saludables y aunque es extremadamente poco probable que se acumulen a niveles peligrosos en una soldadura de aficionado en bajo volumen, sigue siendo una buena idea exponerse lo mínimo que le sea posible.

Además, debe de proteger la superficie de trabajo de alguna manera. No es raro que pequeñas gotas de soldadura fundida caigan, lo que puede quemar y marcar las tablas. Puede comprar una estera de trabajo antiestática (ver Figura 13-4), pero una revista impresa en papel glaseado funciona igual de bien. No se sienta tentado a utilizar unas cuantas hojas de periódico barato (la soldadura puede atravesar el papel fino antes de que se enfríe).

**Figura 13-4**

Un ejemplo del área de trabajo con soldadura, con una estera antiestática protectora.



Si está realizando un trabajo delicado cercano a sus ojos, debe de utilizar gafas protectoras. A veces el flux hirviendo dentro de la soldadura puede causar que salpique hacia arriba y si le golpea un ojo, su mundo será muy doloroso.



Sin embargo, no permita que estas advertencias lo desanimen para soldar. Aunque la soldadura es extremadamente caliente, se enfría rápidamente y las quemaduras son raras pero siempre sin mucha consecuencia. Respete el equipo, pero no le tema.

Con su superficie de trabajo elegida y protegida, prepare su equipo. El soldador debe colocarse al lado de la mano dominante y colocado de modo que el cable no esté arrastrándose a través de su área de trabajo. Cercíorese de poder mover el soldador libremente antes de conectarlo. Si el cable se ve atrapado con algo, puede caerle encima y quemarse.

Humedezca la esponja de soldar en agua (debe estar húmeda pero no empapada). Esto es importante: la esponja húmeda será utilizada para limpiar el cautín, y si se encuentra seca, se quemará y puede dañar la delicada punta del cautín.

Le tomará unos cuantos minutos al cautín alcanzar su temperatura de funcionamiento. Si ha comprado un cautín con control de temperatura, normalmente le avisará que la temperatura se ha alcanzado a través de luces que se encienden y se apagan o mediante una lectura numérica de la temperatura (Consulte el manual de instrucciones que viene con su cautín para averiguar cómo son indicadas las temperaturas ).

Una vez que la temperatura de funcionamiento se ha alcanzado, es momento de preparar el cautín utilizando un proceso conocido como *estañado*. Siga estos pasos:

1. Empuje la punta de la soldadura contra la punta del cautín, permitiendo que una pequeña cantidad de derrita en el cautín. Sea cuidadoso de no derretir demasiado: no sólo se trata de pérdida de soldadura, sino que también el exceso puede caer sobre el área de trabajo.
2. Limpie la punta del cautín sobre la esponja. Si sisea y escupe, la esponja está demasiado húmeda; deje que se enfríe, luego retírela del soporte y exprímala.
3. Siga limpiando la punta del cautín hasta que se revista de una capa de plateada de soldadura (ver Figura 13-5). Si es necesario, aplique más soldadura a la punta.

**Figura 13-5**

Una punta de cautín que está en proceso de ser estañada con soldadura.



Este proceso de estañado de la punta la protege contra el daño y asegura que transferirá eficientemente el calor sobre las superficies a soldar. El estañado inapropiado de la punta del cautín es una de las causas más comunes de las malas uniones de soldadura. Es posible que tenga que repetir este proceso más de una vez si está soldando una gran cantidad de conexiones y necesitará hacerlo de nueva cuenta al finalizar la sesión de soldadura. En general, si la punta del cautín pierde su capa brillante, repita el proceso de estañado.

Con la punta preparada, es hora de empezar a soldar. Coloque los objetos a soldar (tales como una placa de circuito impreso y las patas de un componente) en el soporte de trabajo o tercera mano y asegúrese de que tiene una buena vista. Extienda una longitud de soldadura del contenedor o carrete y comience a soldar los componentes utilizando estos pasos básicos:

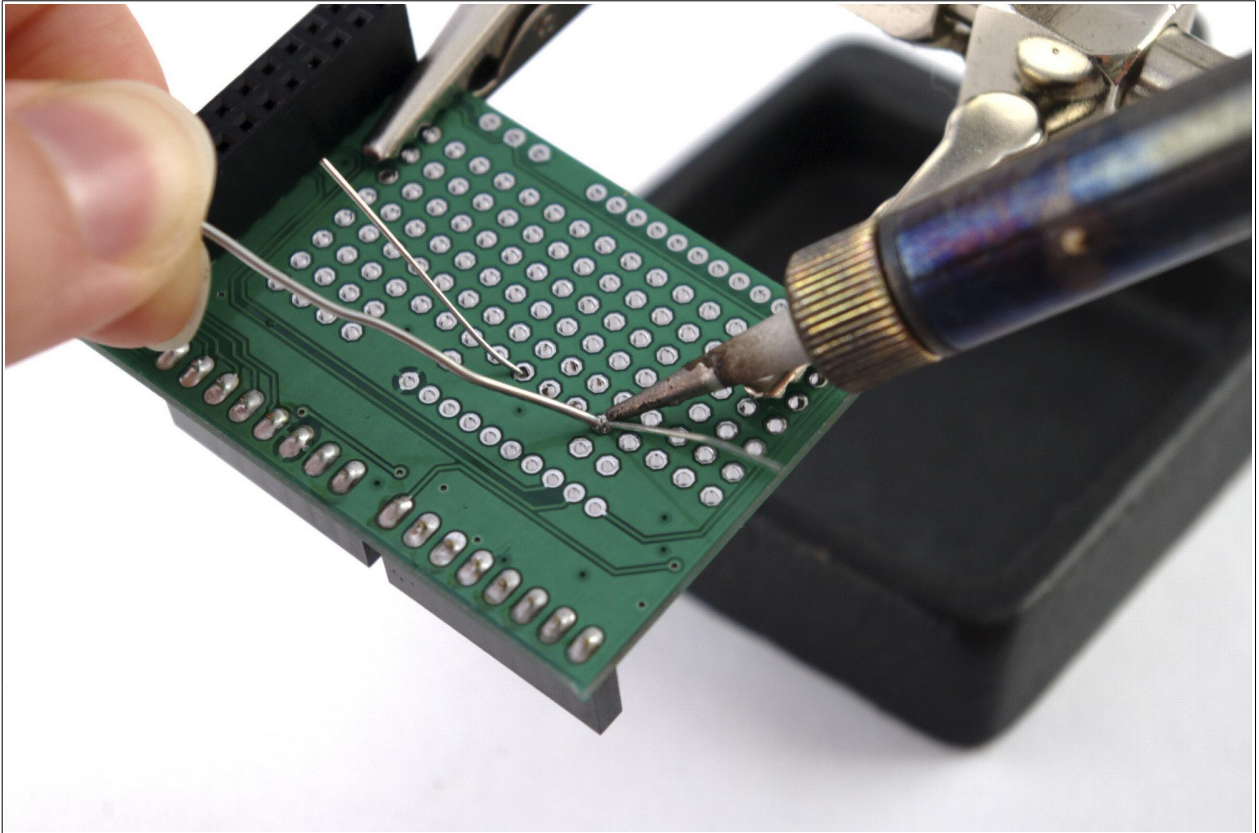
1. Si está soldando componentes through-hole (de inserción) en una placa de circuito impreso, stripboard o alguna placa through-hole similar, coloque las patas del componente a través de los agujeros y dóblelas hacia fuera del componente para que no se caiga cuando la placa sea volteada.
2. Con la placa asegurada en el soporte de trabajo o tercera mano, empuje la punta del cautín contra el componente y el contacto de cobre sobre la placa. Es importante que el cautín toque ambos elementos: si el cautín sólo toca uno, la junta finalizada será mala.
3. Le toma sólo unos cuantos segundos al área calentarse. Cuente hasta tres, y empuje nuevamente el cautín contra el componente y el contacto de cobre (ver Figura 13-6). Si la soldadura no se derrite, retírela, cuente un par de segundos más y luego intente de nuevo. Si todavía no se derrite, intente cambiar la posición del cautín.
4. Mientras la soldadura fluye, la observará caer abajo dentro del agujero en la placa. Esto es una indicación de que el área está lo suficientemente caliente para hacer una buena junta de soldadura. Si la soldadura flota, indica que el área no está lo suficientemente caliente.
5. Retire la soldadura de la unión primero, seguido por el cautín. (¡Si retira el cautín primero, la soldadura se endurecerá y deja su carrete de soldadura pegado al contacto!)

Si todo ha ido bien, le dejará con una unión sólida que le durará muchos años. Si no es así, no se desanime, sólo presione el cautín de nuevo a la unión para volver a convertir en fluido la soldadura y utilice la trenza de desoldar si es necesario para limpiar algún derrame o exceso. Una unión de soldadura perfecta debe tener la forma parecida a la de un volcán, que se eleva desde la superficie de la placa para unirse a la pata del componente.

Nunca permita que el cautín entre en contacto con las partes por más de unos cuantos segundos. Esto es especialmente importante cuando está soldando algunos componentes sensibles al calor como pueden ser los circuitos integrados, ya que pueden dañarse por el contacto prolongado con un cautín caliente. Si está utilizando una estación de soldado de temperatura controlada, asegúrese de que la temperatura se encuentra establecida a un nivel apropiado antes de utilizar el cautín (verifique el empaque de su cautín o la hoja de datos para obtener más información).

Figura 13-6

Soldando un componente through-hole en una placa de circuito impreso



Cuando haya finalizado, no olvide re-estañar la punta del cautín. Si no lo hace, la punta puede corroerse en el almacenaje y necesitará remplazar su cautín mucho antes de lo que debería.

**CONSEJO**

Recuerde calentar ambas superficies con el cautín. Calentar sólo una superficie resulta en lo que se conoce como una *junta seca* o *junta fría*, en donde la soldadura no queda adecuadamente acoplada a una superficie. Con el tiempo, estas uniones fallan y necesitan ser soldadas nuevamente.

Como cualquier otra habilidad, la soldadura requiere de práctica. Muchas tiendas de electrónica venden kits que incluyen una placa de circuito impreso y una selección de componentes sueltos, que pueden utilizarse para practicar la soldadura de componentes through-hole. Algunas placas de expansión para la Raspberry Pi también son proporcionadas en forma de kits y necesitan ser soldados, como la placa Slice of Pi de Ciseco. Aprenderá sobre esta placa en el Capítulo 16, “Placas de Expansión”.



# Capítulo **14**

## El Puerto GPIO

**El puerto** de entrada y salida de propósito general (GPIO) de la Raspberry Pi está localizado en la parte superior izquierda de la placa de circuito impreso, etiquetado con P1. Es un puerto de 26 pines, equipado de fábrica con dos filas de 13 conectores macho de 2.54 mm. La separación de estos pines es particularmente importante: el espaciado de 2.54 mm (0.1 pulgadas en medidas imperiales) es muy común en la electrónica y es el espaciado estándar de las plataformas para prototipos como las stripboards y breadboards.

## Identificando la Revisión de su Placa

El pin layout o diseño de pines del puerto GPIO de la Raspberry Pi difiere ligeramente según el modelo de la revisión de la placa que tenga. Todos los modelos A y modelos B más recientes cuentan con el mismo diseño de pines; el Modelo B original, no obstante, tiene diseño ligeramente distinto.

Antes de utilizar el puerto GPIO para un proyecto de hardware, asegúrese de saber cuál es el modelo de Raspberry Pi que tiene. Si tiene un Modelo A, puede utilizar el primer diagrama en este capítulo; si tiene un Modelo B, necesitará utilizar el primer diagrama (a menos que se trate del modelo original Revisión 1).

La forma más fácil de averiguar si es una placa Modelo B Revisión 1 es mirando en la cara superior de la placa, justo por debajo de los mismo pines GPIO: en donde una placa Revisión 2 tiene dos filas de cuatro orificios bañados en cobre, la Revisión 1 es sólida. Otro método de averiguarlo es mirando cuánta memoria tiene disponible su placa: la Revisión 1 del Modelo B tiene sólo 256 MB de memoria, mientras que la Revisión 2 tiene 512 MB.

Para más detalles sobre las revisiones de la placa y cómo distinguirlas, vea el Capítulo 1, “Conozca la Raspberry Pi”.

## Diagramas Pinout del puerto GPIO

Cada pin del puerto GPIO tiene su propósito, varios pines trabajando en conjunto pueden formar un circuito en particular. El diseño del puerto GPIO puede verse en la Figura 14-1.

El puerto GPIO es el mismo tanto en el Modelo A como en el Modelo B Revisión 2 de la Raspberry Pi. Si tiene una de las placas originales de la Raspberry Pi Modelo B Revisión 1, aparte de tener una pieza de colección en sus manos, los pines sobre el puerto GPIO serán ligeramente diferentes. Estos detalles pueden verse en la Figura 14-2.

---

### CONSEJO

Si no está seguro de qué revisión de la Raspberry Pi tiene, y por lo tanto, no sabe qué pinout aplica a su placa, cargue IDLE y escriba `import RPi.GPIO as GPIO` seguido por `GPIO.RPI_REVISION` para devolver el número de la revisión.

---

La numeración de los pines del puerto GPIO está dividida en dos filas, la fila inferior toma los números impares y la fila superior los números pares. Es importante tener esto en cuenta a la hora de trabajar con el puerto GPIO de la RasPi: la mayoría de los otros dispositivos electrónicos utilizan un sistema diferente de enumeración de pines y debido a que no existen serigrafías sobre la misma placa de la Raspberry Pi, es fácil confundir cuál pin es cuál.

Figura 14-1  
El puerto GPIO de la Raspberry Pi y la definición de sus pines

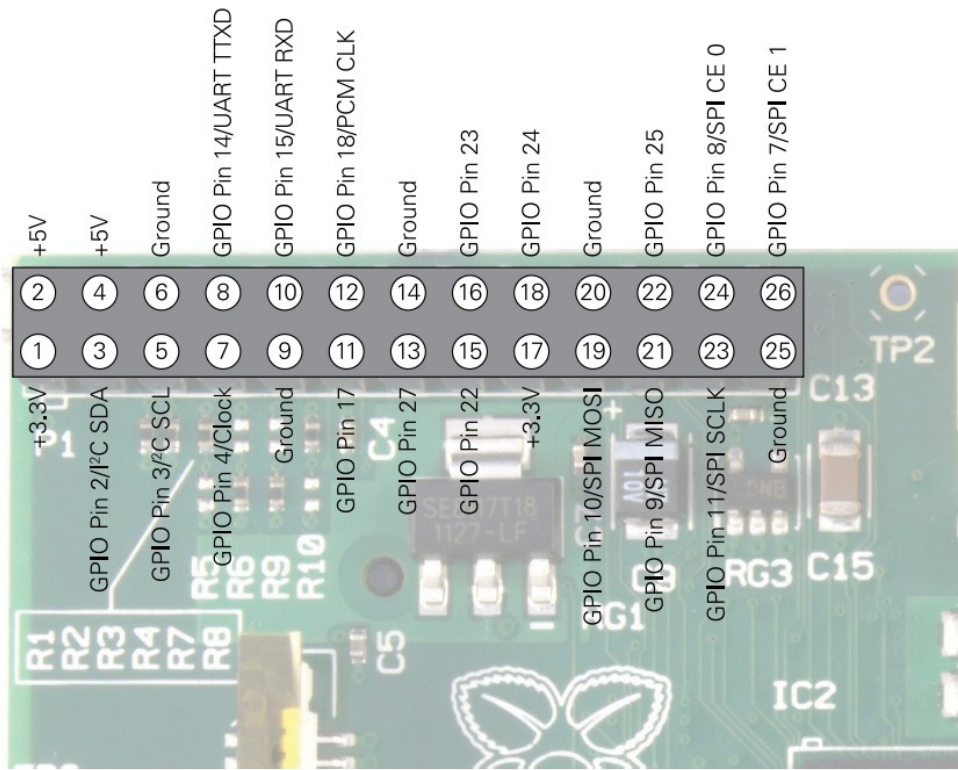
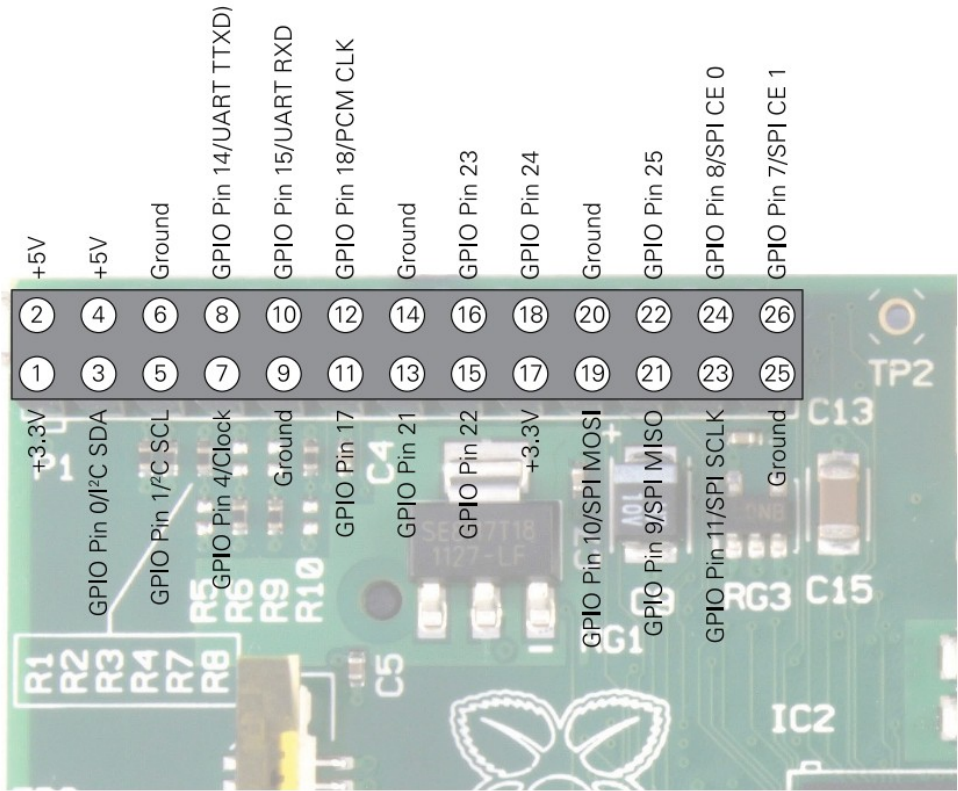


Figura 14-2  
El piout de la Raspberry Pi Modelo B Revisión 1 original



Aunque el puerto GPIO de la RasPi ofrece un suministro de energía de 5 V, proveniente de la energía entrante sobre el hub micro usb, sobre el Pin 2, el funcionamiento interno de la RasPi se basa en lógica de 3.3 V. Esto significa que los componentes en la RasPi funcionan con un suministro de energía de 3.3 V. Si está planeando crear un circuito que haga de interfaz con la RasPi a través del puerto GPIO, asegúrese de que está utilizando componentes compatibles con lógica de 3.3 V o que el circuito está pasando a través de un *regulador de voltaje* antes de llegar a la RasPi.

---

**ADVERTENCIA**

Conectar un suministro de 5 V a cualquier pin del puerto GPIO de la Raspberry Pi, o hacer cortocircuito directamente con cualquiera de los dos pines de alimentación (Pin 1 y Pin 2) hacia cualquier otro pin dañará la RasPi. Debido a que el puerto está directamente conectado a los pines sobre el procesador SoC BCM2835, no será capaz de reparar algún daño cometido. Siempre tenga mucho cuidado cuando trabaje cerca del puerto GPIO.

---

## Características del GPIO

Por defecto, el puerto GPIO dispone de 8 pines para uso general: el Pin 7, Pin 11, Pin 12, Pin13, Pin 15, Pin16, Pin 18 y el Pin 22. Estos pines pueden ser conmutados entre dos estados: *alto (high)*, donde los pines proporciona un voltaje positivo de 3.3 V; y *bajo (low)*, donde los pines son iguales a tierra o a 0 V. Esto equivale al 1 y 0 de la lógica binaria, y puede utilizarse para cambiar el estado de los otros componentes a activos o inactivos. Más adelante en este capítulo aprenderá más sobre esto.

---

**ADVERTENCIA**

La lógica interna de la RasPi funciona a 3.3 V. Esta lógica está en contraste con la de muchos otros microcontroladores comunes, como el popular Arduino y sus variantes, que típicamente operan a 5 V. Los dispositivos diseñados para Arduino pueden no funcionar con la RasPi a menos que un *trasladador de nivel* o *aislador óptico* sea utilizado entre los dos dispositivos. Asimismo, conectar los pines de un microcontrolador de 5 V directamente al puerto GPIO de la Raspberry Pi no funciona y puede dañar de forma permanentemente la RasPi.

---

Además de estos pines de propósito general, el puerto GPIO cuenta con pines dedicados a *buses* en particular. Estos buses se describen en los siguientes apartados.

## Bus Serie UART

El *bus serie Receptor/Transmisor Asíncrono Universal (UART)* proporciona una interfaz serie en dos cables. Cuando un puerto serie es configurado en el archivo `cmdline.txt` (como se describió en el Capítulo 7, “Configuración Avanzada de la Raspberry Pi”), es este bus serie el que está siendo utilizado como el puerto para los mensajes. Conectando en el bus serie UART de la RasPi un dispositivo capaz de desplegar datos, dejará al descubierto los mensajes del núcleo Linux. Si está experimentando problemas al arrancar el núcleo Linux, esta puede ser una herramienta de diagnóstico muy útil (especialmente si no aparece nada sobre la pantalla).

El bus serie UART se puede acceder sobre los Pines 8 y 10, el Pin 8 lleva la señal de *transmisión* y el Pin 10 lleva la señal de *recepción*. La velocidad puede ser establecida en el archivo `cmdline.txt`, y por lo general es de 115,200 bits por segundo (bps).

## El Bus I<sup>2</sup>C

Como su nombre lo indica, el *bus inter-circuitos integrados (I<sup>2</sup>C)* está diseñado para permitir comunicaciones entre múltiples circuitos integrados (ICs). En el caso de la RasPi, uno de estos circuitos integrados es el procesador SoC Broadcom BCM2835 en el interior del sistema. Estos pines incluyen acceso a las *resistencias pull-up* localizadas en la RasPi, lo que significa que no requiere de resistencias externas para acceder a la funcionalidad I<sup>2</sup>C.



El bus I<sup>2</sup>C se puede acceder sobre los Pines 3 y 5, el Pin 3 proporciona la señal *SDA* (*Serial Data Line*) y el Pin 5 proporciona la señal *SCL* (*Serial Clock*). El bus I<sup>2</sup>C disponible sobre estos pines es en realidad sólo uno de los dos buses proporcionados por el propio chip BCM2835, y es conocido como I<sup>2</sup>C0. El segundo bus, I<sup>2</sup>C1, está terminado en las resistencias sobre misma placa de circuito de la Raspberry P y no está disponible para su uso general.

## Bus SPI

El *bus Interfaz de Periférico Serie (SPI)* es un bus serie síncrono diseñado principalmente para la *programación in-system (ISP)* de microcontroladores y otros dispositivos. A diferencia de los buses UART y I2C, este es un bus de cuatro hilos con múltiples líneas *Chip Select*, lo que le permite comunicarse con más de un dispositivo de destino.

El bus SPI de la RasPi se encuentra disponible sobre los Pines 19, 21 y 23, con un par de líneas *Chip Select* sobre el Pin 24 y el Pin 26. El Pin 19 proporciona la señal *MOSI* (*SPI Master Output, Slave Input*); el Pin 21 proporciona la señal *MISO* (*SPI Master Input, Slave Output*); el Pin 23 proporciona la señal *SLCK* (*Serial Clock*) utilizada para sincronizar la comunicación; y los Pines 24 y 26 proporcionan las señales *Chip Select* para un máximo de dos dispositivos esclavo independientes.

Aunque existen otros buses adicionales presentes en el procesador SoC BCM2835 de la RasPi, estos no tiene salida por el puerto GPIO y por lo tanto no están disponibles par su uso por el usuario.

## Utilizando el Puerto GPIO en Python

Con la teoría fuera del camino, ha llegado la hora de pasar a la práctica. En este sección, aprenderá cómo construir y programar dos circuitos electrónicos sencillos que demuestran cómo puede utilizar el puerto GPIO para controlar entradas y salidas.

Como pudo darse cuenta en el Capítulo 12, “Una Introducción a Python”, Python es un lenguaje de programación poderoso y fácil de utilizar. No es, sin embargo, la elección perfecta para todos los escenarios. Aunque Python trabaja bien para los circuitos sencillos que va a crear en este capítulo, no ofrece lo que se denomina como *operación en tiempo real determinista*. Para la mayoría de los usuarios, esto no importa; sin embargo, si está planeando utilizar la RasPi como corazón para un reactor nuclear o una plataforma robótica compleja, es posible que desee incursionar en un lenguaje de bajo nivel como C++ o incluso el *ensamblador* se ejecuta en un tiempo real dedicado del microcontrolador

Si en su proyecto requiere de verdadero funcionamiento en tiempo real, la RasPi puede ser una mala elección. En su lugar, considere utilizar una plataforma de microcontrolador como el popular Arduino de código abierto, o alguno de los microncontroladores de la familia MSP430 de Texas Instruments. Ambos de estos dispositivos pueden interactuar con la RasPi a través del puerto GPIO o mediante la interfaz USB y cuentan con un entorno en tiempo real especializado para el control y la detección.

## Salida GPIO: Parpadeo de un LED

Como primer ejemplo, construirá un circuito simple que consiste de un LED y una resistencia. El LED proveerá una confirmación visual de que el puerto GPIO hace lo que su programa Python le dice que haga y la resistencia limitará la corriente consumida por el LED para protegerlo de que se queme.

## Cálculo del Valor Límite de las Resistencias

Un LED necesita de una *resistencia que limite la corriente* para protegerlo de que se quemé. Sin una resistencia, un LED probablemente sólo funcionaría por un corto tiempo antes de fallar y necesitaría ser reemplazado. Es importante saber que las resistencias son necesarias, pero también es importante saber escoger la resistencia adecuada para el trabajo. Un valor demasiado alto y el brillo del LED será extremadamente débil o no encenderá en absoluto; un valor demasiado bajo y se quemará.

Para calcular el valor de la resistencia requerida, necesitará conocer la *corriente de trabajo (forward current)* de su LED. Esta es la máxima intensidad de corriente que puede circular a través de su LED sin que éste sea dañado y se mide en miliamperios (mA). También necesitará conocer la *tensión directa (forward voltage)* del LED. Este último valor, se mide en voltios y debería ser de 3.3 V o más bajo, un valor más alto y el LED requerirá de una fuente de alimentación externa y un dispositivo de conmutación conocido como *transistor* antes de poder trabajar con la RasPi.

La manera más fácil de calcular cuán grande una resistencia necesita ser es utilizando la fórmula  $R=(V-F)/I$ , en donde R es la resistencia en ohmios, V es la tensión aplicada al LED, F es la tensión directa del LED e I es la máxima corriente de trabajo del LED en amperios (con mil mA equivaliendo a un amperio).

Tomando un típico LED rojo con una corriente de trabajo de 25 mA y una tensión directa de 1.7 V y alimentándolo mediante el suministro de 3.3 V del puerto GPIO, se puede calcular la resistencia requerida con  $(3.3 - 1.7) / 0.025 = 64$ . Por lo tanto, una resistencia de 64  $\Omega$  o mayor protegerá al LED. Las cifras obtenidas raramente coinciden con los valores comunes de las resistencias que se venden, así que cuando elija una resistencia, siempre hay que redondear al valor superior para asegurarse que el LED estará protegido. El valor disponible de resistencia comúnmente más cercano es de 68  $\Omega$ , el cual protegerá apropiadamente al LED.

Si no conoce la tensión directa y la corriente de trabajo de sus LEDs (por ejemplo, si el LED no viene con documentación o fueron rescatados de entre la chatarra electrónica), puede tratar a prueba y error, colocando una resistencia razonablemente grande. Si la luz es demasiado débil, puede probar con una resistencia más baja (pero recuerde que es imposible reparar un LED quemado).

Para montar el circuito, necesitará una breadboard, dos cables jumpers, un LED y una resistencia limitadora de corriente apropiada (como se describió en el “Cálculo del Valor Límite de las Resistencias”). Aunque es posible montar el circuito sin utilizar una breadboard con sólo retorcer los cables juntos, adquirir una breadboard será una buena inversión y nos permitirá que el montaje y desmontaje de circuitos prototipo sea sencillo.

Suponiendo que está utilizando una breadboard, monte el circuito de la siguiente manera para que coincida con el que se muestra en la Figura 14-3:

1. Inserte el LED en la breadboard de manera que la pata larga (el *ánodo*) esté en una fila y la pata corta (el *cátodo*) esté en otra. Si pone las patas del LED en la misma fila, no va a funcionar.
2. Inserte una pata de la resistencia en la misma fila que la pierna corta del LED, y la otra pata de la resistencia en una fila vacía. La dirección en la que se colocan las patas de la resistencia no importa, ya que una resistencia es un dispositivo *no polarizado* (insensible a la dirección).
3. Utilizando un cable jumper, conecte el Pin 11 del puerto GPIO de la Raspberry Pi (o el pin correspondiente en una placa interfaz conectada al puerto GPIO) a la misma fila que la pata larga del LED.
4. Utilizando otro cable jumper, conecte el Pin 6 del puerto GPIO (o el pin correspondiente en una placa interfaz conectada al puerto GPIO) a la fila que contiene sólo una pata de la resistencia y ninguna de las patas de LED.

**ADVERTENCIA**

Tenga mucho cuidado al momento de conectar los cables al puerto GPIO de la Raspberry Pi. Como se discutió anteriormente en este capítulo, puede causar graves daños a la RasPi si conecta mal los pines.

Llegados a este punto, nada sucederá. Esto es perfectamente normal: por defecto, los pines GPIO de la Raspberry Pi se encuentran apagados. Si desea comprobar el funcionamiento del circuito inmediatamente, mueva el cable del Pin 11 al Pin 1 para hacer que el LED ilumine. Sin embargo, tenga cuidado de no conectarlo al Pin 2: una resistencia limitadora de corriente adecuada para una fuente de alimentación de 3.3 V no será la apropiada para proteger al LED cuando se conecte al suministro de 5 V. Recuerde mover el cable de regreso al Pin 11 antes de continuar.

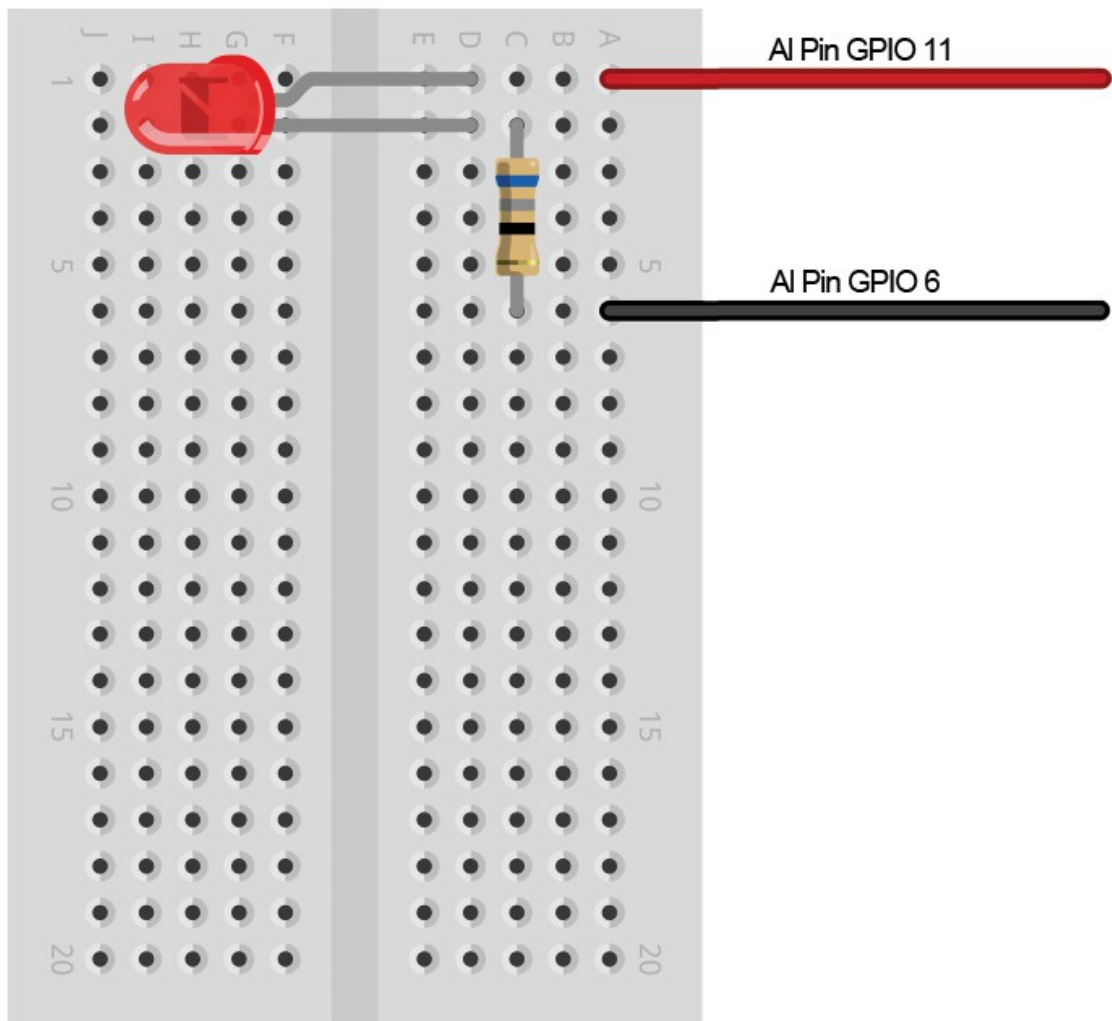
Para hacer que el LED haga algo útil, comience un nuevo proyecto Python. De la misma forma que hizo con los proyectos del Capítulo 12, “Una Introducción a Python”, puede utilizar un editor de texto plano o el software IDLE incluido en la distribución Raspbian para crear este proyecto.

Antes de que pueda utilizar el puerto GPIO de la Raspberry Pi en Python, necesitará importar una librería dentro de su proyecto Python. Para hacerlo, comience el archivo con la siguiente línea:

```
import RPi.GPIO as GPIO
```

Figura 14-3

Un circuito en la breadboard para obtener una salida simple con un LED



Recuerde que Python es sensible a las mayúsculas y minúsculas, así que asegúrese de escribir `RPi.GPIO` exactamente como aparece. Para permitir a Python entender el concepto de tiempo (en otras palabras, hacer que el LED parpadee, encendiéndose y apagándose), necesitará también importar el módulo `time`. Agregue la siguiente línea al proyecto:

```
import time
```

Con las librerías ya importadas, es momento de controlar los puertos GPIO. La librería GPIO nos facilita controlar los puertos de propósito general a través de las instrucciones `GPIO.output` y `GPIO.input`, pero antes de poder utilizarlas, necesitará establecer la librería GPIO al modo *board* (esto se hace para que la enumeración de los pines en la librería GPIO coincida con la enumeración física del puerto GPIO sobre placa Raspberry Pi) y después inicializar los pines ya sea como entradas o salida. En este ejemplo, el Pin 11 es una salida, por lo tanto agregue la siguiente línea al proyecto:

```
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11,GPIO.OUT)
```

La última línea le dice a la librería GPIO que el Pin 11 sobre el puerto GPIO de la Raspberry Pi debe ser configurada como una salida. Si va a controlar dispositivos adicionales, puede agregar más líneas `GPIO.setup` al proyecto. Sin embargo, por ahora, con una sola bastará.

Ya con el pin configurado como una salida, se puede conmutar su suministro de 3.3 V entre encendido y apagado en una simple demostración de lógica binaria. La instrucción `GPIO.output(11, True)` transformará al pin en activo, mientras `GPIO.output(11, False)` lo transformará en inactivo. El pin recordará su último estado, de modo que si sólo da la orden de cambiar al pin a su estado activado y luego se sale del programa Python, el pin mantendrá su estado activo hasta que se le indique lo contrario.

Aunque con sólo agregar `GPIO.output(11,True)` al proyecto Python se puede activar el pin, será más interesante hacer que éste parpadee. Primero, agregue la siguiente línea para crear un bucle infinito en el programa:

```
while True:
```

A continuación, agregue las siguientes líneas para activar el pin, esperar dos segundos y luego desactivarlo volviendo a esperar otros dos segundos. Cerciórese de que cada línea comienza con cuatro espacios para indicarle a Python que estas líneas son parte del bucle `while` infinito:

```
    GPIO.output(11, True)
    time.sleep(2)
    GPIO.output(11, False)
    time.sleep(2)
```

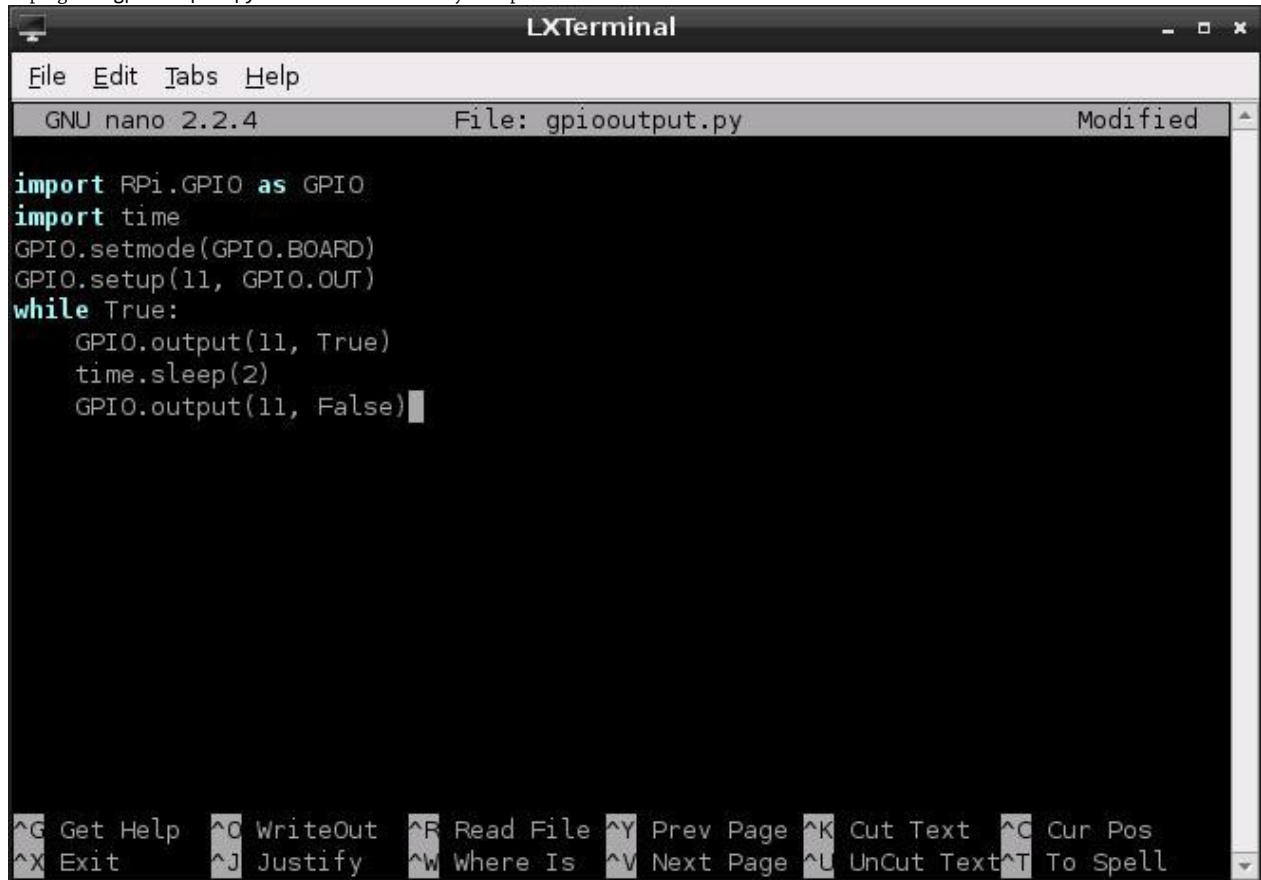
El programa terminado debería verse de esta forma (ver Figura 14-4):

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)
while True:
    GPIO.output(11, True)
    time.sleep(2)
    GPIO.output(11, False)
    time.sleep(2)
```

Guarde el archivo como `gpiooutput.py`. Si está utilizando un entorno de desarrollo Python como SPE, no intente ejecutar el programa desde el editor interno. La mayoría de las distribuciones Linux Raspberry Pi limitan el uso del puerto GPIO al usuario `root`, por lo que será necesario que el programa sea ejecutado utilizando el comando `sudo python gpiooutput.py` en la terminal para ponerlo en marcha. Si todo ha salido bien, debería de observar que el LED comienza a parpadear, prendiéndose y apagándose a intervalos regulares de tiempo (así es como ha creado su primera salida casera a un dispositivo para la Raspberry Pi).

Figura 14-4

El programa `gpiooutput.py` siendo editado en nano y en espera de su línea final



```
File Edit Tabs Help
GNU nano 2.2.4 File: gpiooutput.py Modified
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)
while True:
    GPIO.output(11, True)
    time.sleep(2)
    GPIO.output(11, False)
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos  
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

Si las cosas no funcionan, no se asuste. Primero, verifique todas sus conexiones. Los orificios en la breadboard son bastante pequeños y es fácil pensar que ha insertado un componente en una fila para descubrir que en realidad lo insertó en otra. A continuación, verifique que ha conectado el circuito a los pines correctos en el puerto GPIO (al no contar con etiquetas la Raspberry Pi, desafortunadamente los errores son fáciles de cometer). Finalmente, revise dos veces sus componentes, si la tensión directa de su LED es más alta que los 3.3V o si la resistencia limitadora de corriente es demasiado grande, el LED no se iluminará.

Aunque este ejemplo es básico, resulta una buena demostración de algunos de los conceptos fundamentales. Para extender su funcionalidad, el LED podría reemplazarse con un zumbador o buzzer para crear una alerta audible, o con un servo o motor como parte de una plataforma robótica. El código utilizado para activar y desactivar el pin GPIO puede ser incorporado dentro de otros programas, provocando que un LED se ilumine cuando un nuevo correo electrónico nos llega o cuando un amigo se ha unido a un canal IRC, por ejemplo.

## Entrada GPIO: Lectura de un Botón

Utilizar el GPIO como una salida es indudablemente algo útil, pero es aún más útil cuando se combina con una o más entradas. En el siguiente ejemplo, verá cómo conectar un botón en otro pin del puerto GPIO y leerá su estado desde Python.

Si tiene construido el ejemplo de salida GPIO, puede desconectarlo de su RasPi o dejarlo conectado (este ejemplo utiliza diferentes pines, por lo que ambos pueden coexistir felizmente). Si deja el anterior ejemplo conectado, asegúrese de utilizar diferentes filas en la breadboard para conectar los nuevos componentes o encontrará que las cosas no funcionan como estaba previsto.

Construya el circuito de la manera siguiente, para que coincida con la imagen mostrada en la Figura 14-5:

1. Inserte el botón en la breadboard. La mayoría de estos interruptores tienen dos o cuatro patas. Sólo necesita preocuparse por dos de las patas de este circuito. Si el botón tiene cuatro patas, serán configuradas en pares: verifique la hoja de datos de su botón para averiguar que patas están emparejadas.
2. Conecte una de las patas de una resistencia de 10 K $\Omega$  en la misma fila que una de las patas del botón y la otra en una fila sin usar. Esta es una resistencia pull-up, y proporcionará a la RasPi con un voltaje de referencia de tal forma que ésta sabrá cuando el botón ha sido presionado.
3. Conecte la pata sin utilizar de la resistencia pull-up al Pin 1 del puerto GPIO de la Raspberry Pi. Esto proporciona el voltaje de referencia de 3.3 V.
4. Conecte la pata sin utilizar del botón al Pin 6 del puerto GPIO de la Raspberry Pi. Esto proporciona la conexión a tierra.
5. Por último, conecte el Pin 12 del puerto GPIO de la Raspberry Pi a la otra pata del botón en la misma fila que la resistencia de 10 K $\Omega$ . Su breadboard debe ahora verse como en la Figura 14-5.

El circuito que acaba de construir crea una situación en la que el pin de entrada (que en este caso, es el Pin 12 del puerto GPIO), alcanza constantemente su estado *alto* (*high*) gracias a la resistencia pull-up conectada a un suministro de 3.3 V. Cuando el botón es presionado, el circuito conecta a tierra y se vuelve a su estado *bajo* (*low*), proporcionando la señal para que su programa Python sepa que el botón ha sido activado.

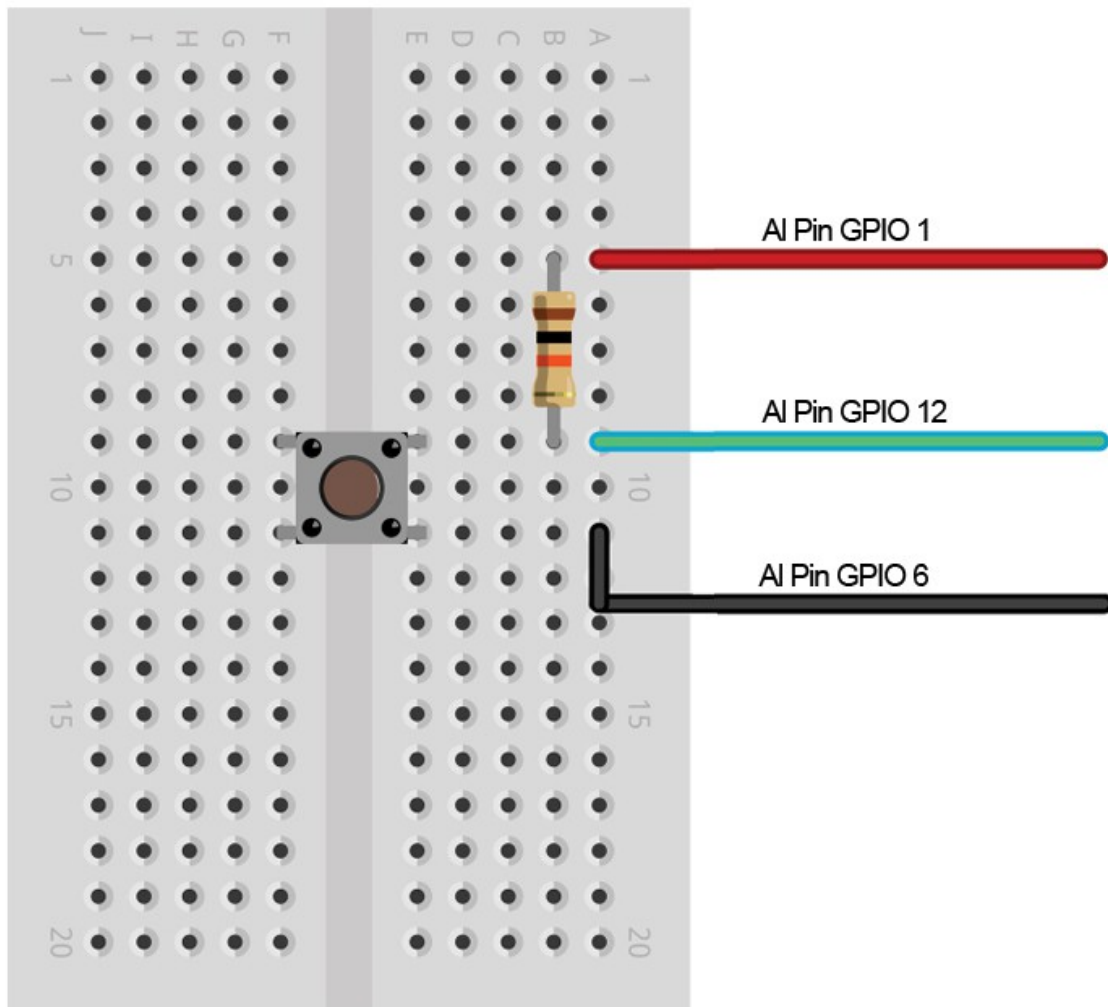
Puede preguntarse por qué la resistencia es absolutamente necesaria y por qué el interruptor no conecta simplemente el Pin 12 al Pin 6 o al Pin 1 de manera directa. Aunque esto es posible, esto crea lo que se conoce como *un pin flotante*, el cual es un pin que no sabe si está en estado alto o bajo. Como resultado, el circuito actuaría como si el botón estuviese siendo presionado incluso cuando no lo está, y puede fallar en detectar cuando el botón está siendo presionado cuando así es.

Abra un nuevo archivo Python ya sea en el editor de texto o utilizando uno de los entornos de desarrollo integrado Python (IDLEs) disponibles en la Raspberry Pi. Para comenzar, necesitará importar la misma librería GPIO que en el ejemplo pasado de la salida GPIO:

```
import RPi.GPIO as GPIO
```

Figura 14-5

El diseño de la breadboard del ejemplo para un sencillo botón de entrada



Esta vez, no necesita importar la librería `time`, porque en este ejemplo no requiere de instrucciones relacionadas con el tiempo. En vez de eso, puede habilitar adecuadamente el modo `board` y configurar el Pin 12 como una entrada. Esto se hace de la misma manera que hizo cuando configuró un pin de salida, sólo que con un ligero cambio en la parte final de la instrucción:

```
GPIO.setmode(GPIO.BOARD)
GPIO.setup(12, GPIO.IN)
```

Si no desea emplear el Pin 12 para tarea, asegúrese de cambiar el número del pin en la instrucción anterior.

Al igual que en el ejemplo pasado, el siguiente paso es crear un bucle infinito que constantemente comprobará si el pin de entrada se encuentra en su estado *bajo* (en otras palabras, si ha sido presionado). Comience el bucle con la siguiente línea de código:

```
while True:
```

Leer el estado de un pin de entrada es muy similar a establecer el estado de un pin de salida, con una excepción: antes de poder hacer algo útil con el valor obtenido, necesitará almacenar este valor en una variable. La siguiente instrucción le dice a Python que cree una nueva variable llamada `input_value` (como se describe en el Capítulo 12, “Una Introducción a Python”) y que le asigne el valor actual del Pin 12:

```
input_value = GPIO.input(12)
```

Aunque el programa puede ser ejecutado ahora y funcionar, no hace nada útil. Para asegurarse de saber qué es lo que está pasando, agregue la siguiente instrucción `print` para obtener retroalimentación:

```
if input_value == False:
    print("The button has been pressed.")
    while input_value == False:
        input_value = GPIO.input(12)
```

Las dos últimas líneas (el segundo `while` y el segundo `input_value`, un *bucle anidado*) son importantes. Aún en el procesador de la Raspberry Pi (que es relativamente menos potente si se compara con los procesadores de las laptops y PCs de escritorio de alto rendimiento), Python se ejecuta muy rápidamente. Este bucle anidado le dice a Python que se mantenga comprobando el estado del Pin 12 hasta que ya no sea *bajo*, en cuyo caso sabrá que el botón ha sido liberado. Sin este bucle, el programa se repetiría mientras el botón esté siendo presionado (y no importa cuán rápidos sean sus reflejos, verá en la pantalla imprimirse el mensaje varias veces, lo cual no es lo correcto).

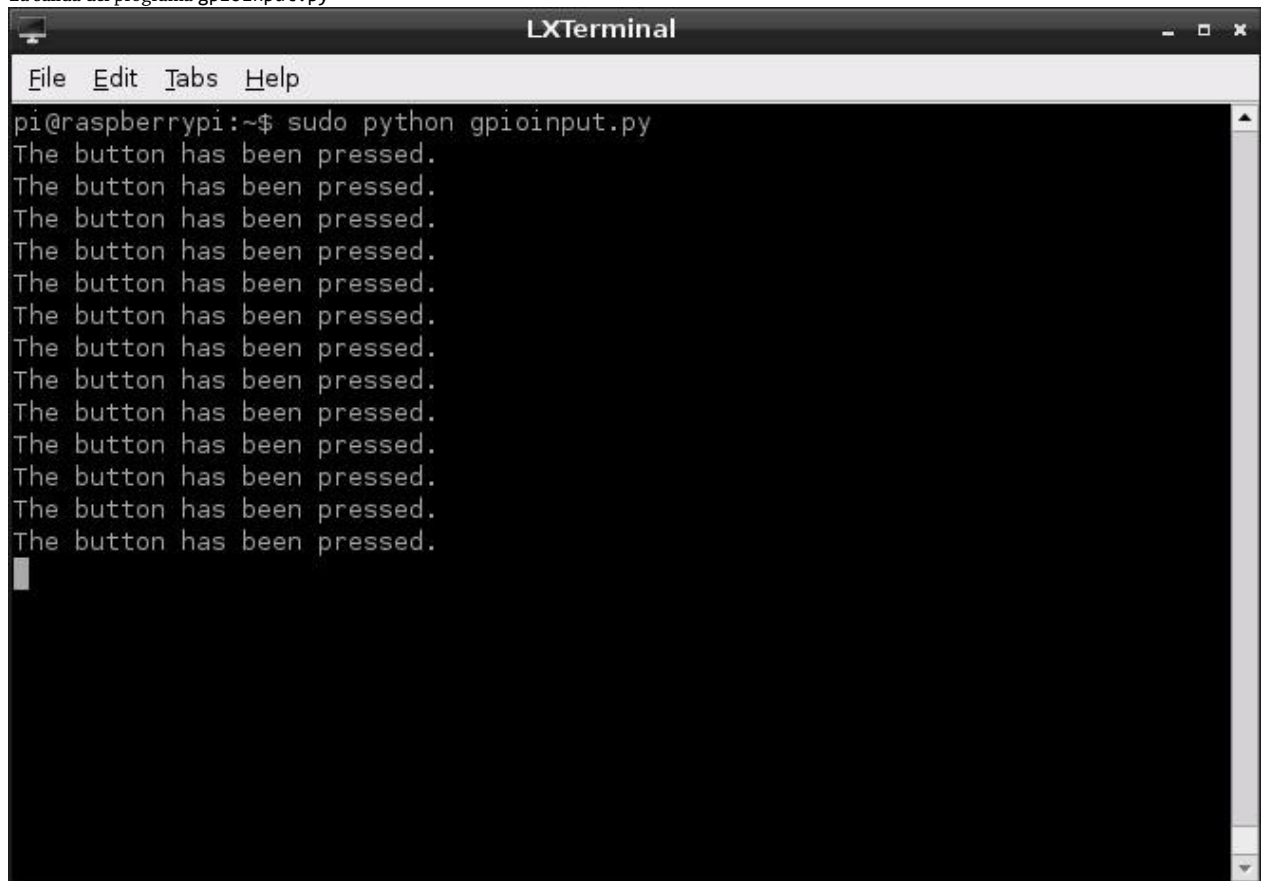
El programa final debe verse como el siguiente:

```
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BOARD)
GPIO.setup(12, GPIO.IN)
while True:
    input_value = GPIO.input(12)
    if input_value == False:
        print("The button has been pressed.")
        while input_value == False:
            input_value = GPIO.input(12)
```

Guarde el archivo como `gpioinput.py` y luego ejecútelo desde la terminal con `sudo python gpioinput.py`. Al comenzar el programa, nada se muestra sobre la pantalla, pero si presiona el botón, el programa empezará a imprimir en la terminal el mensaje de la línea número siete de nuestro programa (ver Figura 14-6). Suelte el botón y vuelva a presionarlo, y el mensaje se repetirá.



Figura 14-6

La salida del programa `gpioinput.py`

```
pi@raspberrypi:~$ sudo python gpioinput.py
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
The button has been pressed.
```

Del mismo modo que el ejemplo pasado de la entrada GPIO, este ejemplo es aparentemente un simple programa que puede ser utilizado para muchos propósitos. Además de ser capaz de leer cuando un botón es presionado, el mismo código se puede utilizar para leer cuándo los pines de un dispositivo aparte (como puede ser un sensor o microcontrolador externo) han sido cambiados al estado *alto* (*high*) o *bajo* (*low*).

Al extender el código para que utilice más botones (cada uno sobre un pin GPIO individual), incluso podría ser capaz de crear un sencillo control de cuatro botones para videojuegos. Por ejemplo, puede combinar el código anterior con el juego de la Serpiente del capítulo 12, “Una Introducción a Python”, para convertir a la Raspberry Pi en una sencilla consola de videojuegos. También podría combinar ambos ejemplos de entrada y salida en un solo programa, que esperaría a que un botón sea presionado para luego iluminar el LED a través del envío del estado *alto* al pin de salida. Para asegurarse de que ha entendido los conceptos en esta sección, intente crear ese programa combinado ahora. Si se queda atorado, o desea comparar su método, vaya al Apéndice A, “Python Recetas”, para ver una solución de muestra.

# Capítulo 15

El Módulo de Cámara de la Raspberry Pi

**Diseñado por los ingenieros** que trabajan para la Fundación Raspberry Pi, el Módulo de Cámara de la Raspberry Pi es la forma más compacta de añadir la capacidad de grabar imágenes estáticas y video a su proyecto. Diseñado para conectarse al conector *CSI* (*Camera Serial Interface*) de la Raspberry Pi, el Módulo de Cámara sólo mide 25 mm sobre su borde más largo y pesa sólo 3 gramos (ver Figura 15-1).

**Figura 15-1**  
El Módulo de Cámara de la Raspberry Pi, sin su cable cinta en la fotografía



Desde su lanzamiento, el Módulo de Cámara ha encontrado un lugar entre una variedad de proyectos que van desde un sistema casero de vigilancia hasta los experimentos más complejos de visión computarizada que rastrean los gestos del rostro y manos del usuario. El módulo incluso ha alcanzado el espacio cercano, como resultado de proyectos fotográficos en los que atan una Raspberry Pi a un globo meteorológico y lo liberan para tomar imágenes a gran altitud.

El módulo de cámara está construido a base de un sensor de cinco megapíxeles de un tipo comúnmente utilizado en los teléfonos inteligentes, ubicado detrás de una lente de foco fijo. El módulo funciona en conjunto con el procesador gráfico de la Raspberry Pi para garantizar que el video de alta resolución y las imágenes fijas puedan ser capturadas suavemente sin sobrecargar el procesador principal de la Raspberry Pi o requerir demasiada memoria.

## ¿Por qué utilizar el Módulo de Cámara?

Si no tiene necesidad de capturar imágenes o video, no necesita el Módulo de Cámara. Es un componente opcional y la RasPi trabaja igual de bien sin ella. También es posible añadir visión a su RasPi utilizando otros productos alternativos, como cámaras web, que se conectan a un puerto libre USB.

Sin embargo, la RasPi tiene sólo un número limitado de puertos y éstos por lo general son utilizados para funciones más críticas, tales como un teclado, ratón o adaptador de red inalámbrico. Esto es particularmente importante en la Raspberry Pi Modelo A, la cual sólo tiene un puerto USB disponible para su uso.

El Módulo de Cámara tiene otras ventajas sobre una cámara web tradicional: ésta consume considerablemente menos energía, lo que significa que no ejerce presión sobre la fuente de alimentación de la Raspberry Pi o no agota

las baterías en un proyecto portátil o de energía solar; el módulo ofrece la captura de imágenes de hasta cinco megapíxeles, y captura de video con una resolución Full HD y 30 fotogramas por segundo; y es considerablemente más pequeño que las cámaras de conexión USB.

El Módulo de Cámara es compatible con todos los modelos y revisiones de la Raspberry Pi, de manera que si todavía no sabe si requerirá de una cámara en su proyecto, no se preocupe; consiga primero cualquier modelo de RasPi y más adelante agregue el Módulo de Cámara.

## Instalación del Módulo de Cámara

El Módulo de Cámara de la Raspberry Pi (como la misma Raspberry Pi) se presenta como una placa de circuito desnuda. Aunque esto es razonablemente robusto, debe tener cuidado cuando lo manipule para no dañar alguno de sus componentes, particularmente el lente de plástico ubicado sobre el propio sensor de la cámara.

El Módulo de Cámara se conecta a la RasPi a través de un *cable de cinta*, un cable delgado, semirrígido que debe ser insertado en el conector CSI de la Raspberry Pi. Cuando recibe la cámara, un extremo del cable ya se encuentra insertado en el propio módulo; el otro extremo, por un lado está pintado en color azul y por el otro lado muestra unos contactos de plata. Estos contactos se conectan a los pines del conector CSI de la RasPi, transfiriendo la energía y los datos entre la RasPi y el Módulo de Cámara.

El conector CSI está etiquetado con S5 y está situado a mano derecha de la placa, cerca de donde los puertos USB se encuentran (ver Figura 15-2). El conector CSI de una Raspberry Pi Modelo B está situado a la izquierda del puerto Ethernet; en una Modelo A, se encuentra en el mismo lugar, justo a la izquierda del hueco en la placa por debajo de los puertos USB. Algunos modelos pueden tener el conector CSI cubierto por una pieza protectora de película plástica; así que debe desprender esta película antes de tratar de instalar la cámara.

Hay un conector de aspecto similar situado al lado izquierda de la Raspberry Pi. Este es el conector *DSI (Display Serial Interface)* y está diseñado para conectar la Raspberry Pi a las pantallas de cristal líquido. Los dos puertos no son intercambiables; si conecta la Cámara al puerto DSI en lugar del puerto CSI, no va a funcionar.

Antes de insertar el cable, pequeños *agarraderas* en ambos lados del conector deben levantarse suavemente. Esto puede hacerse con las uñas sin necesidad de ninguna herramienta, pero tenga cuidado: deben levantarse con facilidad y dejar de levantarlas una vez que se encuentran a unos milímetros por encima del resto del conector CSI (ver Figura 15-3).

Figura 15-2

El conector CSI de la Raspberry Pi, fotografiado en un Modelo A

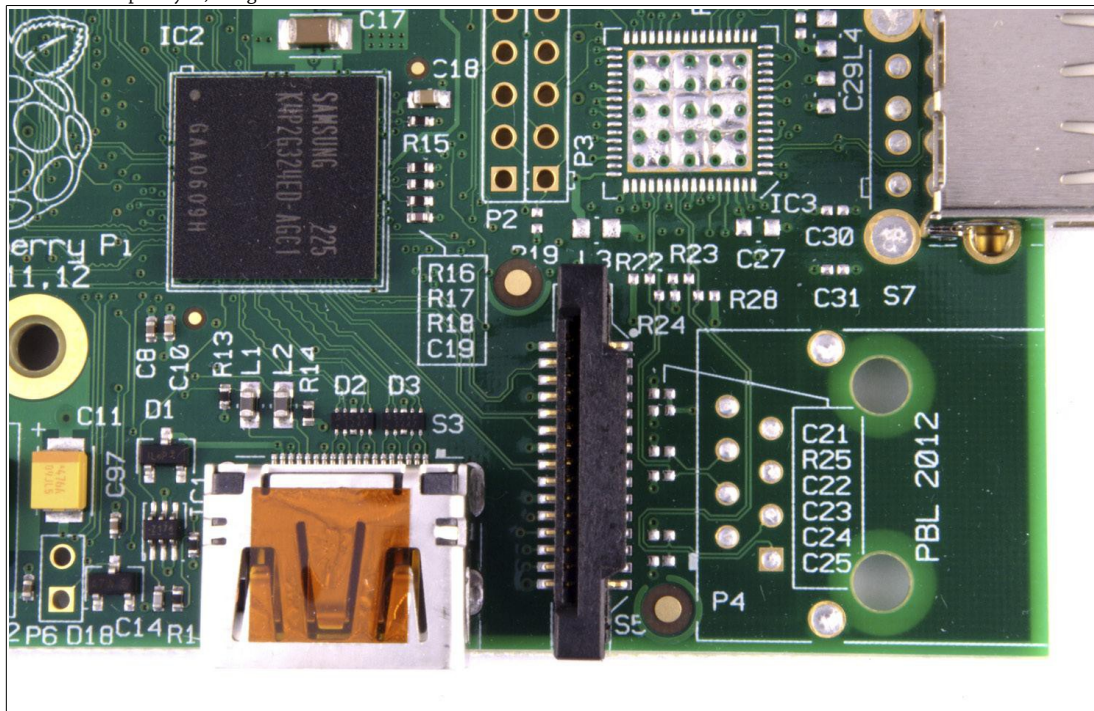
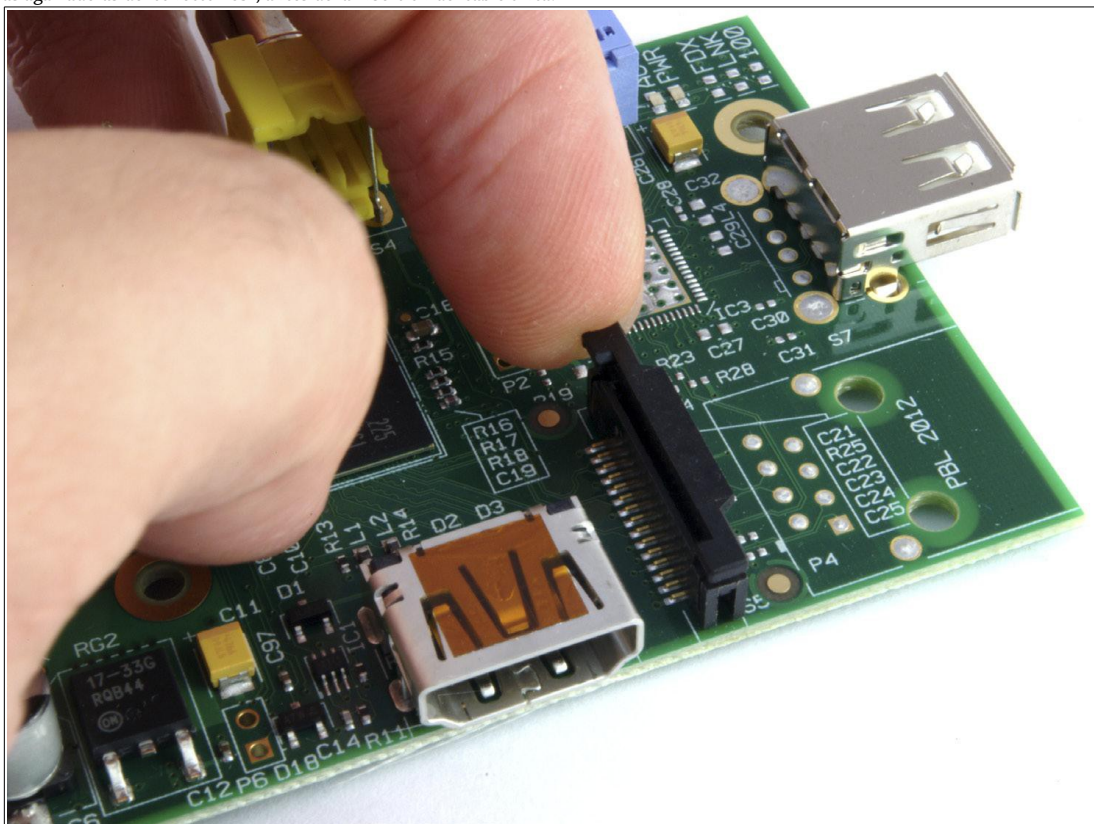


Figura 15-3

Alzando las agarraderas del conector CSI, antes de la inserción del cable cinta.

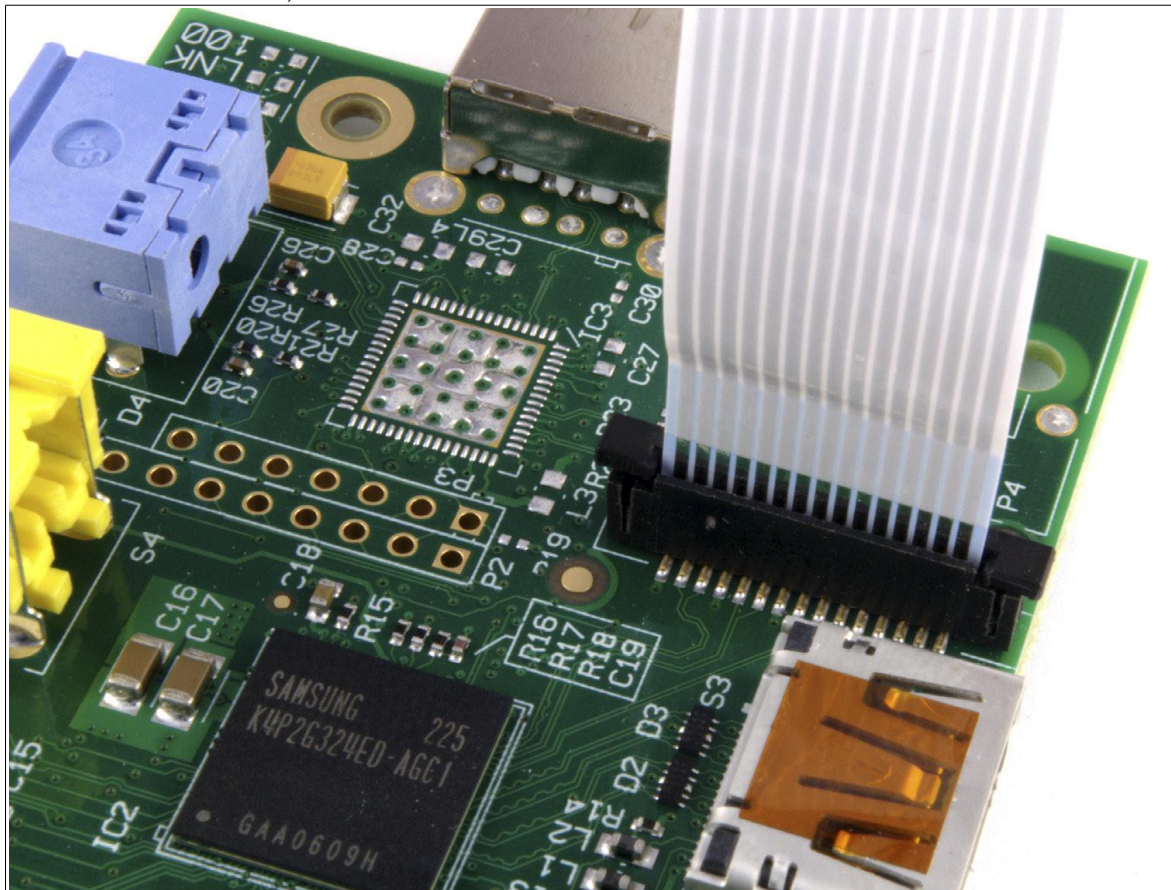




Inserte el extremo libre del cable de cinta en el puerto CSI, asegúrese de que los contactos plateados estén de cara hacia la izquierda de la RasPi y la parte azul del cable a la derecha. Empuje suavemente en su lugar, teniendo cuidado de no torcerlo, y luego empuje las agarraderas hacia abajo para asegurar el cable en su lugar (ver Figura 15-4). Cuando está insertada de forma correcta, el cable sale del conector derecho y debe resistir un suave tirón; sin embargo, cuando haga esta prueba no tire demasiado fuerte ya que puede dañar el cable.

Figura 15-4

El cable de cinta del Módulo de Cámara, insertado correctamente en el conector CSI



#### ADVERTENCIA

El cable de cinta incluido con el Módulo de Cámara de la Raspberry Pi es razonablemente robusto y probará ser confiable incluso para el uso diario. Sin embargo, es susceptible de dañarse si se dobla. Al insertar el cable o mover la cámara, asegúrese de no doblar el cable de cinta sobre sí mismo. Si daña su cable de cinta, pueden encontrar reemplazos en la mayoría de tiendas que comercializan la Raspberrys.

Cuando utilice el Módulo de Cámara con una Raspberry Pi con carcasa, usualmente será necesario pasar el cable de cinta a través de una ranura delgada o entre las capas de plástico. Algunas carcasas no son plenamente compatibles con el Módulo de Cámara; en estos casos, es posible que necesite dejar la tapa de la carcasa fuera con el fin de dar salida al cable de cinta. Posicione el Módulo de Cámara con su cable de cinta saliendo por la parte inferior; si esto no es posible, existe una opción de software para voltear la imagen capturada de arriba hacia abajo como se describe más adelante en este capítulo.

El paso final en la instalación física de la cámara es quitar la pequeña película de plástico protectora que se encuentra sobre la lente. Este plástico simplemente lo puede desprender y desechar jalando suavemente hacia arriba la pestaña que se extiende fuera de la lente. Aunque es tentador dejar la película en su lugar para proteger la lente, esto provocará que todo lo que capture aparezca con una tonalidad azul desagradable.

## Activación del Modo de Cámara

El software que controla el Módulo de Cámara de la Raspberry Pi está incluido en la distribución Raspbian por defecto. Si está utilizando una versión muy vieja, puede darse cuenta que los archivos faltan; si es así, actualice su sistema utilizando el siguiente comando en la terminal o consola (ver Capítulo 3, “Administración del Sistema Linux” para más detalles):

```
sudo apt-get update && sudo apt-get upgrade
```

Sin embargo, necesitará cambiar algunos ajustes del sistema para hacer que el Módulo de Cámara funcione correctamente. Particularmente la grabación de video requiere que la parte del procesamiento gráfico del BCM2835 de la RasPi cuente con al menos 128MB de memoria disponible; con algo menos que esto, la captura de imagen fija trabajará, pero la grabación de video fallará. Para conocer los detalles sobre cómo cambiar la división de la memoria, vea el Capítulo 6, “La Herramienta de Configuración del Software de la Raspberry Pi”.

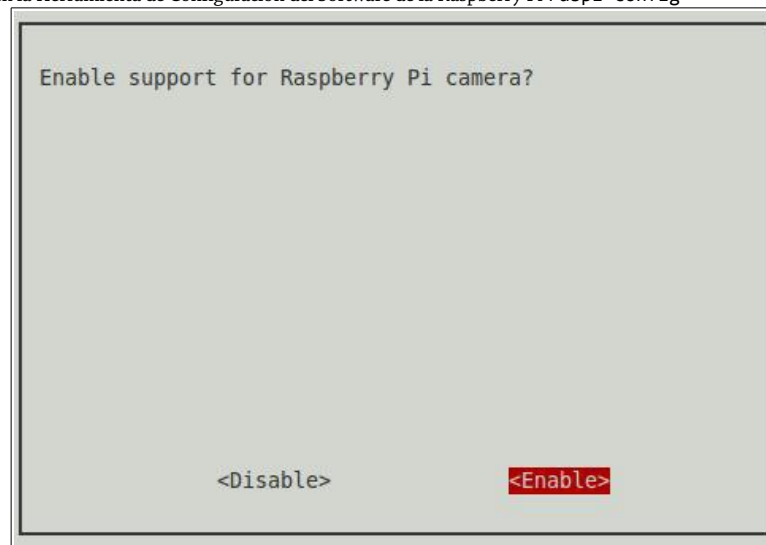
La forma más fácil de asegurarse de que su RasPi se encuentra lista para la cámara es utilizando la Herramienta de Configuración del Software de la Raspberry Pi: **raspi-config**. En la terminal, escriba el siguiente comando para cargar la herramienta:

```
sudo raspi-config
```

En el menú que aparece, seleccione la opción 5 (Enable Camera) utilizando las teclas de cursor y la tecla Intro. Elija Enable en la pantalla que aparece y presione Intro nuevamente (ver Figura 15-5). Si el Modo Cámara fue desactivado anteriormente, se le pedirá que reinicie la RasPi; confirme esto presionando Intro.

**Figura 15-5**

Activando el Modo Cámara en la Herramienta de Configuración del Software de la Raspberry Pi **raspi-config**



---

### CONSEJO

Si no puede encontrar una opción de cámara en **raspi-config**, puede que esté ejecutando una versión obsoleta de la aplicación. Utilice el comando `sudo apt-get update && sudo apt-get upgrade` para actualizar su software, luego reinicie la RasPi e intente ejecutar de nuevo **raspi-config**.

---

Una vez que la RasPi se haya reiniciado, su Módulo de Cámara estará listo para utilizarse.

Un último paso opcional, si le gustaría ver las imágenes capturadas desde la consola Raspbian sin tener que cargar la interfaz gráfica de usuario, instale el *visor de imágenes en framebuffer* **fbi**. Para hacer esto, escriba el siguiente comando:

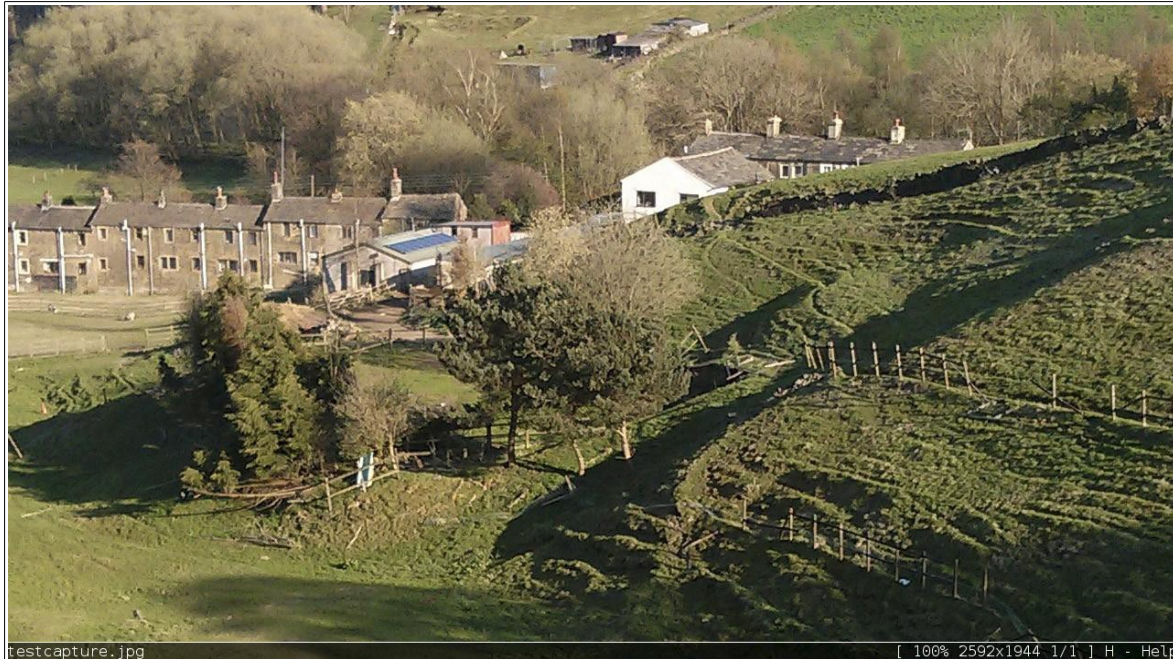
```
sudo apt-get install fbi
```

Para visualizar las imágenes utilizando esta herramienta (ver Figura 15-6) simplemente escriba el comando seguido por el nombre de la imagen:

```
fbi nombreimagen.jpg
```

**Figura 15-6**

Visualizando las imágenes capturadas mediante el paquete fbi



Para salir de **fbi**, presione la tecla Q o Escape.

## Captura de Imágenes Fijas

La mejor forma de probar que la cámara está funcionando correctamente es tratando de capturar una imagen. El Módulo de Cámara utiliza un paquete de software personalizado llamado **raspistill** para capturar imágenes fijas que son almacenadas, por defecto, utilizando el formato de archivo *JPEG* (*Joint Photographers Expert Group*), el mismo formato que utilizan la mayoría de las cámaras digitales y smartphones.

El software para la cámara de la Raspberry Pi se ejecuta desde la terminal o consola. Para obtener mejores resultados, ejecute el software en la consola sin utilizar el comando **startx** para cargar la interfaz gráfica de usuario (ver Capítulo 3, “Administración del Sistema Linux” para más detalles).

El comando **raspistill** acepta un número de argumentos opcionales que controlan ajustes tales como, la resolución vertical y horizontal de la imagen capturada, el modo de orientación de la cámara, el tipo de archivo guardado y el nivel de compresión que será aplicado a la imagen final. Si estas opciones no son indicadas al momento de invocar el comando, se utilizarán los valores predeterminados.

Para probar la cámara, simplemente ejecute la aplicación **raspistill** desde la consola escribiendo su nombre:

```
raspistill -o pruebacaptura.jpg
```



Esto mostrará una vista previa en vivo de cinco segundos (ver Figura 15-7), tiempo durante el cual una luz roja en la parte frontal de la cámara se iluminará. Este es un indicador de actividad, diseñado para proporcionar la confirmación de que la cámara está funcionando. Si mueve sus dedos frente a la cámara durante este periodo de cinco segundos, los verá aparecer en la ventana de vista previa sobre la pantalla. Cuando los cinco segundos hayan transcurrido, la luz se apagará y la ventana de vista previa desaparecerá.

Figura 15-7

La aplicación de captura `raspistill` mostrando una imagen de vista previa en vivo



---

#### CONSEJO

A veces la posición del cable del Módulo de Cámara hace que sea difícil colocar la cámara en la posición correcta. Si su vista previa aparece al revés, utilice la opción `-vf` (vertical flip/volteo vertical) para corregir la imagen. Sólo agregue la opción a la línea de comandos junto con cualquiera de las otras opciones que esté utilizando.

---

Cuando la vista previa haya finalizado, la opción `-o` (output) guardará un archivo JPEG con el nombre `pruebacaptura.jpg`. Cuando se utiliza la opción `-o`, puede cambiar el tipo de formato del archivo guardado con `-e`. Los tipos de formato de archivo soportados actualmente incluyen el *BMP* (*bitmap*), *PNG* (*Portable Network Graphics*) y el *GIF* (*Graphic Interchange Format*). Para guardar un archivo PNG, por ejemplo, utilice el siguiente comando:

```
raspistill -o pruebacaptura.png -e png
```

Modificar `png` para que se lea `bmp`, `gif` o `jpg` permiten acceder a los otros formatos de archivo soportados. Si olvida colocar la opción `-e`, un archivo será guardado pero independientemente de su extensión el contenido de los datos será JPEG.

Otro par de opciones frecuentemente utilizadas son las que tienen el cometido de ajustar el ancho y el alto de la imagen capturada. Esto resulta bueno para capturar imágenes en baja resolución como parte de un proyecto de visión computarizada o simplemente para ahorrar espacio en disco.

El ancho de la imagen capturada se ajusta con la opción `-w`, mientras que la altura se ajusta con `-h`. Estas opciones con frecuencia son utilizadas juntas, para establecer una resolución global de la imagen. Para capturar una imagen de 1,920 píxeles de ancho por 1,080 píxeles de alto (la misma resolución que tiene una televisión Full HD de alta definición o película Blu-ray) escriba el siguiente comando:

```
raspistill -w 120 -h 1080 -o fullhdcaptura.jpg
```

La última opción básica a conocer es **-t**, la cual controla el tiempo de espera de la vista previa. Por defecto, **raspistill** muestra una vista previa en vivo de cinco segundos antes de capturar la imagen; la opción **-t** hace caso omiso de esta configuración predeterminada. El retardo requerido debe especificarse en milisegundos. Para capturar una imagen después de un retraso de 10 segundos, utilice el siguiente comando:

```
raspistill -t 10000 -o diezsegundoscaptura.jpg
```

Para deshabilitar el retraso y capturar la imagen inmediatamente utilice un valor de 0 como se muestra en el siguiente comando:

```
raspistill -t 0 -o capturainstantanea.jpg
```

El nombre del archivo después de la opción de salida (**-o**) puede ser cualquiera que usted desee, en los comandos anteriores los nombres de los archivos son simplemente de ejemplo. Cuando escriba el nombre del archivo, siempre asegúrese de añadir la extensión apropiada para su tipo de archivo: **.jpg** para imágenes JPEG, **.png** para Portable Network Graphics, **.bmp** para imágenes bitmap y **.gif** para Graphics Interchange Format.

Para una completa descripción de todas las opciones disponibles para **raspistill**, vaya al Apéndice B, “Rápida Referencia del Módulo de Cámara de la Raspberry Pi”. Como alternativa, escriba el siguiente comando para ver una lista de opciones:

```
raspistill --help
```

## Grabación de Video

Así como hay una aplicación dedicada a la captura de imágenes fijas mediante el Módulo de Cámara de la Raspberry Pi, hay otra para la captura de video: **raspivid**. El uso de **raspivid** es muy similar al de **raspistill**, pero existen algunas diferencias que necesitará conocer antes de seguir adelante.

La diferencia más importante entre **raspivid** y **raspistill** es la opción **-t**: en **raspistill**, la opción **-t** contiene el tiempo de espera para la vista previa antes de que la imagen sea capturada; en **raspivid**, la opción **-t** proporciona un límite para la duración total del video que será grabado. Si se le proporciona un valor de 0, la cámara grabará por siempre, sin un límite de tiempo (lo que llenará rápidamente su tarjeta SD o dispositivo de almacenamiento externo).

Al igual que **raspistill**, el valor predeterminado para la opción **-t** es de cinco segundos. Por lo tanto, para comprobar que la cámara está funcionando y graba un video corto, sólo tiene que escribir el nombre del comando con la opción **-o** (output) y un nombre de archivo:

```
raspivid -o defaultvideo.h264
```

Este comando graba un video de cinco segundos y lo guarda en un formato conocido como **h.264**. A diferencia de **raspistill**, no existe una opción para guardar el archivo en un formato diferente: **raspivid** utiliza la aceleración por hardware disponible en el procesador BCM2835 de la RasPi para llevar a cabo la grabación de video y sólo admite **h.264** como formato de captura.

Si reproduce el video grabado, se dará cuenta que no tiene sonido. Desafortunadamente, la Raspberry Pi no incluye una entrada de micrófono. Aunque es posible grabar audio conectando ciertos micrófonos o tarjetas de sonido USB, será necesario capturar este audio a través de un software por separado y combinarlo después con el archivo de video utilizando un paquete de edición de video.

Las otras opciones soportadas por `raspistill` también lo son por `raspivd`. Por ejemplo, para modificar el ancho y el alto de la grabación de video a 1,920x1,080 (Full HD) utilice las opciones `-w` y `-h` como en el siguiente comando:

```
raspivd -w 1920 -h 1080 -o fullhdvideo.h264
```

Para grabar un vídeo más largo, ajuste la opción `-t` especificando la duración del video en milisegundos. Para grabar un minuto de video, escriba el siguiente comando:

```
raspivd -t 6000 -o videominuto.h264
```

---

**CONSEJO**

Aunque h.264 es un formato de video razonablemente eficiente, grabar a altas resoluciones puede consumir un espacio considerable del disco. Si está grabando videos de larga duración, asegúrese de que cuenta con espacio suficiente en su tarjeta SD o considere conectar un dispositivo de almacenamiento USB, como puede ser un disco duro externo.

---

Puede leer más opciones de `raspivd` en el Apéndice B, “Rápida Referencia del Módulo de Cámara de la Raspberry Pi” o vea la lista de opciones disponibles en el siguiente comando:

```
raspivd --help
```

## Fotografía de Lapso de Tiempo desde la Línea de Comandos

Hasta el momento, sólo ha aprendido lo básico sobre cómo utilizar el Módulo de Cámara de la Raspberry Pi, pero ahora es tiempo de encontrar un uso práctico para el módulo. La principal ventaja del Módulo de Cámara de la Raspberry Pi frente a las demás cámaras es que cuenta con la capacidad de ser programada fácilmente para diversas tareas. En este ejemplo, convertirá su Raspberry Pi y el Módulo de Cámara en un sistema fotográfico de lapso de tiempo.

El programa `raspistill` viene con una opción para habilitar un modo de lapso de tiempo, `-tl`, en tal modo el Módulo de Cámara captura de forma continua fotografías de acuerdo a un retardo de tiempo en milisegundos. Esta característica está diseñada para cuando `raspistill` sea invocado a través de un programa externo (como un servidor web o streamer de video) y sobrescribirá el archivo de salida cada vez que una nueva fotografía es capturada.

Para utilizar `raspistill` en la fotografía de lapso de tiempo tradicional, donde un nuevo archivo es creado a cada momento, necesitará disponer de una forma de controlar `raspistill` de manera externa. Esto podría hacerse a través de un programa Python, una aplicación dedicada o incluso con un thread del servidor web, pero la forma más fácil es utilizando un proceso conocido como shell scripting.

Un shell script es simplemente una colección de comandos, escritos exactamente en la misma forma como lo haría si estuviera trabajando en la consola o terminal. Cada comando se ejecuta en orden, de arriba hacia abajo, pero también es posible utilizar ciclos como en los lenguajes de programación como Scratch y Python (ver Capítulo 11, “Una Introducción a Scratch” y el Capítulo 12, “Una Introducción a Python” para más detalles). Aunada a la capacidad de shell de poder encadenar varios programas, el shell scripting es una herramienta poderosa.

Para iniciar un shell script, simplemente cree un nuevo archivo de texto en su directorio actual con el siguiente comando:

```
nano timelapse.sh
```

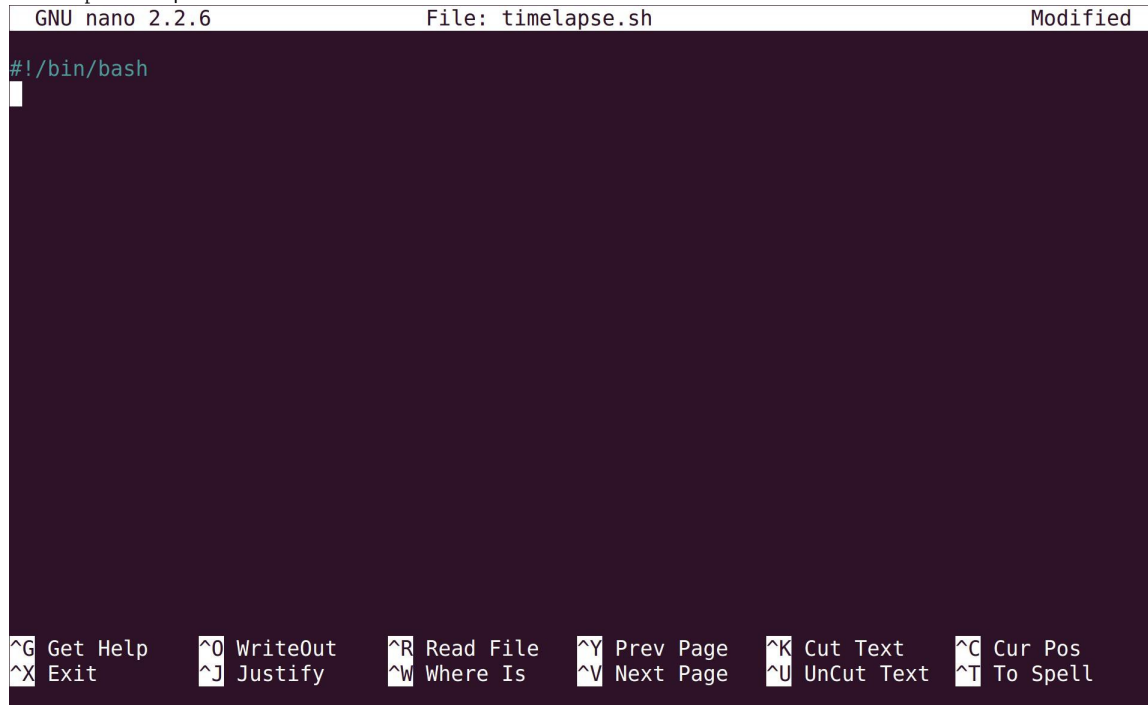
La extensión `.sh` es tradicionalmente utilizada para los shell scripts, aunque no es estrictamente necesaria. Utilizar una extensión de archivo como ésta le ayuda a diferenciar entre un shell script y un archivo ejecutable, por lo que sin duda se recomienda.

En la parte superior del archivo (ver Figura 15-8) introduzca una línea que le dice al sistema operativo que intérprete el comando utilizando el shell Bash por defecto:

```
#!/bin/bash
```

Figura 15-8

Editando el script `timelapse.sh` en el editor nano



Aunque existen otros shells disponibles, Bash es el predeterminado en la mayoría de las modernas distribuciones Linux y ofrece una buena mezcla entre compatibilidad garantizada y características avanzadas. Todo lo que le sigue a la línea anterior será ejecutado por el sistema operativo como un comando tal como si lo hubiera escrito en la misma terminal. La excepción son las *líneas de comentario*, que están precedidas por el símbolo `#`; estas líneas contienen notas que le ayudan a recordar cómo utilizar el script o por qué lo ha escrito en una manera particular. Es una buena práctica incluir líneas de comentarios en su shell script, aunque el script se ejecutará igual de bien sin ellas.

De esta forma usted tiene manera de conocer qué es lo que se supone que hace el script cuando encuentre su archivo algunos meses más tarde, coloque en la siguiente línea un comentario explicando el propósito del script:

```
# Captura de imágenes en lapsos de tiempo utilizando la aplicación raspistill
```

Recuerde colocar el símbolo `#` al comienzo de la línea o el shell intentará ejecutar el comentario como un comando.

Lo siguiente es crear un bucle que nos garantizará que el script continuará su ejecución después de haber tomado la primera fotografía. Introduzca las siguientes dos líneas para iniciar el bucle:

```
while true
do
```

Note que la segunda línea, `do`, tiene sangría de cuatro espacios al inicio; esto ayuda a ilustrar visualmente que el código que aparece abajo es parte del bucle. A diferencia de Python, el cual necesita obligatoriamente la sangría, los comandos en los shell scripts pueden escribirse sin la necesidad de sangrías (pero esto hará que nuestro código sea más difícil de leer y entender). Al igual que los comentarios, la sangría es un paso opcional pero muy recomendado.

A continuación, establezca el nombre del archivo que será guardado introduciendo la siguiente línea:

```
filename=`date +%Y%m%dT%H%M%S`.jpg
```

Esta instrucción se ve algo compleja, se encarga de decirle a shell que tome la salida del comando `date` y lo deposite en una variable llamada `filename`. El símbolo ```, conocido como *acento grave* o *invertido*, le dice a shell que ejecute el comando `date`; sin estos símbolos que se encuentran a cada extremo del comando, el shell ignoraría la instrucción y la trataría como una simple cadena de texto.

El comando `date`, como lo habrá adivinado, le permite averiguar qué fecha y hora actual tiene el sistema. En seguida del símbolo `+` se encuentran una serie de instrucciones que indican cómo la fecha debería ser impresa: `%Y` proporciona el año en formato de cuatro dígitos, `%m` el mes en formato de dos dígitos, `%d` el día en formato de dos dígitos, y `%H`, `%M` y `%S` proporcionan el tiempo en horas, minutos y segundos en formato de 24 horas.

Después de la *comilla invertida* o *acento fuerte* está `.jpg`. Esto completa el nombre del archivo, diciéndole al shell script que guarde el archivo de la imagen en el formato de fecha internacional con la extensión de un archivo JPEG. Por ejemplo, Si captura una imagen exactamente a las dos de tarde del 17 de enero de 2015, el nombre del archivo será `20150117140000.jpg`. Debido a que le toma menos de un segundo a la cámara capturar una fotografía, habrá que hacer una pausa de tiempo más adelante en el shell script para garantizar que cada imagen tenga un nombre de archivo único y nada sea sobrescrito.

Utilizar el formato de fecha internacional tiene una ventaja adicional: si ordena la lista numérica o alfabéticamente, sus fotografías serán ordenadas partiendo de la más antigua a la más nueva. Sin embargo, si prefiere un formato de nombre de archivo diferente, simplemente cambie el orden en el comando.

---

#### ADVERTENCIA

El comando `date` obtiene la información del tiempo del reloj del sistema. Cuando está conectado a una red, la Raspberry Pi utiliza el *NTP (Network Time Protocol)* para encontrar el tiempo actual, pero el Modelo A no puede hacer esto a menos que conecte un adaptador de red USB. En este caso, la fecha en el nombre del archivo estará incorrecta a menos que establezca manualmente la fecha antes de ejecutar el script.

---

Establecido el nombre del archivo, es momento de decirle al script que ejecute `raspistill`. Puede utilizar cualquiera de las opciones que guste aquí, mientras recuerde establecer el nombre del archivo de salida. Para capturar una imagen Full HD, introduzca la siguiente línea:

```
raspistill -w 1920 -h 1080 -t 0 -o $filename
```

El símbolo `$` antes del nombre del archivo le dice a shell que está haciendo referencia a la variable `filename` establecida en la línea anterior. Es importante incluir este símbolo, ya que sin aquel el script escribirá un único archivo llamado `filename`, que se sobrescribirá cada vez que una nueva imagen sea capturada.

Siempre es una buena idea obtener retroalimentación de sus scripts, para conocer cuándo se están ejecutando. Una forma de hacer esto es haciendo que el script imprima mensajes de estado en la terminal o consola, emulando el comando `echo`. Introduzca la siguiente línea como sigue:

```
echo Image captured
```

En seguida, agregue un pausa de tiempo al script mediante el comando `sleep`. Dicho comando necesita que le

especifiquemos un valor en segundos y no en milisegundos como en el caso de `raspistill` y `raspidvid`. Para tomar una foto cada 30 segundos, introduzca la siguiente línea:

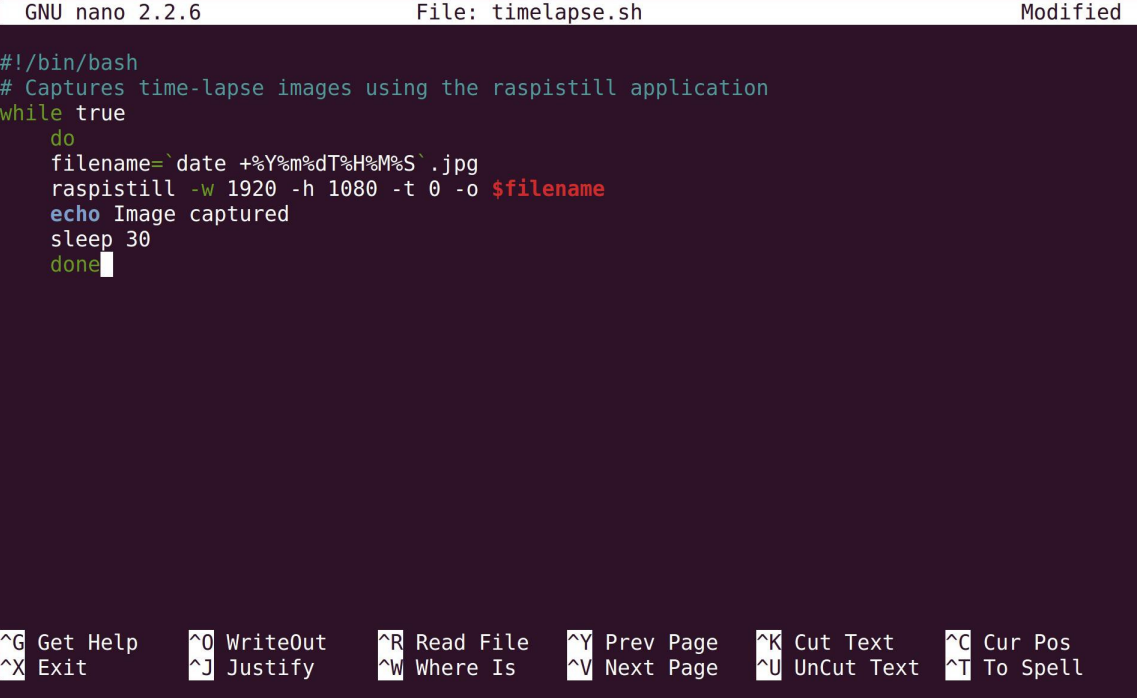
```
sleep 30
```

Por último, cierre el bucle introduciendo la última línea del script (ver Figura 15-9):

```
done
```

Figura 15-9

El script `timelapse.sh` terminado en el editor de texto nano

A screenshot of the nano text editor interface. The title bar at the top shows "GNU nano 2.2.6", "File: timelapse.sh", and "Modified". The editor area has a dark purple background with light-colored text. The script content is as follows:

```
#!/bin/bash
# Captures time-lapse images using the raspistill application
while true
do
    filename=`date +%Y%m%dT%H%M%S`.jpg
    raspistill -w 1920 -h 1080 -t 0 -o $filename
    echo Image captured
    sleep 30
done
```

The cursor is at the end of the `done` line. At the bottom of the screen, there is a status bar with various keyboard shortcuts: `^G` Get Help, `^O` WriteOut, `^R` Read File, `^Y` Prev Page, `^K` Cut Text, `^C` Cur Pos, `^X` Exit, `^J` Justify, `^W` Where Is, `^V` Next Page, `^U` UnCut Text, and `^T` To Spell.

Aunque esto lleve al script a su final, el bucle que se ha utilizado es conocido como un *bucle infinito*: la instrucción `while true` al comienzo le dice al script que mantenga la ejecución del bucle hasta que `true` sea *false*, o 1 sea 0. Dado que esto nunca sucede, el script se ejecuta por siempre (capturando una imagen cada 30 segundos).

Guarde el archivo presionando `Ctrl + O`, y luego cambie los permisos del archivo para hacerlo *ejecutable* con el siguiente comando:

```
chmod +x timelapse.sh
```

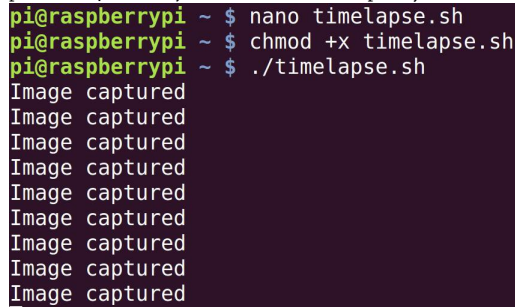
Para más información sobre los permisos de archivos, vea el Capítulo 3, “Administración del Sistema Linux”. Finalmente, ejecute el script con el siguiente comando:

```
./timelapse.sh
```

El script inmediatamente captura una foto e imprime el mensaje de estado “Image captured” a la consola (ver Figura 15-10). Después de 30 segundos, capturará otra foto; después otros 30 segundos de espera, otra tercera foto. Esto continuará por siempre o hasta que su tarjeta SD se llene de imágenes. Para salir del script, presione `Ctrl + C`.

Figura 15-10

El script `timelapse.sh` ejecutándose en una Raspberry Pi Modelo B

A screenshot of a terminal window with a dark purple background. The text is white and green. It shows a user at the 'pi@raspberrypi' prompt running three commands: 'nano timelapse.sh', 'chmod +x timelapse.sh', and './timelapse.sh'. After the third command, the terminal displays ten lines of 'Image captured'.

La fotografía de lapso de tiempo es a menudo utilizada para comprimir y convertir una tarea larga o compleja en un entretenido video. Los equipos comerciales de fotografía de lapso de tiempo (que son extremadamente costosos), son a menudo utilizados para filmar el crecimiento de las plantas, la construcción de edificios o la elaboración de las de carreteras. Una Raspberry Pi ejecutando este script puede hacer todas estas cosas, pero con una fracción del costo de aquellos equipos.

Las imágenes que ha capturado puede ser transformadas en un video, ya sea utilizando un software de edición de video o la herramienta `ffmpeg`. Si bien `ffmpeg` puede ejecutarse en la propia Raspberry Pi, es un programa que consume muchos recursos y le tomará un tiempo considerable ejecutarse si cuenta con un gran número de imágenes en alta resolución a convertir. Si tiene la paciencia para hacer la conversión desde la misma RasPi, instale `ffmpeg` con el siguiente comando:

```
sudo apt-get install ffmpeg
```

Luego, convierta las imágenes escribiendo:

```
ffmpeg -r 1 -i *.jpg timelapse.mp4
```

El comando anterior tomará todas las imágenes JPEG encontradas en el directorio actual y creará un video mostrando imágenes a una tasa de velocidad de una por segundo (o 30 veces más rápido que en tiempo real). Para acelerar el video todavía más, puede ajustar la opción `-r`: `-r 10` mostrará diez imágenes por segundo, `-r 20` veinte imágenes por segundo y así sucesivamente. El video final puede ser compartido directamente, o subirse a un sitio de video streaming como YouTube o Vimeo.

# **Capítulo 16**

## Placas de Expansión



**Aunque puede utilizar** los conectores GPIO de la Raspberry Pi directamente, como aprendió en el Capítulo 13, “Aprendiendo a Hackear Hardware”, una mejor idea es utilizar una placa especializada en prototipado. Las placas de prototipado son placas de expansión diseñadas para tomar asiento entre su proyecto y la Raspberry Pi y las hay en variedades que van desde las relativamente sencillas (como la Slice of Pi de Ciseco) hasta las complejas y poderosas (como la Gertboard).

En su expresión más básica, las placas de expansión simplemente proporcionan un acceso más fácil a los pines GPIO de la Raspberry Pi. Esto puede significar que pueden ser conectadas a una breadboard más fácilmente, o que cuentan con más espacio y están etiquetadas para hacer que la conexión a otros dispositivos sea más simple. Algunas placas incluyen circuitos para conectar dispositivos específicos adicionales (como un pequeño transmisor inalámbrico XBee), y otros proporcionan una pequeña superficie en la cual pueden soldar sus propios componentes para crear una placa personalizada.

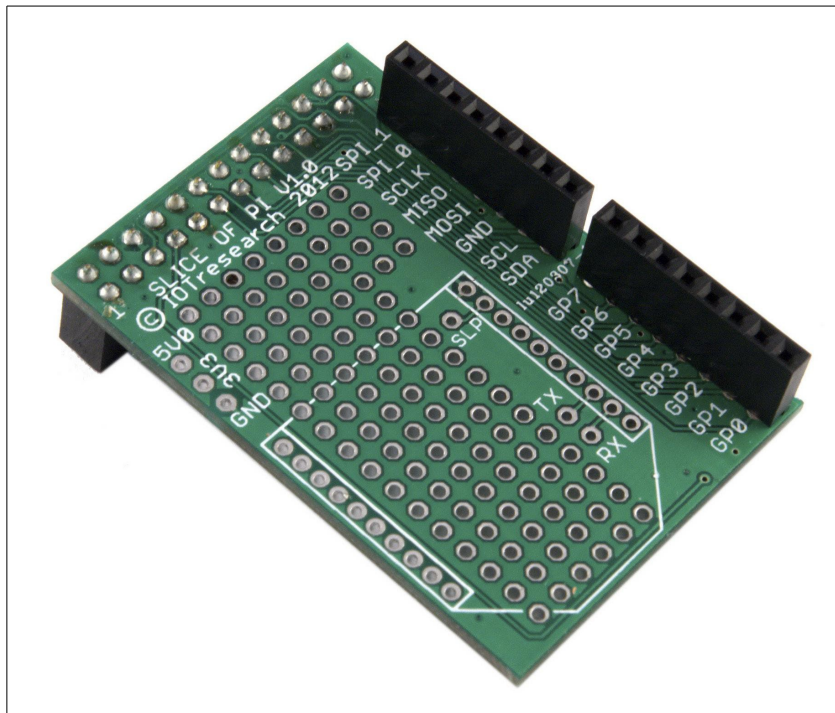
El mercado del hardware está rápida y constantemente cambiando, y la Fundación Raspberry Pi continúa alentando a los desarrolladores a crear más y más dispositivos de expansión. Sería imposible crear una lista exhaustiva de todas las placas de expansión disponibles o planeadas para ser liberadas en un futuro cercano, pero en este capítulo, aprenderá acerca de tres de las placas más comunes y cómo pueden ser utilizadas.

## Slice of Pi de Ciseco

La más básica de las placas, Slice of Pi de Ciseco (ver Figura 16-1) es sin embargo muy útil. Proporcionada en forma de kit, Slice of Pi presenta una pequeña área de prototipado, un espacio para un módulo inalámbrico Xbee y acceso total a los pines GPIO de la Raspberry Pi.

La compacta Slice of Pi está diseñada para ser conectada directamente a los conectores GPIO encima de la Raspberry Pi, con el resto de la placa cubriendo parte de la superficie de la RasPi (pero no extendiéndose más allá de los bordes de la placa ni ocultando algunos de los puertos comúnmente utilizados). Sin embargo, la Slice of Pi se sitúa encima del conector de la salida DSI. En la mayoría de los casos en los que se utiliza este conector, el cable de cinta puede ser encaminado por debajo de la placa Slice of Pi sin ningún problema.

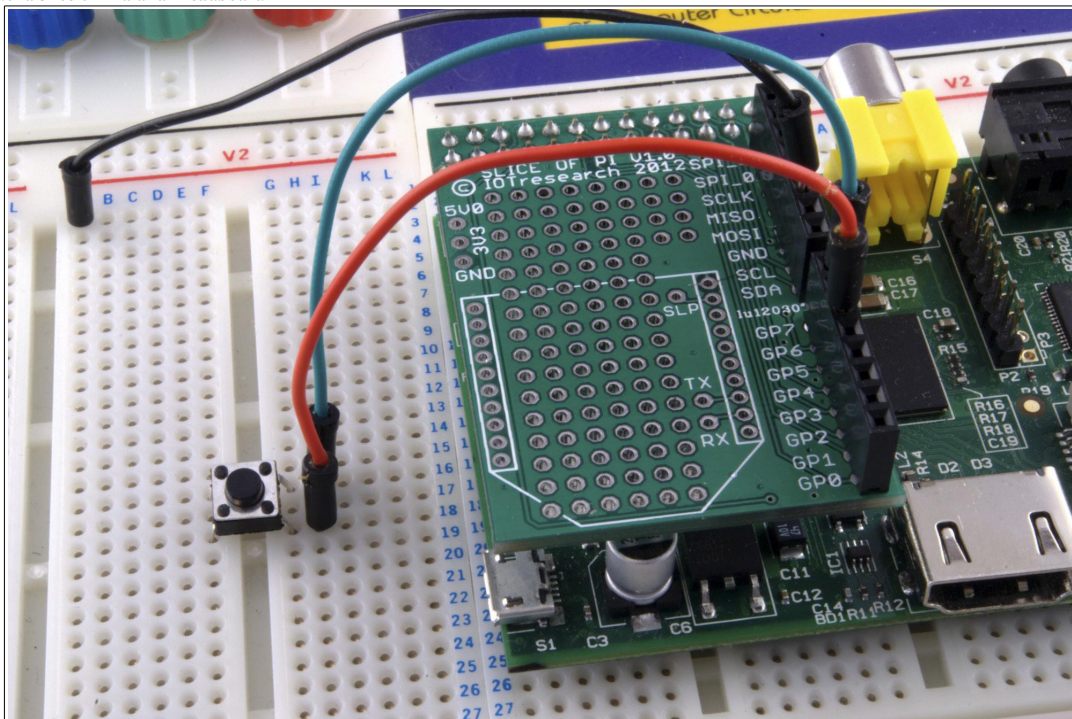
**Figura 16-1**  
La placa Slice of Pi de Ciseco



La Slice of Pi aparte de permitir la comunicación directa entre los conectores del GPIO de la RasPi, tiene como principales ventajas que tiene etiquetados los pines sobre la placa (lo que no tiene el GPIO de la RasPi) y que utiliza conectores hembra. Estos le permiten utilizar *cable jumpers* macho a macho o incluso sólo extensiones de cables cortados para conectar la placa a la breadboard (como se muestra en la Figura 16-2) u otra placa de circuito (o incluso, para conectar componentes directamente en los conectores). Los conectores machos de la RasPi, en contraste, requieren del uso de cable jumpers macho a hembra especializados.

Figura 16-2

Conectando la Slice of Pi a una Breadboard

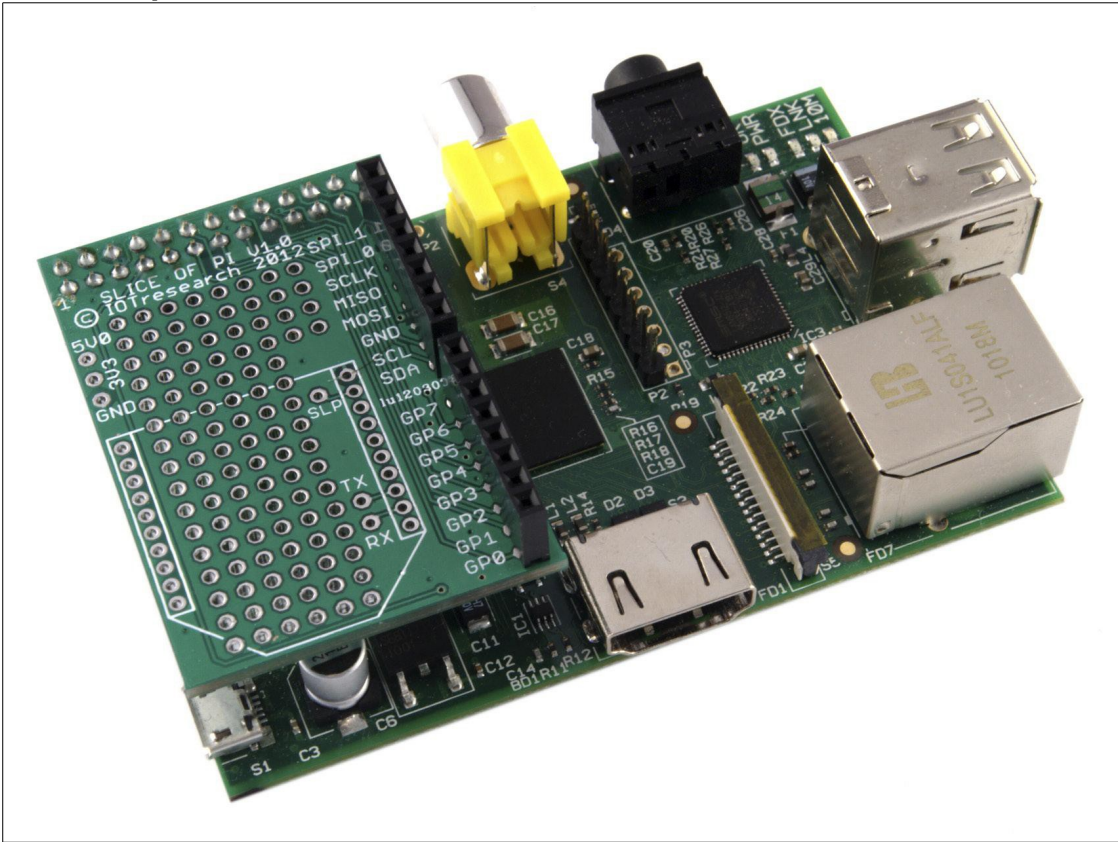


La Slice of Pi también ofrece una pequeña cantidad de protección para la RasPi. El uso de conectores hembra significa que es más difícil provocar un cortocircuito accidentalmente entre dos pines, y el espaciamiento de los conectores (con los ocho pines de propósito general en un conector y los demás pines especializados en otro) hacen que los errores de cableado sean menos probables. Más importante aún, la Slice of Pi no proporciona acceso directo a alguno de los pines etiquetados como Do Not Connect/No Conectar (ver Capítulo 13, “Aprendiendo a Hackear Hardware”), por lo que existe menos riesgo de dañar la RasPi con una conexión incorrecta.

Los conectores para una interfaz inalámbrica XBee también podrían ser útiles si está considerando utilizar la RasPi en algún proyecto de sistemas embebido. Cuando se combina con un transceptor compatible (como el módulo XBee, RF-BEE o RN-XV) permite el acceso inalámbrico al puerto serie UART de la RasPi. Dicho acceso puede ser utilizado para leer los detalles del sensor desde la RasPi, tomando control del software de la RasPi o incluso tomando control de un robot potenciado por la Raspberry Pi desde una ubicación remota.

Finalmente, al área de prototipado proporciona una alternativa a las breadboard o stripboards para el diseño de pequeños circuitos. Si está considerando desarrollar un pequeño módulo de expansión para la Raspberry Pi (como una placa sensor para leer temperaturas o detectar la humedad), los componentes requeridos pueden ser soldados directamente a la Slice of Pi. Gracias a su bajo costo, proporciona una manera fácil de crear módulos de expansión hechos a la medida de la RasPi que son completamente autónomos, sin requerir de cableado, sólo tiene que conectar la Slice of Pi al puerto GPIO de la Raspberry Pi y el dispositivo está listo para andar (ver Figura 16-3).

Figura 16-3  
La Slice of Pi, conectada al puerto GPIO de la RasPi



La desventaja de la Slice of Pi radica en su simplicidad. Su compacto tamaño significa que no es adecuada para la creación de prototipos más grandes o complejos, y aunque el diseño del conector proporciona cierta protección contra los cortos circuitos, no hay protección contra el uso de los componentes de 5 V que pueden dañar los circuitos lógicos de 3.3 V de la RasPi. El etiquetado es también diferente del que se utiliza para describir comúnmente los pines del GPIO de la Raspberry Pi, lo cual puede llevar a confusión cuando se utilice una librería GPIO que no está escrita con la Splice of Pi en mente. La Tabla 16-1 proporciona un mapeo de un sistema de etiquetado a otro.

Table 16-1: Mapeo del GPIO en la Slice of Pi

Número de Pin Físico	Denominación Oficial	Etiqueta de Slice of Pi
7	Pin GPIO 4 (o General Purpose Clock 0)	GP7
11	Pin GPIO 17	GP0
12	Pin GPIO 18	GP1
13	Pin GPIO 27 (21 en las placas Revisión 1)	GP2
15	Pin GPIO 22	GP3
16	Pin GPIO 23	GP4
18	Pin GPIO 24	GP5
22	Pin GPIO 25	GP6



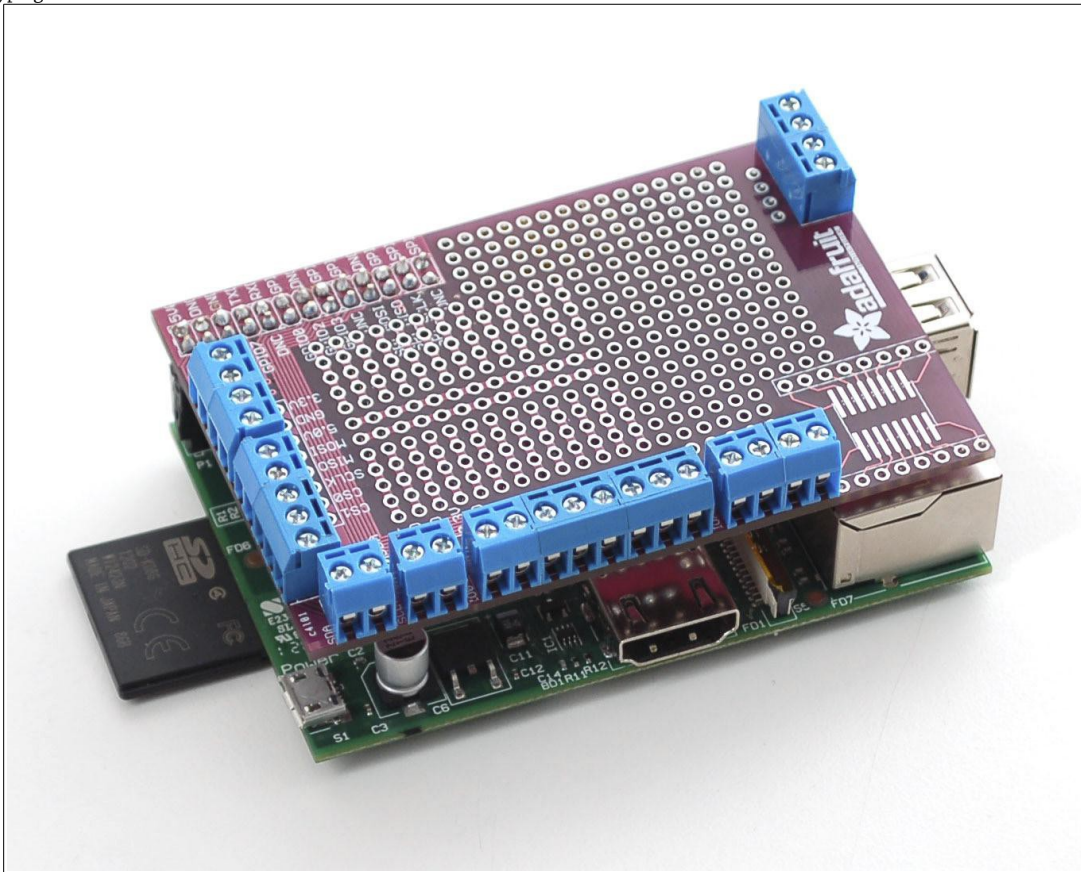
La Slice of Pi se encuentra disponible en la tienda en línea de Ciseco en <http://shop.ciseco.co.uk/slice-of-pi/>.

## Prototyping Pi Plate de Adafruit

La placa de prototipado Pi Plate de Adafruit está basada en el mismo principio general que la Slice of Pi de Ciseco. La placa de expansión se proporciona a manera de kit y le ofrece una superficie de prototipado con fácil acceso a los pines GPIO de la RasPi (ver Figura 16-4). Sin embargo, a diferencia de Slice of Pi, Prototyping Pi Plate es una placa de cobertura total construida del mismo tamaño que la propia RasPi. Por consiguiente, Plate ofrece un área mucho más grande para trabajar que con la Slice of Pi, permitiendo construir circuitos más grandes. La Prototyping Plate también cuenta con *terminales o bornes atornillables* de 3.5 mm para la conexión de los pines, por lo que resulta fácil utilizar cables ordinarios para crear conexiones resistentes de manera temporal (aunque también se proporcionan orificios de 2.54 mm para los conectores macho o hembra en caso de que prefiera utilizarlos en su lugar).

Figura 16-4

La Prototyping Pi Plate de Adafruit



El diseño y distribución de la Prototyping Pi Plate será familiar a cualquier persona que haya utilizado un microcontrolador Arduino. Una placa de expansión con el mismo footprint que el dispositivo de destino, diseñada para conectarse a los conectores a bordo y sentada encima de la superficie de la placa de origen es algo muy visto en el mundo Arduino, en donde tales placas de expansión son conocidas como *shields*. La idea de la Prototyping Pi Plate, en efecto, nace de una placa Protoshield para Arduino de diseño propio de Adafruit.

Como una placa de expansión de cobertura total, la Pi Plate está diseñada para conectarse al conector GPIO y cubrir la superficie entera de la RasPi. Su posicionamiento encima de la RasPi hace difícil acceder a los conectores de la salida de video DSI y la entrada de la cámara MIPI CSI-2, aunque los cables de cinta más largos pueden pasar por debajo de la Pi Plate sin ningún problema. Si está planeando utilizar un módulo de cámara (ver Capítulo 15, “El Módulo de Cámara de la Raspberry Pi”) o una pantalla conectada por DSI, asegúrese de verificar la longitud de su cable de cinta antes de planear utilizarlo en su proyecto al lado de la Pi Plate.

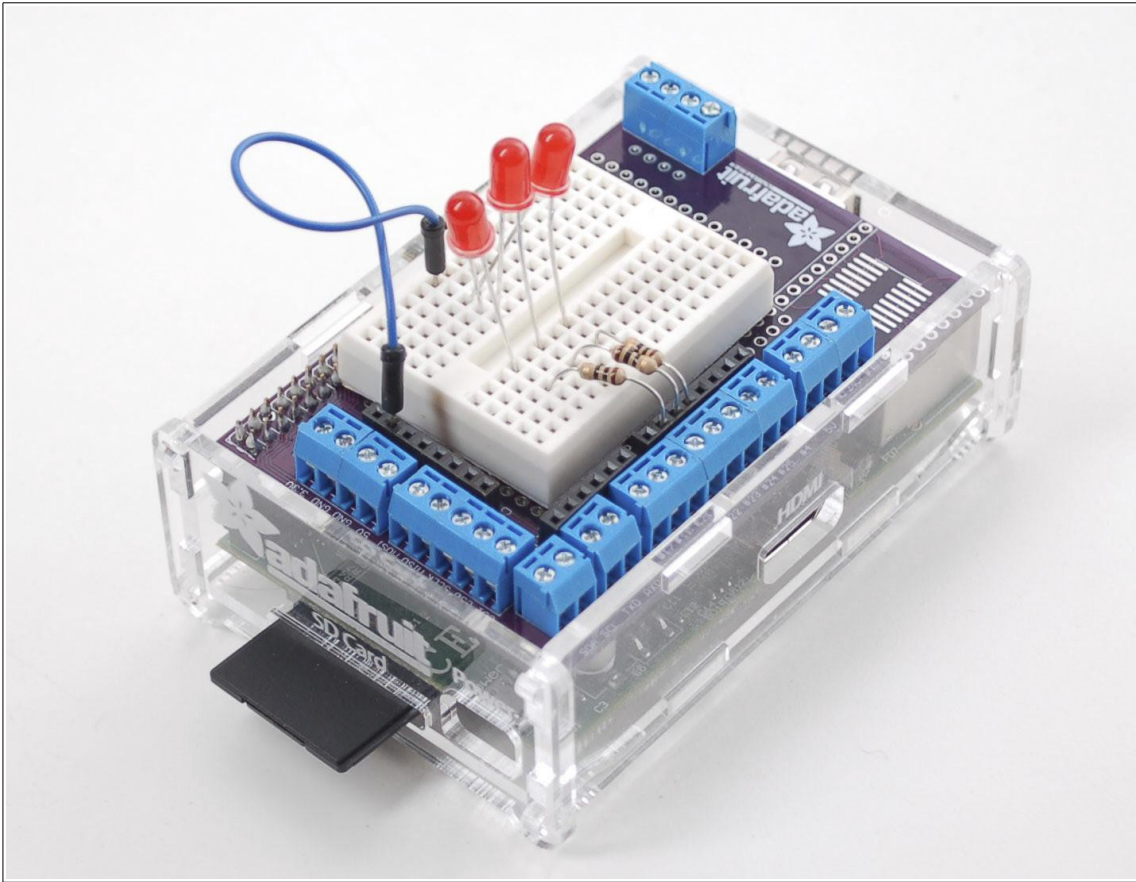
La superficie de prototipado de la Pi Plate está dividida en dos partes, ambas ofrecen construcción through-hole en el espaciado común de 2.54 mm. La primera mitad de la superficie de prototipado está configurada de un modo similar a una breadboard: pistas de cobre sobre la parte inferior enlazan a las filas juntas, y un bus central en el centro proporcionan energía común y conectividad a tierra. La segunda mitad de la superficie de prototipado no cuenta con tales pistas, lo que permite que crear circuitos más personalizados. En general, la superficie que proporciona para la creación de circuitos es significativamente mayor que la de Slice of Pi, convirtiendo a la Pi Plate en la más idónea para proyectos más complejos. Adicionalmente ofrece una *superficie SOIC*, diseñada para soldar algún componente de montaje superficial para aquellas piezas electrónicas que no se encuentran disponibles en formato through-hole.

Gracias a su gran tamaño, también puede utilizarse la Pi Plate junto con una breadboard de un cuarto de tamaño (vendida en la tienda Adafruit como “tiny breadboard”). Esta pequeña breadboard de dos secciones, viene con una espuma autoadherible en su parte de atrás y puede ser adherida en la parte superior de la Pi Plates sobre el área de prototipado (ver Figura 16-5). Si hace esto último significa que será imposible utilizar el área de prototipado para crear un circuito de forma permanente, pero la combinación nos ofrece contar con una placa de expansión autocontenida para el rápido prototipado temporal de circuitos más pequeños.

Al igual que la placa Slice of Pi, la Prototyping Pi Plate es en el fondo un dispositivo básico. El kit no contiene componentes activos, sino simplemente conectores, terminales y la propia placa. Como resultado, proporciona una pequeña protección extra para los puertos GPIO de la Raspberry Pi haciendo que sea más difícil provocar un cortocircuito de manera accidental entre dos conexiones. A diferencia de Slice of Pi, la Pi Plate ofrece acceso a todos los 26 pines del conector GPIO de la RasPi, lo que significa que puede accidentalmente conectar algunas cosas a los pines etiquetados en el diagrama del conector GPIO como Do Not Connect/No Conectar (ver Capítulo 13, “Aprendiendo a Hacker Hardware”). Entrar en comunicación con estos pines de manera directa no es lo más recomendable y puede provocar daños a la RasPi.

Figura 16-5

Una Pi Plate con una breadboard de un cuarto de tamaño

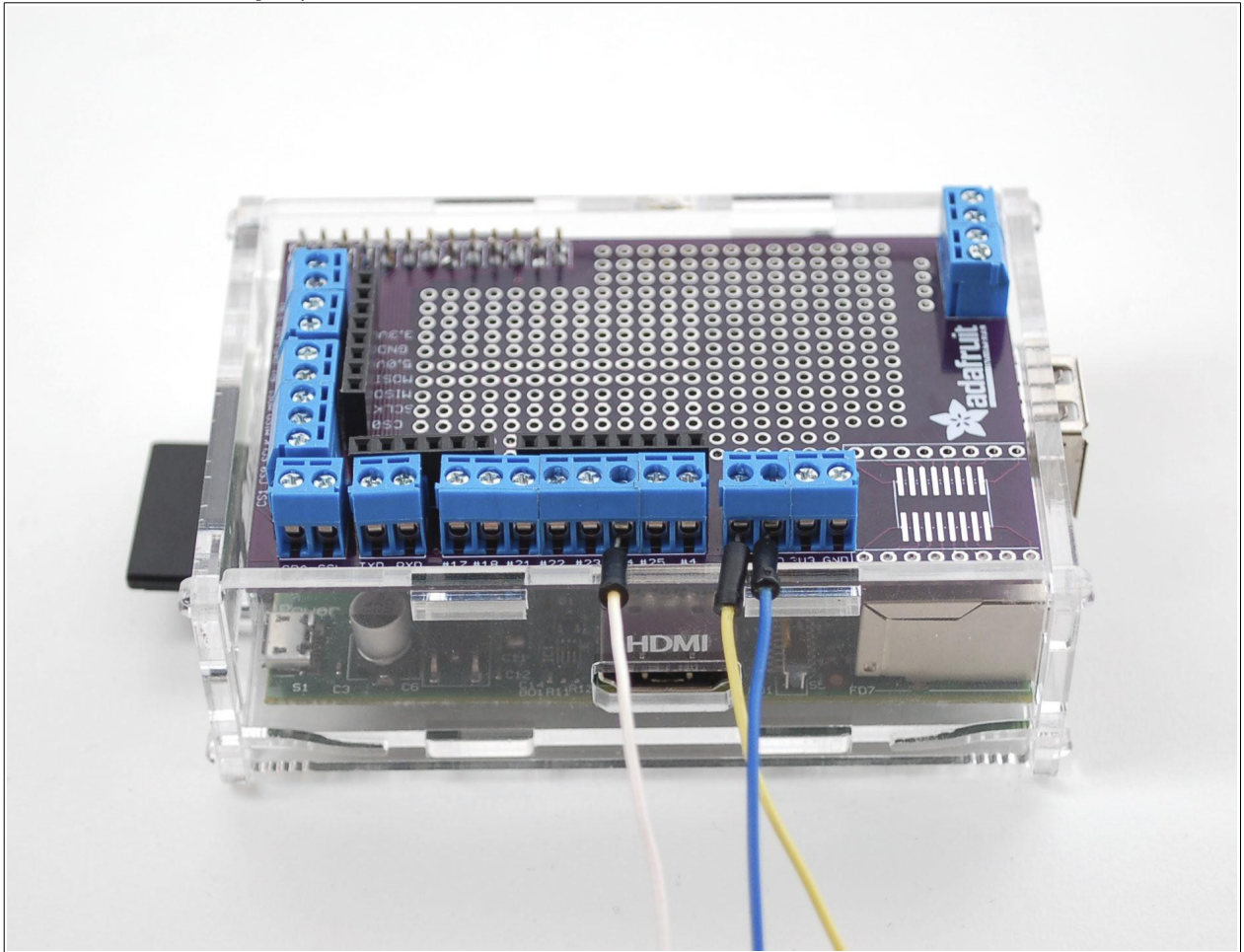


Tal vez la mayor razón para elegir una Pi Plate sobre una Slice of Pi es por su diseño inteligente. Una vez conectada al conector GPIO en la RasPi, hay pocas razones para quitarla a menos que necesite acceder a los conectores DSI o MIPI CSI-2. Debido a que sólo aumenta la altura de la RasPi (antes que el ancho o largo), y porque incluye los bornes atornillables para acceder de lado a los pines GPIO, además es compatible con un número sorprendente de carcasas de Raspberry Pi (ver Figura 16-6). Sin embargo, si está planeando utilizar la RasPi y la Pi Plate en combinación con una carcasa, asegúrese de comprobar que la carcasa pueda ajustar su altura o tenga acceso a los puertos GPIO a través de su tapa.

Puede comprar el Kit Prototyping Pi Plate de Adafruit en <http://www.adafruit.com/products/801>

Figura 16-6

La Pi Plate conectada a una Raspberry Pi con carcasa



## Gerboard de Fen Logic

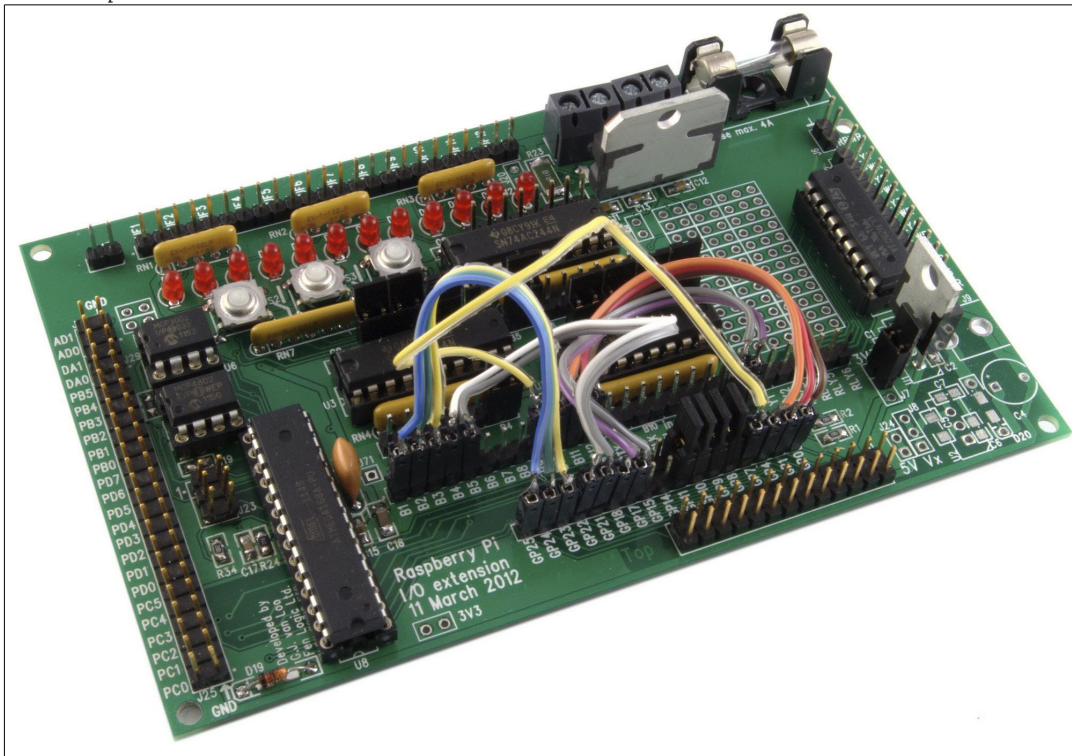
Más exactamente conocida como un Extensor de E/S para la Raspberry Pi, la Gertboard (mostrada en la Figura 16-7) es llamada así por su inventor Gert van Loo. Un empleado de Broadcom y un miembro del equipo que diseñó el procesador BCM2835 en el interior de la Raspberry Pi, van Loo creó la Gertboard como una manera de desbloquear la potencia del chip oculta tras el diseño general de la RasPi y para proveer una plataforma potente y versátil para el retoque electrónico.

A diferencia de Slice of Pi y la Prototyping Pi Plate, la Gerboard es una placa de expansión *activa* con numerosos componentes diseñados para extender la funcionalidad de la RasPi. La Gertboard proporciona 12 puertos *IO buffered* con LEDs de estado, tres botones, seis *controladores de colector abierto*, un *controlador de motor* de 48 V 4 A, un *convertidor de doble canal digital a analógico* (DAC) y un *convertidor de doble canal analógico a digital* (ADC). Además, la Gertboard tiene soporte para un microcontrolador adicional de 28 pines (tal como el Atmel ATmega328 utilizado en el sistema de prototipado Arduino), que puede ser insertado en un zócalo sobre la Gertboard y puede ser programado directamente desde la Raspberry Pi.



**Figura 16-7**

La Gertboard diseñada por Gert van Loo



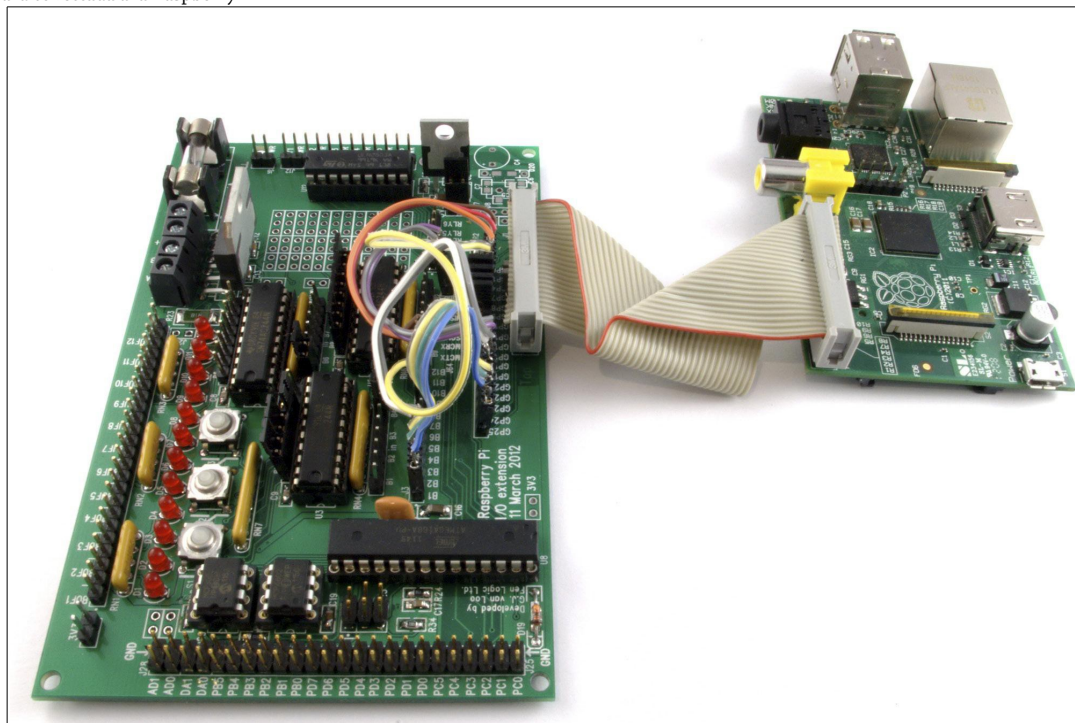
Como su lista de características nos lo dice, la Gertboard es un dispositivo potente y complejo. Su flexibilidad lo hacen una gran opción para proyectos más complejos (pero tal complejidad se manifiesta en su tamaño). El tamaño de la Gertboard es de al menos dos veces y media más grande que la propia Raspberry Pi, convirtiéndose en la placa de expansión más potente y más grande para la RasPi que se puede encontrar al momento de escribir estas líneas. Para contrarrestar su tamaño, la Gertboard se conecta a la RasPi a través de un cable plano con terminales hembra (ver Figura 16-8). Este cable le permite a la Gertboard posicionarse lejos de la RasPi, o ser montado en un proyecto de manera permanente y aún así poder ser separado rápidamente de la RasPi si es necesario.

Quizás la característica más importante de la Gertboard es la protección que ofrece a la RasPi. Utilizando un regulador de 3.3 V, la Gerboard está diseñada para evitar que el puerto GPIO de la RasPi reciba voltajes que no puede manejar. Al igual que la Slice of Pi, intenta restringir el acceso a los seis pines etiquetados como *Do Not Connect/No Conectar* así como también los pines 3.3 V y 5 V para ayudar a prevenir cortocircuitos accidentales. Aunque debe de tener cuidado mientras conecta los circuitos, seguir las instrucciones incluidas en el manual de usuario de la Gertboard le garantizan la seguridad de su Raspberry Pi.



Figura 16-8

La Gertboard conectada a la Raspberry Pi

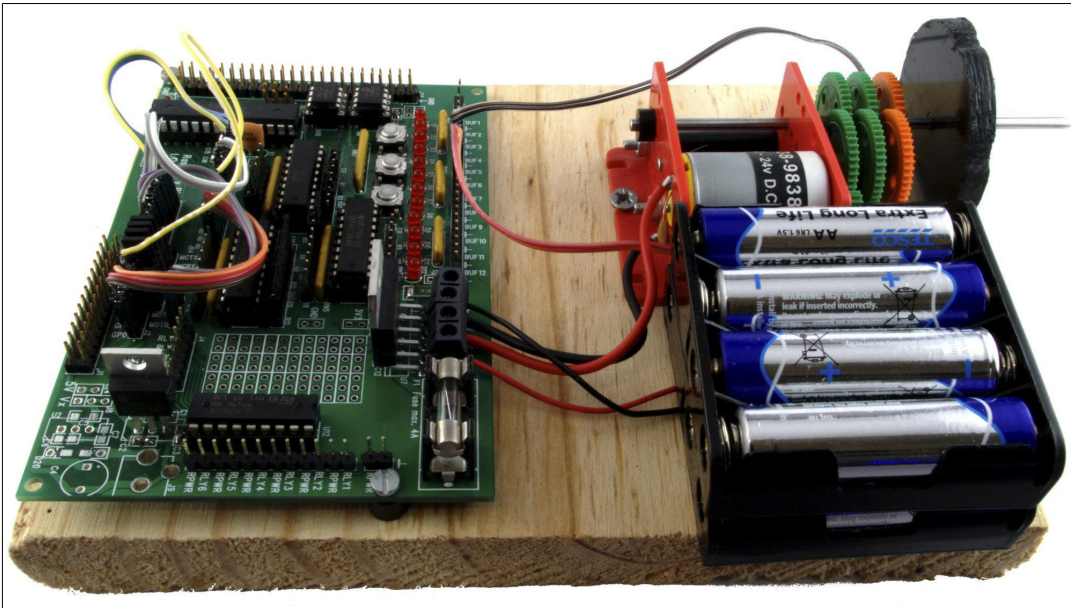


La funcionalidad del controlador de motor de la Gertboard hace que sea fácil de integrar la RasPi en proyectos de robótica. Es adecuado para motores de hasta 48 V a 4 A, que fácilmente tienen la potencia suficiente para conducir un pequeño vehículo robot. Lamentablemente, el motor no se incluye como estándar con la Gertboard. Por lo que tendrá que preguntar a su proveedor si dispone de uno apropiado (siendo el paquete L6203 una opción recomendada) si está planeando utilizar el controlador de motor en su proyecto. Con el controlador de motor soldado en su lugar, la Gertboard puede utilizarse para controlar motores sorprendentemente potentes a través de las instrucciones recibidas por la misma RasPi o desde el módulo de microcontrolador Atmel. La Figura 16-9 muestra un circuito de ejemplo que está utilizando la Gertboard para manejar un motor de 12 V desde una fuente de alimentación por baterías.

Para la construcción de proyectos, la Gertboard ofrece significativamente más funcionalidad que una Raspberry Pi sola. Los 12 puertos de Entrada/Salida buffered, ubicados en la parte superior la placa, pueden ser configurados como entradas o salidas y proporcionar más conectividad que los siete u ocho pines de Entrada/Salida de propósito general del puerto GPIO de la Raspberry Pi. La inclusión de los LEDs, que indican cuando un pin se encuentra en estado alto o bajo, nos ayudan a detectar y solucionar problemas con los circuitos así como también nos son de gran utilidad en la educación electrónica, haciendo que sea fácil observar en un momento determinado qué es lo que las diversas entradas y salidas están haciendo.

Figura 16-9

Utilizando el controlador de motor de la Gertboard

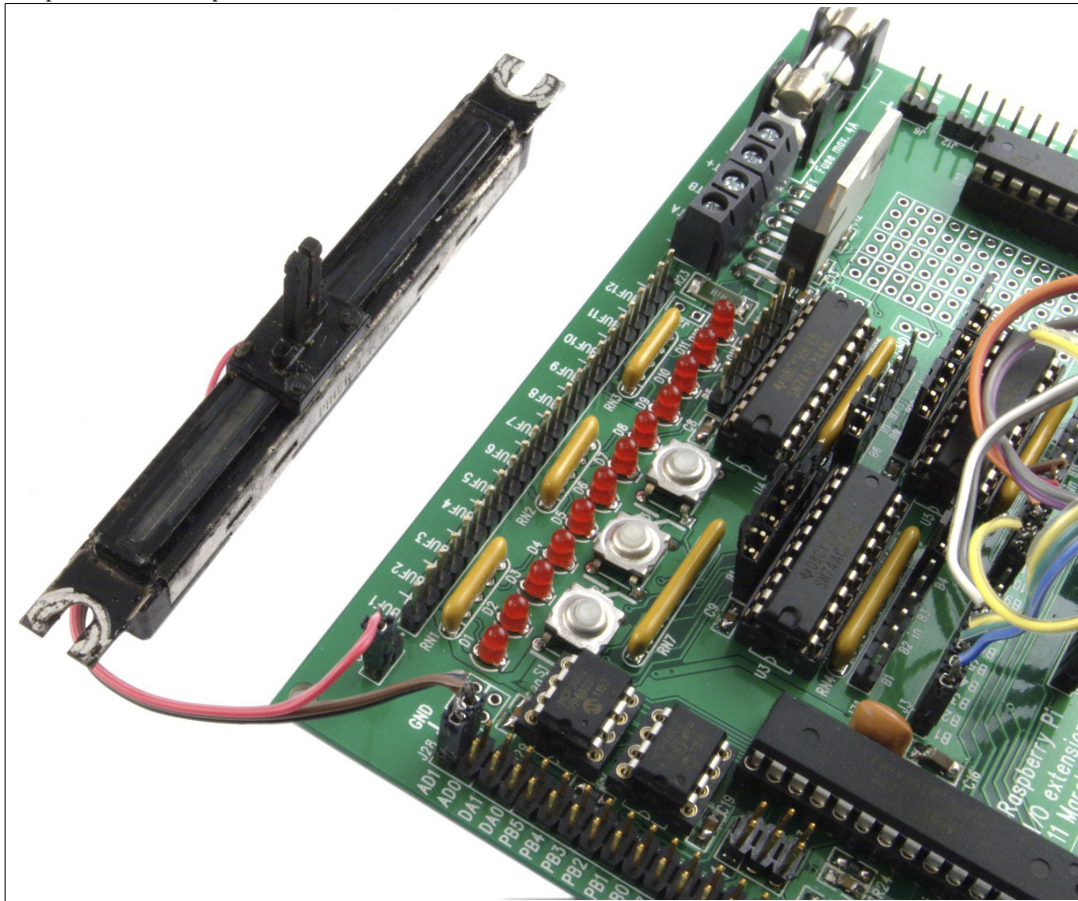


Para los proyectos de detección o retroalimentación, los componentes ADC y DAC de la Gertboard son componentes robustos agregados. A diferencia de los pines GPIO de la propia RasPi, que sólo pueden recibir y enviar señales digitales, la Gertboard incluye dos conexiones ADC y dos conexiones DAC. Los pines ADC, ubicados en la parte superior izquierda de la Gertboard, permiten que las señales de componentes analógicos puedan ser convertidas en señales digitales compatibles con la RasPi. Como un ejemplo, la Figura 16-10 muestra cómo puede utilizar los pines ADC para leer el estado de un *potenciómetro* (un componente que varía su resistencia de acuerdo a la posición de un deslizante o mango). Este circuito puede utilizarse para controlar el volumen de la RasPi en una aplicación de centro multimedia o para alterar la velocidad del motor conectado. Los pines DAC proporcionan la funcionalidad opuesta, tomando una señal digital desde la RasPi y convirtiéndola a analógica. Esto podría nos permitirá controlar un altavoz para crear audio o alterar la velocidad de un motor o el brillo de un LED,

Para proyectos más complejos, la Gertboard proporciona un controlador de colector abierto, que utiliza transistores para encender o apagar los dispositivos que tengan un requerimiento de energía diferente al de 3.3 V utilizado por la Gertboard o que consuman una gran cantidad de corriente y requieran de una fuente de alimentación externa para funcionar. El controlador de colector abierto puede controlar las fuentes de alimentación de hasta seis dispositivos simultáneamente, sin requerir algún otro hardware. Esto le proporciona a la Gertboard una gran flexibilidad (aunque los transistores o relés se pueden utilizar también en las placas de expansión pasivas como la Prototyping Pi Plate para alcanzar el mismo objetivo).

**Figura 16-10**

Conectando un potenciómetro a los pines ADC de la Gertboard



Finalmente, la Gertboard proporciona acceso a un microcontrolador de 28 pines, que puede ser empotrado en un zócalo ubicado sobre la misma placa. Cuando se empotra, la Gertboard puede ser cableada para dirigir ya sea el puerto GPIO de la Raspberry Pi o el microcontrolador, o una combinación de los dos. La Gertboard está diseñada para trabajar con un microcontrolador ATmega168 o ATmega328 (el mismo tipo de microcontrolador utilizado por el proyecto de prototipado Arduino). Cuando está emprotrado, puede usted escribir programas para el microcontrolador Atmega de la Gertboard en el Atmel IDE, proporcionando a la RasPi un nivel útil de compatibilidad con una amplia gama de software orientado a los Atmega disponibles para su descarga. Esto también le proporciona a la RasPi la habilidad de dirigir hardware externo en tiempo real, dando al microcontrolador la responsabilidad para hacer de interfaz con el hardware y simplemente proporcionando retroalimentación a la RasPi.

La Gertboard es un dispositivo muy avanzado y no es adecuado para todo proyecto. Para muchas aplicaciones sencillas, una económica placa de expansión pasiva como la Slice of Pi será normalmente suficiente. Sin embargo, para experimentar en profundidad lo mucho que es posible hacer con el procesador BCM2835 de la Raspberry Pi y/o asegurar que su RasPi será capaz de interactuar con casi cualquier hardware, la Gertboard es una gran elección y un potente extensor de las habilidades ya impresionantes de la RasPi.

La Gertboard está disponible a través de Farnell en <http://www.farnell.com/>

Parte IV

# Apéndices

- Apéndice A** Python Recetas
- Apéndice B** Rápida Referencia del Módulo de Cámara
- Apéndice C** Modos de Pantalla HDMI



## **Apéndice A**

### Python Recetas

**Las siguientes recetas** muestran los códigos completos para los programas del Ejemplo 3 y 4 del Capítulo 12, “Una Introducción a Python”, y muestra una solución para el programa de entrada y salida combinada del Capítulo 14, “El Puerto GPIO”. Cada receta también puede descargarse desde el sitio web de la Raspberry Pi Guía del Usuario en [www.wiley.com/go/raspberrypiuserguide2e](http://www.wiley.com/go/raspberrypiuserguide2e). Si está escribiendo el código a mano, ponga atención al símbolo ¶: este indica que la línea ha sido quebrada en dos líneas debido al espacio del libro. Si ve este símbolo al final de la línea, no presione Intro, continúe escribiendo hasta que haya llegado al final de una línea que no tenga este símbolo.

## Raspberry Pi Serpiente (Capítulo 12, Ejemplo 3)

```
#!/usr/bin/env python
# Raspberry Serpiente
# Escrito por Gareth Halfacree para la Raspberry Pi Guía del Usuario

import pygame, sys, time, random
from pygame.locals import *

pygame.init()

fpsClock = pygame.time.Clock()

playSurface = pygame.display.set_mode((640, 480))
pygame.display.set_caption('Raspberry Snake')

redColour = pygame.Color(255, 0, 0)
blackColour = pygame.Color(0, 0, 0)
whiteColour = pygame.Color(255, 255, 255)
greyColour = pygame.Color(150, 150, 150)
snakePosition = [100,100]
snakeSegments = [[100,100],[80,100],[60,100]]
raspberryPosition = [300,300]
raspberrySpawned = 1
direction = 'right'
changeDirection = direction

def gameOver():
    gameOverFont = pygame.font.Font('freesansbold.ttf', 72)
    gameOverSurf = gameOverFont.render('Game Over', True, greyColour)
    gameOverRect = gameOverSurf.get_rect()
    gameOverRect.midtop = (320, 10)
    playSurface.blit(gameOverSurf, gameOverRect)
    pygame.display.flip()
    time.sleep(5)
    pygame.quit()
    sys.exit()

while True:
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
        elif event.type == KEYDOWN:
            if event.key == K_RIGHT or event.key == ord('d'):
                changeDirection = 'right'
            if event.key == K_LEFT or event.key == ord('a'):
                changeDirection = 'left'
```



```

        if event.key == K_UP or event.key == ord('w'):
            changeDirection = 'up'
        if event.key == K_DOWN or event.key == ord('s'):
            changeDirection = 'down'
        if event.key == K_ESCAPE:
            pygame.event.post(pygame.event.Event(QUIT))
    if changeDirection == 'right' and not direction == 'left':
        direction = changeDirection
    if changeDirection == 'left' and not direction == 'right':
        direction = changeDirection
    if changeDirection == 'up' and not direction == 'down':
        direction = changeDirection
    if changeDirection == 'down' and not direction == 'up':
        direction = changeDirection
    if direction == 'right':
        snakePosition[0] += 20
    if direction == 'left':
        snakePosition[0] -= 20
    if direction == 'up':
        snakePosition[1] -= 20
    if direction == 'down':
        snakePosition[1] += 20
    snakeSegments.insert(0,list(snakePosition))
    if snakePosition[0] == raspberryPosition[0] and snakePosition[1] == raspberryPosition[1]:
        raspberrySpawned = 0
    else:
        snakeSegments.pop()
    if raspberrySpawned == 0:
        x = random.randrange(1,32)
        y = random.randrange(1,24)
        raspberryPosition = [int(x*20),int(y*20)]
    raspberrySpawned = 1
    playSurface.fill(blackColour)
    for position in snakeSegments:
        pygame.draw.rect(playSurface,whiteColour,Rect(position[0], position[1], 20, 20))

    pygame.draw.rect(playSurface,redColour,Rect(raspberryPosition[0], raspberryPosition[1], 20, 20))
    pygame.display.flip()
    if snakePosition[0] > 620 or snakePosition[0] < 0:
        gameOver()
    if snakePosition[1] > 460 or snakePosition[1] < 0:
        gameOver()
    for snakeBody in snakeSegments[1:]:
        if snakePosition[0] == snakeBody[0] and snakePosition[1] == snakeBody[1]:
            gameOver()

    fpsClock.tick(30)

```

## IRC User List (Capítulo 12, Ejemplo 4)

```

#!/usr/bin/env python
# IRC User List
# Escrito por Tom Hudson para la Raspberry Pi Guía del Usuario
# http://tomhudson.co.uk/

```



```

import sys, socket, time

RPL_NAMREPLY    = '353'
RPL_ENDOFNAMES = '366'

irc = {
    'host':      'chat.freenode.net',
    'port':      6667,
    'channel':   '#raspiuserguide',
    'namesinterval': 5
}

user = {
    'nick':      'botnick',
    'username':  'botuser',
    'hostname':  'localhost',
    'servername': 'localhost',
    'realname':  'Raspberry Pi Names Bot'
}

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

print 'Connecting to %(host)s:%(port)s...' % irc
try:
    s.connect((irc['host'], irc['port']))
except socket.error:
    print 'Error connecting to IRC server %(host)s:%(port)s' % irc
    sys.exit(1)

s.send('NICK %(nick)s\r\n' % user)
s.send('USER %(username)s %(hostname)s %(servername)s :%(realname)s\r\n' % user)
s.send('JOIN %(channel)s\r\n' % irc)
s.send('NAMES %(channel)s\r\n' % irc)

read_buffer = ''
names = []

while True:
    read_buffer += s.recv(1024)
    lines = read_buffer.split('\r\n')
    read_buffer = lines.pop();
    for line in lines:
        response = line.rstrip().split(' ', 3)
        response_code = response[1]
        if response_code == RPL_NAMREPLY:
            names_list = response[3].split(':')[1]
            names += names_list.split(' ')
        if response_code == RPL_ENDOFNAMES:
            print '\r\nUsers in %(channel)s:' % irc
            for name in names:
                print name
            names = []
            time.sleep(irc['namesinterval'])
            s.send('NAMES %(channel)s\r\n' % irc)

```

## Entrada y Salida GPIO (Capítulo 14)

```
#!/usr/bin/env python
# Ejemplo de Entrada/Salida con el GPIO de la Raspberry Pi
# Escrito por Gareth Halfacree Raspberry Pi Guía del Usuario

import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)
GPIO.setup(12, GPIO.IN)
GPIO.output(11, False)

while True:
    input_value = GPIO.input(12)
    if input_value == False:
        print "The button has been pressed. Lighting LED."
        GPIO.output(11, True)
        while input_value == False:
            input_value = GPIO.input(12)
        print "The button has been released. Extinguishing LED."
    if input_value == True:
        GPIO.output(11, False)
```

## **Apéndice B**

### Rápida Referencia del Módulo de Cámara de la Raspberry Pi

**El Módulo de Cámara** de la Raspberry Pi viene con un software diseñado para capturar imágenes fijas y videos, cada uno cuenta con una variedad de opciones que se pueden controlar para obtener una salida final. Las opciones para `raspistill`, `raspiyuv` y `raspivid` están incluidas en este apéndice. Para más información, ver el Capítulo 15, “El Módulo de Cámara de la Raspberry Pi”

## Opciones Compartidas

Las siguientes opciones, listadas en orden alfabético, están compartidas entre `raspistill`, `raspiyuv` y `raspivid`. Para ver las opciones específicas de cada aplicación, vea los artículos `raspistill`, `raspiyuv` y `raspivid` individuales más adelante en este apéndice.

- **-? o --help (Ayuda)**-Muestra todas las opciones y sus usos.
- **-awb o --awb (Automatic White Balance o Balance de Blancos Automático)**-Ajusta la temperatura de color de la imagen o video capturado de acuerdo a una configuración elegida de entre una serie de configuraciones preestablecidas. Si su imagen o video aparece azul o naranja, pruebe ajustar esta opción primero. Los valores posibles para esta opción son: `off`, `auto`, `sun`, `cloud`, `shade`, `tungsten`, `fluorescent`, `incandescent`, `flash`, `horizon`.
- **-br o --brightness (Brightness o Brillo)**-Ajusta el brillo de la imagen o video capturado. Los valores posibles para esta opción son números enteros que van de 0 (brillo mínimo) a 100 (brillo máximo).
- **-cfx o --colfx (Colour Effects o Efectos de Color)**-Permite al usuario ajustar el espacio de color YUV para el control de grano fino de la imagen final. Deben darse valores como U:V, en donde U controla la crominancia y V la luminancia. Un valor de 128:128 resultará en una imagen en escala de grises.
- **-co o --contrast (Contrast o Contraste)**- Ajusta el contraste de la imagen o video capturado. Los valores posibles para esta opción son números enteros que van de -100 (mínimo contraste) a 100 (máximo contraste).
- **-d o --demo (Demonstration Mode o Modo de Demostración)**-La opción `-d` ejecuta el modo de demostración tanto en `raspistill` como en `raspivid`, el cual muestra una vista previa haciendo ciclos a través de las variadas opciones de la cámara. En este modo, ninguna imagen es capturada (incluso si se especifica un archivo de salida utilizando la opción `--output`).
- **-ev o --ev(Exposure Value o Valor de Exposición)**- Permite a la cámara incrementar o disminuir su valor de exposición, aclarando o oscureciendo la imagen o video capturado. A diferencia de los ajustes de brillo y contraste, esta opción afecta la captura verdadera de la imagen o video. Los valores posibles son -10 a 10, siendo 0 el valor predeterminado.
- **-ex o --exposure (Exposure Mode o Modo de Exposición)**-Establece el ajuste de exposición automática de la cámara, el cual controla cuánto tiempo la cámara dedica a capturar una imagen o cuadro individual y es en gran medida un factor de la luz disponible o la velocidad del objeto: los objetos en rápido movimiento necesitan un tiempo de exposición corto para poder mantenerse enfocado, mientras que las tomas con poca luz demandan un tiempo de exposición largo. Los valores posibles para esta opción son `off`, `auto`, `night`, `nightpreview`, `backlight`, `spotlight`, `sports`, `snow`, `beach`, `verylong`, `fixedfps`, `antishake`, `fireworks`.
- **-f o --fullscreen (FullScreen Preview o Vista Previa en Pantalla Completa)**-Hace que la vista previa de la imagen llene toda la pantalla, ignorando cualquier otra opción de vista previa que pueda tener establecida.

- **-h o --height (Height o Altura)**-Especifica la altura o resolución vertical de la imagen o video capturado. La altura deseada debe establecerse en pixeles; por ejemplo, una captura Full HD requerirá un valor de altura de 1080. El valor mínimo que admite es 64 y el máximo depende de si está capturando video o imágenes fijas.
- **-hf o --hflip (Horizontal Flip o Volteo Horizontal)**-Voltea la imagen o video capturado sobre su eje horizontal, como si se viera en un espejo.
- **-ifx o --imxfx (Image Effects o Efectos de Imagen)**-Habilita uno de varios efectos especiales preconfigurados sobre la imagen o video. Los valores posibles para esta opción son `none`, `negative`, `solarise`, `sketch`, `denoise`, `emboss`, `oilpaint`, `hatch`, `gpen`, `pastel`, `watercolour`, `film`, `blur`, `saturation`, `colourswap`, `washedout`, `posterise`, `colourpoint`, `colourbalance`, `cartoon`, `whiteboard`, `blackboard`. Estos ajustes se pueden ver en acción utilizando la opción `Demostation Mode`.
- **-ISO o --ISO (ISO Sensitivity o ISO Sensibilidad)**-Ajusta la sensibilidad de la cámara a la luz. Un valor bajo de ISO proporciona una imagen más clara, pero requiere exposiciones más largas; un valor más alto de ISO puede hacer tomas en tiempos de exposición muy bajos para capturar objetos en rápido movimiento o pobremente iluminados, pero crea una imagen 'ruidosa'.
- **-mm o --metering (Metering Mode o El Modo de Medición)**-Ajusta el modo de medición de luz para una imagen o video capturado, el cual controla las opciones de la exposición automática, el balance de blancos y las sensibilidad ISO. Los valores posibles para esta opción son `average`, `spot`, `backlit`, `matrix`.
- **-n o --nopreview (No Preview o Sin Vista Previa)**-No muestra una ventana de vista previa durante la captura; no es necesaria cuando se utiliza un valor `timeout` de 0 (ver `Capture TimeOut` más adelante).
- **-o o --output (Output File o Archivo de Salida)**-Establece el nombre del archivo a ser guardado. El valor para esta opción puede ser un sólo un nombre de archivo, el cual será creado en el directorio actual, o puede ser una ruta absoluta. Si está utilizando `raspistill` o `raspivid` junto con otra aplicación que espera que los datos de imagen sean pasados a través de la entrada estándar, puede usted dirigir la imagen o video a través de la salida estándar utilizando un carácter de guión (-) como nombre de archivo.
- **-op o --opacity (Preview Opacity o Opacidad de la Vista Previa)**-Controla cuán transparente aparece la vista previa. Los valores posibles son cualquier número entero entre 0 y 255, donde 0 significa completamente transparente y por lo tanto invisible y 255 significa completamente visible. Utilizar un valor aproximado a 128 le permitirá ver la vista previa en vivo, pero también poder leer el texto detrás de la vista previa.
- **-p o --preview (Preview Window Control o Control de Ventana de la Vista Previa)**-Ajusta el tamaño de la ventana de la vista previa y en qué lugar aparece. Los valores deben proporcionarse como X, Y, W, H en donde X e Y son las coordenadas en pixeles de la esquina superior izquierda de la ventana que será dibujada, y W y H son el ancho y el alto en pixeles de la ventana de la vista previa.
- **-rot o --rotation (Rotate Capture o Rotar Captura)**-Rota la imagen o video capturado en cualquier ángulo proporcionado. Los valores para esta opción deben ser números enteros que representan los grados en sentido de las manecillas del reloj, en donde 0 significa ninguna rotación y 359 la rotación máxima posible.

- **-sa o --saturation (Saturation o Saturación)**-Ajusta la saturación del color de la imagen o video capturado. Los valores posibles para esta opción son números enteros que van desde -100 (mínima saturación) hasta 100 (máxima saturación).
- **-sh o --sharpness (Sharpness o Nitidez)**-Ajusta la nitidez de la imagen o video capturado. Los valores posibles para esta opción son números enteros que van desde -100 (mínima nitidez) hasta 100 (máxima nitidez).
- **-t o --timeout (Capute Timeout o Tiempo Espera de la Captura)**-Controla el tiempo de espera en milisegundos que la ventana de vista previa aparece. Aunque esta opción se encuentra compartida entre las herramientas, la acción de la opción `--timeout` es diferente en cada una: en `raspistill`, la opción `--timeout` establece el tiempo a esperar antes de capturar la imagen; en `raspivid`, la opción establece el tiempo de grabación del video. Un valor de 0 en `raspistill` hará que capture una imagen de manera inmediata; un valor de 0 en `raspivid` continua la grabación indefinidamente.
- **-v or --verbose (Verbose Messaging o Mensajería Detallada)**-El modo verbose le indica al software de la cámara que imprima lo que está haciendo a la consola o terminal con tanto detalle como le sea posible. Esta opción por lo general, es sólo utilizada para depurar errores en el software, ya que permite al usuario ver en que momento falla la captura.
- **-vf o --vflip (Vertical Flip o Volteo Vertical)**-Voltea la imagen sobre su eje vertical. Utilizado con mayor frecuencia cuando la cámara no puede ser colocada de forma correcta, con el cable de cinta saliendo por su parte inferior. Si la cámara está en otro ángulo que no sea vertical u horizontal, pruebe utilizar la opción de rotación para controlar la imagen final capturada.
- **-vs o --vstab (Video Stabilisation o Estabilización del Video)**-Procura corregir la imagen de las sacudidas o vibraciones del sensor de la cámara. Utilizado frecuentemente cuando la Raspberry Pi o su Módulo de Cámara es sujeta con la mano, o se encuentra adherida a un robot, vehículo u otra plataforma en movimiento.
- **-w o --width (Width o Ancho)**-Especifica la resolución horizontal o ancho de la imagen o video capturado. El ancho deseado debe especificarse en pixeles; por ejemplo, para una captura Full HD se requiere un valor de ancho de 1920. El valor mínimo es de 64; el máximo depende de si está siendo capturada una imagen fija o video.

## Opciones de Raspistill

Diseñada para capturar imágenes fijas, `raspistill` tiene algunas opciones específicas que no aplican a `raspivid`. Estas opciones están listadas a continuación.

- **-e o --encoding (Encoding Format o Formato de Codificación)**-Establece el formato de la imagen de salida. Dicho formato no tiene ningún efecto sobre la extensión del archivo de salida, la cual debe ser modificada de forma manual utilizando la opción `-output`. Los valores posibles para esta opción son `jpg`, `bmp`, `gif`, `png`.
- **-q o --quality (JPEG Quality o Calidad JPEG)**-Establece el nivel de compresión del JPEG a ser guardado y no tiene efecto alguno cuando se utiliza con otro formato de codificación. Cuanto más bajo sea el valor, más pequeño será el archivo de imagen final; un valor de 100 proporcionará la mejor calidad de imagen posible, mientras que un valor de 0 proporcionará el tamaño de archivo más pequeño. Un valor de 90 es una buena correspondencia entre tamaño y calidad.

- **-r o --raw (Save Bayer Data o Guarda Datos Bayer)**-Guarda la salida del filtro de color Bayer como metadatos en la imagen JPEG y no tiene ningún efecto cuando se utiliza con cualquier otro formato de codificación. Este dato adicional y la salida del sensor de la cámara sin interpolación, pueden ser utilizados en las aplicaciones de edición de imágenes para reconstruir una imagen más detallada o a más alta calidad, pero normalmente no es necesaria.
- **-th o --thumb (Thumbnail Settings o Ajustes de Miniaturas)**-Ajusta el tamaño y la calidad de las miniaturas guardadas con las imágenes JPEG y no tiene efecto cuando se utiliza con algún otro formato de codificación. El valor debe proporcionarse como X:Y:Q, en donde X es el ancho, Y el alto y Q un valor de 0 a 100 que indica la calidad de la miniatura JPEG.
- **-tl o --timelapse (Timelapse Mode o Modo de Lapso de Tiempo)**-Pone a raspistill en el modo de lapso de tiempo, en donde las imágenes serán capturadas a un intervalo de tiempo determinado. Muy útil cuando se utiliza raspistill con un script o una aplicación de terceros estableciendo el archivo de Salida a la salida estándar con un carácter de guión (-); cuando se utiliza con un nombre de archivo en la opción --output, el archivo será sobrescrito cada vez que una nueva imagen es capturada. El valor que recibe esta opción indica el retardo de tiempo que habrá entre cada captura (expresado en milisegundos).
- **-x o --exif (EXIF Tag o Etiqueta EXIF)**-Permite personalizar las etiquetas (o *tags*) del Formato de Archivo de Imagen Intercambiable (EXIF) que serán escritas en la imagen JPEG y no tiene ningún efecto cuando se utiliza con los otros formatos de codificación. Los Tags deben tener el formato de 'clave=valor', por ejemplo, sería posible establecer el nombre del fotógrafo mediante -x 'Autor=Gareth Halfacree'.

## Opciones de Raspivid

Diseñada para capturar imágenes en movimiento, raspivid tiene algunas opciones específicas que no se aplican a raspistill. Estas opciones se listan a continuación.

- **-b o --bitrate (Encoding Bitrate o Bitrate de Codificación)**-Establece la tasa de bits (bitrate) del video capturado, expresado en bits por segundos (BPS). Cuanto mayor sea la tasa de bits, mayor será la calidad del video producido (pero también, mayor será el tamaño del archivo). A menos que su video tenga una necesidad específica, la configuración para esta opción debe dejarse en la predeterminada.
- **-e o --penc (Encoding Preview o Codificación de Vista Previa)**-Utiliza la ventana de vista previa para mostrar los frames o cuadros de vídeo después de que éstos hayan sido pasados por el codificador y no antes. Proporciona una vista previa exacta de cómo se verá la imagen del video final y comúnmente es empleado cuando se están haciendo retoques con el Bitrate de Codificación (Encoding Bitrate).
- **-fps o --framerate (Video Framerate o Framerate del Video)**-Establece la tasa de imágenes del video capturado, expresado en cuadros por segundo. Grandes cantidades proporcionarán un movimiento más suave, mientras que cantidades más pequeñas ocuparán menos espacio en disco. La grabación a una tasa superior de 30 cuadros por segundos (la cual puede ser convertida en un video en cámara lenta mediante una aplicación de edición de video) probablemente sólo funcione con las resoluciones más bajas (ajuste la resolución con las opciones --width y --height).
- **-g o --intra (Intra Refresh Period o Periodo de Refresco Intra)**-Establece la frecuencia con la que un fotograma clave (también conocido como imagen intra-codificada o I frame) debe ser capturado. Un fotograma clave es una imagen completa, en vez de sólo ser los cambios registrados desde la última imagen. A mayor frecuencia de fotogramas claves, más alta será la calidad del video cuando se graban escenas que cambian muy rápidamente, pero esto producirá archivos de gran tamaño.

## Opciones de Raspiyuv

Raspiyuv es una aplicación diseñada para capturar uno de los dos formatos específicos de color de imagen (YUV o RGB). La mayoría de los usuarios estarán mejor utilizando `raspistill`, el cual produce formatos de imagen listos para su uso; sin embargo, para aquellos que planean editar sus imágenes aún después de su captura, pueden encontrar que `raspiyuv` es un herramienta muy útil.

La mayoría de las opciones `raspiyuv` son las mismas que tiene `raspistill` con la siguiente excepción:

- **-rgb o --rgb (Capture RGB o Captura RGB)**-Hace que `raspiyuv` capture los datos de pixeles rojo, verde, azul (RGB) en vez de capturar la crominancia y luminancia, la cuales son normalmente utilizadas.



## **Apéndice C**

### Modos de Pantalla HDMI

**Puede utilizar** los valores de las Tablas C-1 y C-2 en la opción `hdmi_mode` dentro del archivo `config.txt` para modificar el flujo de salida de video HDMI. Para más información, vea el Capítulo 7, “Configuración Avanzada de la Raspberry Pi”.

**Tabla C-1 Grupo HDMI 1 (CEA)**

Valor	Descripción
1	VGA (640x480)
2	480p 60Hz
3	480p 60Hz (Relación de aspecto 16:9)
4	720p 60Hz
5	1080i 60Hz
6	480i 60Hz
7	480i 60Hz (Relación de aspecto 16:9)
8	240p 60Hz
9	240p 60Hz (Relación de aspecto 16:9)
10	480i 60Hz (Cuadruplicación del píxel habilitado)
11	480i 60Hz (Cuadruplicación del píxel habilitado) (Relación de aspecto 16:9)
12	240p 60Hz (Cuadruplicación del píxel habilitado)
13	240p 60Hz (Cuadruplicación del píxel habilitado) (Relación de aspecto 16:9)
14	480p 60Hz (Duplicación del píxel habilitado)
15	480p 60Hz (Duplicación del píxel habilitado) (Relación de aspecto 16:9)
16	1080p 60Hz
17	576p 50Hz
18	576p 50Hz (Relación de aspecto 16:9)
19	720p 50Hz
20	1080i 50Hz
21	576i 50Hz
22	576i 50Hz (Relación de aspecto 16:9)
23	288p 50Hz
24	288p 50Hz (Relación de aspecto 16:9)
25	576i 50Hz (Cuadruplicación del píxel habilitado)
26	576i 50Hz (Cuadruplicación del píxel habilitado) (Relación de aspecto 16:9)
27	288p 50Hz (Cuadruplicación del píxel habilitado)
28	288p 50Hz (Cuadruplicación del píxel habilitado) (Relación de aspecto 16:9)
29	576p 50Hz (Duplicación del píxel habilitado)
30	576p 50Hz (Duplicación del píxel habilitado) (Relación de aspecto 16:9)

31	1080p 50Hz
32	1080p 24Hz
33	1080p 25Hz
34	1080p 30Hz
35	480p 60Hz (Cuadruplicación del píxel habilitado)
36	480p 60Hz (Cuadruplicación del píxel habilitado) (Relación de aspecto 16:9)
37	576p 50Hz (Cuadruplicación del píxel habilitado)
38	576p 50Hz (Cuadruplicación del píxel habilitado) (Relación de aspecto 16:9)
39	1080i 50Hz (Reduced blanking)
40	1080i 100Hz
41	720p 100Hz
42	576p 100Hz
43	576p 100Hz (Relación de aspecto 16:9)
44	576i 100Hz
45	576i 100Hz (Relación de aspecto 16:9)
46	1080i 120Hz
47	720p 120Hz
48	480p 120Hz
49	480p 120Hz (Relación de aspecto 16:9)
50	480i 120Hz
51	480i 120Hz (Relación de aspecto 16:9)
52	576p 200Hz
53	576p 200Hz (Relación de aspecto 16:9)
54	576i 200Hz
55	576i 200Hz (Relación de aspecto 16:9)
56	480p 24Hz0
57	480p 24Hz0 (Relación de aspecto 16:9)
58	480i 240Hz
59	480i 240Hz (Relación de aspecto 16:9)

**Tabla C-2 Grupo HDMI 2 (DMT)**

Valor	Descripción
1	640×350 85Hz
2	640×400 85Hz
3	720×400 85Hz
4	640×480 60Hz
5	640×480 72Hz
6	640×480 75Hz
7	640×480 85Hz
8	800×600 56Hz
9	800×600 60Hz
10	800×600 72Hz
11	800×600 75Hz
12	800×600 85Hz
13	800×600 120Hz
14	848×480 60Hz
15	1024×768 43Hz, incompatible con la Raspberry Pi
16	1024×768 60Hz
17	1024×768 70Hz
18	1024×768 75Hz
19	1024×768 85Hz
20	1024×768 120Hz
21	1152×864 75Hz
22	1280×768 (Reduced blanking)
23	1280×768 60Hz
24	1280×768 75Hz
25	1280×768 85Hz
26	1280×768 120Hz (Reduced blanking)
27	1280×800 (Reduced blanking)
28	1280×800 60Hz
29	1280×800 75Hz
30	1280×800 85Hz
31	1280×800 120Hz (Reduced blanking)
32	1280×960 60Hz
33	1280×960 85Hz

34	1280×960 120Hz (Reduced blanking)
35	1280×1024 60Hz
36	1280×1024 75Hz
37	1280×1024 85Hz
38	1280×1024 120Hz (Reduced blanking)
39	1360×768 60Hz
40	1360×768 120Hz (Reduced blanking)
41	1400×1050 (Reduced blanking)
42	1400×1050 60Hz
43	1400×1050 75Hz
44	1400×1050 85Hz
45	1400×1050 120Hz (Reduced blanking)
46	1440×900 (Reduced blanking)
47	1440×900 60Hz
48	1440×900 75Hz
49	1440×900 85Hz
50	1440×900 120Hz (Reduced blanking)
51	1600×1200 60Hz
52	1600×1200 65Hz
53	1600×1200 70Hz
54	1600×1200 75Hz
55	1600×1200 85Hz
56	1600×1200 120Hz (Reduced blanking)
57	1680×1050 (Reduced blanking)
58	1680×1050 60Hz
59	1680×1050 75Hz
60	1680×1050 85Hz
61	1680×1050 120Hz (Reduced blanking)
62	1792×1344 60Hz
63	1792×1344 75Hz
64	1792×1344 120Hz (Reduced blanking)
65	1856×1392 60Hz
66	1856×1392 75Hz
67	1856×1392 120Hz (Reduced blanking)
68	1920×1200 (Reduced blanking)
69	1920×1200 60Hz
70	1920×1200 75Hz

71	1920×1200 85Hz
72	1920×1200 120Hz (Reduced blanking)
73	1920×1440 60Hz
74	1920×1440 75Hz
75	1920×1440 120Hz (Reduced blanking)
76	2560×1600 (Reduced blanking)
77	2560×1600 60Hz
78	2560×1600 75Hz
79	2560×1600 85Hz
80	2560×1600 120Hz (Reduced blanking)
81	1366×768 60Hz
82	1920×1080 (1080p) 60Hz
83	1600×900 (Reduced blanking)
84	2048×1152 (Reduced blanking)
85	1280×720 (720p) 60Hz
86	1366×768 (Reduced blanking)