



Recursosinformáticos

Raspberry Pi 3. Pi Zero

Explote todo el potencial
de su nano-ordenador

François MOCQ

Archivos complementarios
para descarga



Raspberry Pi

1. Introducción	23
2. Presentación	23
3. Historia de la creación de la Raspberry Pi	24
4. Cronología	26
5. Logo	31
6. El futuro de la Raspberry Pi	32

Descripción técnica

1. Introducción	35
2. La Raspberry Pi Zero	36
2.1 Presentación	36
2.2 Los componentes de la Raspberry Pi Zero	37
2.2.1 Vista superior	37
2.2.2 Vista inferior	38
2.2.3 Vista lateral	38
2.3 El SoC de la Raspberry Pi Zero	39
2.3.1 La CPU de la Raspberry Pi Zero	41
2.3.2 La GPU de la Raspberry Pi Zero	44
2.3.3 La memoria de la Raspberry Pi Zero	45
2.4 El puerto USB y de la Raspberry Pi Zero	46
2.5 Las salidas de vídeo de la Raspberry Pi Zero	47
2.5.1 Salida de vídeo analógica	48
2.5.2 Salida de vídeo HDMI (digital)	48

2.6 El bus CSI de la Raspberry Pi Zero	51
2.7 La GPIO de la Raspberry Pi Zero	53
2.8 El LED ACT de la Raspberry Pi Zero	57
2.9 La alimentación de la Raspberry Pi Zero	58
2.10 El conector de tarjeta micro SD de la Raspberry Pi Zero	59
2.11 El circuito ausente	61
2.12 Miniaturización de las tomas y consecuencias	61
2.13 Las dimensiones físicas de la Raspberry Pi Zero	62
2.14 Reloj	63
2.15 Conclusión	64
3. La Raspberry Pi 3	64
3.1 Presentación	65
3.2 Los componentes de la Raspberry Pi 3	65
3.2.1 Vista desde arriba	65
3.2.2 Vista inferior	66
3.3 El SoC de la Raspberry Pi 3	66
3.3.1 La CPU de la Raspberry Pi 3	67
3.3.2 La GPU de la Raspberry Pi 3	67
3.4 Los puertos USB y Ethernet de la Raspberry Pi 3	69
3.5 El Wi-Fi y el Bluetooth de la Raspberry Pi 3	72
3.6 Las salidas de vídeo de la Raspberry Pi 3	75
3.6.1 Vídeo digital de la Raspberry Pi 3	75
3.6.2 Vídeo analógico de la Raspberry Pi 3	76
3.7 Las salidas de audio	79
3.7.1 Audio analógico de la Raspberry Pi 3	79
3.7.2 Audio analógico de la Raspberry Pi 3	79
3.8 El bus CSI de la Raspberry Pi 3	81
3.9 El bus DSI de la Raspberry Pi 3	82
3.10 La GPIO de la Raspberry Pi 3	83
3.11 La alimentación de la Raspberry Pi 3	86
3.11.1 Por la toma de corriente	86
3.11.2 Por la GPIO	88
3.11.3 Por los puertos USB	90
3.12 Los LED de estado de la Raspberry Pi 3	90
3.13 El conector de tarjeta micro SD de la Raspberry Pi	91

3.14 Las dimensiones físicas de la Raspberry Pi 3	93
3.15 El reloj en tiempo real	93
4. Tabla comparativa	95

Sistemas operativos disponibles

1. Introducción	97
2. Linux	98
2.1 De Linus a Linux	98
2.2 El núcleo	100
2.3 Las distribuciones	102
3. Distribuciones Linux para la Raspberry Pi	104
3.1 Raspbian Jessie	105
3.2 Raspbian Jessie Lite	105
3.3 Raspbian "wheezy"	106
3.4 Arch Linux ARM	106
3.5 RISC OS	107
3.6 LibreELEC	107
3.7 OSMC	108
3.8 Ubuntu	108
3.9 Windows 10	109
3.10 PiNet	109
3.11 Otras distribuciones	109

Preparar la tarjeta micro SD

1. Introducción	111
2. Secuencia de boot de la Raspberry Pi	112

2.1 Etapa 1: enchufar a la corriente	112
2.2 Etapa 2: carga de bootcode.bin	114
2.3 Etapa 3: ejecución de bootcode.bin por la GPU	115
2.4 Etapa 4: ejecución de start.elf por la GPU	116
2.5 Etapa 5: ejecución de kernel.img por la CPU ARM	117
2.6 Etapa 6: inicio de Linux	118
2.7 Resumen de la secuencia de boot de la Raspberry Pi	120
3. Instalación del sistema operativo	120
3.1 Acceso a la tarjeta micro SD	121
3.2 Adquirir una tarjeta micro SD lista para usar	121
3.3 Instalación de un OS con NOOBS	122
3.3.1 El sitio de la Fundación	123
3.3.2 NOOBS	124
3.3.3 Descarga de NOOBS	125
3.3.4 Transferencia de NOOBS a la tarjeta micro SD	128
3.3.5 Instalación de Raspbian con NOOBS	135
3.3.6 Conclusión	144
3.4 Instalación de una imagen Raspbian	144
3.4.1 Transferencia de Raspbian a la tarjeta micro SD	147
3.4.2 En Windows 8	148
3.4.3 En Debian 8	152
3.4.4 Conclusión	160

Arrancar Raspbian

1. Preparación de la Raspberry Pi	161
1.1 Establecimiento de la tarjeta micro SD	162
1.2 Conexión de los cables	163
1.2.1 Raspberry Pi 3 modelo B	163
1.2.2 Raspberry Pi Zero	164
1.3 Enchufar a la corriente eléctrica	164
2. Configurar el sistema con raspi-config	165

2.1 Arrancar raspi-config	165
2.2 Usar raspi-config	166
2.2.1 Mover el cursor	166
2.2.2 Selección	166
2.2.3 Selección de las opciones	166
2.2.4 1 Expand Filesystem	167
2.2.5 2 Change User Password	168
2.2.6 3 Boot options	169
2.2.7 4 Wait for network at boot	169
2.2.8 4 Internationalisation Options	170
2.2.9 6 Enable Camera	173
2.2.10 7 Add to Rastrack	173
2.2.11 8 Overclock	174
2.2.12 9 Advanced Options	174
2.2.13 9 About raspi-config	182
2.3 Configuración avanzada de Raspbian	182
2.3.1 El archivo config.txt	182
2.3.2 Configuración del vídeo	192
2.3.3 Formatos soportados por un monitor	193
3. Añadir las licencias	196
4. Utilizar el sistema	199
Utilizar el modo gráfico	
1. Introducción	201
2. Descubriendo la pantalla de Raspbian	203
2.1 Iconos del escritorio de Jessie	203
2.2 La barra de tareas	204
2.2.1 Presentación	204
2.3 Configuración del escritorio en español	207
2.3.1 Mostrar el escritorio de Raspbian en español	207

2.3.2 Configuración de las opciones Locales	209
2.3.3 Configuración del huso horario	210
2.3.4 Configuración del teclado	211
2.3.5 Configuración del país para el Wi-Fi	211
2.3.6 Reinicio de la Raspberry Pi	211
2.4 Configuración de la barra de tareas	212
2.4.1 Barra de tareas (lista de ventanas) Settings	213
2.4.2 Agregar/Eliminar elementos al panel	214
2.4.3 Argumentos del panel	218
2.4.4 Crear un nuevo panel	219
2.4.5 Eliminar un panel	220
2.4.6 Acerca de	221
3. Los menús de Raspbian Jessie	221
3.1 Programación	222
3.1.1 BlueJ Java IDE	223
3.1.2 Geany	224
3.1.3 Greenfoot Java IDE	225
3.1.4 Mathematica	226
3.1.5 Node-RED	229
3.1.6 Python 2 (IDLE)	233
3.1.7 Python 3 (IDLE)	234
3.1.8 Scratch	234
3.1.9 Sense HAT	236
3.1.10 Sonic Pi	237
3.1.11 Wolfram	241
3.2 Herramientas de oficina	242
3.2.1 LibreOffice Base	243
3.2.2 LibreOffice Calc	243
3.2.3 LibreOffice Draw	244
3.2.4 LibreOffice Impress	245
3.2.5 LibreOffice Math	246
3.2.6 LibreOffice Writer	247
3.3 Internet	248
3.3.1 Claws Mail	249
3.3.2 Navegador web Chromium	257

3.3.3 Raspberry Pi Resources	263
3.3.4 The MagPi	264
3.3.5 VNC Address Book	266
3.3.6 VNC Viewer	268
3.4 Juegos	271
3.4.1 Minecraft Pi	271
3.4.2 Python Games	273
3.5 Accesorios	275
3.5.1 Calculator	275
3.5.2 Administrador de tareas	276
3.5.3 LXTerminal	277
3.5.4 Administrador de archivos	279
3.5.5 Xpdf	280
3.5.6 SD Card Copier	281
3.5.7 Editor de texto (Text Editor)	283
3.5.8 Visor de imágenes	284
3.5.9 Xarchiver	285
3.6 Help	291
3.6.1 Guía de referencia para Debian	291
3.6.2 Raspberry Pi Help	292
3.7 Preferencias	293
3.7.1 Add/Remove Software	294
3.7.2 Appearance Settings	295
3.7.3 Audio Device Settings	297
3.7.4 Teclado y ratón	300
3.7.5 Configuración de la Raspberry Pi	302
3.7.6 Main Menu Editor	308
3.8 Run	310
3.9 Shutdown	310
4. Iconos a la derecha de la barra de tareas	311
4.1 Bluetooth	312
4.1.1 El Bluetooth en la Raspberry Pi	312
4.1.2 Hacer visible a la Raspberry Pi	312
4.1.3 Agregar un dispositivo Bluetooth	313
4.1.4 Distribuir la música en un altavoz Bluetooth	315

4.2 Red	317
4.2.1 Visualización de la configuración de red	317
4.2.2 Configuración de la red	317
4.3 Volumen sonoro	321
4.4 Indicador de actividad	322
4.5 Reloj	322
4.6 Ejecución de periférico USB	325
4.6.1 Inserción de periféricos USB	325
4.6.2 Expulsar periférico USB	325
5. Conclusión	328

Usar la línea de comandos

1. Introducción	329
2. La arborescencia de Linux	332
3. La línea de comandos	333
3.1 Conexión a Raspbian	333
3.1.1 Conexión como usuario normal	334
3.1.2 Conexión como root	339
3.2 Moverse en la arborescencia	342
3.2.1 Identificar el directorio actual	342
3.2.2 Listar el contenido de un directorio	343
3.2.3 Cambiar de directorio	345
3.2.4 Ruta relativa y ruta absoluta	347
3.2.5 Modificar la arborescencia	348
3.3 Administrar los archivos	350
3.3.1 Copiar los archivos	351
3.3.2 Mover y renombrar los archivos	353
3.3.3 Eliminar archivos	354
3.3.4 Mostrar el contenido de un archivo	355
3.3.5 Modificar el contenido de un archivo	355

3.3.6 Comprimir y descomprimir un archivo	357
3.4 Acelerar la escritura de los comandos	359
3.4.1 Recordar los comandos anteriores	359
3.4.2 Función autocompletar	361
3.5 Administrar el sistema	364
3.5.1 Administrar los usuarios y los grupos	364
3.5.2 Administrar los repositorios	370
3.5.3 Tener el sistema actualizado	372
3.5.4 Instalar/eliminar un programa	374
3.5.5 Administrar los permisos	376
3.5.6 Conocer la ocupación de la tarjeta micro SD	384
3.5.7 Escribir un script shell	384
3.5.8 Planificar las tareas	388
3.5.9 Administrar los procesos	390
3.5.10 Configurar la fecha del sistema	395
4. Guardar su configuración	396
4.1 Hacer una copia de seguridad de la tarjeta micro SD en Windows	397
4.2 Hacer una copia de seguridad de la tarjeta micro SD en Debian 7	398
4.3 Copia de seguridad de la tarjeta SD en modo gráfico	399
5. Conclusión	399

Conectarse a la Raspberry Pi en remoto

1. Introducción	401
2. En modo texto	402
2.1 En SSH con putty en Windows	402
2.2 En SSH en Debian	405
3. En modo gráfico	406
3.1 Principio	406

3.2 Con VNC Server	407
3.3 Conectarse con VNC Viewer en Windows	410
3.4 Transferir archivos con VNC Viewer	413
3.4.1 Transferencia de archivos de la Raspberry Pi al PC	413
3.4.2 Transferencia de archivos del PC a la Raspberry Pi	416
3.5 Conectarse con VNC Viewer en Linux	418
3.6 Instalar TightVncServer en la Raspberry Pi	421
3.7 Arranque automático de TightVncServer	423
3.8 Conclusión	430

Gestión de la red

1. Conceptos básicos de red	431
1.1 Introducción	431
1.2 Como una carta en el correo	432
1.2.1 Asignación de direcciones	432
1.2.2 Dirección física	433
1.2.3 ¿Qué es una dirección IP?	434
1.2.4 Utilidad de la máscara de sub-red	435
1.2.5 Dirección del host	437
1.2.6 Las direcciones particulares	438
1.2.7 Las excepciones	439
1.2.8 Dirección pública	441
1.3 Llegada a buen puerto	441
1.3.1 Puertos TCP/IP	441
1.3.2 Utilización de los puertos	443
1.3.3 Socket TCP/IP	443
1.4 Roles de la unidad de conexión	444
1.4.1 Servidor DHCP	444
1.4.2 Nombre de dominio	446
1.4.3 DNS	447
1.4.4 Pasarela	449
1.4.5 Router NAT	452
1.5 Acceder a una Raspberry Pi desde Internet	454

1.5.1 DNS dinámico	455
1.5.2 Redirección de puerto	456
1.6 Box Internet	458
2. Gestionar la conexión de red alámbrica de la Raspberry Pi	459
2.1 Mostrar la información de la conexión	459
2.2 Modificar los argumentos de la red	460
2.2.1 Configuración automática por DHCP	461
2.2.2 Configurar una dirección estática	461
2.2.3 Reinicio de la red para tener en cuenta las modificaciones	462
3. Configurar una conexión Wi-Fi	462
3.1 Mostrar la información de la conexión	462
3.1.1 Comprobación de la detección de la llave USB	463
3.1.2 Modificación del archivo /etc/wpa_supplicant/wpa_supplicant.conf	463
3.1.3 Asignar una dirección fija a una llave USB Wi-Fi	464
Utilizar una memoria de almacenamiento externa	
1. Introducción	467
2. ¿ Por qué una memoria externa ?	468
2.1 Características de la tarjeta microSD	468
2.2 Comparativa de los dispositivos de almacenamiento	469
2.3 Elección de una memoria de almacenamiento	471
3. Discos y particiones en Linux	473
3.1 Identificación de los discos en Linux	473
3.2 Identificación de las particiones en Linux	473
3.3 Label y UUID	475
4. Definir un punto de conexión	481
4.1 La arborescencia en Linux	481

4.2 Crear un punto de conexión	482
5. Utilizar una memoria externa	483
5.1 Conexión manual de la llave USB	483
5.2 Desmontar una partición	484
5.3 Conexión automática al inicio	486
5.4 Uso de los UUID	488
6. Modificación de los permisos	489
6.1 Partición FAT o NTFS	489
6.2 Partición Linux	490
7. Conclusión	491

Arrancar sobre un disco externo

1. Introducción	493
2. Secuencia de boot de la Raspberry Pi	494
3. Preparación del disco	496
3.1 Eliminar la partición origen	496
3.1.1 Identificación de la partición a eliminar	496
3.1.2 Eliminación de la partición	499
3.2 Creación de dos particiones	500
3.3 Formateo de las particiones	503
3.3.1 Partición Linux	503
3.3.2 Partición FAT	503
4. Copia del sistema de archivos	504
5. Modificación de los archivos de inicio	506

5.1 cmdline.txt	506
5.2 fstab	507
6. Comprobación del funcionamiento	507
7. Ir más allá	508
8. Arrancar en una llave USB sin tarjeta micro SD	508
8.1 Principio	509
8.2 Actualización del sistema operativo	509
8.3 Autorización del boot USB	510
8.4 Elección de la llave USB	511
8.5 Preparación de la llave USB	512
8.6 Transferencia de Raspbian a la llave USB	514
8.7 Regeneración de las claves SSH	515
8.8 Modificación del archivo cmdline.txt	516
8.9 Modificación del archivo fstab	516
8.10 Desmontar el sistema de archivos	517
8.11 Encender la Raspberry Pi 3	517
8.12 Reinicio de la Raspberry Pi 3	517
9. Conclusión	518

¿Qué hacer con la Raspberry Pi?

1. Introducción	519
2. Un puesto de escritorio	520
2.1 Usar LibreOffice	520
2.2 Imprimir	521
3. Un centro multimedia	527
3.1 Instalar LibreElec	527

3.2 Configuración	532
3.3 Utilizar el centro multimedia	536
3.3.1 Escuchar la música en local	536
3.3.2 Ver una película en modo remoto	540
3.4 Utilizar el mando a distancia	542
4. Un servidor web	543
4.1 Instalar el servidor	543
4.1.1 Objetivo de la instalación	544
4.1.2 Instalación de Raspbian Lite	544
4.1.3 Modificación del nombre del host	544
4.2 Instalación de lighttpd	545
4.2.1 Modificación de los permisos sobre el sitio web	545
4.2.2 Verificación del servidor web	546
4.3 Instalación de PHP	547
4.3.1 Instalar PHP 5 en la Raspberry Pi	548
4.3.2 Indicar a lighttpd que debe usar PHP	548
4.3.3 Renombrar la página index.html por index.php	548
4.3.4 Comprobación de la instalación de PHP	549
4.4 Instalación de MariaDB	550
4.4.1 Instalar la base de datos MariaDB MySQL en la Raspberry Pi	551
4.4.2 Seguridad de MariaDB	551
4.4.3 Probar la instalación de MariaDB	553
4.4.4 Creación de la base de datos para WordPress	554
4.4.5 Asignación de los permisos en la base de datos	555
4.5 Instalación de WordPress	556
4.5.1 Descarga del archivo WordPress	556
4.5.2 Extracción de WordPress	557
4.5.3 Configuración de WordPress	559
4.6 Comprobación del sitio web	562
4.7 Poner en línea el sitio web	564
4.7.1 Nombre de dominio	564
4.7.2 Configuración del router	565
4.8 Conectarse al servidor web	566

5. Una cámara de vídeo vigilancia con Pi Zero	568
5.1 La distribución motionEyeOS	569
5.2 Instalación de motionEyeOS	569
5.3 Arrancar motionEyeOS	572
5.4 Conexión a motionEyeOS	572
5.5 Configuración de motionEyeOS	574
5.5.1 Cámara Ethernet	575
5.5.2 Configuración del vídeo	575
5.5.3 Ajuste de la detección de movimiento	576
5.5.4 Elección del nombre de los archivos guardados	578
5.5.5 Configuración de las notificaciones	579
5.5.6 Información superpuesta en las imágenes	582
5.5.7 Difusión de las imágenes	582
5.5.8 Registro de imágenes	584
5.6 Conclusión	585

Programar en Scratch

1. Presentación	587
1.1 Una larga historia	587
1.2 Los proyectos	588
1.3 Acceso al hardware	589
2. El entorno de Scratch	590
2.1 La interfaz gráfica	590
2.2 Los bloques de movimiento	598
2.3 Los bloques de apariencia	603
2.4 Las estructuras de control	607
2.5 Los sensores	610
2.6 Los sonidos	613
2.7 Los operadores	616
2.7.1 Operadores aritméticos	616
2.7.2 Operadores lógicos	617

2.8 El lápiz	620
2.9 Las variables	622
2.9.1 Creación de una variable	623
2.9.2 Creación de una lista	624
3. Crear un juego sencillo en Scratch	625
3.1 Definir el escenario	625
3.2 Definir las interacciones	626
3.3 Preparar la escena	627
3.4 Diseñar los duendes	628
3.4.1 Visor del aparato de fotos	629
3.4.2 El logo Raspberry Pi	630
3.4.3 El pingüino	632
3.5 Crear las variables	633
3.6 Animar el juego	633
3.6.1 Los scripts de la escena	634
3.6.2 Los scripts del gato	635
3.6.3 Los scripts del pingüino	637
3.6.4 El script del logo	638
3.6.5 Los scripts del visor	639
3.6.6 La evolución del juego	639
4. Conclusión	640
Programar en Python	
1. Historia de Python	641
2. Particularidades de Python	643
3. Python en modo consola	644
3.1 Inicio de Python	644
3.2 Utilizar Python como una calculadora	645

3.2.1 Operaciones básicas	645
3.2.2 Operadores de comparación	645
3.2.3 Operadores de asignación	646
3.2.4 Operadores sobre los bits	646
3.3 Escribir una cadena de caracteres	647
3.4 Las variables	647
3.5 Las listas	648
3.6 Crear un programa en Python	649
3.7 Ejecutar un programa	651
3.8 Utilizar el bucle for	652
3.9 Recuperar un valor de entrada por el teclado	653
3.10 Juegue usted mismo	654
4. Python en entorno gráfico	655
4.1 El entorno de desarrollo IDLE	655
4.1.1 Ajustar el tamaño de los caracteres	656
4.2 Utilizar el modo prompt	657
4.3 Crear un programa	657
4.4 Ejecutar un programa	658
5. Definir una función	659
5.1 Crear una función	660
5.2 Uso de la función	661
6. Utilizar los módulos	661
6.1 Los módulos internos	662
6.2 Crear un módulo	662
6.3 Utilizar un módulo en un script	664
6.4 Alias del nombre de módulo	665
7. Leer y escribir en un archivo	666
7.1 Creación de un archivo	666
7.2 Escritura en un archivo	667
7.3 Lectura de un archivo	668
7.4 Eliminación de un archivo	668

7.5 Gestión de excepciones	669
8. Ir más allá con Python	670

La GPIO de la Raspberry Pi

1. Descripción de la interfaz GPIO	671
1.1 Funciones disponibles	671
1.2 Valores límite (V e I)	673
1.2.1 Esquema de la interfaz GPIO	673
1.2.2 Umbrales de la señal de entrada	674
1.2.3 Histéresis	675
1.2.4 Limitación de la corriente de salida	675
1.2.5 Resistencia pull-up	675
1.2.6 Configuraciones peligrosas	676
2. Uso de la GPIO	677
2.1 Configuración utilizada	678
2.2 Encender un diodo LED	678
2.3 Leer un nivel de entrada	680
2.4 Administrar la GPIO por línea de comandos	681
2.4.1 Acceso a la GPIO	682
2.4.2 Creación de un acceso a una clavija GPIO	683
2.4.3 Eliminación del acceso a una clavija GPIO	683
2.4.4 Recuperar la información de una clavija de la GPIO	684
2.4.5 Parpadeo de un LED en shell	684
2.5 Administrar la GPIO en Python	685
2.5.1 Controlar un LED	686
2.5.2 Leer el nivel de una entrada	689
2.6 Acceder a la GPIO desde Scratch	693
2.6.1 Configurar los puertos de la GPIO	694
2.6.2 Utilizar una tarjeta adicional	698
2.6.3 Scratch y el aprendizaje del código	701
2.6.4 Conclusión	701

Los periféricos

1. El módulo cámara	703
1.1 Características del módulo Cámara V1.3	703
1.2 Características del módulo cámara V2	705
1.2.1 Una nueva tecnología de sensor	706
1.2.2 Tabla comparativa	708
1.3 Conexión a la Raspberry Pi	708
1.4 Uso de cámaras compatibles	710
1.5 Activación de la cámara	711
1.5.1 Activación en modo texto	711
1.5.2 Activación en modo gráfico	712
1.6 Captura de una imagen fija	712
1.7 Registro de un vídeo	713
1.8 Timelapse	713
2. Las tarjetas de extensión	715
2.1 Las tarjetas HAT	715
2.1.1 Presentación	715
2.1.2 Dimensiones	716
2.1.3 Funcionamiento	717
2.1.4 Conclusión	717
2.2 La tarjeta de sonido HiFiBerry DAC+	718
2.2.1 Presentación de la tarjeta HiFiBerry DAC+	718
2.2.2 Puesta en servicio de la tarjeta HiFiBerry DAC +	719
2.2.3 Uso de mpg123	721
2.3 La tarjeta ADC Pi Plus	723
2.3.1 Presentación de la tarjeta ADC Pi Plus	723
2.3.2 Las entradas analógicas	726
2.3.3 Instalación de la tarjeta	727
2.3.4 Prueba de la tarjeta	729
2.4 La tarjeta IO Pi Plus	731
2.4.1 Presentación de la tarjeta IO Pi Plus	731
2.4.2 Las entradas digitales	734
2.4.3 Instalación de la tarjeta	735

2.4.4 Pruebas de la tarjeta	736
2.5 La tarjeta RTC Pi Plus	737
2.5.1 Presentación de la tarjeta RTC Pi Plus	737
2.5.2 Puesta en hora de la Raspberry Pi	739
2.5.3 Instalación de la tarjeta	739
2.5.4 Configuración del reloj RTC	740
2.5.5 Uso del reloj RTC DS1307	741
2.5.6 Uso de varias tarjetas ABElectronics	742
2.6 PiFace Digital 2	742
2.6.1 Presentación de la tarjeta PiFace Digital 2	742
2.6.2 Esquema de la tarjeta	744
2.6.3 Conexión de la tarjeta	746
2.6.4 Configuración de la tarjeta	747
2.6.5 Comprobaciones de la tarjeta	748
2.6.6 Control de la tarjeta PiFace en Scratch	756
3. Utilizar tarjetas específicas	763
3.1 Tarjeta prototipada	763
3.2 Control del motor PAP	766
3.3 Control de un motor CC	773
Solución de problemas en la Raspberry Pi	
1. Fiabilidad de la Raspberry Pi	779
2. Los LED de la Raspberry 3	780
2.1 LED ACT de la Raspberry Pi 3	780
2.1.1 LED ACT apagado	780
2.1.2 LED ACT encendido de manera fija	780
2.1.3 LED ACT parpadeando cuatro veces	781
2.1.4 LED ACT parpadea siete veces	781
2.2 LED PWR de la Raspberry Pi 3	782
2.2.1 Parpadeo del LED PWR	782
2.2.2 LED PWR apagado	782

2.3 Indicador de tensión baja en la pantalla	783
2.4 Indicador de temperatura máxima en la pantalla	784
2.5 Medida de la tensión en la Raspberry Pi 3	785
2.6 LED del puerto Ethernet	786
3. El LED ACT de la Raspberry Pi Zero	788
3.1 Posición del LED ACT	788
3.2 Estados del LED ACT de la Raspberry Pi Zero	789
3.2.1 El LED de la Raspberry Pi Zero parpadea cuatro veces	789
3.2.2 El LED de la Raspberry Pi Zero parpadea siete veces	789
3.3 Medir la tensión de la Raspberry Pi Zero	789
4. La Raspberry Pi no arranca	791
5. Problema de conexión a la red	791
5.1 Un periférico USB deja colgada la red	792
5.2 Mover la tarjeta micro SD bloquea la red	792
5.3 Bloqueo de la red bajo una fuerte carga	793
6. Visualización anormal en la pantalla	794
6.1 Imagen rodeada de bordes negros	794
6.2 Imagen que se desborda de la pantalla	796
6.3 Puntos de colores e interferencias	796
6.4 No hay visualización en la pantalla	797
6.5 La resolución no es la adecuada	798
7. El sonido no funciona	798
7.1 Sin sonido en un monitor HDMI	798
7.2 Sin sonido completamente	799
8. El teclado es QWERTY	800
8.1 Teclados AZERTY y QWERTY	800
9. Monitorización de la Raspberry Pi	802

9.1 Instalación de EZ Server Monitor	802
10. Conclusión	804
 Anexo	
1. Respuestas al ejercicio del capítulo Arrancar sobre un disco externo	807
2. Respuestas a los ejercicios del capítulo Programar en Python	808
3. Respuestas al ejercicio del capítulo La GPIO de la Raspberry Pi	809
4. Sonic Pi	811
5. Tabla de configuración del vídeo	812
6. Creación de un nuevo cuadro de mandos	820
7. Controlar la tarjeta PiFace desde Scratch	822
 índice	 823

Raspberry Pi 3 o Pi Zero

Explote todo el potencial de su nano-ordenador

El objetivo de este libro es proporcionar al lector **principiante** conceptos sólidos para explorar los recursos que ofrece la **Raspberry Pi**, tanto desde el punto de vista del sistema operativo como del desarrollo y de la interfaz física. En este libro se tratan los modelos **Raspberry Pi 3** y **Raspberry Pi Zero**. No es necesario ningún requisito previo en Linux, programación o electrónica.

Después de una **presentación física** de la Raspberry Pi, se ofrece una visión sobre los **sistemas operativos** compatibles con este ordenador. Se le guiará en la **instalación rápida del sistema operativo** de su elección en una tarjeta micro SD, para que su Raspberry Pi sea operativa. Se explica de manera detallada el uso de **NOOBS**, herramienta de instalación de un sistema, recuperación de la tarjeta micro SD y gestión del multiboot. La presentación del nuevo escritorio gráfico de **Raspbian Jessie PIXEL**, continúa con el uso de Linux en **línea de comandos**. Después de una **iniciación en redes**, aprenderá cómo **conectarse en remoto** a la Raspberry Pi con **VNC**. Verá cómo **usar memorias de almacenamiento externas** (llave USB o disco duro USB) y **arrancar la Raspberry Pi sin tarjeta micro SD** en un soporte de almacenamiento externo. Aprenderá a **usar los entornos de desarrollo** disponibles para la Raspberry Pi: en **Scratch** y en **Python**. La **descripción del GPIO** continúa con ejemplos sobre el **uso de los puertos de entrada y salida** de la Raspberry Pi. La implementación de **tarjetas de interfaz**, abre la vía a aplicaciones en las que la Raspberry Pi interactúa con el mundo físico. Aprenderá cómo transformar su Raspberry Pi en un **puesto de trabajo** con la suite **LibreOffice** (edición e impresión), en media-center con **LibreELEC**, en servidor web con **lighttpd** y **WordPress** o en **cámara de video vigilancia**, capaz de **detectar un movimiento** y **avisarle por medio del correo electrónico**.

Para terminar, en el capítulo sobre la solución de problemas, descubrirá cómo **establecer un primer diagnóstico en caso de funcionamiento incorrecto**, así como las soluciones que se deben implementar.

Los elementos adicionales se pueden descargar en esta página.

Los capítulos del libro:

Prólogo – Raspberry Pi – Descripción técnica – Sistemas operativos disponibles – Preparar la tarjeta micro SD – Arrancar Raspbian – Utilizar el modo gráfico – Usar la línea de comandos – Conectarse a la Raspberry Pi en remoto – Gestión de la red – Utilizar una memoria de almacenamiento externa – Arrancar sobre un disco externo – ¿Qué hacer con la Raspberry Pi? – Programar en Scratch – Programar en Python – La GPIO de la Raspberry Pi – Los periféricos – Solución de problemas en la Raspberry Pi – Anexo

François MOCQ

Formador en redes y telecomunicaciones desde hace muchos años, **François MOCQ** siempre ha sido un apasionado de la electrónica. Desde su llegada al mercado, el potencial que ofrece Raspberry Pi le ha entusiasmado y, para compartir su pasión, rápidamente creó un blog y después un foro dedicado a este nano-ordenador (www.framboise314.fr). La escritura de este libro le permite compartir con el lector toda su experiencia sobre este producto, enriquecida con intercambios en la comunidad Raspberry.

Introducción

Diseñada con el objetivo de facilitar el aprendizaje de la programación informática, la Raspberry Pi ha tenido una acogida que ha sobrepasado ampliamente las expectativas de sus creadores.

Tras su aparición, el 28 de febrero de 2012, la Raspberry Pi vendió un millón de unidades el primer año. Las ventas alcanzaron los 2,3 millones de tarjetas a comienzos de 2014 y los 4 millones a primeros de 2015. La aparición de la Raspberry Pi 2 (el 2 de febrero de 2015) aceleró el movimiento con 500.000 unidades en los dos meses siguientes a su salida. En total, a finales de febrero del 2016, tras la aparición de la Raspberry Pi 3, son más de 8 millones de Raspberry Pi vendidas hasta hoy. Este ordenador, no más grande que una tarjeta de crédito y con un precio de entre 5 € y 40 € según la versión, ha permitido a un gran número de apasionados el acceso a un sistema informático.

La Raspberry Pi es capaz de mostrar vídeos en calidad Blu-ray full HD, servir como puesto de escritorio/Internet, de consola de videojuegos o de herramienta de iniciación a la informática. Gracias a sus puertos de hardware también puede recibir información de dispositivos de almacenamiento externos y controlar las acciones para manejar robots, vehículos, etc.

Este libro se dirige a un público variado, debido a que las aplicaciones de la Raspberry Pi son muy diversas y están en continua evolución:

- El estudiante que desee practicar la programación encontrará en los capítulos dedicados a Scratch y a Python las bases para empezar a escribir programas y adquirir las técnicas de desarrollo.
- El informático acostumbrado a los sistemas operativos propietarios descubrirá de manera rápida los diferentes sistemas libres disponibles en la Raspberry Pi, tanto por línea de comandos como en modo gráfico.
- El profesional audiovisual dispondrá de un sistema silencioso capaz de difundir los vídeos que podrá recuperar con Wi-Fi desde un servidor remoto en alta definición, 1080 líneas de exploración progresiva (1080p) a 30 imágenes por segundo.
- El profesional de la electrónica, el fabricante de aparatos de control remoto, de robots, etc., dispondrá con la Raspberry Pi de un sistema de cálculo potente, con poco consumo, que podrá adaptar a sus operaciones.
- El friki amante de las novedades tecnológicas y apasionado de la informática podrá iniciar rápidamente su Raspberry Pi y probar sus posibilidades.

El objetivo del libro es ofrecer al lector debutante unas bases sólidas para explorar los recursos que ofrece la Raspberry Pi, tanto desde el punto de vista del sistema operativo como del desarrollo y de la interfaz física, etc. No es necesario ningún conocimiento previo en Linux, programación o electrónica.

La Raspberry Pi 3 y la Raspberry Pi Zero se presentan en este libro. Después de una presentación física de la Raspberry Pi tendrá una visión general de los sistemas operativos compatibles con este ordenador. Se le guiará para instalar rápidamente el sistema operativo de su elección en una tarjeta micro SD y hacer que su Raspberry Pi quede operativa.

Se explica en detalle el uso de NOOBS, herramienta de instalación de un sistema, de recuperación de la tarjeta SD y de gestión del multiboot. La presentación del nuevo escritorio gráfico de Raspbian Jessie PIXEL es consecuencia del uso de Linux por línea de comandos.

Después de un inicio en red, aprenderá a conectarse en modo remoto a la Raspberry Pi con VNC. Verá cómo usar las memorias de almacenamiento externo (llaves USB o disco duro USB) e iniciar la Raspberry Pi sin tarjeta micro SD en un soporte de almacenamiento externo. Aprenderá a usar los entornos de desarrollo disponibles para la Raspberry Pi: en Scratch y en Python.

La descripción de la GPIO, seguida de ejemplos de uso de los puertos de entrada/salida de la Raspberry Pi y de la puesta en marcha de tarjetas de interfaz, abren la vía a aplicaciones en las que Raspberry Pi se integra con el mundo físico.

Aprenderá a transformar su Raspberry Pi en un puesto de escritorio con la suite LibreOffice (edición e impresión), en un media center con LibreELEC, en un servidor web con lighttpd y WordPress o en una cámara de vídeo-vigilancia capaz de detectar un movimiento y avisarle por correo electrónico.

Para terminar, en el capítulo de ayuda, descubrirá cómo establecer un primer diagnóstico en caso de funcionamiento erróneo, así como las soluciones que se deben poner en marcha.

Introducción

Para muchas personas, el fenómeno Raspberry Pi parece haber sido brusco. A finales del primer trimestre de 2012, aparece de repente la pequeña tarjeta en muchas revistas, blogs, etc. Sin embargo, el desarrollo de esta pequeña maravilla empezó mucho antes de su aparición. La Fundación Raspberry Pi ha trabajado tanto en la definición del producto como en su puesta en producción. Este capítulo resume el largo camino que va desde una idea a un éxito, a través de los valores defendidos por la Fundación y las etapas que se han sucedido en este recorrido.

Presentación

Imagine un ordenador del tamaño de una tarjeta de crédito que cuesta menos de 40 € y que es capaz de proporcionar vídeos en 1080p (*full HD* = alta definición) a 30 imágenes por segundo...

¿Un sueño? Hoy es una realidad gracias a la Raspberry Pi 3.

La Raspberry Pi 3 reúne en una tarjeta de 85 × 56 mm todos los componentes necesarios para que pueda funcionar un sistema operativo (Linux) y, de esta manera, poder usar la Raspberry Pi 3 como un verdadero PC.



Hay disponibles herramientas básicas en la mayor parte de los distribuidores: tratamiento de texto, hoja de cálculo, juegos o navegadores de Internet.

Los creadores de hardware como impresoras 3D, aparatos teledirigidos (vehículos, drones, helicópteros, etc.), encontrarán decenas de aplicaciones insólitas, útiles... incluso totalmente inútiles o absolutamente necesarias.

Pero lo que ha motivado la creación de la Raspberry Pi es, sobre todo, es una voluntad de poner a disposición de los niños y adolescentes un ordenador de bajo coste que les permita descubrir la informática y la programación.

Historia de la creación de la Raspberry Pi

La Fundación Raspberry Pi explica la génesis del proyecto Raspberry Pi en su sitio web (www.raspberrypi.org/about). El siguiente texto, traducido del sitio web, describe el nacimiento de la Raspberry Pi.

"En los años 90, los estudiantes británicos de "nivel A" (nivel similar al bachillerato en España), eran en su mayor parte apasionados de la informática. Muchos de ellos se pasaban horas programando su Amiga, BBC Micro, ZX Spectrum o Comodore 64. Llegaban a tener un nivel de programador amateur y pasaban sin problemas las pruebas.

Los estudiantes de los años 2000 no tienen el mismo perfil. Como mucho, alguno de ellos habrá construido algunos sitios web.

Se han identificado las causas de esta evolución: los cursos de informática se han transformado en la formación en aplicaciones de escritorio y la creación de páginas web. Los PC han sustituido a los antiguos micro-ordenadores, que toda una generación habían utilizado para formarse en programación. Estos PC, mucho más costosos y misteriosos que la generación anterior de "pequeños ordenadores", han hecho que los padres prohíban a los programadores principiantes "trastear" con el PC familiar. Este se usa para navegar en la web, jugar y más raramente (mucho más raramente) para gestionar la contabilidad familiar. Los jóvenes se han visto obligados a manejar las consolas de videojuegos. No han vuelto a tener acceso a la programación.

Un pequeño grupo de personas no puede hacer mucho contra los programas escolares inadaptados o los medios limitados por la crisis financiera. Pero hemos pensado que era necesario hacer algo contra la situación actual. Era necesario encontrar una plataforma que, al igual que los antiguos ordenadores, pudiera arrancar directamente en un entorno de programación. Para contrarrestar el hecho de que no se puede dejar a un joven experimentar sus programas en un PC, es necesaria una máquina con un precio tal que la noción de "riesgo" desaparezca.

De 2006 a 2008, Eben Upton ha construido varios prototipos de lo que se convertiría en la Raspberry Pi. A partir de 2008, los microprocesadores diseñados para usos móviles fueron más accesibles y lo suficientemente potentes como para ofrecer vídeo de excelente calidad. Pensamos que esta característica podría hacer que la tarjeta estuviera al alcance de los jóvenes, no obligatoriamente interesados por una herramienta destinada a la programación pura.

El proyecto parecía realizable. Eben Upton (hoy diseñador de componentes en Broadcom), Robert Mullins, Jack Lang y Alan Mycroft, junto con Pete Lomas, director de ingeniería en Norcott Technologies, así como David Braben, autor del juego Elite para la BBC Micro, crearon la Fundación Raspberry Pi para que el proyecto fuera una realidad.

Tres años más tarde empezó la fabricación en serie gracias a los acuerdos de fabricación bajo licencia, formados con Farnell element14 y RS Components. Y solo era el comienzo de la historia de la Raspberry Pi.

La comunidad educativa ha manifestado un interés y un apoyo enormes. Hemos estado encantados y sorprendidos de haber recibido peticiones de información de los usos de la Raspberry Pi muy por encima de nuestras expectativas iniciales. Los países en vías de desarrollo están interesados en el uso de la Raspberry Pi como herramienta de productividad en las zonas donde no disponen ni de la alimentación eléctrica ni del hardware necesarios, para mantener un PC. Hospitales y museos nos han contactado para obtener información acerca de la capacidad de la Raspberry Pi para controlar los dispositivos de visualización. Los padres de niños gravemente discapacitados piensan en aplicaciones de vigilancia y accesibilidad. Parece haber miles de personas, armadas con un soldador, que la esperan para hacer un robot.

Sin pretender tener todas las respuestas, ni creer que la Raspberry Pi va a dar una solución a todos los problemas informáticos del mundo, estimamos que puede ser un catalizador. Queremos ver por todos los lados ordenadores baratos, accesibles y programables. Animamos a otras empresas a copiar lo que hacemos. Queremos romper con la creencia de que es necesario gastar centenares de euros para usar Internet. Queremos tener un ordenador realmente personal y que se convierta en algo normal para los niños."

Cronología

De 2006 a 2008, Eben Upton realizó varios prototipos de lo se convertiría en la Raspberry Pi.

Después de 2008, se integra al proyecto un SoC (*System on a Chip* = circuito integrado que agrupa todos los componentes de un ordenador) y se definen las características definitivas de la Raspberry Pi.

Cincuenta prototipos de tarjetas alpha se habían construido en agosto de 2011. Sus características eran idénticas a las de la Raspberry Pi actual, salvo por su tamaño, más grande por la presencia de conectores previstos para la depuración. Hay vídeos de demostración que muestran LXDE (*Lightweight X11 Desktop Environment* = entorno de escritorio ligero en GNU/Linux), con el juego *Quake 3* en 1080p (agosto de 2011) y la reproducción de un vídeo con calidad HD desde la tarjeta SD (septiembre de 2011).

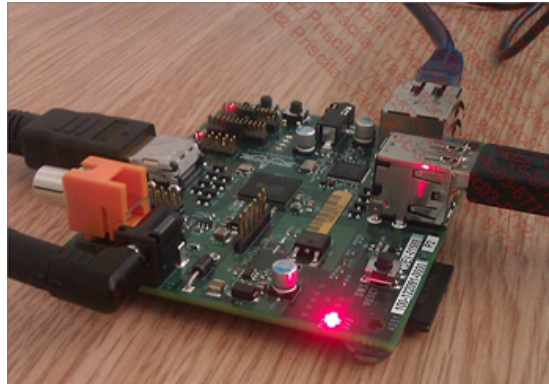


Foto Paul Downey - Licencia Creative Commons - CC-BY-2.0

La foto anterior muestra una tarjeta prototipo de la Raspberry Pi (2011). Los numerosos conectores utilizados para la depuración hacen que su tamaño sea superior al tamaño definitivo.

A finales de octubre de 2011, una versión de RISC OS (*Reduced Instruction Set Computer Operating System* - sistema operativo para microprocesador de juego de instrucciones reducido) funcionaba en la Raspberry Pi, y en diciembre se fabricaron 100 circuitos impresos del modelo B. Veinticinco de estas tarjetas modelo B se llenaron de componentes y se probaron. Los componentes eran idénticos a los previstos para la versión final. Solo se produjo un único error. Varias entradas del microprocesador que se tuvieron que mantener en alto, eran removibles. Los prototipos se corrigieron fácilmente y el error se eliminó cuando la tarjeta definitiva se puso en producción.

A finales de diciembre de 2011, estos prototipos de tarjetas demostraron su capacidad de arrancar en Linux, reproducir un vídeo en 1080p y ejecutar un software de prueba OpenGL, de la empresa Rightware.

Al inicio de 2012, diez de estos prototipos de tarjetas se subastaron en eBay. Se vendieron por un total de 16.000 £ (cerca de 18.600 €). La última tarjeta que se puso a la venta, el prototipo nº01, alcanzó 3.500 £, es decir, más de 4.000 €.

Las 10.000 primeras tarjetas no se fabricaron en Gran Bretaña, sino en China y Taiwán. El motivo fue la cantidad de impuestos de importación (que debían pagarse por cada componente individual y no por los productos terminados), además del plazo de cuatro semanas que necesitaban los fabricantes chinos, frente a las doce semanas de los británicos.

El anuncio de la comercialización de la Raspberry Pi se produjo a finales de febrero y provocó una saturación de los servidores de la Fundación, sobrecargados por los internautas que dejaban sus navegadores conectados a la página y refrescaban la conexión regularmente para ver la fecha exacta de comercialización.

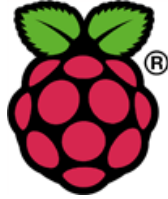
Las grandes etapas de los comienzos de Raspberry Pi se resumen a continuación:

29 de febrero de 2012	07:00h de la mañana (hora española), es decir, 6 años después del comienzo de la aventura, la Fundación indica que la Raspberry Pi ha empezado su comercialización. Los sitios web de las tiendas de Farnell element14 y RS componentes se saturan completamente. El primer lote se vende por Farnell en pocos minutos. RS registra más de 100.000 pedidos anticipados durante el día. Los pedidos se limitan a una tarjeta por persona. Al mismo tiempo, la Fundación revela que el modelo A, inicialmente previsto con 128 MB de memoria, tendrá finalmente 256 MB de memoria RAM.
Marzo de 2012	La Fundación informa que por motivo de la implantación en las tarjetas de un conector Ethernet sin transformador de aislamiento, la entrega del primer lote se retrasa, el puerto Ethernet desprovisto de este transformador integrado no funciona. Farnell y RS participan en la provisión de los componentes y el problema se resuelve rápidamente.
Mayo de 2012	Se han entregado 22.000 Raspberry Pi. A mediados de julio de 2012, se alcanza un ritmo de fabricación de 4.000 tarjetas cada día y se elimina la restricción sobre el número de tarjetas pedidas por persona.
Septiembre de 2012	La Fundación anuncia la aparición de un modelo B Revisión 2, con algunas mejoras y modificaciones menores. Liz Upton declara que la producción de las Raspberry Pi se va a traer a Gran Bretaña, a la fábrica Sony de Pencoed, en Gales. La producción se estima en 30.000 unidades cada mes, lo que permitirá la creación de 30 empleos. Ya se han vendido 500.000 Raspberry Pi.
Octubre de 2012	La Raspberry Pi modelo B tiene 512 MB de memoria, en lugar de 256 MB. El aumento de la producción y la bajada de precios de las memorias han permitido esta evolución, manteniendo un precio estable.
Diciembre de 2012	La Fundación pone a la venta las 12 primeras Raspberry Pi modelo A de preproducción, cuyas ganancias se destinan a fines caritativos.
Enero de 2013	La Fundación indica que se han vendido un millón de Raspberry Pi. Sin embargo, los clientes se quejan de los tiempos de espera, que oscilan entre los cuatro y los seis meses. Esto se debe a las dificultades de aprovisionamiento del microprocesador y las previsiones de venta inferiores a la realidad.
Febrero de 2013	La Fundación anuncia la disponibilidad del modelo A para Europa.
Noviembre de 2013	Liz Upton informa a los lectores del blog de la Fundación que se ha vendido la unidad número dos millones de la Raspberry Pi en la semana del 24 al 31 de octubre de 2013.
Julio de 2014	El 14 de julio de 2014, la Fundación anuncia la aparición de un modelo B mejorado, llamado B+. Tiene 4 puertos USB, se ha mejorado la posición de las tomas y la alimentación se ha rediseñado para mejorar el funcionamiento y reducir el consumo. La tarjeta SD del primer modelo deja paso a una microSD.
Noviembre de 2014	El 10 de noviembre de 2014, la Fundación anuncia la aparición del nuevo modelo A+. Se beneficia de la alimentación desarrollada para el modelo B+. Solo tiene un puerto USB y no está equipada de puerto Ethernet. Se revisa la implantación de los componentes y el modelo A+ tiene las dimensiones más reducidas en relación a todos

los modelos anteriores.

Finales de 2014	Se alcanzan los 4 millones de Raspberry Pi vendidas.
Febrero de 2015	El 2 de febrero de 2015, la Raspberry Pi 2 da la sorpresa. Conservando las mismas dimensiones y puertos que el modelo B+, tiene un procesador de 4 núcleos y 1 GB de memoria.
Abril de 2015	5 millones de Raspberry Pi se han vendido, de las cuales 500.000 se corresponden a la Raspberry Pi 2.
Noviembre de 2015	El 26 de noviembre de 2015, apareció la Raspberry Pi Zero, anunciada a un precio de venta de 5 \$ (cerca de 9 € en España). Mientras que la comunidad imaginaba una evolución hacia una tarjeta más potente todavía, la Fundación provocó de nuevo la sorpresa al sacar una Raspberry Pi de dimensiones reducidas (65mm x 30mm x 5mm) equipada con un procesador mono-núcleo. 10.000 unidades se entregaron de manera gratuita con el N° 40 de la revista The MagPi y otras 20.000 se vendieron en menos de 24 horas.
Enero de 2016	Teniendo en cuenta todos los modelos, se habían vendido más de 7 millones de Raspberry Pi.
Febrero de 2016	El 29 de febrero, fecha del cuarto aniversario de la aparición de la Raspberry Pi, la fundación saca al mercado la Raspberry Pi 3, dotada de un procesador de 64 bits con 4 núcleos y 1 GB de RAM. En ese momento hay 8 millones de Raspberry Pi vendidas; 6 millones fabricadas en Gran Bretaña.
Abril de 2016	El 25 de abril de 2016, se pone a la venta un nuevo modelo de cámara CSI para la Raspberry Pi. Tiene 8 megapíxeles en su versión normal e infrarroja. Está destinada a sustituir al primer modelo (5 megapíxeles), modelo que ya no está disponible.
Septiembre de 2016	La Fundación anuncia que la Raspberry Pi ha alcanzado los 10 millones de unidades.

Logo



Más allá de las consideraciones técnicas y financieras, la Fundación también se preocupa por encontrar una imagen para representarla. Sin ninguna duda, el éxito de la Raspberry Pi desde su aparición ha sido rotundo. Era necesario poder identificar visualmente el producto.

El 5 de agosto de 2011 arranca la competición, poniendo como restricción que el logo debería ser legible en un cuadrado de 1 cm por 1 cm, pero que su calidad también debería permitir usarlo en diferentes productos de tamaños diferentes. Para terminar, se debería identificar fácilmente tanto en color como en blanco y negro.

El 7 de octubre del mismo año, Liz Upton publicó el resultado en el sitio de la Fundación. El logo, diseñado por Paul Beech, había sido elegido por votación del jurado.

La elección de este logo no es casual: representa una frambuesa, pero en forma de molécula C_{60} , compuesta de átomos de carbono. Esta molécula de forma esférica (lo que recuerda al número Pi), se compone de 20 hexágonos y 12 pentágonos, es decir, 32 caras. Y el procesador de la Raspberry Pi es un procesador 32 bits. Además, de estas 32 caras, 11 son visibles en el logo. Y el procesador de la Raspberry Pi es un ARM... 11. Difícil encontrar más coincidencias.

El futuro de la Raspberry Pi

Con la Raspberry Pi, la Fundación ha demostrado que es posible construir un ordenador totalmente operativo, por una cantidad fijada desde el inicio en 35 \$. El entusiasmo provocado es el previsto y responde a una necesidad, a la que no habían sabido responder los Arduino, BeagleBoard y otras tarjetas.

La aparición de la Raspberry Pi ha "agitado" este pequeño mundo. Nuevas tarjetas con rendimientos mejorados y a precios competitivos vienen a perturbar los cimientos de la Raspberry Pi. Intel empezó a comercializar a finales de noviembre de 2013 una tarjeta llamada Galileo, equipada con un Pentium de bajo consumo (SoC Quark X 1000) y compatible con Arduino Uno R3, sus *shields* (tarjetas de extensión) y sus *sketchs* (programas). Esta tarjeta se vende por 55 € y se distribuyen gratuitamente 50.000 unidades a 1.000 universidades en todo el mundo.

Otras tarjetas derivadas de la Raspberry Pi, llamadas Banana Pi o incluso Orange Pi, dotadas de rendimientos más elevados (procesadores A20 o A31, 1 GB de RAM, socket SATA, reloj RTC...), se posicionan en el nicho de la Raspberry Pi. Su precio continúa siendo más elevado (cerca de los 50 € para las versiones básicas y la ausencia de una comunidad real es un freno a su expansión).

Adoptada por muchos directores de proyectos que la integran en sus desarrollos, la Raspberry Pi también ha provocado el nacimiento de un ecosistema formado por empresas que van a producir complementos (tarjetas, cámaras, etc) o que integrarán la tarjeta del nano-ordenador en sus productos (aparatos de fotos, servidores multimedia, gatekeeper SIP, etc.).

Aislada en su rango de precios y con una comunidad muy activa, la pequeña tarjeta de la Fundación continúa siendo atractiva, y el número de revistas que publican regularmente artículos de la Raspberry Pi es el esperado.

- El precio de la Raspberry Pi 3 se ha fijado en un máximo de 40 \$. En Europa se vende alrededor de los 35 €. En los siguientes capítulos se usan de manera indistinta ambas monedas.

Introducción

La Raspberry Pi es el resultado de elecciones técnicas y compromisos económicos. Este capítulo presenta en detalle cada parte de la tarjeta, sin cuestionar las elecciones realizadas por los diseñadores del producto. La Raspberry Pi es lo que es y el comprador de este ordenador lo utilizará con las características seleccionadas por sus creadores.

A pesar de su reducido tamaño, la Raspberry Pi tiene una gran cantidad de componentes. Las numerosas imágenes que ilustran este capítulo permiten observar las partes que se describen.

La Raspberry Pi Zero

1. Presentación



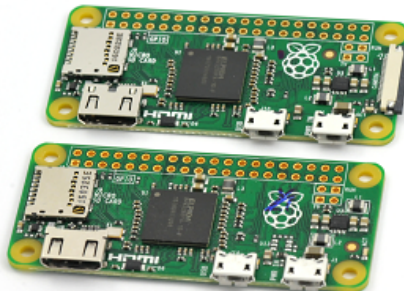
La Raspberry Pi Zero v1.2 ha generado sorpresa desde su salida. Con la donación a la fundación a través de un descuento por parte de Google de un millón de dólares para distribuir Raspberry Pi a los escolares, Eben Upton anunció a Eric Schmidt (PDG de Google) que la fundación trabajaba en la Pi 2, que sería mucho más potente, pero más cara. Eric Schmidt pensaba que era una tontería y le aconsejó que intentara reducir el precio. Finalmente, la Raspberry Pi 2 se vendió prácticamente al mismo precio que los primeros modelos, aunque esto orientó a la Fundación hacia un nuevo modelo basado en el precio más bajo posible. (<https://www.raspberrypi.org/maapi/making-pi-zero/>)

El precio de la Raspberry Pi Zero es la primera fuente de asombro, pues se ha anunciado a 5 \$ (se la encuentra por unos 9 € en España).

El tamaño de la Raspberry Pi Zero es la segunda fuente de asombro: 65 mm x 30 mm x 5mm. Esto permite incorporar la potencia de cálculo de la Raspberry Pi en muchos objetos (siguiente imagen).



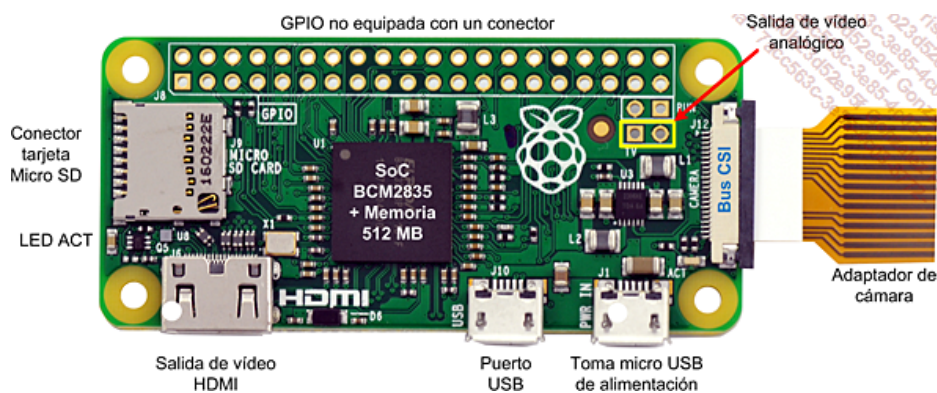
Casi seis meses después de la salida de la Raspberry Pi Zero, y después de que los usuarios trabajaran mucho con ella, la fundación sacó al mercado una Raspberry Pi Zero v1.3, idéntica en todo al modelo anterior, pero con un puerto C.S.I. que puede albergar una cámara, por el mismo precio (a continuación los dos modelos).



La demanda fue tal que en junio de 2016 resultaba realmente complicado hacerse con uno de estos reducidos ordenadores. Las tarjetas se compran una vez puestas a la venta y es realmente difícil hacerse con una.

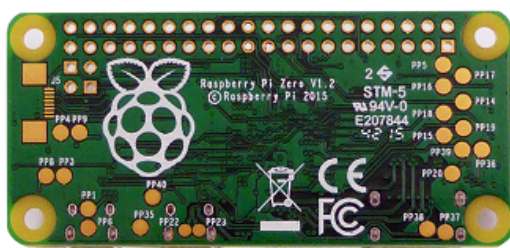
2. Los componentes de la Raspberry Pi Zero

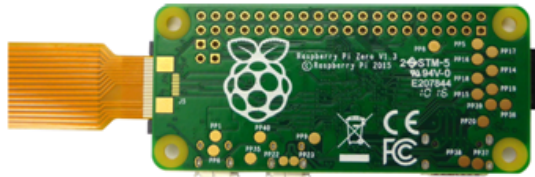
a. Vista superior



Los componentes principales de la Raspberry Pi Zero se pueden ver en la imagen anterior. Todos los componentes se montan en el mismo lado de la tarjeta de circuito impreso, con el objetivo de reducir la altura de la tarjeta.

b. Vista inferior





La Raspberry Pi Zero no tiene ningún componente por debajo de la tarjeta del circuito impreso. Los únicos elementos localizables son las pastillas de los puntos de comprobación identificados por las letras PP, utilizadas para las pruebas en producción y/o la búsqueda de errores.

Se prevé una ubicación J5 para una toma, situada en el extremo izquierdo de las imágenes anteriores, pero no está ocupada. Los contactos se usan para las pruebas en la fase de fabricación. Es un conector JTAG (*Joint Test Action Group*).

c. Vista lateral



En la imagen anterior se ve la primera versión de la Raspberry Pi Zero sin puerto para cámara CSI; en la siguiente, la versión con puerto CSI.



Solamente con 5 mm de espesor, la Raspberry Pi Zero se puede incorporar a muchos objetos o proyectos. Todos los conectores se ensamblan en el mismo lado (salvo el puerto de la cámara). Esto simplifica su embalado.

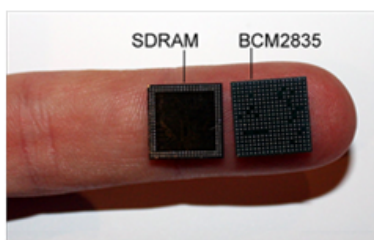
3. El SoC de la Raspberry Pi Zero

El SoC seleccionado para la Raspberry Pi es el BCM2835 de la empresa Broadcom. Este componente integra en un único chip todos los elementos necesarios para la realización de un ordenador: el microprocesador, el procesador gráfico y todos los circuitos anexos implementados en un sistema informático. El BCM2835 no es un SoC de última generación. Quizás estaba destinado a aplicaciones multimedia. Utiliza un microprocesador de una generación anterior, pero dispone de un procesador gráfico con un rendimiento muy elevado.

Aquí, como en cualquier otro producto industrial, la elección de cada componente es un compromiso entre rendimiento, espacio, precio, etc.

En el caso de la Raspberry Pi, la elección del BCM2835 ha estado guiada principalmente por el bajo precio de sus componentes, incluso aunque su rendimiento es peor que el de las nuevas generaciones de SoC disponibles. La restricción era que el precio de la tarjeta permaneciera inferior a 5 \$ para la Raspberry Pi Zero y a 35 \$ para la Raspberry Pi 3. Este compromiso se ha mantenido.

Es un sistema muy integrado, diseñado originalmente para utilizarse en dispositivos móviles y sistemas embebidos. Se pensó y optimizó para ahorrar energía y tiene un procesador para aplicaciones multimedia full HD (1080p a 30 imágenes por segundo).



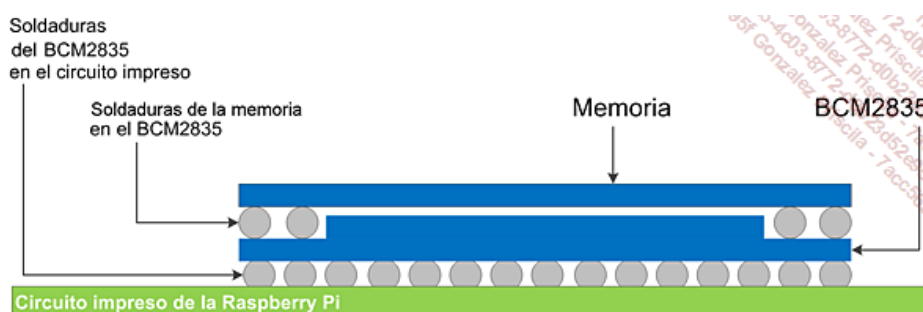
El SoC mide aproximadamente 12 mm x 12 mm (foto de James H. - con la autorización de [raspberrypi.org](http://www.raspberrypi.org)). Está conectado al circuito impreso por puntos de soldadura, dispuestos siguiendo un cuadrado de 18 por 18 contactos. Veinticinco puntos no están conectados, lo que deja 299 uniones por realizar. Broadcom no publica toda la información de este circuito. La familia BCM2708 incluye los procesadores BCM 2835, 2836 y 2837. Se puede descargar una nota técnica de más de 200 páginas del sitio web de la Fundación Raspberry Pi (<http://www.raspberrypi.org/wp-content/uploads/2012/02/BCM2835-ARM-Peripherals.pdf>). Corresponde a los periféricos accesibles por el microprocesador ARM (*Acorn Risc Machine convertido después en Advanced Risc Machine limited*). Los periféricos gestionados por el procesador gráfico no se documentan y Broadcom desaconseja su uso.

La información proporcionada en este documento está lo suficiente detallada, desde Broadcom, para permitir a un desarrollador adaptar un sistema operativo a la familia de SoC BCM2835, BCM2826 o BCM2837.

Están implicados los timers, controladores de interrupción, GPIO (*General Purpose Input/Output* = puertos de entrada/salida de uso general), puertos USB...

La ausencia de información de las conexiones del SoC está orientada, según la Fundación Raspberry Pi, a proteger las inversiones de las empresas que han arriesgado fondos para permitir la producción de la Raspberry Pi. La distribución de las clavijas "no oficiales" está en fase de desarrollo por parte del circuito impreso de la Raspberry Pi, según observaciones realizadas con rayos X...

El SoC BCM2835 no tiene memoria SDRAM (*Synchronous Dynamic Random Access Memory* = memoria dinámica síncrona de acceso aleatorio). Ya que el SoC está previsto para dispositivos móviles de pequeño tamaño y para reducir el espacio, esta SDRAM no está soldada al circuito impreso, sino al SoC directamente. Esta conexión POP (*Package-on-Package* = circuito sobre circuito) permite administrar de manera separada la memoria y el procesador. De esta manera es posible tener varios tamaños de memoria disponibles y varios proveedores, todo ello conservando un tamaño reducido, en particular para los teléfonos móviles. El siguiente esquema muestra la conexión POP del BCM2835 y de la memoria SDRAM de la Raspberry Pi.



Visto el tamaño del BCM2835 equipado con cerca de 300 contactos soldados en el circuito impreso, más la memoria que se suelda por encima con 168 puntos de soldadura, el montaje está totalmente automatizado y excluye cualquier intervención humana. Esto fija igualmente el tamaño de la memoria SDRAM disponible, porque está soldada en POP al SoC. Por tanto es imposible sustituir o extender la memoria de la

Raspberry Pi.

Esta configuración equipa todas las Raspberry Pi que salieron hasta febrero de 2015. A partir de esta fecha, la Raspberry Pi 2 lleva un SoC BCM2836 y la memoria está soldada debajo de la tarjeta. También será el caso para la Raspberry Pi 3, equipada con un SoC BCM2837. Con la Raspberry Pi Zero (noviembre de 2015), la Fundación vuelve a un SoC sobre el que está soldada una memoria SDRAM.

a. La CPU de la Raspberry Pi Zero

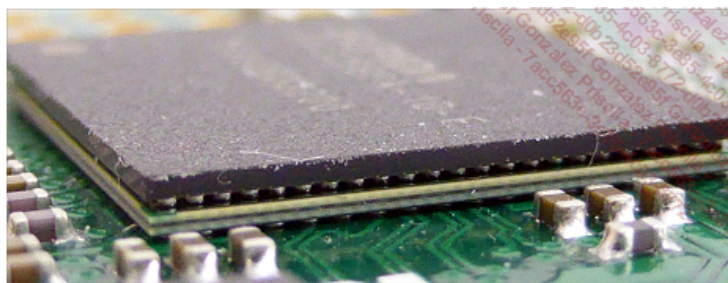
La CPU (*Central Processing Unit* = unidad central de procesamiento) es un sub-conjunto del SoC. Es un microprocesador de la familia ARM11, un ARM1176JZF-S que funciona originalmente con la frecuencia de 1 GHz. La familia de microprocesadores ARM11 agrupa los microprocesadores que usan una arquitectura RISC (*Reduced Instruction Set Computer* = microprocesador con un juego de instrucciones reducido) de 32 bits. La familia ARM11 es del 2002. El ARM1176 de la Raspberry Pi utiliza el juego de instrucciones ARMv6.

Esta CPU es un BCM2835 como el que equipaba las primeras Raspberry Pi. Esta CPU funcionaba a 700 MHz, pero podía aumentar su frecuencia con *overclocking*. Este aumento de velocidad y el excesivo calentamiento podían deteriorar la CPU rápidamente. La evolución de las técnicas de fabricación ha permitido proporcionar BCM2835 que funcionan de manera segura a 1 GHz. Este último modelo es el que equipa la Raspberry Pi Zero.

La empresa británica Acorn Computers está en el origen de estos microprocesadores RISC. En los años 80, Acorn fabricaba microordenadores, entre ellos el célebre BBC Micro. Entonces cambió para dedicarse a la fabricación de microprocesadores RISC, llamados ARM. ARM ya no fabrica circuitos, sino que vende licencias a los fabricantes que quieran implementar esta tecnología.

Encontramos este microprocesador o sus homólogos en una cantidad impresionante de smartphones, tabletas, libros electrónicos, TV digitales, consolas de videojuegos e incluso ordenadores portátiles. Por ejemplo, equipa a la primera generación del iPhone.

Debido a la gran difusión de esta gama de procesadores, muchos sistemas operativos gestionan los procesadores ARM: iOS, GNU/Linux, BlackBerry OS, ciertas versiones de Windows, Google Chrome...



La imagen anterior muestra el apilamiento del SoC BCM2835 y de la memoria en la tarjeta de la Raspberry Pi Zero. Se distinguen las soldaduras que se realizan a la conexión entre el SoC y el circuito impreso, así como las situadas entre la memoria y el SoC.

Overclocking

El *overclocking* es el aumento de la frecuencia de reloj de un microprocesador, más allá de su frecuencia normal de funcionamiento (fuente: wikipedia). Cuando se fabrica un circuito integrado, se le somete a una prueba para garantizar que funciona conforme a las especificaciones previstas. Entre estas pruebas figura la medición de la frecuencia máxima de funcionamiento del circuito, es decir, la frecuencia a partir de la cual se convierte en inestable.

En una misma serie, dos componentes que salen de la misma cadena de fabricación, con algunos minutos de diferencia, pueden tener frecuencias máximas de funcionamiento diferentes. Esto se debe a los defectos que existen en las placas de silicio utilizadas para la fabricación, o a ínfimas diferencias producidas durante los procesos de fabricación de los circuitos. Como consecuencia de estas pruebas, el fabricante certifica que su circuito funciona a una frecuencia determinada (por ejemplo, la Raspberry Pi Zero funciona a 1 GHz), mientras que la mayor parte de los circuitos funcionan en frecuencias superiores (1.1 GHz) y, en algunos casos concretos, en frecuencias muy superiores (incluso 1.2 GHz). Garantizando el funcionamiento a 1 GHz, el fabricante no toma ningún riesgo, pero es de prever que intente aumentar la frecuencia de reloj del circuito para mejorar el rendimiento. En general, la garantía del fabricante no cubre el *overclocking*.

Con un ajuste inicial a 1 GHz, la frecuencia de la CPU se puede aumentar ligeramente. Se monitoriza la temperatura de funcionamiento del SoC, y si alcanza los 85 °C, la frecuencia de funcionamiento desciende automáticamente para conservar el tiempo de vida del componente.

Como para cualquier *overclocking*, la estabilidad del sistema puede quedar comprometida, así como el tiempo de vida. Pero con una tarjeta de 35 €, el riesgo financiero es bajo en comparación a la mejora en el rendimiento, del orden del 30 % (en función de las pruebas realizadas).

Puede comprobar esta posibilidad añadiendo en el archivo config.txt. "arm_freq=1100", para intentar aumentar un poco la frecuencia de la Raspberry Pi Zero (por su cuenta y riesgo...).

b. La GPU de la Raspberry Pi Zero

La GPU (*Graphics Processing Unit* = unidad de procesamiento gráfico) de la Raspberry Pi es la Broadcom VideoCore IV @ 250 MHz. Ofrece soporte a varias versiones de OpenGL (*Open Graphics Library* = librería gráfica de código abierto). Se utiliza la norma OpenGL para el cálculo de imágenes 2D y 3D por muchas aplicaciones (CAO-DAO, juegos de vídeo, modelización 3D, etc.).

La aplicación envía a la GPU la información de los objetos a mostrar, es decir, las coordenadas de los puntos, vectores o polígonos, así como las texturas a aplicar a estos objetos.

Entonces OpenGL utiliza la potencia de cálculo de la GPU para mostrar la escena en la pantalla, teniendo en cuenta las distancias, los objetos ocultos por otros objetos, la luz y los reflejos, etc.

En febrero de 2016, se publicó una versión experimental de la OpenGL que utiliza la aceleración gráfica. Permite usar juegos o software como FreeCAD y OpenSCAD.

La GPU también es compatible con OpenMAX, que es una interfaz que permite a las aplicaciones utilizar de manera estandarizada los diferentes codecs de vídeo.

Es capaz de mostrar vídeo 1080p a 30 imágenes por segundo y decodificar el H.264 High Profile, que es la norma de los discos Blu-Ray y de la TV digital HD española.

La GPU también es capaz de decodificar el MPEG-2, utilizado por la TNT y el DVD. Puede tratar el formato Microsoft VC-1, junto al H.264, utilizado en los discos Blu-ray de Sony. Es posible utilizar estos dos codecs añadiendo una clave de licencia al sitio de la Fundación. Hay que proporcionar el número de serie del procesador de su Raspberry Pi, porque esta licencia está relacionada con el procesador de su máquina y, por tanto, no es transferible a otra Raspberry Pi. La licencia MPEG-2 cuesta alrededor de 3 €. La licencia VC-1 cuesta 1,50 € (en función del tipo de cambio de la libra).

También puede administrar una cámara HD en directo y codificar vídeo.

Originalmente, la GPU VideoCore no era libre, es decir, había información necesaria para su uso que Broadcom no difundía.

A finales de octubre de 2012, la Fundación Raspberry Pi puso en línea la información de la liberación de la pila gráfica de la GPU. De hecho solo la capa que permite hacer llamadas al programa binario (no libre) funcionará en la GPU publicada en ese momento.

El 28 de febrero de 2014, Eben Upton anunció la disponibilidad de la documentación de la GPU por Broadcom. Actualmente, toda la descripción de la GPU, así como el código fuente de los drivers OpenGL, son de libre distribución.

c. La memoria de la Raspberry Pi Zero

La memoria disponible está compartida entre la CPU y la GPU. Esta operación se llama *split* (división). No existe un valor por defecto para la compartición de la memoria entre la CPU y la GPU. Es la distribución Linux utilizada la que fija la cantidad de memoria asignada a cada procesador. Una distribución orientada al contenido multimedia favorecerá a la GPU asignándole, por ejemplo, la mitad de la memoria disponible. Una distribución clásica que utilice poca GPU reservará la parte más grande de la memoria para la CPU con el objetivo de favorecer la ejecución de los servicios del sistema.

Elpida, perteneciente a la estadounidense Micron, proporciona la memoria de la Raspberry Pi Zero. Es de tipo LPDDR2. Es una SDRAM (*Synchronous Dynamic Random Access Memory* = memoria dinámica síncrona de acceso aleatorio) DDR2 (*Double Data Rate*) y de bajo consumo (LP = low power), prevista para equipar a los smartphones.

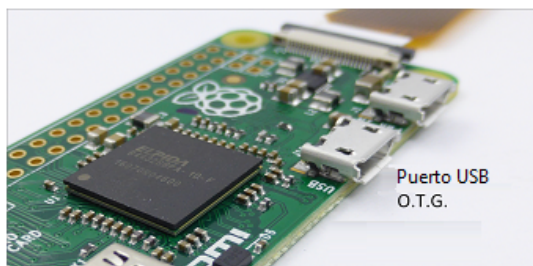


Las memorias de Elpida (imagen anterior) son del tipo B4432BBPA-1D-F. El número 44 presente en la referencia indica el tamaño de la memoria: 4 Gbits = 512 MB. El número 32 significa que la memoria está organizada en bloques de 32 bits. Son memorias del tipo DDR2 (*Double Data Rate* = frecuencia de transferencia doble), alimentadas con 1,8 V y 1,2 V y especialmente diseñadas para un bajo consumo. La documentación proporcionada por Elpida indica que se utilizan en una franja de temperaturas entre -25 °C a +85 °C, y que pueden trabajar a una frecuencia de 533 MHz con un ancho de banda de 1066 Mbit/s.

4. El puerto USB y de la Raspberry Pi Zero

Con el objetivo de reducir el tamaño de este modelo, la Fundación ha optado por tomas en miniatura en los puertos USB y HDMI.

Todos los modelos anteriores de Raspberry Pi estaban equipados con tomas USB de tipo A. Esto permite utilizar la misma conexión que en los ordenadores personales. En la Raspberry Pi Zero, la reducción de tamaño ha conducido a la selección de una toma micro USB para equipar al puerto USB.



El puerto USB es único y está directamente unido al SoC. Es un puerto USB On-The-Go (OTG). El BCM2835 que equipa la Raspberry Pi originalmente estaba destinado a utilizarse en los smartphones o tabletas. Estos disponen de un puerto USB único, que también permite la conexión a un PC que añade periféricos al smartphone o a la tableta.

La conexión al PC permite transferir archivos desde (o hacia) el smartphone. La conexión del periférico a un aparato móvil permite el uso de un verdadero teclado en una tableta, o la transferencia de archivos desde (o hacia) una llave USB o un disco USB externo.

En general cualquier periférico soportado por Linux se puede usar con la Raspberry Pi Zero. Entre todos los sistemas operativos, Linux tiene probablemente la cantidad más grande de controladores de hardware. En ocasiones puede haber cierto retraso hasta que se de soporte a un dispositivo reciente, pues es necesario escribir los controladores open source para Linux para que reconozca el dispositivo.

En modo gráfico, si desea utilizar un periférico con la Raspberry Pi, conéctelo. Hay muchas opciones de que funcione "sin más". Si se usa en una interfaz gráfica (como el entorno de escritorio LXDE en Raspbian) es probable que un icono o una ventana emergente le avise de que el nuevo periférico está listo para funcionar.

Puede tener problemas con antiguas webcam que usen USB 1.1, tarjetas un poco exóticas o hubs USB que usen la especificación USB Single-TT. TT significa Transaction Translator (Traductor de Transacción). Estos hubs usan un único TT para todos los puertos, al contrario que los hubs Multi-TT (un TT por puerto).

En caso de duda, consulte la lista de compatibilidad disponible en http://elinux.org/RPi_VerifiedPeripherals. Esta lista contiene el hardware verificado e indica si funcionan con la Raspberry Pi.

La Raspberry Pi Zero no tiene límite de corriente en el puerto USB. La tensión de 5 voltios disponible se envía directamente a la alimentación. Por tanto, tampoco hay protección. Si enchufa "en caliente" un periférico USB que consuma mucha corriente hay muchas posibilidades de que la Raspberry Pi Zero falle.

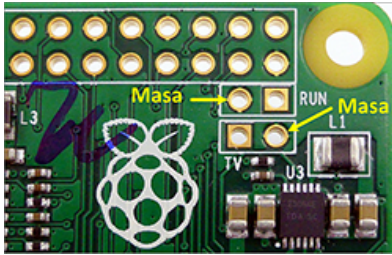
5. Las salidas de vídeo de la Raspberry Pi Zero

La Raspberry Pi Zero dispone de dos salidas de vídeo:

- Una salida de vídeo compuesto.
- Una salida HDMI (High Definition Multimedia Interface = interfaz multimedia de alta definición).

Estas dos salidas no pueden funcionar al mismo tiempo. Si se detecta una conexión HDMI, se activa la salida HDMI y la salida de vídeo compuesto se desactiva. Si la Raspberry Pi no detecta conexión HDMI, el vídeo se dirige a la salida compuesta (analógica).

a. Salida de vídeo analógica



Debido al tamaño reducido de la Raspberry Pi Zero, la Fundación ha renunciado a la toma jack 3,5 mm, que proporcionaba el vídeo compuesto en los demás modelos. La salida de vídeo compuesto de la Raspberry Pi Zero es accesible con dos pastillas de soldar.

Las pastillas están en el interior de un rectángulo xerografiado e identificado con **TV**. La masa está unida a la pastilla redonda. La señal de vídeo compuesto está disponible en la pastilla cuadrada. Es responsabilidad del usuario soldar directamente un cable a esta salida. Para obtener un resultado óptimo, se recomienda utilizar un cable coaxial para esta aplicación. La trenza del blindaje del cable coaxial se conecta a la masa, mientras que el interior se conecta a la salida de vídeo compuesto (pastilla cuadrada).

La salida de vídeo compuesto se utiliza para enviar la señal de vídeo saliente de la Raspberry Pi a una pantalla de TV analógica o a una pantalla digital que tenga este tipo de entrada.

b. Salida de vídeo HDMI (digital)

El conector HDMI (*High Definition Multimedia Interface* = interfaz multimedia de alta definición) es una interfaz audio/vídeo totalmente digital. En el 2005 sustituyó a la toma de Euroconector analógico en los equipamientos de gran consumo, capaz de mostrar vídeo HD.



El puerto HDMI es capaz de transportar vídeo de hasta 340 megapíxeles por segundo a través de varios canales digitales, así como 8 canales de audio digitales a 192 kHz.

Las señales de audio y vídeo se combinan en serie con banda ancha. La codificación utilizada minimiza las interferencias y permite el uso de cables largos y baratos. Estas características son importantes para el equipamiento de gran consumo.

Un argumento del archivo config.txt permite dirigir el vídeo a la salida HDMI o a la salida de vídeo compuesto. Como para el vídeo compuesto, es posible ajustar la definición de la visualización en HDMI. La gama de ajuste es mucho más amplia que para el vídeo compuesto, porque hay 60 modos diferentes, desde 640 x 350 píxeles hasta 2560 x 1600 píxeles, pasando por 1080p (1920 x 1080 píxeles) en full HD. También aquí es posible ajustar el tamaño de la imagen eliminando las bandas negras presentes alrededor de la imagen o devolviendo la visualización a la dimensión de la pantalla, si existen partes de la imagen invisibles que se extiendan más allá de los bordes de la pantalla.

Para los procesadores de pantalla provistos de una toma DVI (*Digital Video Interface* = interfaz de vídeo digital), pero que no tengan toma HDMI, observe que no hay diferencia entre las señales transportadas por una toma DVI y las transportadas por una toma HDMI. Ambas tomas, más allá de su diferente forma, son totalmente compatibles. Por tanto, para utilizar una Raspberry Pi en una pantalla provista de un conector DVI basta suficiente con usar un adaptador HDMI/DVI (siguiente imagen). El precio de estos adaptadores, compuestos solo de dos tomas y un cableado entre ellas, sigue siendo barato (unos pocos euros). Sin embargo, algunos cables adaptadores HDMI/DVI dorados, blindados, provistos de ferritas antiparásitos, son más caros que la propia Raspberry Pi...



También es posible convertir las señales HDMI en VGA (Video Graphics Array), pero el estándar VGA es un formato analógico mientras que HDMI es digital. Por tanto, hace falta un adaptador "activo".

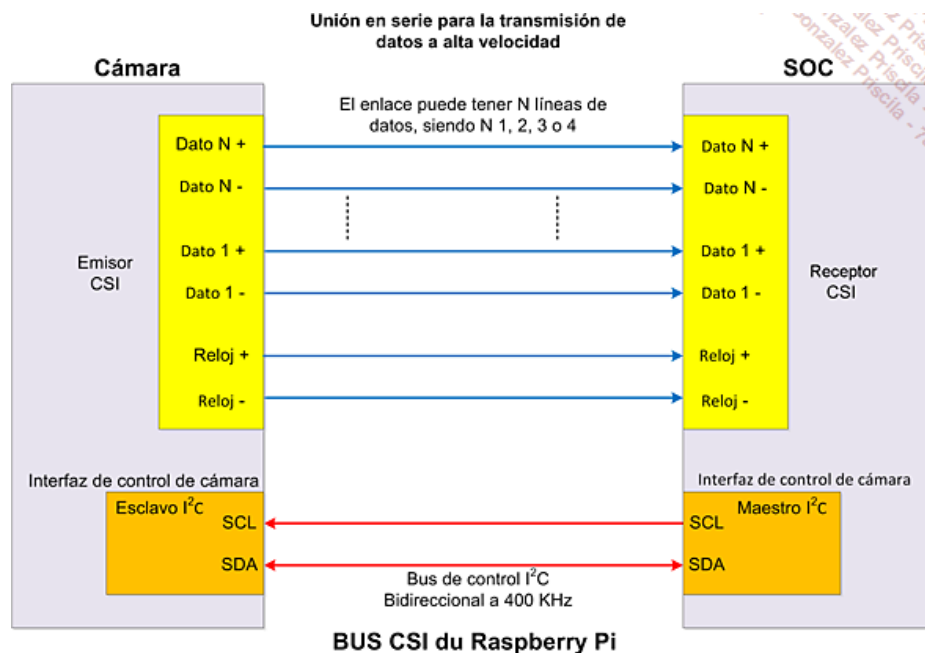


Este adaptador usa un circuito integrado encargado de convertir los flujos de datos digitales en señales adaptadas a la interfaz VGA. El circuito integrado trata y separa las señales roja/verde/azul, así como las señales de sincronización horizontales y verticales. Después las envía a una toma VGA de 15 clavijas.

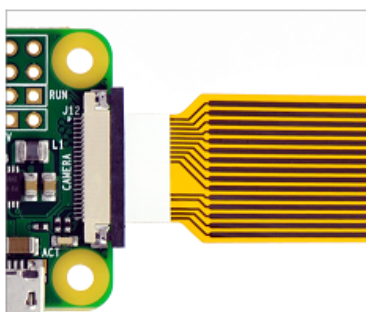
Algunos adaptadores, como el que se representa en la imagen anterior, también separan el sonido que envían a una toma jack 3,5 mm. Un adaptador de este tipo tiene circuitos electrónicos. Es más caro (de 25 a 75 €) que el convertidor HDMI a DVI. Por el contrario, permite reutilizar pantallas VGA antiguas, que no tienen toma DVI o HDMI. En la Raspberry Pi la corriente máxima permitida en la línea de 5 voltios es de 400 mA. Por encima de esta intensidad entra en acción un circuito de protección. El mismo circuito también sirve para desconectar la línea de 5 voltios HDMI de la alimentación de la tarjeta en caso de corto-circuito, evitando un reinicio de la Raspberry Pi en caso de sobre-consumo de corriente en la toma HDMI.

6. El bus CSI de la Raspberry Pi Zero

El bus CSI-2 (*Camera Serial Interface* = bus de serie para cámara) es un bus estándar, definido por la alianza MIPI (*Mobile Industry Processor Interface* = interfaz de procesador para la industria móvil), que agrupa a más de 200 empresas que intervienen en el dominio de los terminales móviles. MIPI define los buses que enlazan los diferentes componentes incorporados en los aparatos móviles.



El SoC de la Raspberry Pi nació de la industria del móvil y dispone de un bus CSI. Este bus puede recibir una capa de conexión hacia una cámara que responda a esta norma.



El bus CSI está situado en el extremo de la tarjeta, entre los agujeros de fijación (imagen anterior). Para alojar el conector en el sitio disponible, la Fundación ha seleccionado un conector más corto que en los anteriores modelos. El cable plano de la cámara de la Raspberry Pi ya no se adapta y es necesario utilizar un adaptador que adapte la salida de la cámara a la longitud del puerto CSI de la Raspberry Pi Zero.

El bus CSI tiene los siguientes elementos:

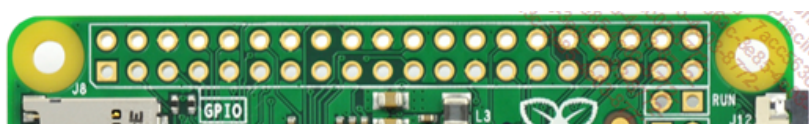
- Un bus de control estándar I2C (*Inter Integrated Circuit* = bus inter-circuitos integrados), que se usó en los años 1980 por Philips para la comunicación entre los microprocesadores y los circuitos anexos, que equipaban sus televisores, magnetoscopios, etc... Se convirtió en un estándar industrial para la comunicación entre los circuitos integrados. El bus I2C tiene dos señales: SDA (*Serial Data Line* = línea de datos en serie) y SCL (*Serial Clock Line* = línea de reloj en serie), suficientes para garantizar una comunicación bidireccional entre el maestro y el esclavo.
- Un bus de datos unidireccional en serie diferencial. Los datos solo van de la cámara al SoC. Cada línea de datos está formada por un par de cables que funcionan en diferencial (un cable positivo, el otro negativo) para reducir el efecto de las perturbaciones exteriores. Las uniones de datos pueden estar formadas por cuatro pares de cables, según el ancho de banda que la cámara debe enviar al procesador.



La presencia del conector CSI en la Raspberry Pi Zero abre la vía a aplicaciones de tamaño reducido, que usan una cámara. En la imagen anterior, una cámara PiNoIR infrarroja v2.1 (8 mega píxeles) está conectada a una Raspberry Pi Zero v 1.3.

7. La GPIO de la Raspberry Pi Zero

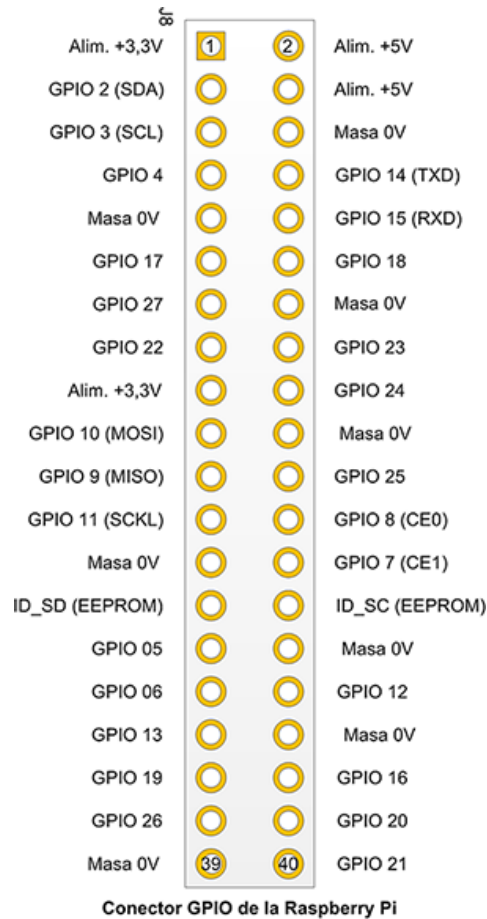
La GPIO (*General Purpose Input/Output* = entrada/salida de uso general) es una interfaz que permite conectar la Raspberry Pi Zero al mundo real. Es la ubicación identificada con J8 y GPIO en la siguiente imagen.



El puerto GPIO está compuesto por 40 pastillas repartidas en dos franjas de 20, separadas por 1/10e de pulgada (2,54 mm). El borne nº1 está identificado por una pastilla cuadrada. La Raspberry Pi se entrega sin conector GPIO soldado a la tarjeta. Para utilizar tarjetas HAT u otras extensiones habría que soldar un conector que tuviera dos franjas de 20 clavijas en la ubicación J8.

La GPIO está situada entre dos agujeros de fijación de la Raspberry Pi Zero del lado de los componentes, con el puerto GPIO en la parte superior. La clavija numerada como 1 es la más cercana a la marca J8. La clavija 2 se sitúa en el borde de la tarjeta, junto a la clavija 1. En el otro extremo de la franja de conectores, la clavija 39 está junto a la marca RUN. La clavija 40 está en el borde de la tarjeta. Todas las clavijas impares están situadas entre la clavija 1 y la clavija 39. Las clavijas pares están situadas a lo largo del borde de la tarjeta.

El siguiente esquema presenta el conector GPIO (J8), visto desde arriba.



Hay doce clavijas dedicadas a la alimentación. Las tensiones de alimentación de 3,3 y 5 V, así como la masa, están accesibles en las clavijas de la GPIO para alimentar los circuitos externos. De manera inversa, es posible alimentar la Raspberry Pi con 5 V a partir de las clavijas de la GPIO. La GPIO ofrece dos clavijas de 3,3 V, dos clavijas de 5 V y ocho clavijas de masa.

La siguiente tabla detalla el uso de las clavijas de la GPIO.

Clavija	Uso	Descripción
6-9-14-20-25-30-34-39	Alimentación	Masa
2-4	Alimentación	+ 5 V
1-17	Alimentación	+ 3,3 V
8-10	UART	Estas dos clavijas dan acceso a un UART (<i>Universal Asynchronous Receiver Transmitter</i> = emisor-receptor asíncrono universal), que podrá hablar con un terminal o cualquier otro dispositivo que tenga un puerto en serie. Estas clavijas son la 8: TXD (<i>Transmitted Data</i> = datos emitidos) y la 10: RXD (<i>Received Data</i> = datos recibidos).
19-21-23-24-26	Bus SPI	Estas cinco clavijas proporcionan un bus SPI (<i>Serial Peripheral Interface</i> = interfaz de periférico en serie). Es un bus de transmisión de datos en serie síncrono. Se utiliza para comunicar circuitos integrados. Las señales son SCLK (23) (<i>Serial Clock</i> = reloj en serie), MOSI (19) (<i>Master Out-Slave In</i> = emisión del maestro-recepción del esclavo), MISO (21) (<i>Master In-Slave Out</i> = recepción del maestro-emisión del esclavo) y dos señales CE0 (24) (<i>Chip Enable</i> = selección del circuito) y CE1 (26) que van a permitir seleccionar el esclavo con el que el maestro se va a comunicar. La Raspberry Pi solo funciona en modo maestro y puede gestionar de manera nativa dos esclavos. Es posible gestionar más esclavos que usen otros puertos GPIO para controlar las entradas CS de los circuitos utilizados.
3-5	Bus I2C	Estos dos contactos proporcionan un bus I2C. Se trata de SDA (3) (<i>Serial Data Line</i> = línea de datos en serie) y SCL (5) (<i>Serial Data Clock</i> = reloj de datos en serie). El bus I2C permite controlar numerosos circuitos integrados, compatibles con este tipo de bus.
27-28	I2C EEPROM	Estas dos clavijas no se deben utilizar. Se reservan para un diálogo con la EEPROM, presente en las tarjetas de extensión que responden a las especificaciones HAT (<i>Hardware Attached on Top</i> = hardware que viene en la parte superior de la Raspberry Pi).
7-11-12-13-15-16-18-22-29-31-32-33-35-36-37-38-40	GPIO	Estas diecisiete clavijas son entradas/salidas digitales.

Por supuesto, todas las clavijas de la GPIO se pueden reconfigurar como se desee. Solo las clavijas TXD y RXD (GPIO 14 y GPIO 15) se configuran durante el arranque para ofrecer las señales de UART. Según la necesidad, se podrán transformar en clavijas GPIO estándares (como las clavijas SPI e I2C) para ofrecer hasta 26 entradas/salidas GPIO.

Por el contrario, las clavijas 27 y 28 (ID_SD e ID_SC) se reservan solo para la lectura de las EEPROM presentes en las tarjetas de extensión. Cuando se arranca la Raspberry Pi, pregunta a esta interfaz I2C para encontrar una EEPROM que le permita identificar la tarjeta conectada a la GPIO. El contenido de esta EEPROM también define la configuración de los puertos de entrada/salida de la GPIO (y eventualmente, los controladores Linux a cargar). No hay que utilizar estas clavijas para otra cosa diferente a la unión con una EEPROM I2C. Deje estas clavijas no conectadas si no hay ninguna EEPROM.

Se pueden utilizar diferentes tipos de conexiones para unirse a la GPIO:

- cables soldados directamente a las pastillas de la GPIO,
- implantación de un conector de 2x40 clavijas, que habría que soldar a la tarjeta del circuito impreso.

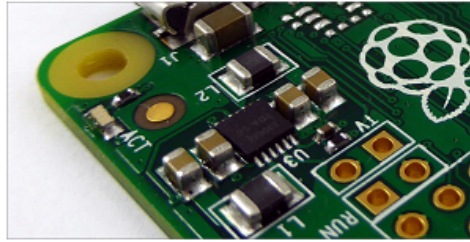
Es posible modificar el estado de las clavijas de la GPIO por línea de comandos, usando librerías como WiringPi (descargable en www.wiringpi.com), o incluso RPi.GPIO (<https://pypi.python.org/pypi/RPi.GPIO>), que proporciona una clase para controlar la GPIO de la

Raspberry Pi con Python (el capítulo La GPIO de la Raspberry Pi aporta más detalles acerca del uso de este puerto de entrada/salida).

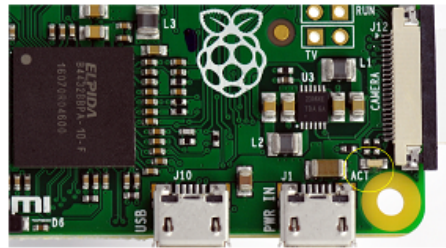
La GPIO no tiene entrada analógica aunque el SoC de un teléfono móvil no lo necesita realmente. Para acceder a medidas de valores analógicos habría que utilizar un convertidor analógico digital, un microcontrolador Arduino, y adquirir esta información en la Raspberry Pi.

➤ La GPIO es idéntica para la Raspberry Pi Zero y la Raspberry Pi 3.

8. El LED ACT de la Raspberry Pi Zero



En la Raspberry Pi Zero v1.2, el LED ACT está en el borde de la tarjeta. La versión v1.3 con el puerto CSI situado en el borde de la tarjeta hace que el LED ACT quede desplazado y menos visible (véase a continuación).



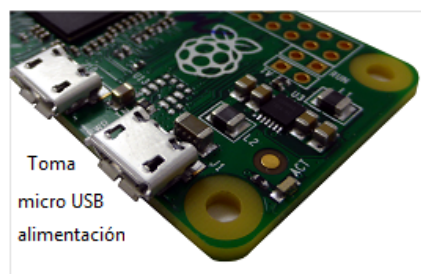
El número de diodos de LED también se ha reducido para ganar espacio. Actualmente solo hay un LED. El LED ACT indica la actividad de la Raspberry Pi, igual que en los primeros modelos. Esto quiere decir que durante el arranque de la Raspberry Pi Zero no se puede utilizar como indicador de encendido. Solo sucede cuando el sistema arranca, momento en el que se gestiona y se enciende para indicar la presencia de tensión. Se apaga (parpadea) para indicar un acceso a la tarjeta micro SD.

Si la Raspberry Pi Zero arranca sin tarjeta micro SD o con una tarjeta defectuosa o no conforme, el LED ACT no ofrece ninguna indicación. Permanece apagado.

9. La alimentación de la Raspberry Pi Zero

La Raspberry Pi Zero se alimenta a través de una toma micro USB, como todos los modelos de Raspberry Pi.

Junto a la toma J1 (PWR IN), se sitúa la alimentación de la Raspberry Pi Zero.



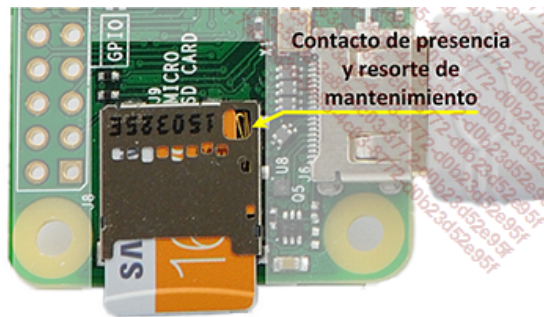
El núcleo de esta alimentación es el circuito U3, un PAM2306. Es un doble downconverter de conmutación, que acepta una tensión máxima de 5,5 voltios. Con una frecuencia de conmutación de 1,5 MHz y un rendimiento que puede alcanzar el 96 % esta alimentación, que solo necesita una decena de componentes, es ideal para el equipamiento embebido, en particular con una alimentación por baterías.



Las salidas de esta alimentación proporcionan tensiones de 3,3 v y de 1,8 v, utilizadas por el SoC. La tensión de 3,3 v también está disponible en la GPIO. Cada salida puede ofrecer 1 A y la alimentación está protegida contra los cortocircuitos y el sobrecalentamiento.

10. El conector de tarjeta micro SD de la Raspberry Pi Zero

El conector de la tarjeta micro SD de la Raspberry Pi Zero se sitúa en la parte superior de la tarjeta de circuito impreso. Esta elección favorece el bajo espesor global de la tarjeta del nano-ordenador.



La tarjeta micro SD se sitúa en el conector, con los contactos dirigidos hacia la tarjeta del circuito impreso. Por tanto, debe ver aparecer la marca de la tarjeta en la parte superior como en la imagen anterior, realizada con una tarjeta Samsung EVO de 16 GB.

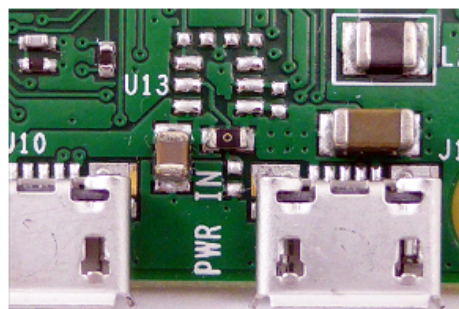
Con la Raspberry Pi Zero, la Fundación cambia el tipo de conector utilizado para recibir la tarjeta micro SD. En las versiones anteriores, el conector era un modelo llamado "push-push". Para situar la tarjeta micro SD, hay que empujarla en el conector hasta escuchar (y sentir) un clic. Para sacar la tarjeta, será necesario empujarla de nuevo en el conector, lo que provoca de nuevo un clic y se desbloqueará la tarjeta micro SD, haciéndola salir del conector. Algunas veces, se produce una verdadera expulsión de la tarjeta micro SD.

En la Raspberry Pi Zero el conector de la tarjeta micro SD es de tipo "push-pull". Es decir, para situar la tarjeta micro SD simplemente hay que empujar la tarjeta micro SD dentro del conector, hasta el fondo. En la Raspberry Pi Zero, no hay un clic que garantice la correcta ubicación de la tarjeta y su bloqueo. En la imagen anterior se ve que la sujeción de la tarjeta se realiza mediante la presión de las lengüetas, que se apoyan en los contactos de la tarjeta micro SD, así como por la presión lateral del contacto de presencia, que también hace el papel de resorte de sujeción.

- En caso de uso de una Raspberry Pi Zero en el equipamiento incorporado (coche, barco, dron, etc...) el usuario debería asegurarse de que las vibraciones que sufre la Raspberry Pi Zero no puedan provocar a la larga la expulsión de la tarjeta micro SD del conector.

11. El circuito ausente

Los participantes de los foros han disertado acerca del motivo de la presencia de una ubicación inutilizada marcada con U13 (siguiente imagen) en la versión 1.2 de la Raspberry Pi Zero. La conclusión de estas discusiones es que esta ubicación está prevista para un circuito de tipo AP255x, que la Fundación ya debería haber utilizado como limitador de corriente en los puertos USB de los anteriores modelos. Una resistencia de 0 ohmios (un corto-circuito) lleva los 5v de la alimentación directamente a la salida correspondiente de la toma USB, sin limitación de corriente.



El circuito AP255x solo cuesta algunas decenas de céntimos cuando se compran muchos. Puede que su montaje se haya abandonado para mantener el precio de la Raspberry Pi Zero por debajo de los 5 \$.

Esta ubicación vacía ha desaparecido de la versión 1.3 de la Raspberry Pi Zero. La adición del conector CSI destinado a enchufar una cámara ha desplazado los componentes y eliminado esta ubicación vacía.

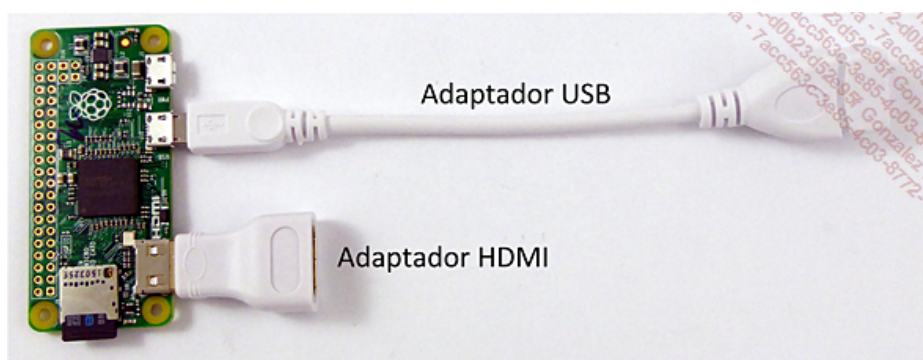
12. Miniaturización de las tomas y consecuencias

Con el objetivo de reducir el tamaño de este modelo, la Fundación ha optado por tomas en miniatura en los puertos USB y HDMI.

La toma de alimentación de 5v es un modelo micro USB, como en todos los modelos de la familia Raspberry Pi. Otra toma micro USB da acceso al puerto USB único de la Raspberry Pi Zero. Sustituye las tomas USB de tipo A utilizadas previamente.

- A pesar de su forma idéntica, estas dos tomas tienen un uso bien definido por la Fundación. Es imposible acceder al bus USB a través de la toma de alimentación. Por el contrario, en los modelos probados, la conexión de la Raspberry Pi mediante la única toma USB a un hub USB alimentado permite hacer funcionar la tarjeta ahorrando una toma.

La toma HDMI también se ha sustituido por un modelo en miniatura de tipo mini-HDMI.

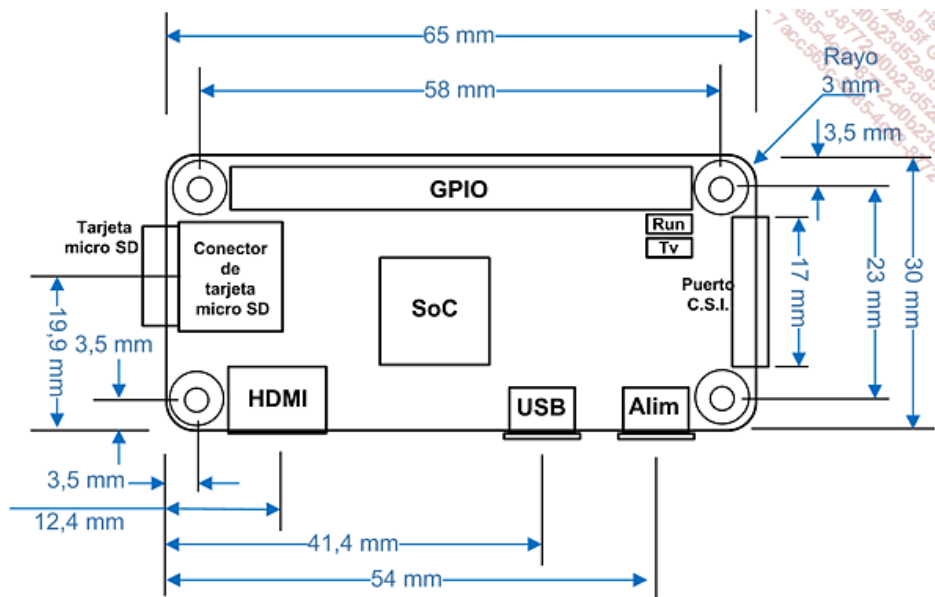


La consecuencia de estas modificaciones es la obligación de utilizar un juego de adaptadores mini-HDMI hacia HDMI y USB micro-B hacia USB A hembra, para conectar la tarjeta a los puertos estándares (imagen anterior).

El tamaño de la Raspberry Pi Zero la destina a aplicaciones en las que no estará unida a una pantalla HDMI o a un dispositivo USB. Será suficiente con tener un juego de adaptadores para preparar las Raspberry Pi Zero, antes de integrarlas al proyecto al que están destinadas.

13. Las dimensiones físicas de la Raspberry Pi Zero

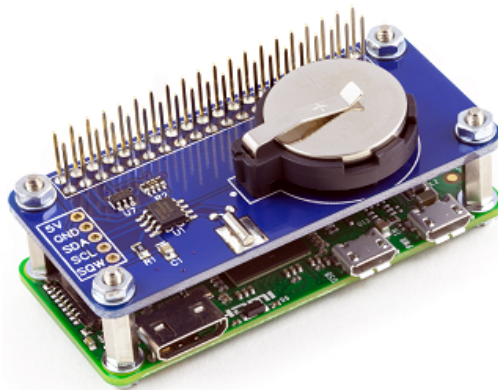
Igual que su precio, el tamaño de la Raspberry Pi Zero ha provocado asombro. Con unas dimensiones tan reducidas como 65 mm x 30 mm x 5 mm y su potencia de cálculo, la Raspberry Pi se puede incorporar fácilmente en muchos objetos.



El modelo que se ha presentado anteriormente, es la Raspberry Pi Zero v1.3. La versión anterior es idéntica pero no está equipada con el puerto C.S.I. situado a la derecha de la tarjeta.

14. Reloj

Como sucede con otras tarjetas Raspberry Pi, la Raspberry Pi Zero no tiene un reloj capaz de ofrecer una hora precisa. Esta función puede ser necesaria para algunas aplicaciones que necesiten un horario concreto de eventos.



El uso de una tarjeta de extensión (imagen anterior) puede aportar una respuesta a este problema, en detrimento del volumen y del peso del conjunto (ver el capítulo Los periféricos para más detalles).

15. Conclusión

La Raspberry Pi Zero es un concentrado de la Raspberry Pi. En un volumen mínimo, con un espesor reducido de 5 mm y un peso de solamente 9 gramos, este modelo ofrece el mismo rendimiento que la primera generación de Raspberry Pi. Ante todo, se destina a equipar proyectos embebidos u objetos de comunicaciones que es posible dotar de una sorprendente potencia de cálculo en un volumen reducido, con un consumo inferior a 0,8 watt.

La Raspberry Pi 3

La Raspberry Pi 3 modelo B marca una nueva evolución importante en la serie. De hecho, este modelo está equipado con un nuevo SoC de tipo BCM2837. Provisto de cuatro núcleos, el procesador de 64 bits de la Raspberry Pi 3 funciona a 1,2 GHz. Es el primer modelo de Raspberry Pi equipado de un procesador de 64 bits. Los sistemas operativos disponibles durante la salida de la Raspberry Pi 3 funcionaban con 32 bits, pero la posibilidad de utilizar sistemas de 64 bits abre nuevas posibilidades de desarrollo.

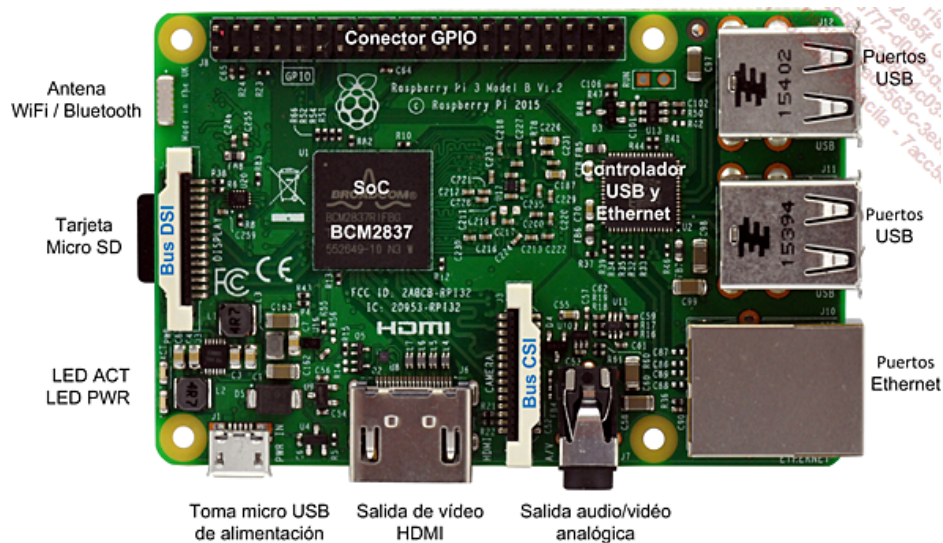
1. Presentación



El factor de forma es idéntico al del resto de modelos anteriores. La compatibilidad mecánica con los modelos B+ y Raspberry Pi 2 anteriores está garantizada. Todos los accesorios serán compatibles.

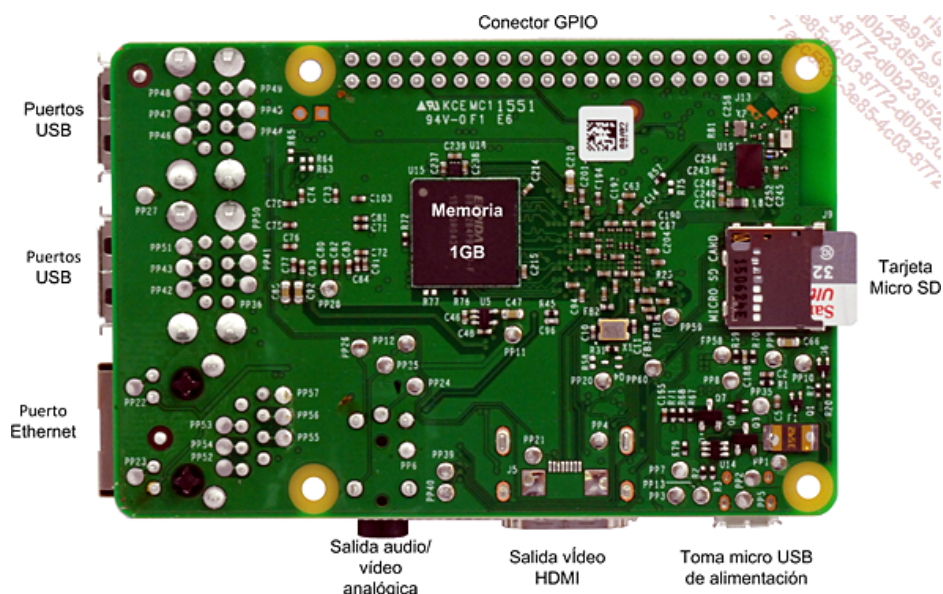
2. Los componentes de la Raspberry Pi 3

a. Vista desde arriba



Los componentes principales de la Raspberry Pi 3 están reunidos en la imagen anterior. Observe sin embargo que la adición de la Wi-Fi a la Raspberry Pi 3 ha implicado mover los LED de estado de un lado al otro del conector DSI. Los espacios cerrados opacos destinados a los antiguos modelos ya no permiten observar el estado de los LED.

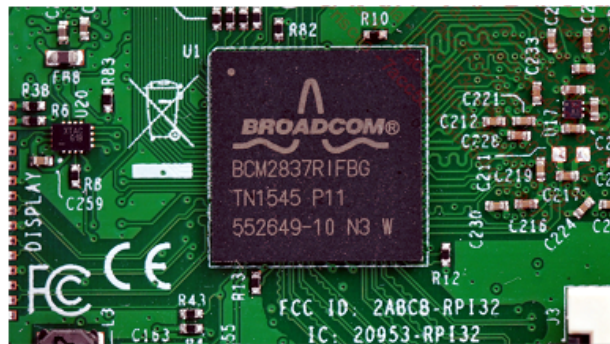
b. Vista inferior



Debajo de la tarjeta Raspberry Pi 3 los componentes principales son la memoria de 1 GB y el conector de tarjeta micro SD. Estos dos elementos tienen un espesor reducido, inferior a la longitud de las clavijas de la GPIO y de las diferentes tomas soldadas a la tarjeta. Por tanto, su presencia no molesta el empaquetado de la Raspberry Pi 3.

3. El SoC de la Raspberry Pi 3

La Raspberry Pi 3 está equipada con un SoC de la familia BCM2708, que integra todos los periféricos del SoC (bus SPI, I2C...). Este núcleo es la base de toda la gama de procesadores utilizados sucesivamente en los diferentes modelos de Raspberry Pi.



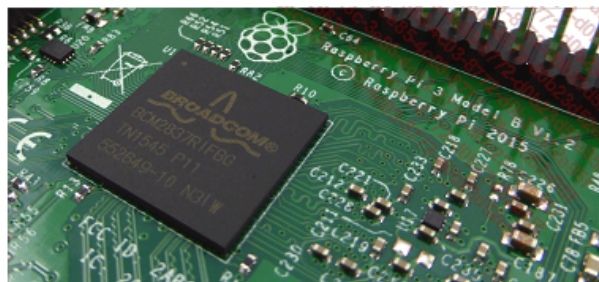
El SoC es un modelo Broadcom BCM2837 (imagen anterior).

a. La CPU de la Raspberry Pi 3

La CPU es un modelo de 64 bits, compatible con 32 bits, basado en una arquitectura ARMv8. Es un microprocesador de cuatro núcleos Cortex A53 que funcionan a 1.2GHz.

Con la introducción de esta CPU la Fundación anunció un rendimiento que duplicaba el de la Raspberry Pi 2 y que era diez veces mejor que el de la Raspberry Pi original. Las pruebas muestran que si se usa un único núcleo, la ganancia es cercana a 3 veces. Con el uso de los 4 núcleos es cuando la ganancia se multiplica por 10, incluso más. Sin embargo, es necesario que el sistema operativo y las aplicaciones gestionen el uso de los 4 núcleos.

b. La GPU de la Raspberry Pi 3



La GPU es un procesador gráfico VideoCore IV de doble núcleo que funciona a 400 MHz. Soporta OpenGL ES 2.0 (*Open Graphics Library* = Librería gráfica abierta), que es una interfaz que permite a los programas generar objetos en 2D o 3D, en forma de puntos, vectores o polígonos. OpenGL se encarga de calcular la imagen que se debe mostrar en la pantalla, teniendo en cuenta el punto de referencia seleccionado, la claridad, textura, reflejos, objetos ocultos por otros, etc. Muchas aplicaciones usan la norma OpenGL para el cálculo de imágenes 2D y 3D (CAO-DAO, videojuegos, modelización 3D...).

VideoCore IV también soporta la aceleración de hardware OpenVG. Diseñada originalmente para los dispositivos móviles, OpenVG deja a la GPU los cálculos necesarios para la visualización de los diseños vectoriales 2D. Esto permite crear interfaces en formato SVG (*Scalable Vector Graphics* = Gráficos vectoriales redimensionables), que permiten redimensionar una imagen o una tarjeta, conservando la calidad original, a diferencia de las imágenes bitmap, formadas por puntos (siguiente ejemplo *Bitmap_VS_SVG.svg*: Yug, CC BY-SA 2.5).



La GPU de la Raspberry Pi 3 también decodifica el 1080p30 H.264 high-profile. High Profile (HiP) es el perfil utilizado para la difusión de la televisión HD y para el almacenamiento de vídeos en Blu-ray. También puede mostrar el 1080p H.264 a 60 fps (*frames per second* = frames por segundo) usando una decodificación por software.

El reloj de la GPU VideoCore IV trabaja a 400 MHz y el nodo de cálculo 3D funciona a 300 MHz. Las versiones anteriores lo hacían a 250 MHz. Por tanto, esta nueva versión de la Raspberry Pi implica una aceleración en el tratamiento de imágenes, que hace más fluida la difusión de los vídeos. La memoria de la Raspberry Pi 3 funciona a 900 MHz, y también participa en la mejora del rendimiento.

La GPU también es capaz de decodificar el formato MPEG-2, utilizado por la TDT y el DVD. Puede tratar el formato Microsoft VC-1, competencia del H.264, utilizado en los discos Blu-ray de Sony. El uso de estos dos codecs se puede hacer comprando una licencia en el sitio de la Fundación. Hay que introducir el número de serie del procesador de su Raspberry Pi, porque esta licencia está relacionada con el procesador de su máquina, de modo que no es transferible a otra Raspberry Pi. La licencia MPEG-2 cuesta cerca de 3 €. La licencia VC-1 cuesta 1,50 € (junio de 2016 - variable en función de la evolución de la libra).

También puede gestionar una cámara HD en directo y codificar vídeo. En el mercado hay tarjetas que convierten una señal HDMI en CSI, para guardar un vídeo o mostrarlo en la pantalla de la Raspberry Pi.

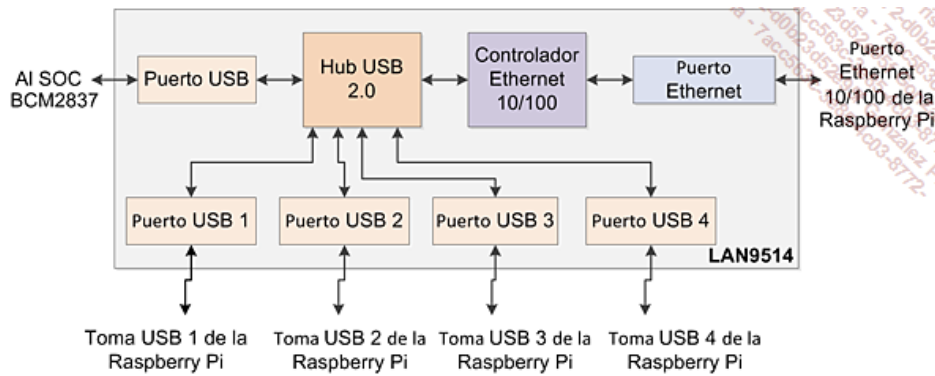
Originalmente, la GPU VideoCore no era libre, es decir, que la información necesaria para su uso no se difundía por Broadcom.

A finales de octubre de 2012, la Fundación Raspberry Pi pone en línea información correspondiente a la liberación de la pila gráfica de la GPU. De hecho, en ese momento solo se publicó la capa que permite llamar al programa binario (no libre) que funciona en la GPU.

El 28 de febrero de 2014, Eben Upton anunció la disponibilidad de la documentación de la GPU por parte de Broadcom. Se ha liberado la descripción completa de la GPU, así como las fuentes de los drivers OpenGL.

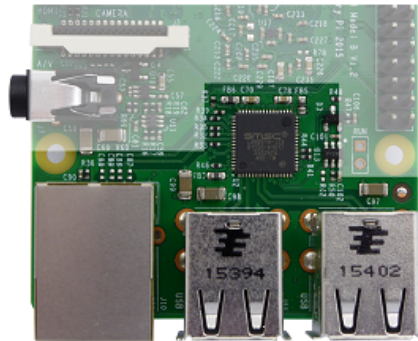
4. Los puertos USB y Ethernet de la Raspberry Pi 3

En la Raspberry Pi 3, el puerto USB del SoC BCM2837 está conectado a un circuito LAN9514 del fabricante SMSC en lugar de estar conectado directamente a la toma USB como en la Raspberry Pi Zero.



Este circuito integrado LAN9514 encierra un hub de 6 puertos USB 2.0. Cuatro de los puertos USB están conectados físicamente a dos tomas USB dobles de la Raspberry Pi. Un quinto puerto está conectado en modo interno a un controlador Ethernet 10/100. El sexto puerto está conectado al SoC BCM2837. El resumen anterior presenta la organización de la LAN9514.

Incluso aquí, se puede discutir la opción de este componente, en el que la unión Ethernet pasa por un puerto USB de la BCM2837, porque el tráfico de red y la carga de los puertos USB van a tomar la misma ruta hacia el SoC y causar eventuales cuellos de botella. De nuevo, es una opción barata que ha permitido, aceptando una posible degradación del rendimiento, mantener el precio de la tarjeta por debajo de los 35 €.



En la imagen anterior figuran, de abajo hacia arriba y de izquierda a derecha: la toma Ethernet y los cuatro puertos USB, el circuito LAN9514 y los componentes utilizados para administrar la corriente necesaria para los puertos USB. Los LEDs de estado de la LAN9514 (Link y 100 Mb/s) se sitúan sobre el conector Ethernet (siguiente imagen). El LED amarillo situado en la parte inferior izquierda del conector Ethernet indica la velocidad de transmisión y se enciende para una velocidad de 100 Mb/s. El LED verde, en la parte inferior derecha del conector Ethernet, se ilumina cuando el enlace físico es estable y parpadea cuando hay actividad en la red.



El controlador Ethernet de la LAN9514 ofrece una dirección MAC (*Media Access Control*) y una conexión que cumplen las normas IEEE 802.3 en 10BASE-T (10 Mbit/s) y 802.3u en 100BASE-T (100 Mbit/s). La norma 802.3 se conoce generalmente con el nombre de Ethernet.

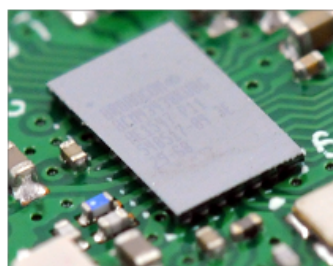
El puerto Ethernet gestiona la auto negociación, la configuración half o full-duplex y la auto-MDIX (*Medium Dependent Interface Crossover*). En otras palabras, esto significa que es capaz de detectar solo la configuración del enlace de red que se le conecta y se adapta rápidamente (normalmente en menos de 0,5 segundos). Esto evita que el usuario tenga que administrar el tipo de unión de red que se debe establecer (switch, PC, router...) y adaptar el cable Ethernet en consecuencia (cable recto o cruzado).

Se soportan muchos métodos de gestión del despertador, como los *Magic Packets* (paquetes mágicos), *Wake on LAN* (despertador en red) y *Link Status Change* (cambio de estado de la conexión). Esto puede permitir despertar una Raspberry Pi desde otra máquina para que ejecute tareas que no necesitan un funcionamiento permanente de la tarjeta, o cuando la alimentación de la Raspberry Pi se realiza con baterías o con paneles solares.

La documentación del fabricante indica que el consumo de la LAN9514 es de solo 180 mA cuando la interfaz Ethernet transfiere los datos en 10BASE-T, frente a 288 mA en 100BASE-T. Esto puede tener importancia para las aplicaciones embebidas, donde el consumo de energía es un elemento crítico.

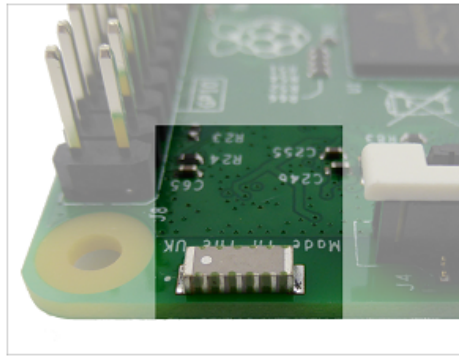
5. El Wi-Fi y el Bluetooth de la Raspberry Pi 3

Con la adopción de un microprocesador de 64 bits, la adición de la conectividad sin cables es la otra gran novedad de la Raspberry Pi 3.



El circuito seleccionado por la Fundación para asegurar esta función es el Broadcom BCM43438 (imagen anterior). Este circuito, que mide menos de 5 mm x 3 mm, está implementado en la tarjeta. Es un circuito destinado a los dispositivos móviles, poco consumidores de energía. Ofrece conectividad Wi-Fi 802.11 b/g/n hasta 150 Mbps, así como Bluetooth 4.1 LE.

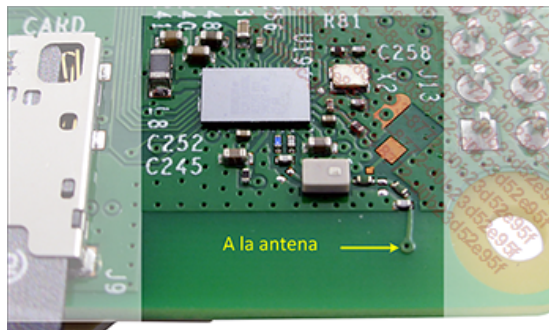
El circuito funciona a 2,4 GHz. La Fundación ha implementado una antena cerámica de 2,4 GHz, compartida entre el Wi-Fi y el Bluetooth (siguiente imagen).



Esta antena, de dimensiones reducidas, ofrece Bluetooth con un rendimiento aceptable. Es cercano a 10 m en línea recta. En Wi-Fi, por el contrario, el alcance obtenido también es de 10 m en campo abierto, lo cual resulta inferior al rendimiento de una mini llave USB Wi-Fi, habitualmente utilizada hasta 30 m. Además, la antena de dimensiones reducidas de la Raspberry Pi 3 es muy sensible al entorno (carcasa de metal, por ejemplo). La posición de la antena también influye en la calidad de la unión, el Wi-Fi usa principalmente ondas polarizadas verticalmente, mientras que la antena de la Raspberry Pi 3 polariza las ondas horizontalmente. Situar la Raspberry Pi 3 en posición vertical puede aumentar ligeramente el alcance del Wi-Fi.

El BCM43438 permite recibir radio FM. La entrada prevista para la antena FM está en el circuito y no ha sido conectada.

Debido al espacio limitado disponible en la tarjeta de la Raspberry Pi y a las dimensiones limitadas similares a las de una tarjeta de crédito, fue necesario volver a reacondicionar la implantación de los componentes. Los LED de estado y sus transistores de control se han movido para dejar sitio al circuito Wi-Fi/Bluetooth, que ocupa algo más de un centímetro cuadrado (siguiente imagen).



En esta imagen se reconoce el BCM43438, que es un circuito **WL-CSP**. Esta tecnología, empleada en la construcción de los teléfonos móviles, consiste en utilizar el circuito desnudo, sin protegerlo por una caja. Durante la fabricación, las bolas de soldadura se colocan directamente en el chip que, en ese momento, se suelda igual que en el circuito impreso.

WL = *Wafer-level packaging* (empaquetado en el wafer) es una tecnología que consiste en acondicionar un circuito integrado mientras que todavía está sobre el wafer, soporte en silicio que se utiliza para la fabricación. Tradicionalmente se corta el chip y después se deposita en un zócalo antes de que las lengüetas del zócalo se unan por finos cables al chip. A continuación, se cierra y se sella el zócalo. Aquí ocurre a la inversa, el circuito se prepara para el montaje mientras que todavía está en el wafer.

CSP = *Chip-Scale Package* (zócalo del tamaño del chip), implica que el componente tiene prácticamente el mismo tamaño que el chip.

Esta tecnología permite, en el lugar de fabricación del wafer, condicionar, probar y presentar los componentes en un plazo de tiempo para poder depurar. Esto permite encadenar en un único lugar toda la cadena de fabricación, desde el silicio hasta el envío a los clientes.

En la imagen anterior se puede ver una ubicación vacía marcada como J13. Podría permitir dotar a la Raspberry Pi de un conector U.FL que permitiera conectar un cable y una antena a la salida, para aumentar el alcance. Posiblemente, la Fundación ha renunciado a montar este conector para ajustarse a la reglamentación Wi-Fi.

- El Wi-Fi y el Bluetooth pueden funcionar al mismo tiempo. Por ejemplo, esto permite ver un vídeo en línea a través de Wi-Fi, enviando el sonido a un altavoz Bluetooth.

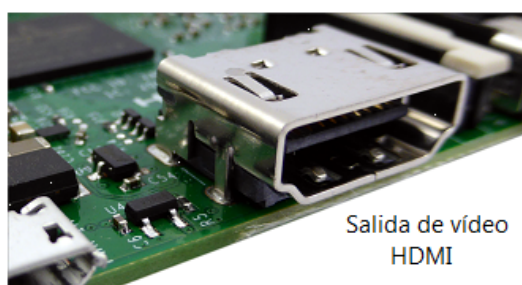
6. Las salidas de vídeo de la Raspberry Pi 3

La Raspberry Pi 3 dispone de dos salidas de vídeo:

- una salida HDMI (*High Definition Multimedia Interface* = interfaz multimedia de alta definición),
- una salida de vídeo compuesto.

Estas dos salidas no pueden funcionar al mismo tiempo. Si se detecta una conexión HDMI, es la salida HDMI la que se activa y la salida de vídeo compuesto se desactiva. Si la Raspberry Pi no detecta conexión HDMI, el vídeo se dirige a la salida compuesta (analógica).

a. Vídeo digital de la Raspberry Pi 3

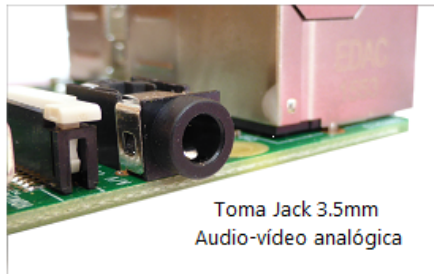


La salida de vídeo HDMI de la Raspberry Pi 3 es idéntica a la de la Raspberry Pi Zero. Encontrará una descripción detallada de esta interfaz HDMI en la parte dedicada a la Raspberry Pi Zero, en este mismo capítulo.

b. Vídeo analógico de la Raspberry Pi 3

Está disponible a través de un conector jack de 3,5 mm con cuatro contactos y proporciona señales de audio y vídeo analógicas. La salida de vídeo compuesto se usa para enviar la señal de vídeo de salida de la Raspberry Pi a una pantalla de TV analógica o a una pantalla digital

que tenga este tipo de entrada.



El conector jack es un modelo de cuatro contactos llamado también TRRS (*Tip-Ring-Ring-Sleeve*).



Hay disponibles diferentes cables adaptadores. La siguiente tabla presenta los modelos más actuales.

Raspberry Pi Modelo B+ Conector audio/vídeo Jack 3,5 mm

Raspberry Pi B+	Vídeo	Masa	Audio D	Audio G	Apropiado
Apple	Vídeo	Masa	Audio D	Audio G	Apropiado
Zune - Microsoft	Vídeo	Masa	Audio D	Audio G	Apropiado
Videocámara	Audio D	Masa	Vídeo	Audio G	Atención
MP3	Masa	Vídeo	Audio D	Audio G	No apropiado

Vemos que los modelos previstos para los dispositivos de marca Apple o Microsoft pueden ser convenientes para un uso con la Raspberry Pi B+. Los cables proporcionados con determinadas videocámaras pueden ser convenientes, con la condición de invertir las tomas RCA vídeo (amarilla) y audio derecho (rojo), situadas en el otro extremo del cable. Por el contrario, los cables proporcionados con determinados lectores MP3 no funcionarán, porque la masa no está situada en el lugar correcto. En caso de duda es mejor comprobar las conexiones con un multímetro.

El nivel de las señales de audio y vídeo son parecidas y el intercambio de cables como consecuencia de las pruebas no debería provocar daños. La siguiente imagen muestra un modelo de cable adaptador TRRS hacia RCA, disponible en los sitios web de venta en línea.



El uso de esta salida, que agrupa audio y vídeo analógicos, permite reciclar un televisor antiguo o conectar la Raspberry Pi a una pantalla pequeña, como las que se venden a bajo precio para realizar las pantallas de la cámara trasera en los coches, por ejemplo. La siguiente imagen muestra un adaptador audio/vídeo compuesto hacia SCART (también llamado Euroconector), en el que se podría conectar el cable que viene de la Raspberry Pi. La salida de vídeo se identifica por conectores de color amarillo.



Compuesto por un único cable central protegido con un blindaje, el cable coaxial conectado a la toma de vídeo compuesta transporta, en este único cable central, la totalidad de las señales de vídeo analógicas destinadas al monitor o a la TV. Las señales de audio no se envían al mismo tiempo. En la señal de vídeo hay información de luminosidad y de color para cada punto de la pantalla. Se añaden las señales destinadas a sincronizar el monitor o la TV con el escaneo del origen que ha generado las señales de vídeo: las señales de sincronización horizontal, que indican el fin de cada línea de la imagen, y las señales de sincronización vertical, que indican el fin de cada imagen en la pantalla. El término compuesto indica la presencia de esta información en la señal.

Utilizado desde el inicio de la televisión, este formato de transmisión de imágenes tiene el defecto de ser analógico y ver sus señales deformadas por las perturbaciones atmosféricas, etc. Por otra parte, las normas utilizadas limitan la definición de las imágenes:

- **576i** (576 líneas entrelazadas) para el PAL (*Fase Alternating Line* = alternancia de fase por línea) y el SECAM (color secuencial de memoria) europeos.
- **480i** (480 líneas entrelazadas) para el NTSC (*National Television System Committee* = comité nacional del sistema de televisión) americano.

Estas definiciones, suficientes para la difusión de imágenes animadas, se hacen difícilmente aceptables comparadas con las definiciones actuales de los sistemas informáticos, TV digitales, etc.

La salida de vídeo compuesta se configura en el archivo *config.txt*, presente en la tarjeta micro SD de la Raspberry Pi, que se lee durante el inicio del sistema. Los estándares NTSC (versión US o versión Japón) y PAL (versión Europa o versión Brasil) están disponibles en tres formatos 4:3, 14:9 y 16:9.

También es posible ajustar el tamaño de la imagen eliminando las franjas negras presentes alrededor de la imagen o llevando la visualización a la dimensión de la pantalla si hay partes de la imagen invisibles porque se extienden más allá de los bordes de la pantalla.

7. Las salidas de audio

a. Audio analógico de la Raspberry Pi 3

La salida de audio digital se realiza por el puerto HDMI en formato 5.1. El audio se combina con el vídeo.

b. Audio analógico de la Raspberry Pi 3

La salida de audio analógico se realiza mediante un jack estéreo de 3,5 mm de cuatro polos (TRRS: consulte la sección Vídeo analógico de la Raspberry Pi 3, en este capítulo, que presenta el reparto de las señales en los contactos del jack). Es el jack estándar para la salida de sonido de muchos dispositivos (smartphones, PC, consolas de juegos, lectores DVD portátiles, lectores MP3, etc.). Esta salida ofrece una señal a nivel "línea" y se debe conectar a un amplificador o a altavoces previstos para un PC, por ejemplo. La potencia de salida de audio no está prevista para unos auriculares de baja impedancia, que provocan una caída en la tensión de salida.

En caso del audio analógico, la Raspberry Pi B+ entrega una señal digital de tipo PWM (*Pulse Width Modulation* = modulación de longitud de impulso). La señal se codifica con 11 bits (los CD de audio se codifican con 16 bits), lo que podría ser conveniente para las aplicaciones de voz IP o para efectos sonoros básicos (sonidos de sistema). Pero es insuficiente para escuchar música, porque está lejos de la calidad de los CD de audio. La salida se pone sucesivamente a 0 y a 1, dependiendo del tiempo más o menos largo. La señal se filtra y el resultado es un sonido más o menos bueno, según la calidad de los circuitos implementados. Una vez más, han prevalecido las opciones económicas, en detrimento de la calidad del sonido obtenido.

El circuito de alimentación de audio de las últimas generaciones de la Raspberry Pi se ha mejorado respecto a sus predecesores. El circuito de salida dispone ahora de su propia alimentación, lo que disminuye el ruido que podría estar provocado por las subidas de corriente de otros componentes de la tarjeta. Para beneficiarse de la calidad óptima de esta salida de sonido, suba el volumen al máximo en la Raspberry Pi y modifique el volumen final con los ajustes disponibles en el amplificador exterior. Si disminuye el sonido a nivel del software, comprobará un aumento del nivel de ruido.

Para obtener un sonido de calidad, sin duda será necesario añadir una tarjeta de audio conectada al puerto USB equipada con un verdadero convertidor digital-analógico, y capaz de restituir sonidos de calidad correcta. También existen tarjetas de extensión que entregan un sonido Hi-Fi. Se enchufan al conector GPIO. Hay un ejemplo en el capítulo Los periféricos.

En este caso, es importante comprobar que el equipamiento seleccionado está bien gestionado por la distribución utilizada en la Raspberry Pi. La lista está disponible en la dirección: http://elinux.org/RPi_VerifiedPeripherals#USB_Sound_Cards

8. El bus CSI de la Raspberry Pi 3

El funcionamiento del bus CSI de la Raspberry Pi 3 es idéntico al que equipa la Raspberry Pi Zero. Vaya al ese párrafo de la descripción de la Raspberry Pi Zero.



El bus CSI se sitúa entre el conector de audio/vídeo analógico y el conector HDMI (imagen anterior). Está previsto para recibir el cable que proviene del módulo de la cámara.



El módulo cámara de la Raspberry Pi se entrega con una capa que se inserta en el conector de bus CSI. Aparecieron en el mercado cámaras compatibles (ArduCam: imagen anterior), ofreciendo la posibilidad de cambiar de objetivo, a diferencia de la cámara "oficial" cuyo objetivo era fijo. Utilizando el mismo sensor que la cámara original se utilizan sin modificación del sistema o del software. En la cámara situada a la derecha de la imagen, se ha desmontado el objetivo de montura CS, dejando aparecer el sensor equipado con su filtro de infrarrojos.

9. El bus DSI de la Raspberry Pi 3

Como el CSI, el bus DSI (*Display Serial Interface* = interfaz en serie para pantalla) es una especificación de la alianza MIPI. Su objetivo es reducir el precio de las pantallas utilizadas en los móviles. Su estructura es parecida a la del bus CSI, con la excepción de que las líneas de datos parten del SoC y van hacia la pantalla. El conector DSI (siguiente imagen) se sitúa en un extremo de la tarjeta de la Raspberry Pi, cerca de la GPIO.



El bus DSI se controla mediante la GPU. A primeros de septiembre de 2015, la Fundación puso en el mercado una pantalla táctil oficial que usa este puerto DSI. Esta pantalla táctil tiene una definición de 800 x 480 píxeles a 60 ps y ofrece una buena calidad de imagen.

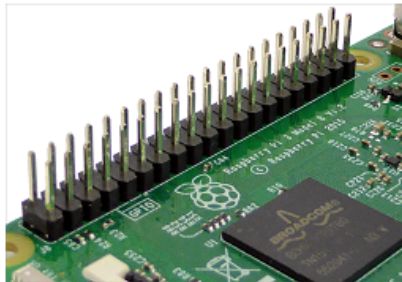


En la imagen anterior, la Raspberry Pi 3 está conectada a través del puerto DSI a la tarjeta de control de la pantalla táctil oficial. Es una pantalla de 7 pulgadas (18 cm) de diagonal, con un tamaño de visualización en el entorno de 155 mm x 86 mm.

La pantalla consume hasta 470 mA además del consumo de la Raspberry Pi. Será necesario prever una alimentación en serie que ofrezca al menos 2,5 A.

10. La GPIO de la Raspberry Pi 3

La GPIO de la Raspberry Pi 3 es idéntica a la de la Raspberry Pi Zero. Vaya a esta parte del capítulo para obtener información acerca de esta interfaz.

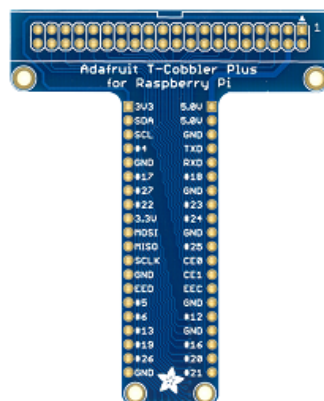


La Raspberry Pi 3 se entrega con el conector GPIO montado. Por tanto, es posible utilizar estas entradas/salidas soldando directamente cables al conector, usando conectores individuales o incluso enchufando un adaptador para unir el puerto GPIO a una tarjeta externa.

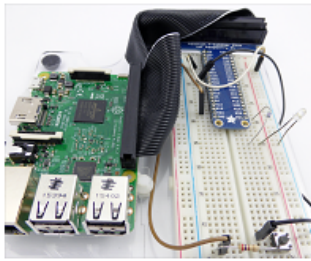
Existen cables para conectar al puerto GPIO de la Raspberry Pi a una tarjeta de prueba (*breadboard*). Por ejemplo, el kit "Pi T-Cobbler" propuesto por Adafruit tiene un cable de unión plano con dos tomas de 40 contactos engarzados en la capa.



Una tarjeta de circuito impreso en forma de T contiene inscripciones xerografiadas que indican el nombre de cada una de las 40 clavijas. Se proporciona un conector de 40 puntos, así como conectores soldados en fila. El conector de 40 puntos se suelda en la parte superior de la barra horizontal de la T y garantiza la unión con el cable plano. Los conectores distribuidos en fila se sueldan debajo de la tarjeta del circuito impreso, a cada lado de la barra vertical de la T. Permiten enchufar el adaptador a una tarjeta de prueba.



El cable obtenido después de la soldadura de las clavijas permite conectar limpiamente la Raspberry Pi a una tarjeta de prueba. La unión firmemente establecida entre la GPIO y la placa garantiza la ausencia de falsos contactos. La siguiente imagen muestra una maqueta destinada a probar un programa que lee el estado de un interruptor y controlar un LED a partir de esta información.



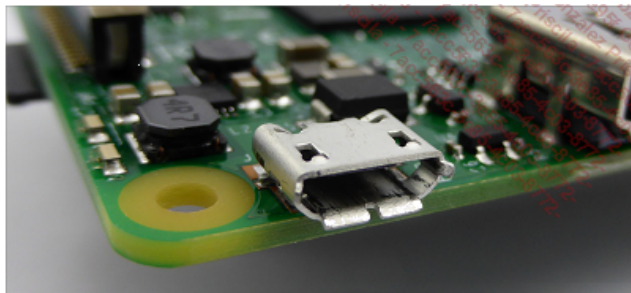
El LED, el botón de encendido y las resistencias están cableados a la placa de control. Esto permite trabajar en todo momento de manera segura con la GPIO, sin provocar un cortocircuito de manera involuntaria.

- Las clavijas de la GPIO están previstas para trabajar con 3,3 V y no soportan una tensión de 5 V. No existe ninguna protección contra las subidas de tensión. Es responsabilidad del usuario implementar en la tarjeta externa los amplificadores o convertidores de nivel de tensión necesarios para la protección de la tarjeta Raspberry Pi.

11. La alimentación de la Raspberry Pi 3

a. Por la toma de corriente

La alimentación de la Raspberry Pi normalmente se realiza mediante una toma micro-USB (siguiente imagen). No existe un interruptor arranque/parada y es la inserción o retirada de la toma micro-USB lo que hace el papel de interruptor. Algunos cargadores de teléfonos están provistos de una toma micro-USB de este tipo. Sin embargo, es obligatorio que la tensión de alimentación proporcionada a la Raspberry Pi sea de 5 V, con una intensidad de 1,5 A como mínimo. La Raspberry Pi modelo 3 consume hasta 800 mA. Si alimenta a periféricos USB es necesario añadir este consumo al de la tarjeta. La opción es una alimentación capaz de proporcionar al menos 2 A. Una alimentación como la oficial (2,5 A máximo) es una buena opción para evitar problemas relacionados con las caídas de tensión.



La alimentación entrega únicamente la corriente que necesita el dispositivo al que está conectada. No por estar etiquetada con 5 V/2 A significa que una alimentación va a enviar 2 A a la Raspberry Pi. Esto significa simplemente que es capaz de proporcionar una corriente de 2 A como máximo, si la Raspberry Pi lo solicita. No hay ningún riesgo en conectar una alimentación que pueda proporcionar una corriente superior a 1 A.

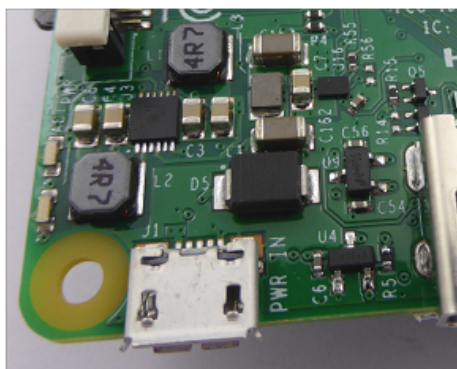
Por el contrario, es importante que la tensión sea de 5 V. Algunos adaptadores entregan una tensión ligeramente inferior a 5 V. La conexión de periféricos USB en la Raspberry Pi crea una fuerte subida de corriente, que hace caer momentáneamente la tensión proporcionada. En determinados casos esto podría provocar un reinicio de la Raspberry Pi a pesar de las protecciones puestas en marcha.

Además, el uso de un cable USB de mala calidad entre el bloque de alimentación y la toma microUSB de la Raspberry Pi puede agravar la situación, provocando una caída de tensión adicional. Las alimentaciones sobre las que está integrado el cable de salida no tienen este tipo de problema en general.

Si usa un cargador de smartphone o tableta, la aparición de un cuadrado coloreado en la esquina superior derecha de la pantalla o el parpadeo del LED de alimentación (consulte más adelante la sección Los LED de estado de la Raspberry Pi 3) le deben alertar y en este caso es mejor sustituir el bloque de alimentación afectado, porque no está adaptado para un uso con la Raspberry Pi.

La alimentación de la Raspberry Pi es una de las partes de la tarjeta que ha sufrido más evoluciones: se ha revisado totalmente, respecto a los primeros modelos. Los reguladores analógicos de las versiones anteriores a la Raspberry Pi B+ bajan los 5 V a 3,3 V y 1,8 V, disipando la energía excedente en forma de calor. Esto hacía necesario algunas veces instalar en el regulador principal un pequeño radiador destinado a facilitar la disipación del calor y a reducir así el aumento de temperatura.

En la Raspberry Pi actual los reguladores son modelos de conmutación que tienen un rendimiento mucho más elevado (96 %) y no disipan prácticamente calor.



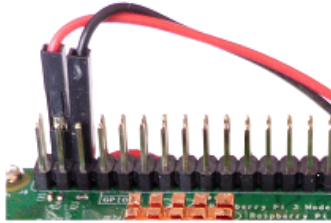
La imagen anterior muestra los componentes de la alimentación de la Raspberry Pi B+. No se ve ningún componente de potencia. Los componentes marcados 4R7 son las bobinas utilizadas para la regulación de las tensiones 3,3 V y 1,8 V.

b. Por la GPIO

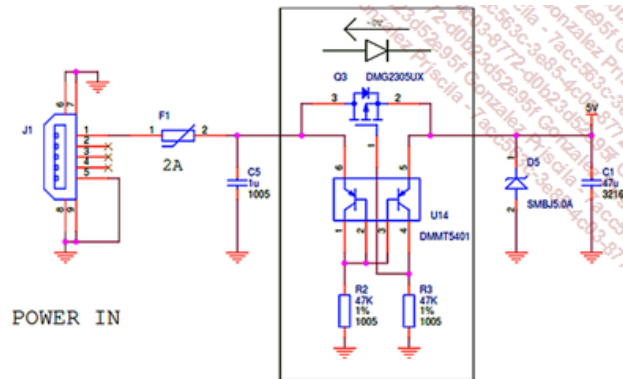
Es posible alimentar la Raspberry Pi en 5 V a través de la GPIO.

Alim. +3,3V	1	2	Alim. +5V
GPIO 2 (SDA)	3	4	Alim. +5V
GPIO 3 (SCL)	5	6	Masa0V
GPIO 4	7	8	GPIO 14 (TXD)

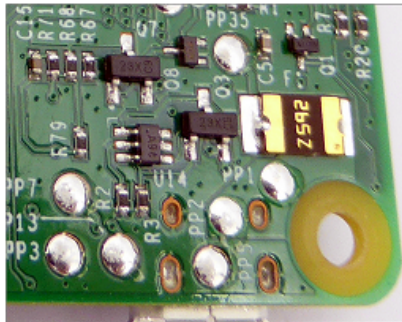
El cable rojo que proporciona los + 5 V se debe conectar a la clavija 4 y la masa (cable negro) a la clavija 6.



Este método requiere muchas precauciones, porque la tensión se aplica directamente a los circuitos de la Raspberry Pi sin ninguna seguridad ni protección.



En el esquema anterior de la entrada de la alimentación (POWER IN) se observa cómo cuando la tensión de alimentación llega a la tarjeta a través de la toma USB J1, inicialmente se aplica a un fusible reajustable (polifusible) F1 de 2A (marcado como Z592 en la siguiente imagen). El bloque que sigue tiene un circuito equivalente a un diodo. Solo deja pasar la corriente en el sentido autorizado. Este montaje sustituye a un diodo cuya caída sería perjudicial. Aquí el transistor Q3 garantiza esta función, provocando solo una caída de tensión muy débil.



El bloque de protección contra las inversiones de tensión puede observarse en la imagen anterior (U14 y Q3). El diodo D5, situado justo después de etapa de protección, se encarga de eliminar posibles sobretensiones transitorias que se pudieran provocar. La tensión observada de 5V a la derecha del esquema se destina a la alimentación de conmutación, produciendo 3,3V y 1,8V. También alimenta a los puertos USB, así como a las clavijas de 5V de la GPIO.

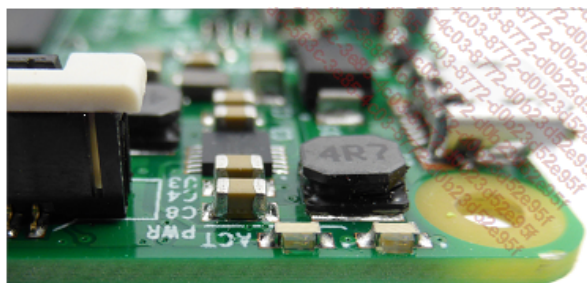
Esto significa que si elige alimentar la Raspberry Pi a través de la GPIO, enviará los 5V directamente a los circuitos de la Raspberry Pi, después al fusible y después a la protección contra las inversiones de tensión. Cualquier sobretensión o inversión de tensión a nivel de la GPIO será fatal para la Raspberry PI.

c. Por los puertos USB

En principio, no hay que alimentar una tarjeta a través de uno de sus puertos USB. También es el caso de la Raspberry Pi.

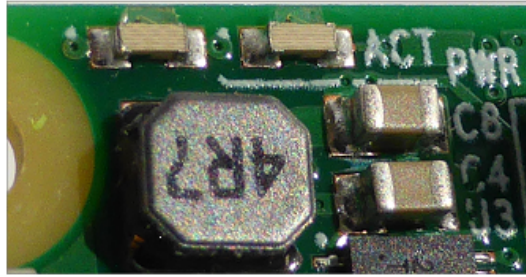
Por supuesto, es posible hacerlo de manera indirecta. Si conecta el puerto USB cuando la Raspberry Pi está apagada, no arranca. Sin embargo si aplica la alimentación de manera clásica, por ejemplo por el puerto micro USB de alimentación, como se conecta la alimentación a través del puerto USB y al final corta la alimentación original, la Raspberry Pi seguirá funcionando. Conviene observar que la corriente de salida de los puertos USB está limitada a 1,2 A. No hay que intentar proporcionar más intensidad a un periférico conectado al puerto USB.

12. Los LED de estado de la Raspberry Pi 3



La Raspberry Pi 3 está equipada con 2 LED de estado visibles en la imagen anterior en el borde de la tarjeta. El LED situado más a la derecha es el LED PWR (Power = Alimentación). Se enciende cuando se aplica una tensión de 5 voltios en la toma micro USB de alimentación. Cuando la tensión disminuye por debajo del valor de 4,75 voltios se apaga, lo que indica un problema de alimentación. Puede ser el caso cuando se

enchufa un periférico que consume mucha corriente, o si se usa una alimentación insuficientemente estable para la Raspberry Pi o cuando se usa un cable de alimentación inadecuado.



La marca PWR se ha movido por falta de espacio, el LED correspondiente se indica mediante una línea que parte de la base de la P.

El LED ACT situado a la derecha de la imagen anterior, indica la actividad de la Raspberry Pi, es decir el acceso a la tarjeta micro SD. El LED se enciende durante un acceso a la tarjeta. Si no introduce una tarjeta micro SD, permanecerá apagado.

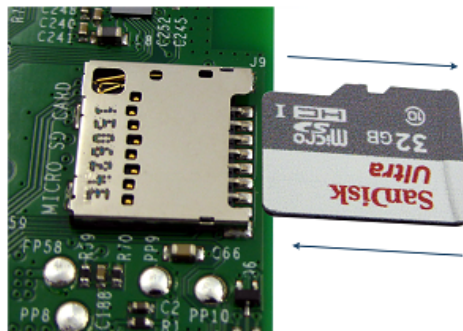
13. El conector de tarjeta micro SD de la Raspberry Pi

Situado debajo de la tarjeta de la Raspberry Pi, el conector está preparado para acoger una tarjeta micro SD.



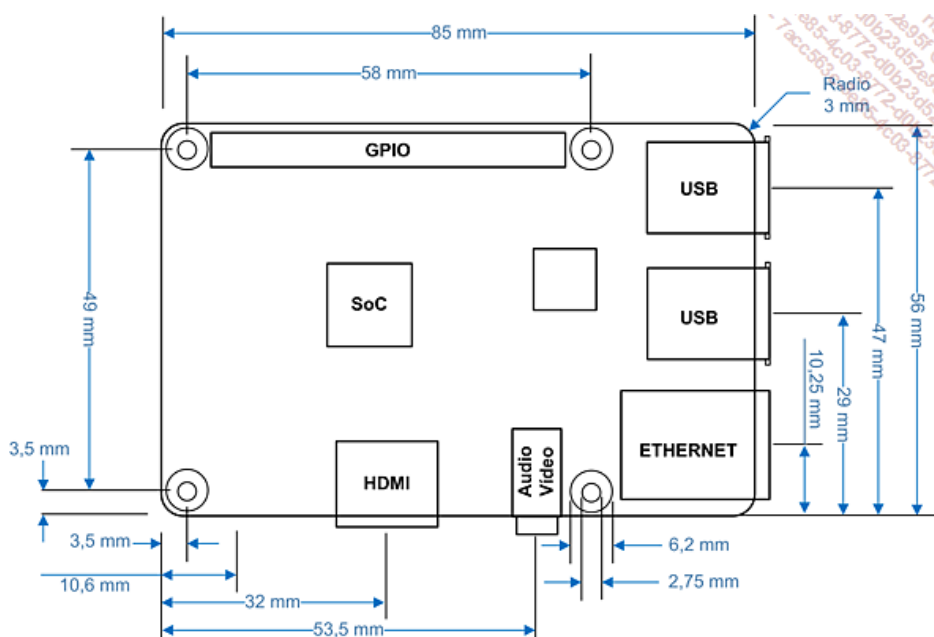
Respecto al soporte de tarjeta SD en plástico de las primeras versiones de la Raspberry Pi, este modelo metálico es más robusto. Por otro lado, el pequeño saliente de la tarjeta micro SD (2 mm) evita que una pulsación en ella afecte al conector.

El conector es de tipo "push-pull". No se produce bloqueo mecánico durante la inserción de la tarjeta. Simplemente hay que empujar a fondo en el conector. La tarjeta se mantendrá a continuación en su posición mediante unas lengüetas que se apoyan en los contactos de la tarjeta micro SD y gracias al resorte que forma el contacto presente en la tarjeta, visible en la imagen anterior en el ángulo superior derecho del conector.



Para sacar la tarjeta, tire de ella. Según la configuración en la que se haya instalado la Raspberry Pi, la extracción de la tarjeta puede necesitar el uso de una pequeña pinza.

14. Las dimensiones físicas de la Raspberry Pi 3



El factor de forma de la Raspberry Pi 3 es idéntico al de sus predecesoras Raspberry Pi B+ y Raspberry Pi 2. Tiene cuatro agujeros de fijación dispuestos en los ángulos de un rectángulo de 58 x 49 mm. La fijación de la tarjeta en un soporte se simplifica y esta disposición de los agujeros también permite montar de manera muy estable las tarjetas de extensión.

15. El reloj en tiempo real

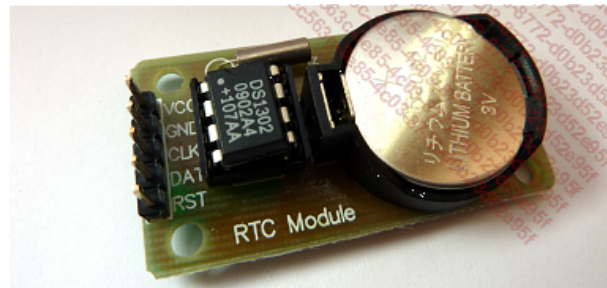
Este párrafo no debería existir porque en la Raspberry Pi... no hay reloj en tiempo real. El sistema se pone en hora cuando se conecta a un servidor NTP (*Network Time Protocol* = protocolo de red de sincronización horaria). El protocolo NTP sincroniza la hora del ordenador local con la de un servidor en tiempo real. Los servidores primarios, sincronizados con un reloj atómico, envían la hora al árbol de servidores secundarios. Esto garantiza una gran precisión.

El protocolo NTP gestiona:

- La periodicidad de la solicitud de la hora enviada por el ordenador local al servidor.
- El cálculo del espacio de tiempo entre la hora local y la del servidor.
- El cálculo del tiempo de ida/vuelta incurrido por los mensajes en la red.
- El cálculo de las correcciones que hay que aplicar al reloj local para conservar la sincronización.

Si la Raspberry Pi no se conecta a Internet, la sincronización no puede tener lugar y la hora mostrada por la Raspberry Pi no tendrá sentido. Hay que ajustar la hora manualmente.

Si la aplicación necesita marcar la información tratada con la fecha y la hora (estación meteorológica, grabación de ruta GPS...), se impone la implementación de un reloj en tiempo real RTC (*Real Time Clock* = reloj en tiempo real) mantenido por una pila.



La tarjeta anterior es un ejemplo. Equipado con un circuito de reloj RTC DS1302, un reloj de cuarzo (el cilindro gris situado encima del componente DS1302) y una pila de litio CR2032, permite conservar la hora mientras que la Raspberry Pi no tenga corriente.

Durante el inicio del sistema, se lee la hora de este reloj RTC y se utiliza para poner en hora el reloj del sistema.

La unión con la Raspberry Pi se hace con los puertos de la GPIO. No es una unión estandarizada I²C, SPI o 1-wire (bus utilizado principalmente en domótica para conectar componentes solo con dos cables), sino una sencilla unión en serie. Hay módulos para administrar este tipo de tarjeta con el sistema operativo de la Raspberry Pi.

Existen muchos otros modelos de tarjeta RTC, algunos se conectan directamente a la GPIO. Es necesario un estudio inicial de las diferentes soluciones para seleccionar el reloj RTC que mejor se adapte a sus necesidades.

Tabla comparativa

La Raspberry Pi Zero y la Raspberry Pi 3 tienen características bien diferentes, no destinadas a los mismos usos. En la siguiente tabla encontrará un resumen de sus características.

Modelo	Raspberry Pi Zero	Raspberry Pi 3
Procesador	BCM2836	BCM2837
CPU	ARM1176JZF-S 1 núcleo - 32 bits	ARM Cortex-A53 4 núcleos - 64 bits
Frecuencia Reloj	1 GHz	1,2 GHz
RAM	512 Mo	1 GB
GPU	VideoCore IV @ 250 MHz	VideoCore IV @ 400 MHz
Tarjeta de memoria	Micro SD	Micro SD
Salida vídeo	1 salida mini HDMI y 2 pastillas de vídeo compuesto	1 salida HDMI Jack audio/vídeo compuesto
Salida audio	En mini HDMI	En salida HDMI Jack audio/vídeo compuesto
Conexión alámbrica	-	1 puerto RJ45 10/100 MHz
Conexión inalámbrica	-	Wi-Fi 802.11n (150 Mbits/s) Bluetooth 4.1
Puerto USB	1 puerto USB 2 micro USB	4 puertos USB 2
GPIO	2 x 20 pastillas	Conector 2 x 20 clavijas
Reset	2 pastillas: RUN	2 pastillas: RUN
Interfaz cámara	Conector MIPI-CSI2 22 contactos	Conector MIPI-CSI2 15 contactos
Interfaz de pantalla	-	Conector MIPI-DSI 15 contactos
LED	1 LED verde ACT	1 LED rojo PWR 1 LED verde ACT
Corriente consumida	~ 150 mA	~800 mA

Introducción

Para hacer funcionar un ordenador, es necesario un sistema operativo. Este OS (*Operating System* = sistema operativo) es el director de orquesta que gestiona los recursos del ordenador y los asigna a las diversas tareas que se están ejecutando, ya sean aplicaciones ejecutadas por el usuario o servicios propios del sistema.

Cuando un software necesita un recurso de hardware, es el sistema operativo el que se lo asigna. El OS gestiona el hardware (memoria y periféricos), el acceso a los archivos almacenados en los dispositivos de almacenamiento, así como el intercambio de información en red.

Cuando la Fundación creó la Raspberry Pi, el mantenimiento del precio en el rango fijado al inicio implicó que no había que pagar licencia por el uso del sistema operativo. Por lo tanto, es natural que los creadores pensaran en Linux.

Este capítulo ilustra rápidamente la historia de Linux y presenta las distribuciones compatibles con la Raspberry Pi.

Linux

Linux se debería llamar GNU/Linux. Se inscribe en el proyecto GNU (*GNU's NOT Linux* = GNU no es Linux), iniciado en 1983 por Richard Stallman. Es un sistema operativo libre bajo licencia GPL (*GNU General Public License* = licencia pública general GNU). Es decir, el código fuente de los programas está a disposición de todo el mundo. Independientemente de quién use Linux, puede coger el código fuente, estudiarlo, modificarlo y redistribuirlo. Son millares de desarrolladores, algunas veces pagados por empresas, los que participan en la evolución de Linux.

1. De Linus a Linux

En los años 90, los estudiantes universitarios trabajaban en terminales conectados a un potente ordenador central que funcionaba con UNIX. Los ordenadores personales que se podían comprar tenían microprocesadores 80386, fabricados por Intel, y funcionaban con MS-DOS, Windows (Microsoft) u OS/2 (Microsoft y e IBM).

Linus Torvalds, un estudiante finlandés de la universidad de Helsinki, compró un PC e instaló Minix, un sistema operativo parecido a UNIX, creado por el profesor Tanenbaum y utilizado para la enseñanza. Insatisfecho con el funcionamiento de Minix, y en particular con el emulador de terminal que utilizaba para conectarse al ordenador de la universidad, inicia la escritura de su propio emulador de terminal. Finalmente, en algunos meses desarrolla el núcleo de un sistema operativo, donde potenciaba aquellos aspectos que mejor dominaba. En el servidor FTP de la universidad, el directorio en el que se almacenaba el trabajo de Linus se llama "LINUX". Este es el nombre que se ha mantenido para denominar al sistema operativo que estaba naciendo.

El 25 de agosto de 1991, Linus anuncia en Usenet (sistema de foros que funcionan con UNIX) que ha creado un OS e invita a los desarrolladores a contribuir a su desarrollo. Inicialmente, Linux no es libre. Linus autoriza solamente su difusión gratuita. El feedback recibido le anima a continuar con el desarrollo y a introducir mejoras. Algunos años más tarde, estaba asegurado que su nombre estaría definitivamente ligado al núcleo Linux y Linus lo "liberó" renunciando a usar la licencia GPL.



Larry EWING inicia en 1996 un concurso de diseño que permite a Linux adquirir una mascota, el pingüino Tux. Linus declara en el "Manifiesto de Linux" (<http://linux-es.org>) que Tux no tienen ningún significado particular, pero recuerda un viaje a Australia durante el que un pingüino que trataba de domesticar le mordió. Cuando tuvo que buscar un logo y algunos propusieron logos comerciales aburridos, Linus prefirió asociar Linux a un logo simpático y un poco irreverente, en lugar de a algo estático y aburrido.

En lo que respecta al nombre, Tux viene del inglés *tuxedo*, que hace referencia a un smoking que se usa para cenar, con camisa blanca y pajarita. La persona que lleva esta indumentaria se llama... "pingüino".

El hecho de que Linux fuera libre coincide con el desarrollo de Internet. De esta manera miles de desarrolladores de todo el mundo, seducidos por el concepto, se unen al proyecto. El desarrollo del núcleo continúa y Linus sigue siendo la persona que decide en última instancia sobre las modificaciones introducidas en el núcleo.

Después de defender su tesis en Helsinki, Linus viaja a Estados Unidos. De 1997 a 2003 trabaja para Transmeta (un fabricante de microprocesadores). Puede seguir ocupándose del núcleo Linux en una parte de su tiempo de trabajo, con la autorización de su empleador. En 2003 se une a la Fundación Linux, donde se dedica plenamente a Linux. Obtiene la nacionalidad estadounidense en 2010.

2. El núcleo

Descripción del núcleo Linux

El núcleo Linux es la parte central del sistema operativo. Gracias a él, el software accede al hardware. Es quien gestiona y asigna los recursos del sistema. Desarrollado inicialmente para máquinas basadas en procesadores x86, es decir, para máquinas con un microprocesador compatible con el 8086 de Intel, el núcleo Linux ha ido progresivamente adaptándose a otras arquitecturas.

La escritura del núcleo se realiza en lenguaje C, un lenguaje ya utilizado para escribir UNIX. A continuación se compila el código fuente producido por los desarrolladores (se transforma en un programa ejecutable por un microprocesador). A partir de un único programa fuente, el compilador y las librerías utilizadas permiten compilar el núcleo para diferentes microprocesadores. Por este motivo, hoy existen versiones del núcleo Linux adaptadas a una gran variedad de sistemas.

Además, la disponibilidad del código fuente permite la adaptación a máquinas particulares mediante la modificación de las líneas de código afectadas por las diferencias que se deben tener en cuenta en el núcleo.

Usos del núcleo Linux

Linux equipa súper calculadoras (máquinas con cientos de miles de procesadores), teléfonos móviles, ordenadores personales y una amplia gama de hardware en el que están embebidas. En un sistema embebido, el usuario no ve la máquina como un ordenador, sino que tiene otras funciones: box Internet, copiadoras digitales, grabadoras de vídeo digitales, lectores Mp3/Mp4, teléfonos IP...

Existen versiones adaptadas de Linux ocultas en el sistema Android de Google, pero también en webOS, anteriormente utilizado por Palm y Hewlett Packard, que LG implementó en sus TV digitales y TV Internet. La plataforma Maemo de los teléfonos Nokia también se basa en un núcleo Linux.

En lo que respecta a las súper calculadoras utilizadas por las grandes instituciones civiles y militares, o por las empresas globales, Linux literalmente ha borrado del mapa a la competencia, porque equipa a más del 92 % de las súper calculadoras del TOP500 (las 500 máquinas más potentes del mundo). En comparación, UNIX con 5 % y Windows con 0,4 % son muy poco representativas.

Empresas y organizaciones utilizan Linux a diario en sus servidores y puestos de trabajo. Un poco más lejos, a 400 km de la Tierra, todos los portátiles de la estación espacial internacional (ISS) han migrado a Debian (mayo del 2013), y el robot humanoide R2 destinado a sustituir a los astronautas en las maniobras peligrosas también usa Linux. La elección de Debian (consulte a continuación la sección Seguridad del núcleo) está motivada por la presencia de virus informáticos en los ordenadores portátiles del ISS. Es un cosmonauta ruso quien los subió a bordo en su ordenador portátil, que no usaba Linux.

Es reconfortante y divertido pensar que este mismo sistema operativo es el que da vida a la minúscula tarjeta Raspberry Pi, y que todo lo que se adopta en la Raspberry Pi es transportable a otros sistemas...

Seguridad del núcleo

El diseño mismo del núcleo, la separación entre lo que se puede ejecutar por el administrador del sistema y lo que se puede ejecutar por un simple usuario impide la ejecución y, sobre todo, la propagación de virus en Linux.

El sistema es libre, como la mayoría de las aplicaciones que lo acompañan. Miles de personas han leído y releído el código para descubrir fallos. Como en todo sistema, los fallos existen, pero en general la publicación de las correcciones suele llegar algunas horas después de descubrir el fallo. De la misma manera, es posible modificar las fuentes para reforzar la seguridad del sistema.

Por ejemplo, la armada estadounidense publica LPS (*Lightweight Portable Security* = Seguridad portable ligera), una versión segura de Linux destinada a los empleados del departamento de defensa, que arranca desde un CD o una llave USB, cifrando los datos y que es capaz de leer las tarjetas de acceso a los sitios del gobierno.

Sin embargo, existen antivirus para Linux, lo que parece contradecir las líneas anteriores... En realidad, los antivirus Linux se instalan en los servidores (mensajería, compartición de archivos, FTP...) a los que acceden las máquinas que funcionan con Windows, y que son susceptibles de contaminarse por los virus que pueden viajar en los e-mails o en los archivos. Los antivirus Linux no protegen a la máquina Linux, sino a los clientes Windows.

Versiones del núcleo

El núcleo original, escrito por Linus Torvalds en 1991, tiene alrededor de 10.000 líneas de código. Tiene el número de versión 0.01. El primer núcleo estable y realmente usable apareció en marzo de 1994. Equipa el nº 1.0 y tiene cerca de 200.000 líneas de código.

La numeración de las versiones obedece a una regla simple:

- La primera cifra es el número de versión mayor (versión del núcleo).
- La segunda cifra es el número de versión menor (revisión del núcleo).
- La tercera cifra es un número de revisión correspondiente a la corrección de errores, fallos de seguridad o adición de funcionalidades.

El núcleo actual (aparecido en agosto de 2016) es el 4.5. Tiene más de 15 millones de líneas de código y se mantiene por más de 5.000 desarrolladores y 500 empresas de todo el mundo. Algunas distribuciones como Raspbian utilizan núcleos anteriores por razones de estabilidad o seguridad. En agosto de 2016 Raspbian usa el nodo 4.4.16+.

3. Las distribuciones

El núcleo Linux es, de alguna manera, el motor del sistema operativo. Pero igual que para fabricar un coche, el motor por sí mismo no es suficiente. Es necesario "vestir" este núcleo con aplicaciones para permitir su uso.

A partir del mismo motor es posible crear un 4x4, un familiar, un monovolumen, un cupé, etc. De la misma manera, a partir del núcleo Linux, personas, comunidades y empresas han construido un conjunto de software que comparte un objetivo concreto y tiene una filosofía particular.

Este conjunto de software, a menudo construido por los actores del mundo libre, se llama una distribución. Las distribuciones proporcionan:

- La facilidad de instalación.
- La facilidad de uso.
- La rapidez y la baja necesidad de recursos.
- La cantidad de software disponible.
- El acceso al contenido multimedia.
- La calidad de sus herramientas de escritorio.
- La frecuencia de las actualizaciones.
- La disponibilidad de una comunidad para la asistencia.
- La adaptación a una necesidad concreta (desarrollo, cálculo científico...).
- La seguridad ofrecida.

Por tanto, la elección de una distribución es determinante y se debe adaptar al nivel del usuario (principiante, experto), a los recursos disponibles (Raspberry Pi, servidor multiprocesador), a la aplicación prevista (de escritorio, centro multimedia)...

Las distribuciones pueden derivar de otras distribuciones. De esta manera, Linux Mint deriva de Ubuntu, que deriva de Debian...

En el sitio <http://distrowatch.com> se enumeran las distribuciones y se clasifican en base al número de visitas a la página que se les dedica en el sitio. Por tanto, esta clasificación se debe tomar con precaución, porque no tiene en cuenta el número de sistemas realmente instalados. Las cuatro primeras distribuciones son (en agosto de 2016):

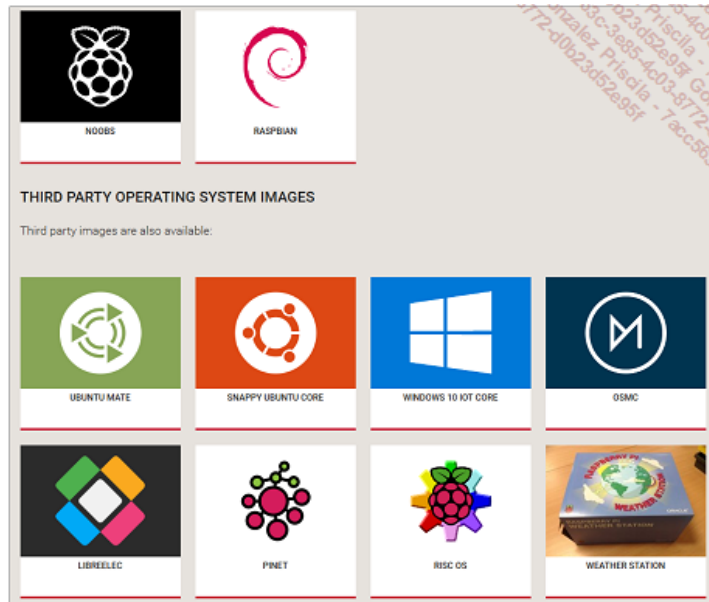
- Linux Mint (derivado de Ubuntu)
- Debian
- Ubuntu (derivado de Debian)
- OpenSUSE

Es decir tres distribuciones de cada cuatro usan Debian o sus derivados.

La distribución esperada para la Raspberry Pi por la Fundación Raspberry es Raspbian, una distribución de Debian compilada para la Raspberry Pi. La página oficial de descarga de las distribuciones adaptadas a la Raspberry Pi es www.raspberrypi.org/downloads.

Distribuciones Linux para la Raspberry Pi

Las distribuciones "oficiales" propuestas por la Fundación o por otros proveedores, se encuentran en la página <https://www.raspberrypi.org/downloads/>:



El sitio <http://elinux.org> enumera cerca de 50 distribuciones disponibles para la Raspberry Pi. Este número debería continuar creciendo. Las principales distribuciones (en agosto de 2016) se presentan a continuación. Los identificadores y contraseñas se indican para las distribuciones en las que están definidas por defecto (http://elinux.org/RPi_Distributions).

1. Raspbian Jessie

La llegada de Simon LONG a la fundación Raspberry Pi ha acelerado el proceso de evolución del sistema utilizado por la Raspberry Pi. A finales de octubre de 2015, sale la primera versión de Raspbian Jessie, una Debian 8 adaptada a la Raspberry Pi. La innovación principal de esta versión es el arranque en modo gráfico por defecto, lo que abarca el uso de la Raspberry Pi desde la educación hasta el mundo empresarial.

El entorno gráfico se ha revisado, en particular el escritorio se ha liberado de los iconos, excepto el de la papelería. A comienzos de febrero de 2016, aparece una nueva versión de Jessie. Más allá de la actualización de las principales aplicaciones (Sonic Pi, Scratch, Mathematica...) esta versión ofrece un control OpenGL experimental. El uso de la GPU por el controlador sustituye al control por software encargado de calcular las imágenes en 2D y en 3D. Esto debe facilitar el uso de los juegos y las aplicaciones de diseño 2D y 3D, cuando el controlador esté estable.

En septiembre de 2016, aparece una nueva versión de Raspbian Jessie llamada PIXEL y que aporta un entorno gráfico modernizado. El escritorio recibe imágenes de fondo y aparecen nuevas aplicaciones como Chromium y VNC Server. Uno de los argumentos destacados de Raspbian es que hay disponibles más de 35.000 programas en forma de paquetes precompilados fáciles de instalar. Cada uno debería encontrar lo que busca.

Identificador: **pi** contraseña: **raspberr**y

➤ Esta distribución es la recomendada por la fundación Raspberry Pi.

2. Raspbian Jessie Lite

Simultáneamente a la distribución Raspbian Jessie, aparece una versión Lite. Esta versión no tiene entorno gráfico. Está destinada a aplicaciones embebidas o como servidor.

Identificador : **pi** contraseña: **raspberr**y

3. Raspbian "wheezy"

Raspbian original

Raspbian se creó por un pequeño equipo de desarrolladores, fans de la Raspberry Pi, partiendo de Debian 7 (también llamado "wheezy"). La Fundación Raspberry Pi ha compilado su propia versión de Raspbian, disponible como versión "oficial", en el sitio raspberrypi.org. Integra una versión optimizada de Debian, el entorno de escritorio LXDE (*Lightweight X11 Desktop Environment* = entorno ligero de escritorio en XWindow), un navegador web libre y ligero, herramientas de desarrollo (Scratch y Python)...

Sustituido por Jessie, esta versión puede no obstante ser útil. Algunos tutoriales realizados en Wheezy ya no funcionan en Jessie. La instalación de esta distribución puede permitir utilizarlos.

Identificador: **pi** contraseña: **raspberr**y

➤ Esta distribución está disponible para su descarga en el sitio <https://downloads.raspberrypi.org/raspbian/images/raspbian-2015-05-07/>.

4. Arch Linux ARM

Aquí se presenta la filosofía de Arch Linux: combinar simplicidad y control completo por parte del usuario. Esta distribución ofrece una infraestructura ligera en la que el usuario puede construir un sistema según sus necesidades.

La distribución soporta el juego de instrucciones de la calculadora en coma flotante del procesador ARMv7, que equipa la Raspberry Pi 2.

La distribución no tiene versión, sino que utiliza un sistema llamado *rolling-release* (actualización continua), que permite obtener una actualización diaria de la distribución.

La imagen comprimida ocupa 235 MB, la imagen representa 1,9 GB. La versión para Raspberry Pi arranca en 10 segundos aproximadamente.

Identificador: **root** contraseña: **root**

➤ Esta distribución está disponible para su descarga en la dirección: <https://archlinuxarm.org/about/downloads> ArchLinuxARM-rpi2-latest.tar.gz

5. RISC OS

Creado en Cambridge, en Gran Bretaña por la empresa Acorn, RISC OS apareció en 1987.

Su origen se remonta al primer equipo que desarrolló el microprocesador ARM. En la actualidad RISC OS pertenece a Castle Technology y se mantiene por RISC OS Open Limited.

RISC OS no es un Linux. El nodo del sistema es mono tarea, es la interfaz gráfica WIMP (Windows, Iconos, Menús, Puntero), la que proporciona la multitarea. Estas funcionalidades son bastante básicas, pero se traducen en una gran ligereza, que permite a RISC OS funcionar en sistemas que dispongan de pocos recursos.

La versión 5 para la Raspberry Pi está disponible gratuitamente para los usuarios de la Raspberry Pi. Soporta las Raspberry Pi 1 y Raspberry Pi 2 y, como experimento, la Raspberry Pi 3. Las aplicaciones disponibles para RISC OS no se pueden utilizar con Raspbian.

➤ Esta distribución está disponible para su descarga en el sitio raspberrypi.org.

6. LibreELEC

LibreELEC está diseñado como un sistema operativo embebido. El usuario no "ve" el sistema operativo, sino únicamente una pantalla de centro multimedia. Es una distribución construida *from scratch* (a partir de los programas fuentes), que transforma la Raspberry Pi en un centro multimedia gracias a KODI, ex XBMC (*XBox Media Center*). La versión compilada para la Raspberry Pi 3 ocupa menos de 16 MB.

Una vez creada la tarjeta micro SD, el sistema arranca directamente y necesita una configuración mínima. Es posible controlar el centro multimedia con el mando a distancia del televisor, porque los comandos CEC (*Consumer Electronic Control* = control electrónico por el cliente) se interpretan por KODI. Se reconoce la mayor parte de los formatos; algunos, como MPEG2, necesitan una licencia que es posible comprar en el sitio de la Fundación.

KODI puede leer vídeos, fotos, músicas y podcasts. Puede guardar programas y servir de PVR (*Private Video Recorder* = reproductor de vídeo digital). Es compatible con uPnP (*universal Plug and Play* = conexión universal de periféricos) y puede leer datos que provengan de hardware compatible uPnP (un ordenador, una box Internet, etc.).

➤ Esta distribución, así como RaspBMC, están disponibles para su descarga en el sitio <http://raspberrypi.org>.

7. OSMC

Como LibreELEC, OSMC (*Open Source Media Center*) es una distribución basada en Debian Jessie y KODI. Existe una imagen disponible para las Raspberry Pi 3 (<https://osmc.tv/download/>). Ocupa 156 MB.

8. Ubuntu

Hay disponibles dos versiones de Ubuntu para la Raspberry Pi: *Ubuntu Mate* disponible en versión 16.04 (agosto de 2016), que da acceso a un escritorio gráfico, y *Snappy Ubuntu Core*, una versión en modo texto destinada a los desarrolladores.

9. Windows 10

Incluso si dispone de un modo gráfico, esta versión no tiene escritorio, navegador web ni tratamiento de texto. *Windows 10 IoT (Internet of Things* = Internet de las cosas). La edición IoT es mono tarea y se configura en modo remoto, por línea de comandos.

10. PiNet

PiNet es un sistema destinado a facilitar la gestión de una clase. Se instala en un PC en Ubuntu.

PiNet permite centralizar la gestión del sistema operativo de las Raspberry Pi. Genera una imagen de la tarjeta micro SD destinada a la Raspberry Pi. Esta imagen arranca en la Raspberry Pi que se ha descargado su sistema operativo en el PC central. De esta manera, se integra automáticamente una actualización de la imagen en el servidor para todas las Raspberry Pi de la clase durante el próximo inicio.

11. Otras distribuciones

Existen actualmente muchas distribuciones disponibles para la Raspberry Pi. Otras se están migrando y cualquier día se liberarán. Funcionan en modo de prueba, como Android o Chromium OS. La Fundación Mozilla también prepara una versión de Firefox OS para la Raspberry Pi, con una hoja de ruta bien definida hasta 2015 (<https://wiki.mozilla.org/FirefoxOS/RPiB2>).

Otros proyectos trabajan para transformar la Raspberry Pi en:

- Cortafuegos domestico como *IPFire*.
- Cliente ligero para Microsoft, Citrix o VMware View: *Raspberry Pi Thin Client*.
- Cliente ligero para Edubuntu u otro servidor LTSP (*Linux Terminal Server Project* = proyecto Linux de servidor de terminales): *BerryTerminal*.
- TV Internet: *RasPlex*.
- Herramienta de seguridad de BackTrack: *Kali*.
- Emulador de consolas de videojuegos: *PiMAME*, *Raspicade*, *RecalBox*, *HAPPI*.

Introducción

La Raspberry Pi es un ordenador cómodo y de bajo coste. Sus diseñadores, con el objetivo de simplificar y reducir el precio máximo del producto, eligieron hacer que su ordenador arrancara en uno de los tipos de memoria más extendidos: la tarjeta micro SD.

Después de una explicación detallada del arranque de la Raspberry Pi, este capítulo explica cómo recuperar el sistema operativo después de instalarlo en una tarjeta micro SD, tanto para los principiantes como para los más expertos.

La descarga se realiza en Windows 8. Los usuarios de Linux en modo gráfico lo harán fácilmente en este sistema. La instalación del sistema en la tarjeta micro SD se detalla en Windows 8 y en Debian 7. El sistema operativo elegido es el que recomienda la Fundación Raspberry Pi para los usuarios principiantes: Raspbian.

La sintaxis de los comandos que se presentan está limitada a las opciones utilizadas en este capítulo. Para obtener más información, diríjase al capítulo Usar la línea de comandos, y consulte también el `man` (manual) de cada comando.

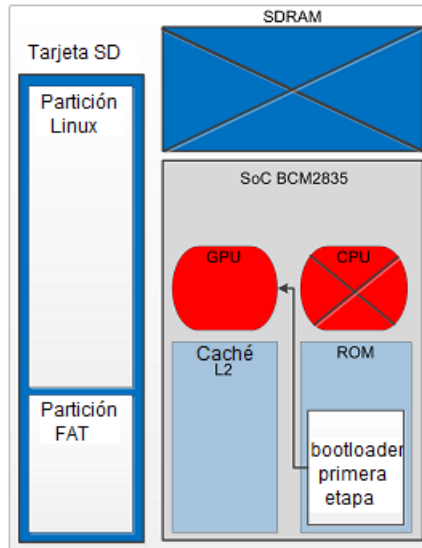
Las versiones de los sistemas operativos y de las herramientas, evolucionan. Tiene que adaptar la siguiente información, en función de las versiones disponibles cuando prepare su tarjeta micro SD.

Secuencia de boot de la Raspberry Pi

El *boot* (inicio) de la Raspberry Pi es estrictamente idéntico en los diferentes modelos Pi Zero y Raspberry Pi 3. Este proceso de inicio implica un determinado número de operaciones. El correcto entendimiento de esta secuencia es primordial si se intenta modificar voluntariamente el inicio del sistema operativo (iniciar en una llave USB o desde un disco duro). Pero mientras el sistema no arranca, el usuario está en presencia de un enigma que solo el conocimiento del desarrollo exacto de la secuencia de boot permite resolver.

1. Etapa 1: enchufar a la corriente

El siguiente esquema representa las diferentes partes implicadas en el inicio de la Raspberry Pi. A la izquierda está la tarjeta micro SD dividida en dos partes. El SoC integra el microprocesador ARM (CPU) y el procesador gráfico (GPU). Está situado a la derecha, bajo la memoria (SDRAM).

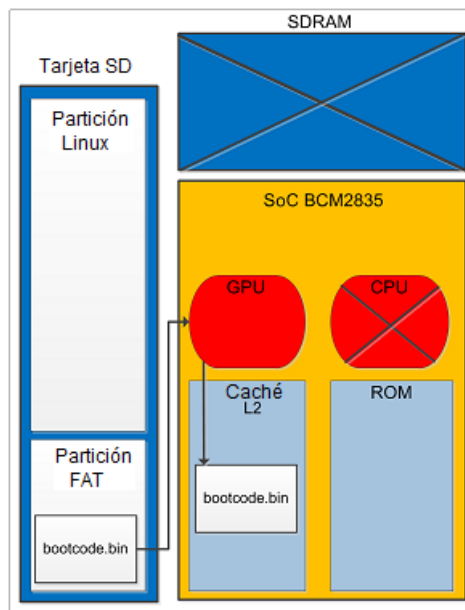


Cuando se enchufa a la corriente, solo está activa la GPU. Mantiene la CPU parada. La memoria SDRAM también está desactivada. El SoC contiene una ROM (*Read Only Memory*), en la que el fabricante ha registrado un programa que forma parte de la cadena de boot. Este programa es el primero de la cadena de boot (*bootloader* = gestor de arranque). No es accesible para el usuario y, por tanto, no se puede modificar. La GPU ejecuta este primer programa, cuyo único papel es acceder a la partición FAT de la tarjeta micro SD para cargar el archivo *bootcode.bin* en memoria. Esta partición contiene los siguientes archivos:

Nombre	Fecha de modifica...	Tipo	Tamaño
overlays	25/11/2016 17:24	Carpeta de archivos	
bcm2708-rpi-b.dtb	22/09/2016 9:07	Archivo DTB	14 KB
bcm2708-rpi-b-plus.dtb	22/09/2016 9:07	Archivo DTB	14 KB
bcm2708-rpi-cm.dtb	22/09/2016 9:07	Archivo DTB	14 KB
bcm2709-rpi-2-b.dtb	22/09/2016 9:07	Archivo DTB	15 KB
bcm2710-rpi-3-b.dtb	22/09/2016 9:07	Archivo DTB	16 KB
bcm2710-rpi-cm3.dtb	24/10/2016 12:41	Archivo DTB	15 KB
bootcode.bin	22/06/2016 8:06	Archivo BIN	18 KB
cmdline.txt	25/11/2016 17:52	Documento de tex...	1 KB
config.txt	25/11/2016 17:24	Documento de tex...	2 KB
COPYING.linux	21/08/2015 17:04	Archivo LINUX	19 KB
fixup.dat	25/11/2016 16:35	Archivo DAT	7 KB
fixup_cd.dat	25/11/2016 16:35	Archivo DAT	3 KB
fixup_db.dat	25/11/2016 16:35	Archivo DAT	10 KB
fixup_x.dat	25/11/2016 16:35	Archivo DAT	10 KB
issue.txt	25/11/2016 18:09	Documento de tex...	1 KB
kernel.img	25/11/2016 16:35	Archivos de imagen	4.032 KB
kernel7.img	25/11/2016 16:35	Archivos de imagen	4.133 KB
LICENCE.broadcom	18/11/2015 16:01	Archivo BROADC...	2 KB
LICENSE.oracle	25/11/2016 18:09	Archivo ORACLE	19 KB
start.elf	25/11/2016 16:35	Archivo ELF	2.756 KB
start_cd.elf	25/11/2016 16:35	Archivo ELF	619 KB
start_db.elf	25/11/2016 16:35	Archivo ELF	4.839 KB
start_x.elf	25/11/2016 16:35	Archivo ELF	3.813 KB

2. Etapa 2: carga de bootcode.bin

El siguiente esquema explica la primera fase de inicio de la Raspberry Pi, durante la que GPU carga el código de inicio en memoria.



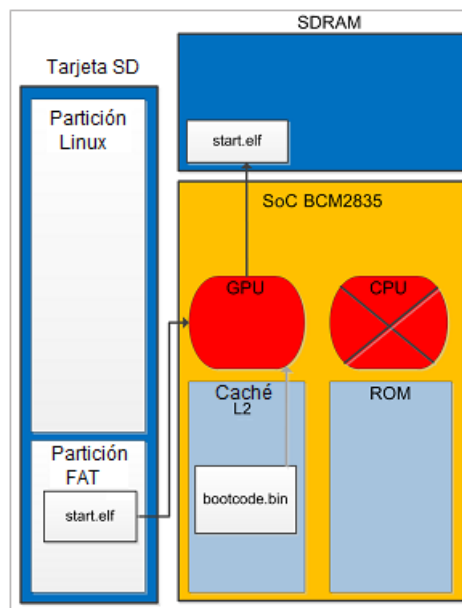
La GPU ejecuta el programa contenido en la ROM del SoC. Accede a una partición FAT de la tarjeta micro SD y encuentra el archivo *bootcode.bin*, que carga en la memoria caché L2 (*Level 2* = memoria caché de nivel 2).

El programa en la ROM, así como el programa *bootcode.bin*, se escriben en código específico para la GPU. Las especificaciones de la GPU no se han publicado, y *bootcode.bin* solo puede distribuirse en formato binario. Este no es un programa libre (licencia Broadcom).

Como la memoria SDRAM montada en el SoC todavía no está activada, es la memoria caché de la GPU, la memoria L2, la que se usa para la carga de *bootcode.bin*.

3. Etapa 3: ejecución de *bootcode.bin* por la GPU

El siguiente esquema explica la segunda fase de inicio de la Raspberry Pi, cuando la GPU carga el firmware (*start.elf*) en memoria.

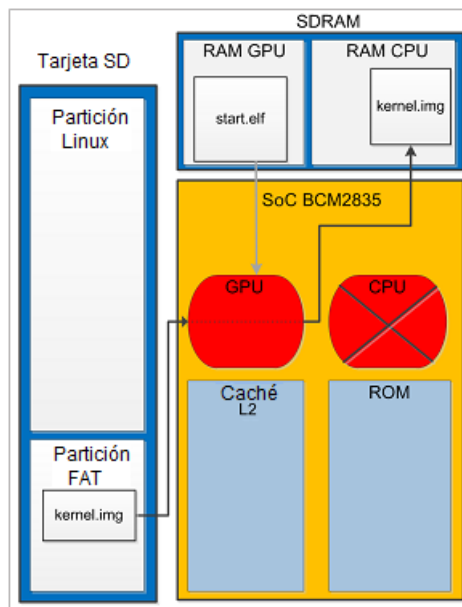


Una vez que *bootcode.bin* está cargado en la memoria caché L2, la GPU ejecuta este programa (flecha gris), que es el segundo nivel del bootloader. El objetivo final es recuperar el programa *start.elf*, que se sitúa también en la tarjeta micro SD, en la partición FAT. *start.elf* es el firmware de la GPU. Como *bootcode.bin*, este programa no es libre y se distribuye en formato binario.

La GPU, bajo las órdenes de *bootcode.bin*, activa la memoria SDRAM de la Raspberry Pi y transfiere una copia de *start.elf* a la parte superior de la memoria. Una vez que *start.elf* está cargado en memoria, *bootcode.bin* le pasa el testigo.

4. Etapa 4: ejecución de *start.elf* por la GPU

El siguiente esquema indica cómo la GPU comparte la memoria con la CPU en función de los argumentos y después carga el núcleo Linux en memoria.



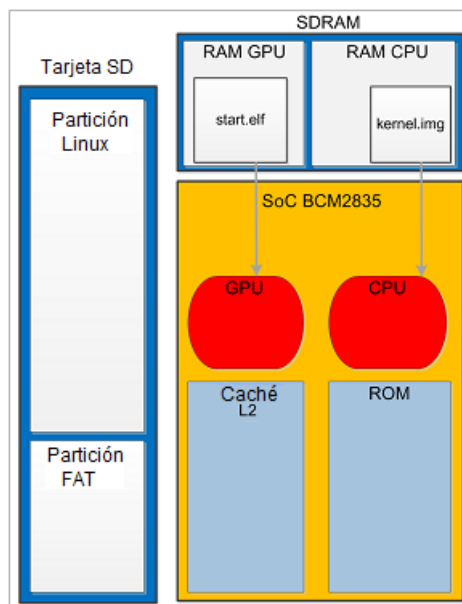
La GPU ejecuta ahora su firmware: *start.elf* (flecha gris).

start.elf reparte la memoria entre la GPU (parte superior) y la CPU ARM (parte inferior), en función del argumento presente en el archivo de configuración (*config.txt*). El argumento *gpu_mem* indica, en MB, la cantidad de memoria que se debe asignar a la GPU. Si este argumento no está presente en el archivo *config.txt*, el valor se fija por defecto a 64 MB. El valor de *gpu_mem* es, como mínimo, igual a 16 MB. Debe ser múltiplo de 16 MB.

Después del establecimiento de las zonas de memoria, el programa *start.elf* transfiere el núcleo Linux *kernel.img* a la parte inferior de la memoria, zona reservada a la CPU ARM. A continuación, lee el archivo *cmdline.txt*, que contiene los argumentos que debe pasar al núcleo durante su ejecución.

5. Etapa 5: ejecución de kernel.img por la CPU ARM

El siguiente esquema explica el inicio de la CPU y el inicio del núcleo Linux.

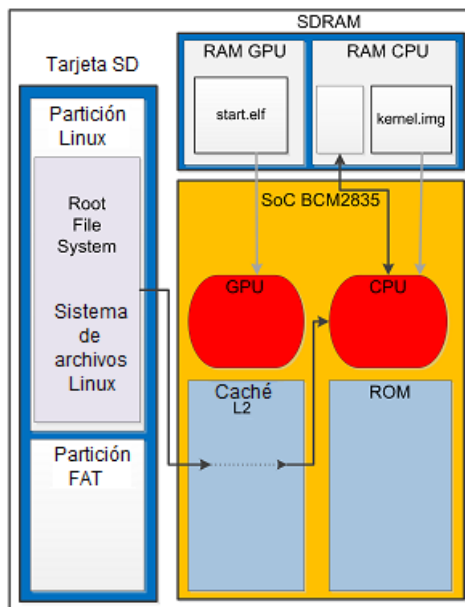


Antes de convertir simplemente el firmware de la GPU, lo que garantizará en adelante el funcionamiento como procesador gráfico (flecha gris entre GPU y *start.elf*), el programa *start.elf* va a "despertar" a la CPU ARM relanzando el reset (puesta a cero) que activó cuando se enchufó a la corriente.

La GPU queda autorizada para funcionar, la CPU arranca y ejecuta el programa *kernel.img*, que se encuentra en su zona memoria. El núcleo se carga y Linux arranca.

6. Etapa 6: inicio de Linux

El siguiente esquema explica el montaje del sistema de archivos raíz (*Root File System*).



El núcleo Linux ya arrancado busca ahora el RFS (*Root File System* = sistema de archivos raíz), que es el sistema de archivos básico de Linux, ya que contiene la raíz del sistema de archivos llamado / o *root directory* (= directorio raíz). Es en este sistema de archivos donde se montan (o añaden) los demás sistemas de archivos. Es decir, que incluso aunque físicamente los demás sistemas de archivos estén en soportes diferentes (llave USB, memoria, disco duro, etc.), todos forman parte de un único sistema de archivos después del inicio del sistema.

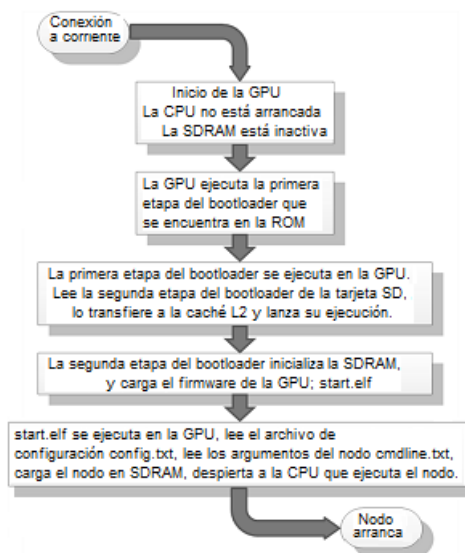
El acceso a la tarjeta micro SD por la CPU ARM se hace a través de la memoria caché de nivel 2, cuya función es anticipar los accesos a la tarjeta micro SD de la CPU y poner a su disposición las franjas de memoria que le interesan en L2. Esta memoria, cercana al procesador, es más rápida que la tarjeta micro SD y su uso acelera el acceso de la CPU a los programas y datos almacenados en la tarjeta micro SD. Un controlador de memoria caché L2 (*Level 2 Cache Controller*) se encarga de gestionar esta memoria y de la transferencia de las franjas de memoria.

El correcto inicio de Linux quedará confirmado por la aparición de la línea de comandos que le pide que se identifique.

Cada una de las etapas anteriores puede dar lugar a algún error. Puede suceder que el archivo esté corrupto (errores durante la transferencia a la tarjeta micro SD), ausente, incorrectamente llamado, etc. En cualquiera de los casos la cadena se interrumpe. Algunas veces, el mensaje de error es lo suficientemente claro para descubrir el defecto, pero en otras ocasiones solo el conocimiento de las etapas de la cadena de inicio permite localizar el error.

La observación de los LED de la Raspberry Pi también es una fuente de conocimiento ya que, por ejemplo, el número de destellos emitidos por los LED indica el defecto, interrumpen el arranque. El capítulo Solución de problemas en la Raspberry Pi da pistas para explorar en caso de problema.

7. Resumen de la secuencia de boot de la Raspberry Pi



El organigrama resume las diferentes fases de inicio de la Raspberry Pi. Menos detallado que las seis etapas anteriores permite, sin embargo, darse cuenta del encadenamiento de las operaciones necesarias para obtener una línea de comandos en pantalla.

Instalación del sistema operativo

Debido a su precio, limitado voluntariamente, la Raspberry Pi se entrega sin instrucciones, sin disco duro ni sistema operativo. El usuario es el que debe suministrar la alimentación, periféricos (teclado y ratón), memoria de almacenamiento externa (tarjeta micro SD), pantalla y cable necesarios para unirla a la Raspberry Pi. También tiene que preparar la tarjeta micro SD para que la Raspberry Pi pueda iniciar el sistema operativo.

1. Acceso a la tarjeta micro SD

Para acceder a la tarjeta micro SD es necesario usar un lector de tarjetas, que puede estar integrado si su ordenador es una torre o un portátil. Adquiera un lector multi-tarjetas como el de la siguiente imagen. Este lector acepta varios tipos de tarjetas de memoria y se conecta con un puerto USB al PC. Permite preparar tarjetas micro SD para las diferentes Raspberry Pi.



Algunas tarjetas micro SD se entregan con un adaptador al formato SD en el que la tarjeta se circunscribe. Esto permite su uso en un lector sin puerto micro SD.

- Si tiene una tarjeta micro SD que funciona gracias a un adaptador de este tipo en un modelo A o B, actualice su tarjeta con la última imagen si desea usarla en una Raspberry Pi 3 o en una Raspberry Pi Zero. Solo las últimas versiones de Raspbian tienen en cuenta las particularidades de estas versiones.

2. Adquirir una tarjeta micro SD lista para usar

Es posible conseguir tarjetas micro SD listas para usar. En general, los vendedores de Raspberry Pi ofrecen una tarjeta de este tipo, que puede ser muy útil para un usuario que desee iniciar su Raspberry Pi sin tener que gestionar la compra de la tarjeta, la descarga y la instalación del sistema. Los precios se convierten a euros según el tipo de cambio oficial a junio de 2016.

Tarjetas micro SD pre-programadas para la Raspberry Pi			
Vendedor	OS instalado	Tamaño de la tarjeta	Precio
Adafruit USA	Raspbian Jessie	8 GB / Cl. 4	10,50 €
Adafruit USA	Raspbian Wheezy	4 GB / Cl. 4	8,75 €
Adafruit USA	NOOBS	8 GB / Cl. 4	10,50 €
Amazon ES	NOOBS	8 GB / Cl. 4	6,99 €
Kubii FR	NOOBS	8 GB / Cl. 10	9,90 €
Amazon GB	NOOBS	16 GB / Cl. 10	14,00 €

Esta solución, rápida, no necesita conocimientos en Linux. Se reserva a los casos urgentes o cuando la adquisición de conocimientos no es el principal objetivo de la puesta en marcha de la Raspberry Pi.

También puede servir como "rueda de repuesto" para un usuario principiante que ha preparado personalmente una tarjeta micro SD y cuya Raspberry Pi no arranca. La inserción de esta tarjeta micro SD de referencia puede ayudar a localizar el error. Si la Raspberry Pi no arranca con ninguna de las dos tarjetas es necesario buscar un fallo de hardware. Si la Raspberry Pi arranca con la tarjeta pre-programada, pero no con la tarjeta preparada por el usuario, es esto lo que es necesario cuestionarse.

- En el mercado encontramos tarjetas con versiones anteriores de los sistemas operativos. No funcionan con los últimos modelos de Raspberry Pi. Compruebe la compatibilidad con su modelo de tarjeta.

3. Instalación de un OS con NOOBS

Los sistemas operativos se entregan generalmente en los CD-ROM o DVD-ROM, con un procedimiento de instalación destinado a preparar el soporte que va a recibir el sistema.

En caso de la Raspberry Pi, el principio es un poco diferente. La tarjeta micro SD debe contener una partición FAT y una partición Linux. La partición FAT recibe los programas de boot, así como el núcleo Linux, mientras que la partición Linux alberga el *Root File System* o sistema de archivos raíz de Linux.

El sistema operativo se entrega como una imagen. Un archivo imagen *.img* es una copia binaria de un sistema instalado. Es decir que la etapa de instalación del sistema ya se ha realizado y a continuación una copia de cada byte del sistema al archivo imagen ha permitido obtener una copia idéntica del sistema de origen. Este archivo imagen está listo para ser copiado a la tarjeta micro SD, y es el que se ofrece para su descarga.

Las imágenes de los sistemas operativos recomendados por la Fundación Raspberry Pi están disponibles en el sitio <http://raspberrypi.org/downloads>. El resto de imágenes se descargan desde las páginas web que están dedicadas a esto. El sitio http://elinux.org/RPi_Distributions (respete las minúsculas y mayúsculas) enumera las distribuciones disponibles y ofrece enlaces a las páginas que permiten obtener información o descargar estas distribuciones.

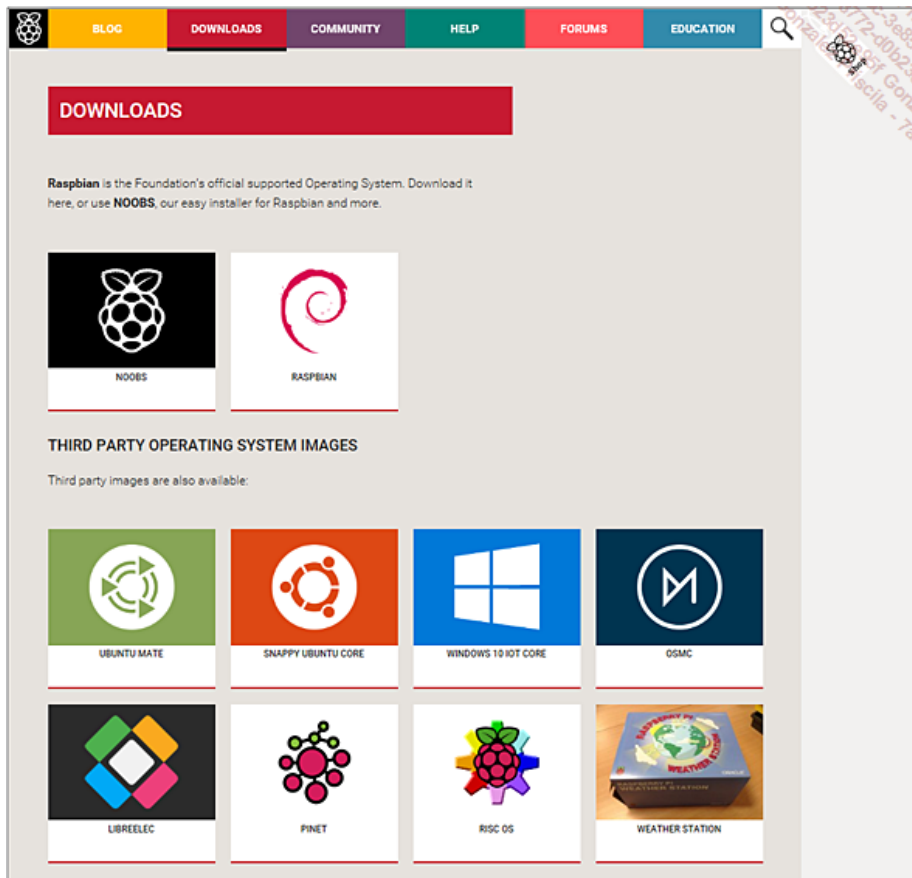
Para empezar, la solución más sencilla es confiar en la Fundación y, una vez adquiridas las competencias necesarias probar otros OS, experimentar e incluso, por qué no, realizar su propia distribución.

a. El sitio de la Fundación

El sitio de la Fundación ofrece la descarga de NOOBS o de las imágenes de sistemas operativos a transferir a una tarjeta micro SD.

NOOBS (*New Out Of Box Software*) se puede descargar desde junio de 2013. Se trata de un "nuevo software que funciona desde el primer momento", lo que casi es el caso aquí; es necesario hacer un mínimo de operaciones. La Fundación recomienda NOOBS para los usuarios principiantes de la Raspberry Pi.

La elección del nombre no es inocente, ya que es así como se llaman los principiantes o *newbies*. En el dominio de los videojuegos, un *newbie* es un neófito. La palabra ha evolucionado en *Newb* y después en *Noob*, que hace referencia a un jugador experimentado que comete errores de principiante...



- La página de descarga del sitio <http://raspberrypi.org> se actualiza regularmente y su presentación puede diferir de la que aparece en la captura de pantalla anterior.

La página **Downloads** del sitio *raspberrypi.org* reúne todos los sistemas ofrecidos para su descarga para la Raspberry Pi por la Fundación.

b. NOOBS

NOOBS no es un sistema operativo. Es un sistema de recuperación de la tarjeta micro SD y un administrador de multiboot. Algunos fabricantes de PC ofrecen la posibilidad de una reinstalación de fábrica de un PC, activando una combinación de teclas durante el inicio del PC. Esta operación permite reinstalar un sistema "limpio" en caso de grave infección por algún virus, imposibilidad de reiniciar el sistema operativo, etc. De la misma manera, NOOBS deja al usuario la posibilidad de pulsar en la tecla [Shift] de su máquina durante el inicio e instalar uno o varios sistemas operativos, elegidos de entre las imágenes ofrecidas por NOOBS.

Si el usuario no pulsa la tecla [Shift] durante el inicio de NOOBS, alguno de los sistemas instalados anteriormente en la tarjeta micro SD arranca con normalidad. Es el sistema que ha arrancado durante el comienzo anterior el que arranca de nuevo, salvo si el usuario indica otro.

Los sistemas operativos presentes en NOOBS (a primeros de junio de 2016) son: Raspbian, LibreELEC para Pi2, OSMC para Pi2, RISC OS y Windows 10 IoT Core. Aquí incluso se recomienda la instalación de Raspbian.

También es posible añadir una partición vacía de 512 MB a las particiones que NOOBS creará.

NOOBS existe en dos versiones. La versión *Lite* es ultra ligera (27 MB), aunque todos los sistemas operativos se cargan sistemáticamente desde Internet. Por tanto, la Raspberry Pi debe tener acceso a Internet.

La versión completa (más de 1 GB) incluye únicamente Raspbian y permite realizar la instalación en una máquina no conectada a Internet. La instalación de los demás sistemas ofrecidos por NOOBS durante la instalación se hará obligatoriamente descargando los sistemas de Internet.

c. Descarga de NOOBS

NOOBS se ofrece para su descarga en formato .zip, de manera directa o en .torrent. La descarga directa se hace desde alguno de los servidores espejo del servidor de la Fundación. La versión .torrent utiliza el protocolo BitTorrent, que permite la descarga desde ordenadores de otras personas conectadas, poniendo las partes ya descargadas en su máquina a disposición de los demás usuarios. Un cliente .torrent se debe instalar en la máquina prevista para realizar la descarga.

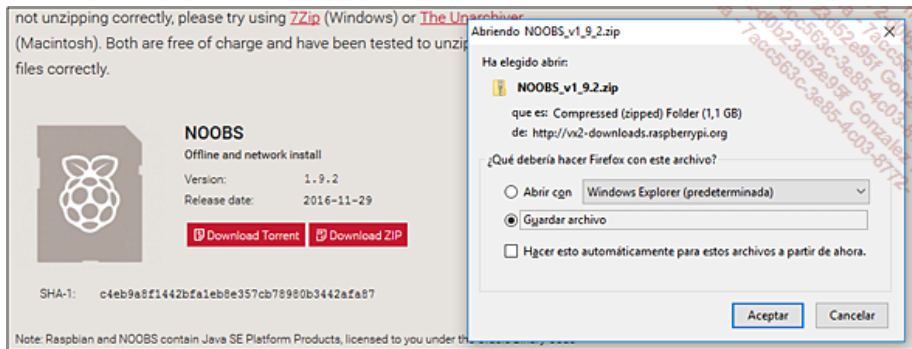
- La versión de NOOBS disponible en el momento de escribir este libro es la versión la 1.9. Tiene que adaptar las instrucciones en función de la versión disponible cuando usted lea este capítulo.

Para realizar una descarga del archivo .zip de NOOBS:

Pulse en el botón **Download ZIP** situado debajo de la fecha de aparición de la versión de NOOBS (*Release date*).



Aparece la ventana de descarga. El servidor ofrece la descarga desde un servicio espejo que selecciona.



Pulse en **Guardar archivo**. En la ventana que se abre a continuación, indique dónde desea guardar el archivo y termine pulsando el botón **Guardar**. NOOBS ocupa en torno a 1 GB. Espere hasta el final de la descarga.

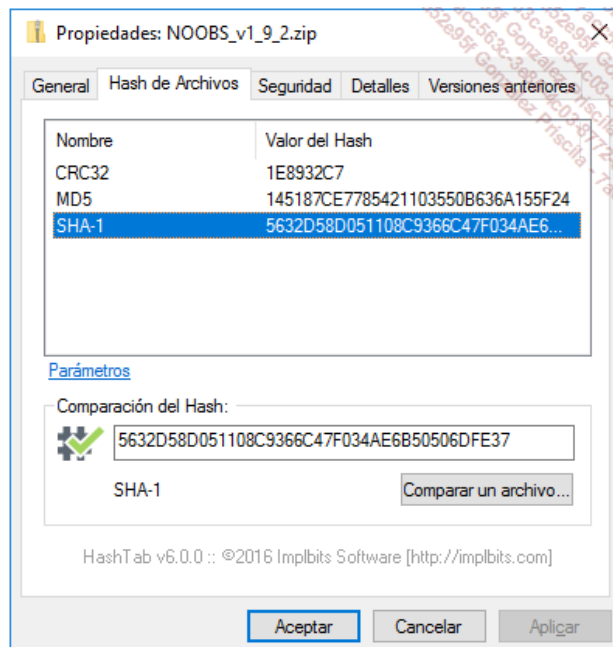
Hash SHA-1

Se ofrece un hash SHA-1 en la página de descarga que permite comprobar si el archivo recibido ha llegado sin haber sido corrompido ni modificado. SHA-1 o *Secure Hash Algorithm* hace referencia a una función que, partiendo de un conjunto de datos (aquí nuestro archivo *NOOBS_v1_9_2.zip*), realiza un determinado número de operaciones. Al final proporciona un conjunto de 160 bits, es decir, 20 bytes o incluso 40 caracteres hexadecimales, que es un tipo de firma digital del archivo, que permite identificarlo de manera segura. La modificación de un solo byte en el archivo genera un código SHA-1 completamente diferente.

Comprobación de la descarga

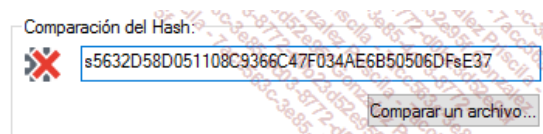
Un software como *Hashtab* disponible para Windows y Mac (<http://implbits.com/products/hashtab/>), gratuito para los particulares, estudiantes y asociaciones, permite comprobar la integridad del archivo. Aunque existen otras soluciones por línea de comandos en Linux o en modo gráfico en Windows.

Hashtab calcula el hash de un archivo y lo muestra, haciendo clic con el botón derecho del ratón en el nombre del archivo, seleccionando la opción **Propiedades** en el menú que se abre y, para terminar, en la pestaña **Hash de Archivos** de la ventana **Propiedades**.



El sitio raspberrypi.org indica que el hash SHA-1 del archivo debe ser: *5632D58D051108C9366C47F034AE6B50506DFE37*. Para comparar el resultado obtenido por *Hashtab* con este valor, basta con copiar el hash mostrado por el sitio web en la zona de texto **Comparación del Hash** de la ventana de *Hashtab*. Si ambos valores son idénticos, una marca de verificación indica el éxito de la comparación.

En caso de diferencia, aparece una cruz para indicar que los hash son diferentes:



➤ Aquí, el "7" con el que termina la cadena proporcionada por raspberrypi.org se ha modificado por un "8" por necesidades de la demostración. El resultado sería el mismo si el archivo recibido se hubiera corrompido o fuera diferente al archivo original en un único bit.

d. Transferencia de NOOBS a la tarjeta micro SD

Tipo de tarjeta micro SD a usar

La Fundación recomienda en las instrucciones de instalación de *NOOBS* una tarjeta micro SD de clase 4, de 8 GB como mínimo. Es preferible usar tarjetas de marca, porque son más fiables. El sitio linux.org/RPi_SD_cards enumera las tarjetas micro SD comprobadas y validadas, así como las tarjetas que no funcionan con la Raspberry Pi. Una tabla resume los rendimientos medios en modo lectura y escritura de estas tarjetas.

Descarga de la herramienta SDFormatter

Para la instalación de *NOOBS* en la tarjeta micro SD, la Fundación Raspberry Pi recomienda usar una herramienta de formateo: *SDFormatter 4.0 for SD/SDHC/SDXC*. Este programa se ofrece de manera gratuita por SD Association, la asociación que agrupa todas las empresas implicadas en el desarrollo de tarjetas micro SD.

➤ En general, las tarjetas micro SD nuevas funcionan sin necesitar el uso de *SDFormatter*. Por el contrario, una tarjeta que se haya usado en un aparato de fotos o una Raspberry Pi se debería formatear con esta herramienta.

¿Por qué usar este programa en lugar de formatear la tarjeta micro SD con Windows?

- *SDFormatter4* se ha creado específicamente para las tarjetas micro SD. Según SD Association, el formateo genérico realizado por un sistema operativo puede reducir el rendimiento de una tarjeta micro SD.
- Windows 8 solo reconoce una única partición en una tarjeta micro SD, a condición de que esté en formato FAT. Si la tarjeta micro SD ya se ha utilizado en Linux y se ha particionado con un formato Linux o contiene varias particiones, la opción de formateo de Windows 8 solo tendrá en cuenta una única partición FAT, si existe. La visualización en Windows del contenido de una tarjeta micro SD de 4 GB que contiene una partición FAT de 50 MB y una partición Linux de 3,9 GB significa que el tamaño de la tarjeta es de... 50 MB.

Esta herramienta está disponible para todos los sistemas Windows, de XP a Windows 8, y para Mac de la versión OS X 10.5 a la 10.8. Se puede descargar en la página web https://www.sdcard.org/downloads/formatter_4/

Seleccione la versión que desee (Windows o Mac).

Acepte la licencia pulsando en el botón **I Accept** en la parte inferior de la página.

Una ventana de descarga ofrece abrir o guardar el programa *SDFormatter4.exe.zip*, que tiene un tamaño de 6 MB.

Seleccione **Guardar el archivo** y después pulse en **Aceptar**.

En la ventana que aparece, seleccione la carpeta en la que desea guardar el archivo y después pulse en **Guardar**.

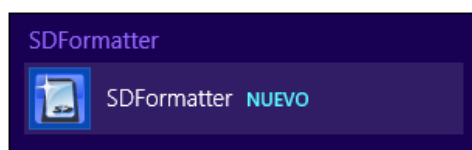
Un archivo que contiene *SDFormatter4* está disponible en la carpeta elegida. Descomprima este archivo en la carpeta que prefiera.

Instalación de SDFormatter

Haga doble clic en el archivo *setup.exe* en la carpeta que se ha creado.

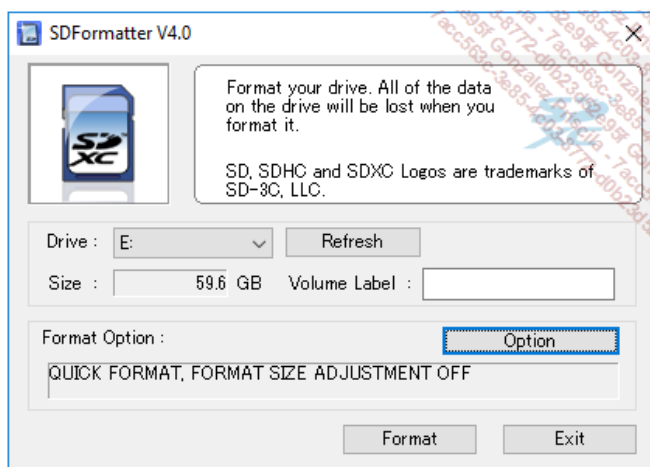
La instalación comienza. Valide el resto de opciones por defecto, eventualmente autorice la instalación de la herramienta si el sistema la solicita.

SDFormatter aparece ahora en la lista de aplicaciones instaladas.



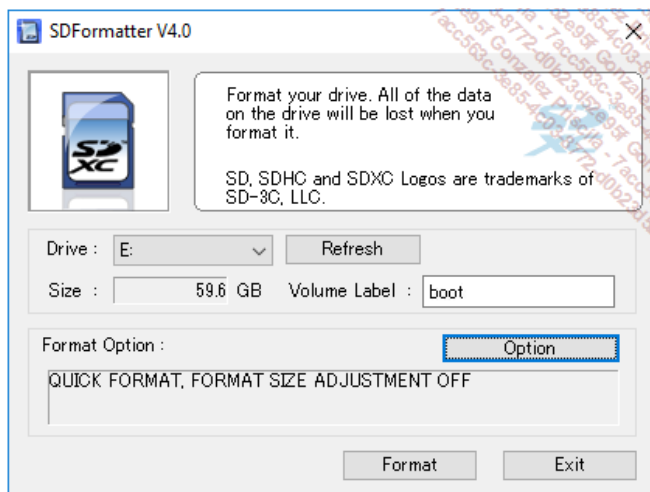
Formateo de la tarjeta micro SD

Haga clic con el botón derecho del ratón en **SDFormatter** en la lista de aplicaciones, y pulse en **Ejecutar como administrador**.

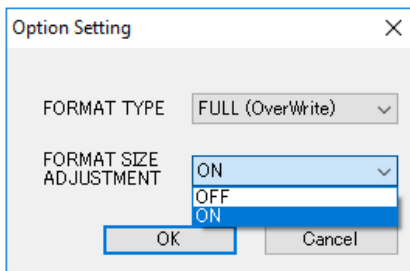


Se abre la ventana de la aplicación. Compruebe que el lector que se indica a la derecha de la etiqueta **Drive** se corresponde con su tarjeta micro SD. Aquí es el lector **E:** y se trata correctamente de la tarjeta micro SD. El tamaño de la tarjeta micro SD está justo debajo, a la derecha de la etiqueta **Size**.

Si la tarjeta ya se ha utilizado en una Raspberry Pi, es posible que este tamaño sea inferior al tamaño real y que la zona **Volume Label** contenga **boot**, como en la siguiente imagen.



Pulse el botón **Option**.



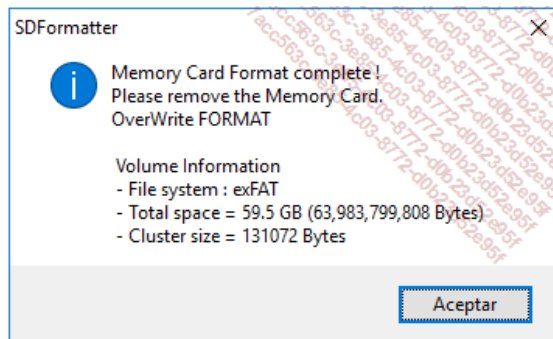
Para formatear la tarjeta completamente, seleccione **FORMAT TYPE = FULL (OverWrite)** y configure **FORMAT SIZE ADJUSTMENT a ON**.

Un formateo parcial elimina simplemente la tarjeta sin reinicializarla totalmente. Estas opciones garantizan un formateo completo y una re-inicialización de todas las zonas de la tarjeta micro SD. La opción **FORMAT SIZE ADJUSTMENT** ajusta la capacidad de la tarjeta formateada a un múltiplo exacto de un sector (512). Esta opción puede reducir ligeramente la capacidad de la tarjeta. Algunas máquinas tienen problemas para acceder a la tarjeta si su tamaño no es un múltiplo de un sector (512).

Valide las opciones pulsando **OK** y después pulse el botón **Format** de la ventana **SDFormatter V4.0**.

Una ventana le advierte que no quite la tarjeta micro SD durante las operaciones de formateo y le pregunta si desea formatear: responda **OK**.

El formateo arranca, puede durar mucho tiempo en función del tamaño de la tarjeta micro SD. Espere hasta que **SDFormatter** muestre la siguiente ventana:

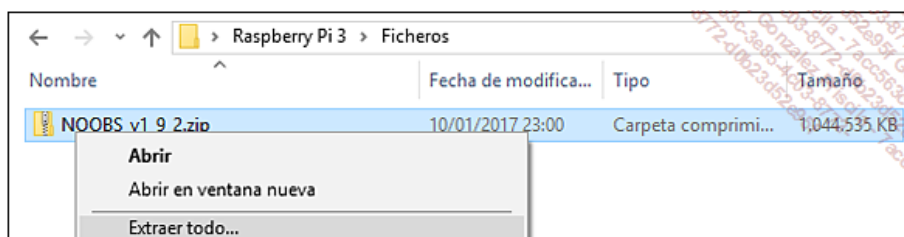


Compruebe la información que se muestra. Su tarjeta micro SD está lista para recibir NOOBS.

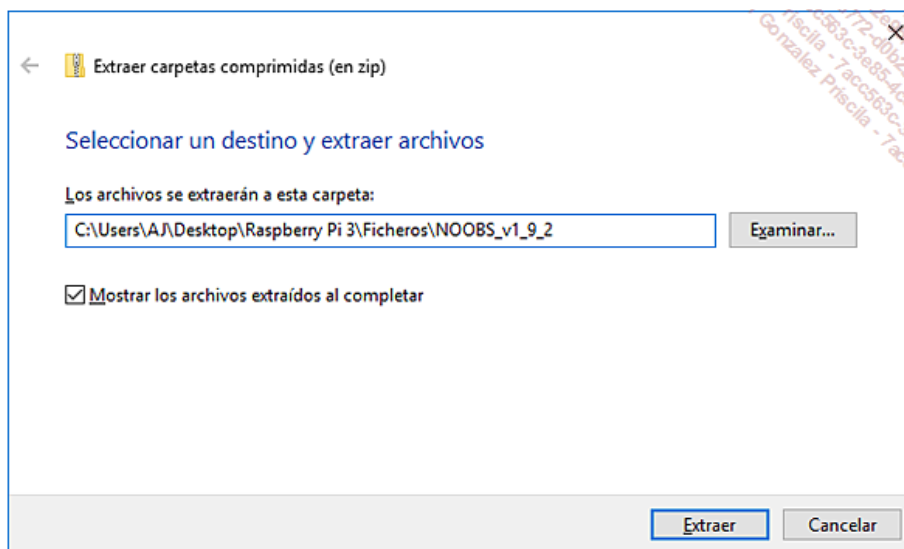
- Tarjeta 64 GB: SDFormatter formatea las tarjetas micro SD cuyo tamaño sobrepasa 32 GB en exFAT. Este formato no se reconoce por la Raspberry Pi. Convierta la tarjeta en FAT32 con una herramienta como FAT32Formatter, por ejemplo (<http://www.ridgecrop.demon.co.uk/download/fat32format.zip>).

Copia de NOOBS en la tarjeta micro SD

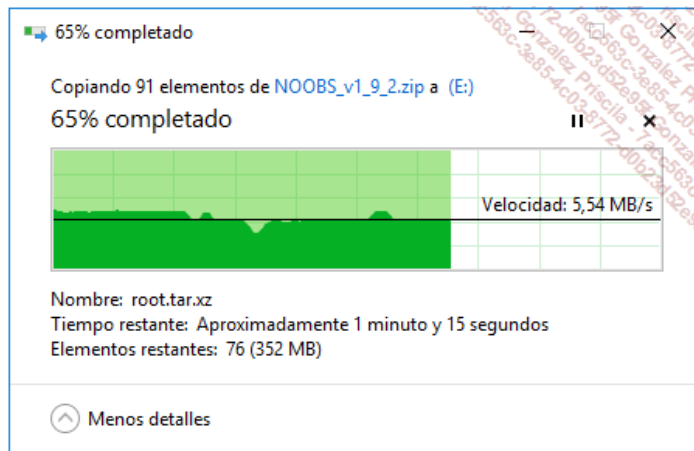
En la carpeta donde ha almacenado el archivo *NOOBS_v1_9_2.zip*, haga clic con el botón derecho del ratón en el archivo y seleccione **Extraer todo...**



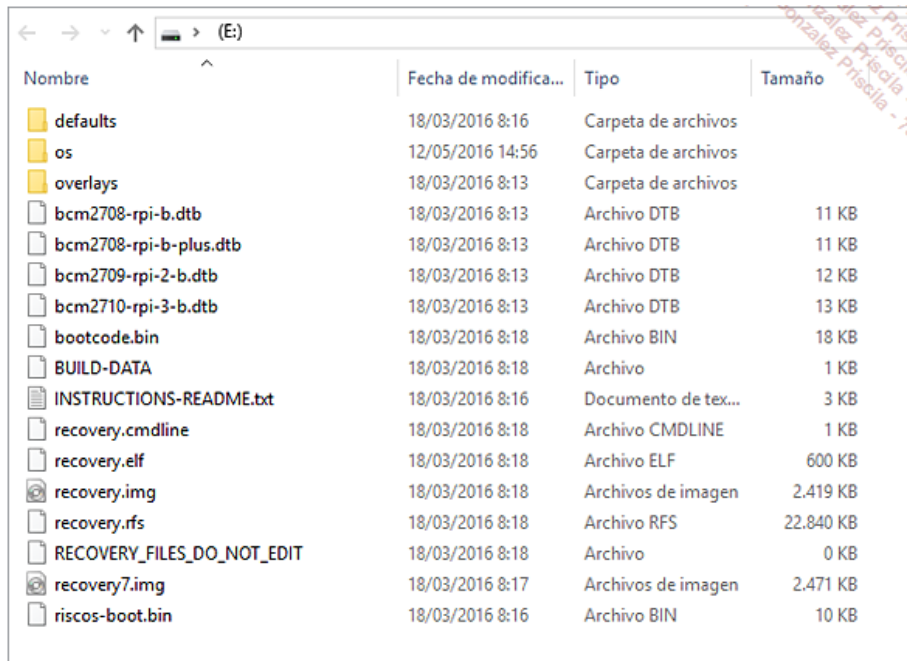
Windows le ofrece un destino por defecto. Se trata de la carpeta en la que ha guardado el archivo de NOOBS.



Valide el destino que le ofrece Windows o modifíquelo según sus necesidades. Inicie la descompresión pulsando el botón **Extraer**.

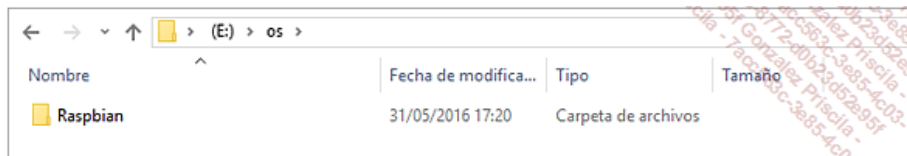


Cuando la operación termina, obtiene una carpeta con los archivos de NOOBS. Es el conjunto de estos archivos lo que necesita copiar a su tarjeta micro SD. Seleccione todo el contenido de la carpeta en la que Windows ha descomprimido el archivo y haga copiar-pegar a la tarjeta micro SD.

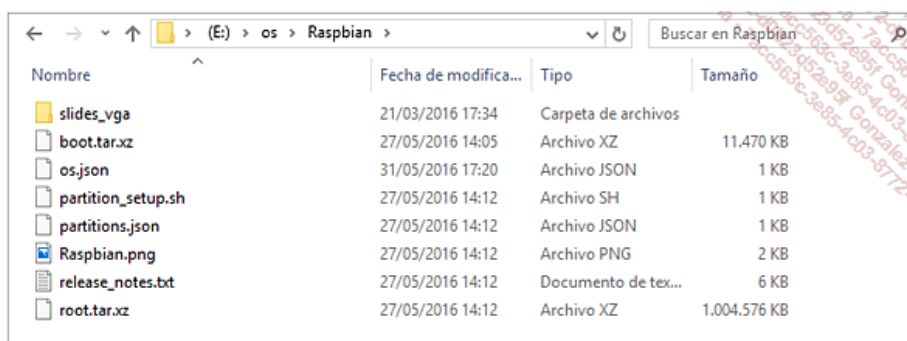


Como resultado de la copia, su tarjeta micro SD está lista para el uso en la Raspberry Pi.

La carpeta contiene un directorio llamado *os*. Es en esta carpeta donde se almacena la que contiene la imagen utilizada por NOOBS para instalar Raspbian, cuando el usuario elige este sistema operativo.



Veamos el contenido de la carpeta *Raspbian*:



La carpeta *slides_vga* contiene las imágenes del slideshow que se mostrará durante la instalación para entretener al usuario y ofrecerle información del sistema operativo. La imagen de Raspbian ocupa en torno a 1 GB.

➤ Las imágenes de los sistemas operativos están en formato **.xz**, que es un formato abierto de compresión de datos.

e. Instalación de Raspbian con NOOBS

NOOBS permite instalar varios sistemas operativos. El sistema operativo recomendado por la Fundación es Raspbian. A continuación se muestra cómo instalarlo desde NOOBS.

Inicio de NOOBS

Para iniciar NOOBS:

Introduzca la tarjeta micro SD que contiene NOOBS en el conector situado bajo la Raspberry Pi.

Conecte el teclado y el ratón.

Conecte la pantalla con el cable HDMI o el cable de vídeo.

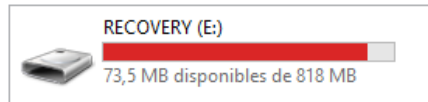
Conecte el cable Ethernet (es opcional para Raspbian si ha descargado la versión completa, pero obligatorio para el resto de sistemas operativos).

Conecte la alimentación a la toma microUSB.

Enchufe a la corriente una toma de salida.

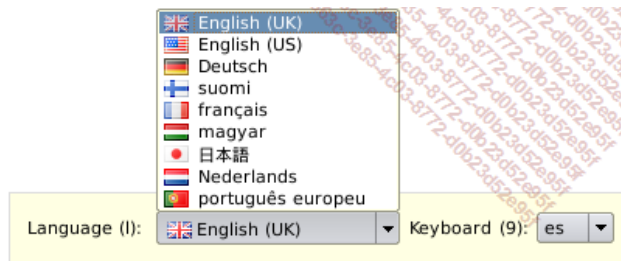
Los LED de la Raspberry Pi se deberían encender y parpadear.

Durante el primer inicio de NOOBS, se ejecuta una herramienta de instalación. Este instalador prepara la tarjeta micro SD, que solo contiene inicialmente una única partición FAT. Después de la instalación, la tarjeta contiene la partición FAT que contiene NOOBS y las particiones Linux para los OS. Si vuelve a leer ahora la tarjeta micro SD en una máquina Windows, solo verá una partición de 1 GB llamada *RECOVERY*. El resto de la tarjeta está en formato Linux y se ignora por Windows, que solo acepta una única partición en una tarjeta micro SD o micro SD.



Después de la secuencia de inicio, NOOBS se inicializa y le ofrece una pantalla en la que puede seleccionar el sistema que desea instalar.

En la parte inferior de esta pantalla, una lista le permite seleccionar el idioma utilizado. Seleccione **English(UK)** o el idioma que desee. El teclado se configura automáticamente en inglés. Pero NOOBS va mucho más lejos, porque la elección del idioma va a influir en los sistemas operativos que instale desde la interfaz: cualquier instalación de Raspbian se hará con una configuración automática en inglés.



Ahora puede instalar uno o varios sistemas operativos.

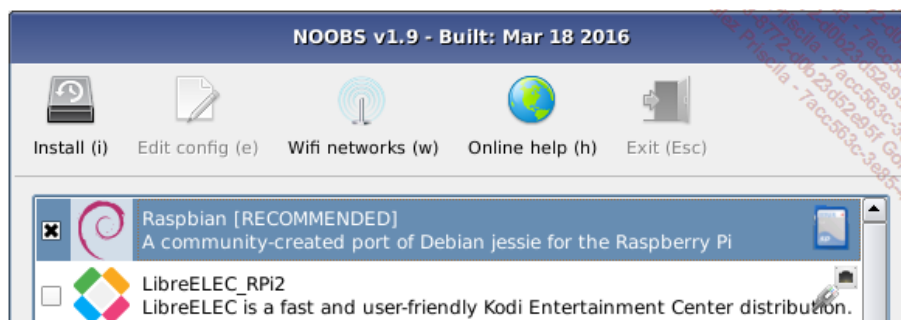
Instalación de Raspbian

Por defecto, NOOBS ofrece la instalación de Raspbian. Como es la distribución recomendada por la Fundación, es la que vamos a instalar.

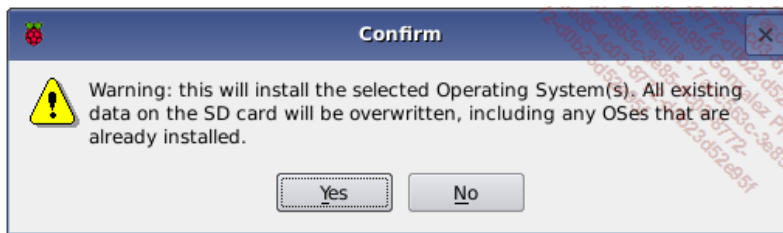


La captura de pantalla anterior tiene a la derecha de cada fila un icono. Si este icono representa un conector RJ45, significa que el sistema operativo que figura en la fila se debe descargar en línea. Vemos aquí que solo se usa la primera línea en local y las cinco filas siguientes implican que la Raspberry Pi esté conectada a un acceso a Internet.

Marque la casilla de selección situada a izquierda de la fila **Raspbian [RECOMMENDED]** o haga doble clic en la fila para marcar esta casilla. El icono **Install (i)** se debe activar.

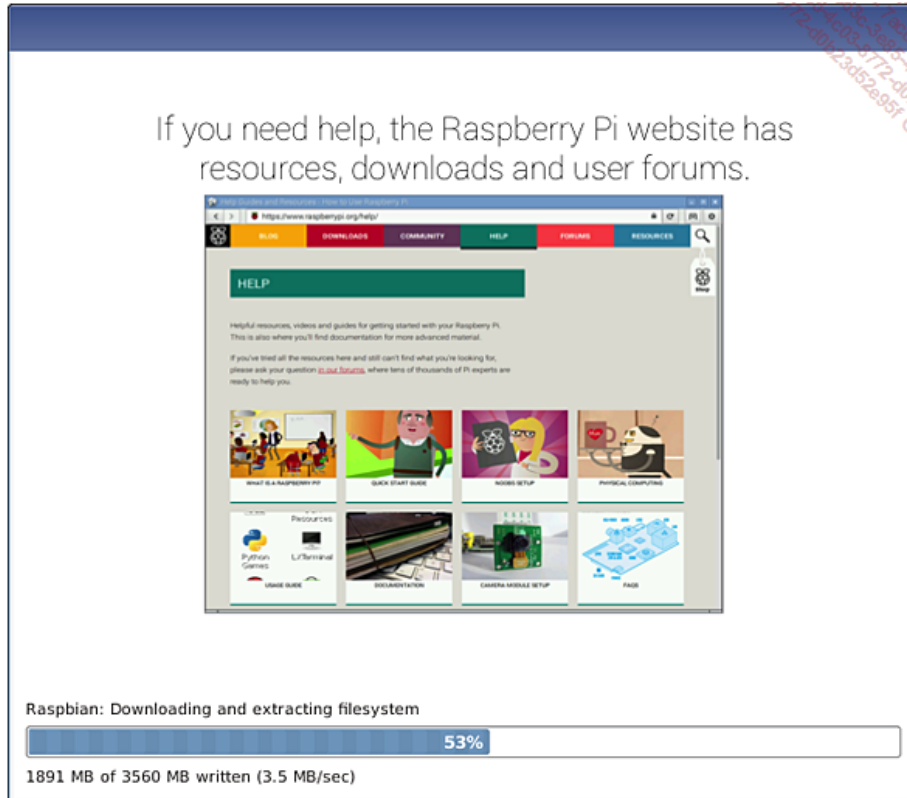


Pulse ahora en el icono **Install (i)** en la parte superior de NOOBS.



La ejecución de las operaciones implica que NOOBS va a instalar la distribución seleccionada en la tarjeta micro SD y a eliminar los datos que pueda haber. Una ventana le solicita autorización para borrar los datos presentes en la tarjeta. Pulse el botón **Yes**.

La instalación de Raspbian arranca. Una barra de progreso le indica el avance de la operación.



La Raspberry Pi debe descomprimir el archivo de Raspbian, que ocupa 1 GB cuando está comprimido, crear la imagen del OS (3,2 GB) a partir de este archivo y después copiarlo en la tarjeta micro SD. El OS está compuesto por varios miles de archivos de pequeño tamaño, lo que explica la duración de la operación. Para hacer que el usuario espere pacientemente durante el (largo) tiempo de instalación, se muestran varias pantallas de información en inglés. Estas son las que están en la carpeta *slides_vga* que hemos visto anteriormente.

Después de algunos minutos, Raspbian está instalado en su Raspberry Pi.



Valide la instalación pulsando en **OK**.

NOOBS arranca de nuevo.

En cada inicio, NOOBS le presenta una pantalla que permite lanzar una recuperación de la tarjeta micro SD.

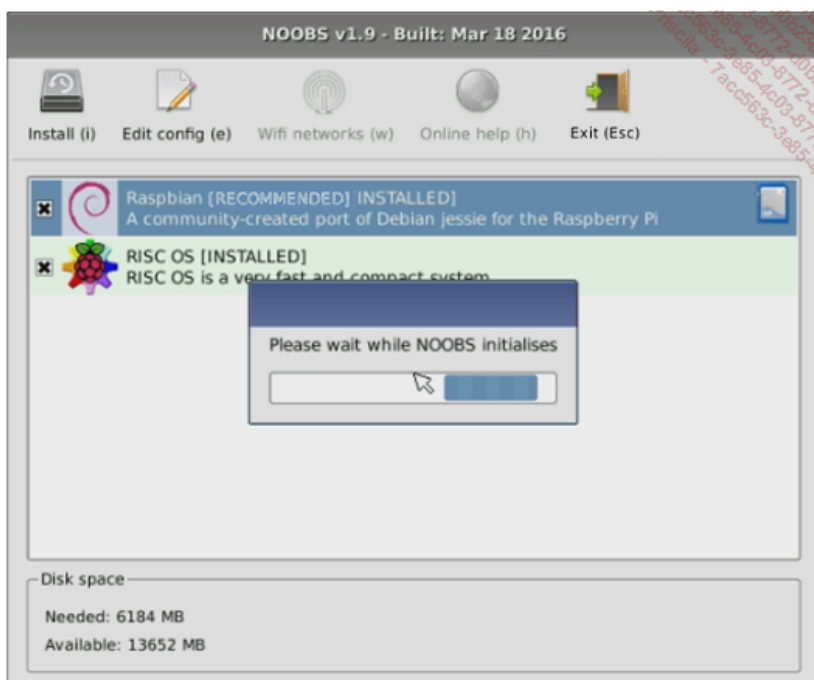


NOOBS invita al usuario a pulsar la tecla [Shift] durante la visualización de esta pantalla. Si el usuario no hace nada, el inicio de Raspbian se producirá normalmente, pasados algunos segundos.

Algunos teclados llevan más tiempo para su instalación, que no dura el inicio de NOOBS. Si la tecla [Shift] de su teclado parece inactiva, intente con otro teclado (otra marca o modelo).

Modo recuperación

Si el usuario pulsa la tecla [Shift], aparece la ventana de bienvenida de NOOBS y es posible regenerar la tarjeta micro SD con el mismo sistema operativo o con uno o varios de entre los demás ofrecidos por NOOBS. La siguiente pantalla se muestra durante una solicitud de recuperación de una tarjeta micro SD en la que se han instalado Raspbian y RISC OS. Estos dos sistemas están marcados para una eventual reinstalación y la mención **[INSTALLED]** figura junto al nombre del OS Raspbian que se instala en la tarjeta. La Raspberry Pi no está conectada a la red y vemos que únicamente Raspbian está disponible para una reinstalación (icono en forma de tarjeta micro SD). Para reinstalar RISC OS, hay que volver a conectar la Raspberry Pi a la red para que pueda acceder a Internet.



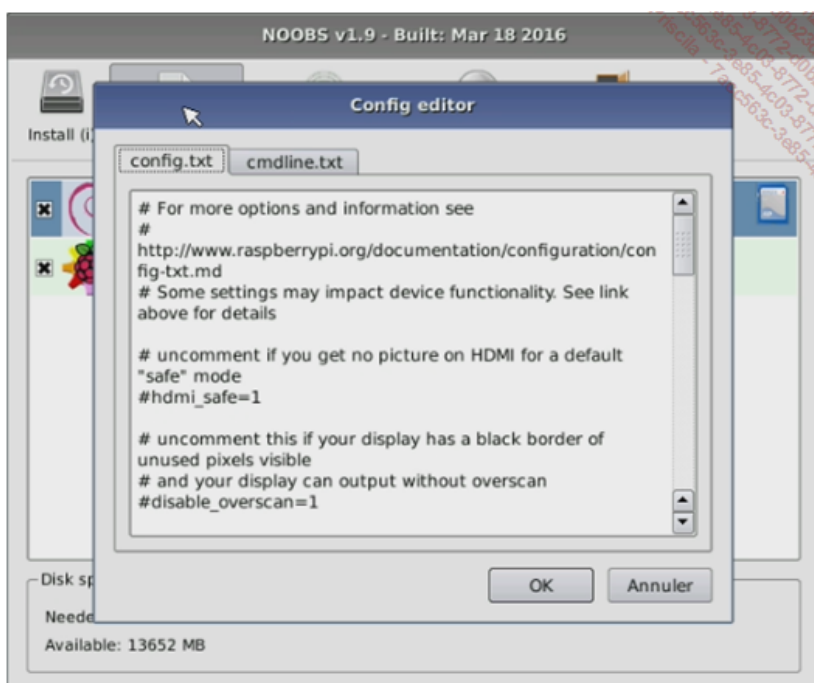
Los iconos de la parte superior de la ventana permiten:

- Instalar el o los OS seleccionados.
- Editar los archivos de configuración del OS seleccionado.
- Abrir una página web en el foro del sitio de la Fundación.
- Salir de NOOBS.

Edición de los archivos de configuración

Los archivos de configuración accesibles mediante el icono **Edit config** son *config.txt* y *cmdline.txt*. El ajuste de los argumentos de los sistemas es realmente fácil con **Config editor**.

Seleccione el sistema operativo a configurar pulsando en su fila. Esta queda destacada. Pulse en el icono **Edit config**. Se abre un editor de texto con dos pestañas, cada una apuntando a uno de los archivos a modificar.

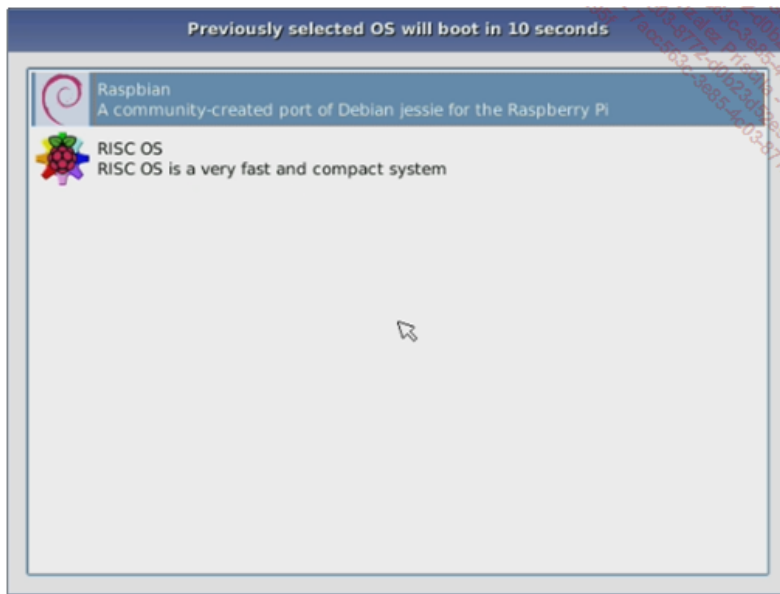


Después de realizar las modificaciones que desee a los archivos, un clic en el botón **OK** actualiza los archivos de configuración del OS implicado.

Multiboot

Si el usuario no pulsa la tecla [Shift] y solo hay instalado un sistema, es este sistema operativo el que arranca pasados algunos segundos (impuestos por NOOBS).

Si se instalan varios sistemas, una pantalla permite seleccionar de una lista el OS que va a cargar la Raspberry Pi.



Si el usuario no interviene durante 10 segundos, el sistema utilizado anteriormente es el que se va a cargar de nuevo (aquí Raspbian, es decir, la fila seleccionada). Para iniciar otro sistema presente en la tarjeta micro SD basta con pulsar en la fila correspondiente. Esto permite arrancar en el sistema seleccionado por el usuario. Esta funcionalidad es particularmente útil para experimentar diferentes sistemas operativos.

f. Conclusión

La Fundación, después de la aparición de la tarjeta Rev. 2 de la Raspberry Pi, había anunciado que el acento se pondría en el desarrollo de software. La aparición de NOOBS en junio de 2013 es una primera etapa en la puesta a disposición de un medio sencillo para iniciar la Raspberry Pi. Preparación de la tarjeta micro SD, instalación de un sistema elegido entre varios, regeneración rápida de la tarjeta micro SD: se reunieron todos los ingredientes para facilitar el inicio de una Raspberry Pi, incluso por un principiante.

La aparición de una versión 1.3 de NOOBS en septiembre de 2013 ha marcado una nueva etapa en el desarrollo de software con la aparición del multiboot. Los principiantes y los usuarios que habitualmente necesitan arrancar en OS diferentes adoptaron NOOBS.

4. Instalación de una imagen Raspbian

La Fundación ofrece las imágenes RAW. RAW significa bruto en inglés. Por tanto, el archivo a descargar es una copia "en bruto" de un sistema instalado. Cada byte del archivo es exactamente igual a cada byte correspondiente del sistema de origen. Si descendemos a nivel de bit, podemos decir que cada bit del archivo es estrictamente equivalente al bit correspondiente del sistema de partida. De ahí el nombre de imagen binaria que también se da a este tipo de archivo.

La Fundación ofrece para su descarga Raspbian Jessie, un OS basado en Debian 8 Jessie. También están disponibles Ubuntu Mate, Snappy Ubuntu Core, Windows 10 IoT Core, OSMC, LibreElec, Pinet y RISC OS. Su instalación a partir de una imagen se hará de la misma manera que para Raspbian.

Estas distribuciones están disponibles para su descarga como un archivo, directamente desde el servidor de la Fundación o desde el servidor del proveedor.

Para alguna de entre ellas, como la Raspbian anterior, se proporciona el hash SHA-1 que permite comprobar la integridad del archivo descargado.

En la página de descarga del sitio raspberrypi.org, seleccione la versión que le interesa.

Opte por Raspbian si comienza con la Raspberry Pi.

Pulse en el enlace **Download ZIP** de Raspbian y después proceda como anteriormente durante la descarga de NOOBS.

Guarde la imagen del sistema en una carpeta.

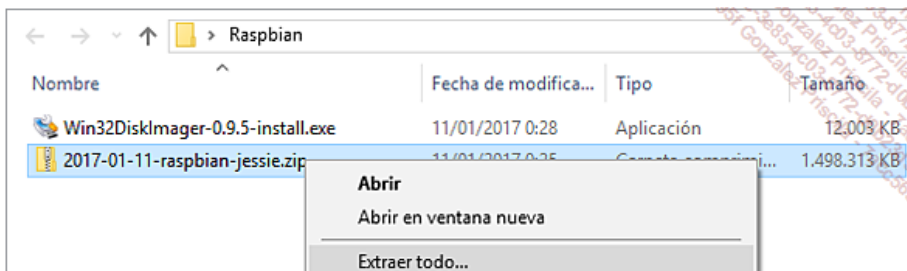
Compruebe a continuación que el hash SHA-1 se corresponde con el que figura en el sitio Web, de la misma manera que anteriormente con NOOBS.

Descompresión del OS

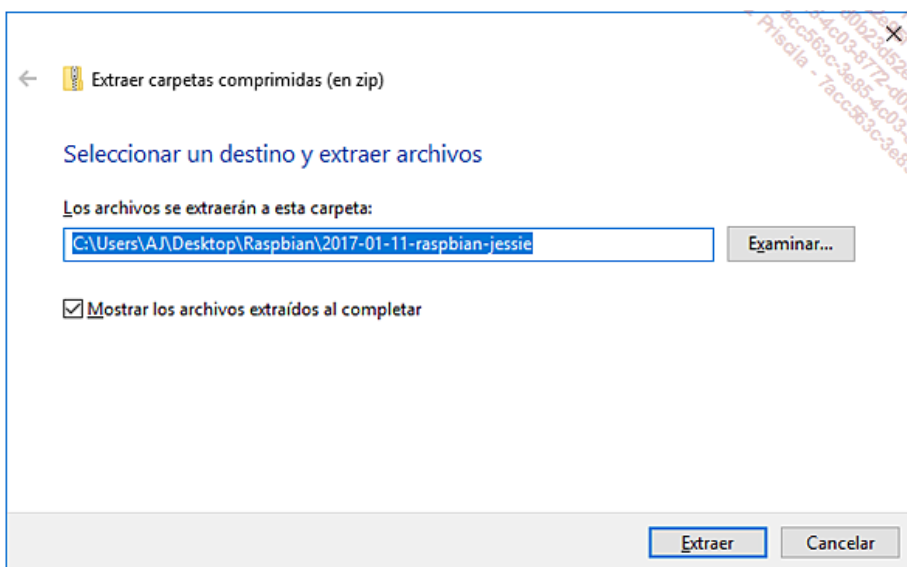
Los archivos descargados están en formato .zip. Este formato de compresión permite reducir el tamaño de los archivos sin pérdida de datos, para los archivos, por ejemplo.

Por tanto, en primer lugar hay que descomprimir los archivos descargados para poder transferirlos a la tarjeta micro SD de la Raspberry Pi. Esta operación es idéntica tanto en NOOBS como en un OS. A continuación se muestra cómo realizar esta operación para Raspbian.

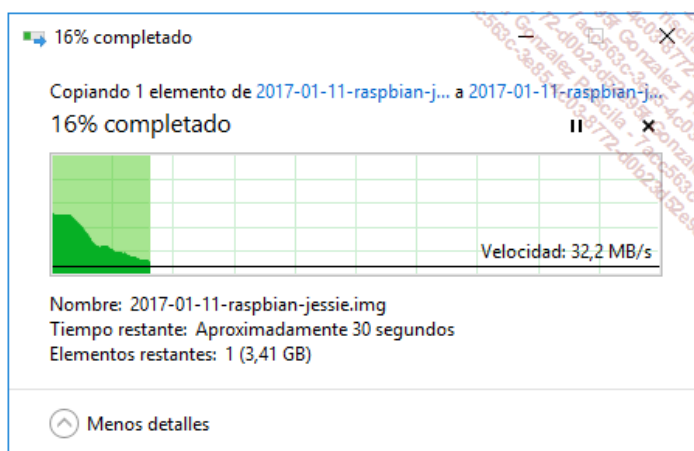
En Windows 8, haga clic con el botón derecho del ratón en el archivo. En el menú contextual, pulse en **Extraer todo...**



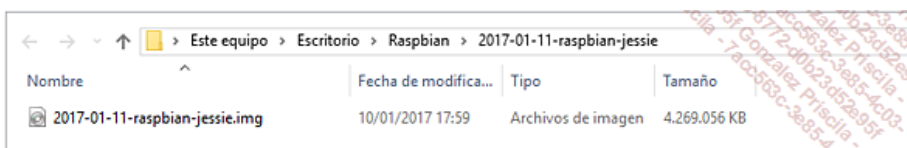
Debe seleccionar la carpeta en la que Windows almacenará el contenido extraído del archivo. Por defecto, la carpeta tiene el mismo nombre que el archivo. Generalmente esto es conveniente. Modifique este destino si es necesario o acéptelo pulsando en **Extraer**.



Windows procede ahora a la extracción del archivo contenido en el archivo .zip. La duración de la operación depende del tamaño del archivo.



Como resultado de la descompresión del archivo, se abre la carpeta creada por Windows para almacenar el archivo descomprimido. Contiene la imagen de Raspbian, lista para ser transferida a la tarjeta micro SD.



Dispone ahora de una imagen RAW de Raspbian, que es necesario copiar en la tarjeta micro SD de la Raspberry Pi. Hay que observar que el archivo de origen en .zip ocupa 955 MB, mientras que la imagen descomprimida representa 3,2 GB de espacio en disco.

a. Transferencia de Raspbian a la tarjeta micro SD

NOOBS es una buena solución para los novatos que descubren la Raspberry Pi y Linux. La instalación directa del sistema en la tarjeta micro SD se reserva para los usuarios más experimentados.

Tipo de tarjeta micro SD a usar

La Fundación recomienda una tarjeta micro SD de clase 4. Es preferible una tarjeta de clase 10, ya que mejorará el rendimiento. La imagen de Raspbian ocupa cerca de 4 GB, por lo que es necesario prever una tarjeta de 8 GB como mínimo para la instalación de las imágenes RAW. Si no quiere que su tarjeta se llene rápidamente, opte por una tarjeta de 16 GB. Como sucede con NOOBS, opte por tarjetas de marca y consulte la lista en el sitio elinux.org (elinux.org/RPI_SD_cards).

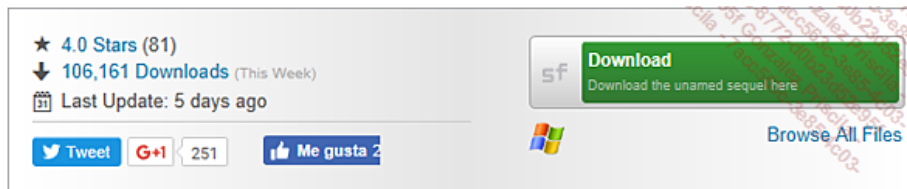
b. En Windows 8

La Fundación indica que la instalación en Windows no se puede hacer simplemente copiando o pulsando y arrastrando en la imagen RAW del sistema operativo. La tarjeta micro SD obtenida de esta manera **no funciona en la Raspberry Pi**.

El único método válido en Windows es la instalación con la herramienta *Win32DiskImager*. Este programa está especialmente diseñado para escribir la imagen RAW de un disco en un soporte extraíble, como la tarjeta micro SD. También puede realizar una copia en modo RAW de un soporte extraíble a un archivo. Un enlace en la página de descarga del sitio raspberrypi.org redirige al sitio de *Win32DiskImager* (<http://sourceforge.net/projects/win32diskimager/>).

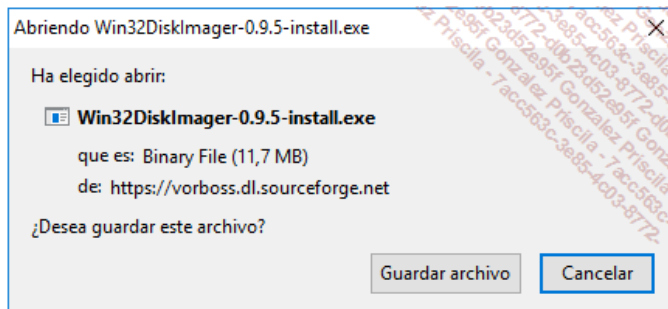
Descarga de Win32DiskImager

La página SourceForge de *Win32DiskImager* ofrece numerosos botones **Download**. El único que inicia la descarga de la herramienta y no muestra publicidad es el que está situado sobre la imagen que se muestra a continuación: **Download the unnamed sequel here**.



Pulse en el botón y espere... Se abre una nueva ventana, no pulse en ningún botón para no ver su pantalla invadida de publicidad.

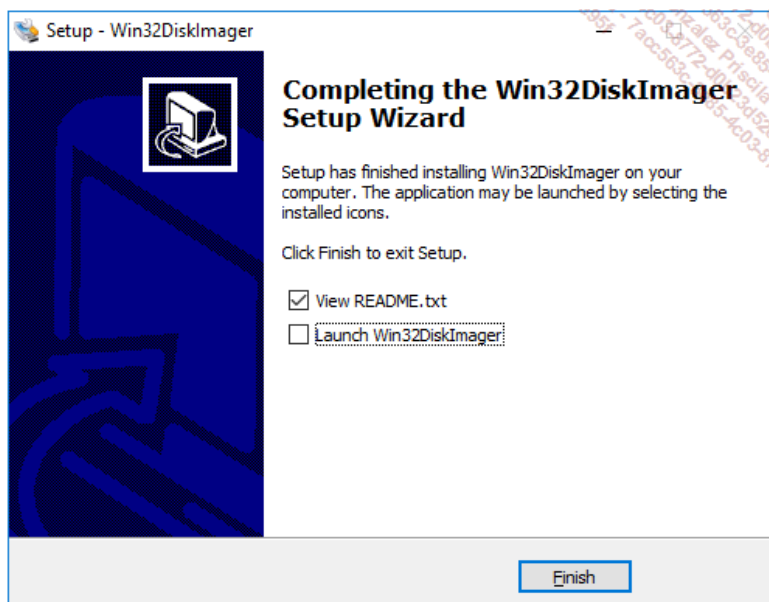
La ventana que se abre al cabo de un momento debe ser parecida a la que se muestra a continuación. Permite ejecutar o guardar *Win32DiskImager*.



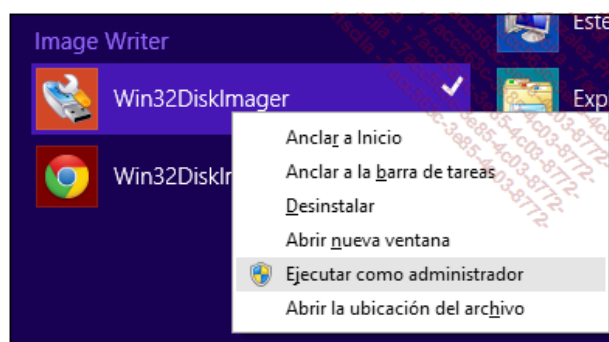
Pulse en **Guardar archivo**.

En la siguiente ventana, indique la carpeta en la que desea guardar *win32diskimager-0.95-install.exe* y después pulse el botón **Guardar**. La descarga de los 12 MB debería ser rápida.

En la carpeta donde se guarda *Win32DiskImager*, haga doble clic en el archivo *Win32DiskImager-0.9.5-install.exe*. Esto inicia la instalación de *Win32DiskImager* en su máquina. Se abre una primera ventana de bienvenida (Welcome...). Pulse el botón **Next** para continuar. En la siguiente ventana, acepte la licencia de uso (GPLv2) marcando **I accept the agreement** y después pulse el botón **Next**. Acepte la instalación de *Win32DiskImager* en la carpeta que sugiere el software de instalación (a menos que tenga una razón para no instalar ahí...) pulsando en **Next** y después acepte las opciones siguientes pulsado de nuevo en **Next** hasta que llegue al inicio de la instalación. Pulse el botón **Install**. En la ventana que aparece al final de la instalación, desmarque la opción **Launch Win32DiskImager**. Hay que dar permisos de ejecución como administrador a esta aplicación para que pueda acceder a la tarjeta micro SD en modo escritura.



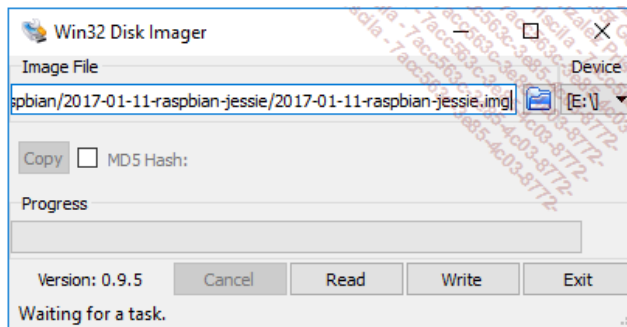
En la pantalla de bienvenida de Windows 8, haga clic con el botón derecho del ratón en el icono de *Win32DiskImager*. En el menú que aparece, pulse en **Ejecutar como administrador**.



➤ En los sistemas operativos posteriores a Windows XP, los programas se ejecutan con los permisos de un usuario normal para limitar las infecciones por virus. *Win32DiskImager* accede directamente al hardware y su ejecución necesita permisos de administrador para que pueda realizar su tarea. Si usa frecuentemente este programa, modifique sus propiedades para que siempre se ejecute con permisos de administrador.

Escritura de la imagen del OS en la tarjeta micro SD

Se abre la ventana de la herramienta:



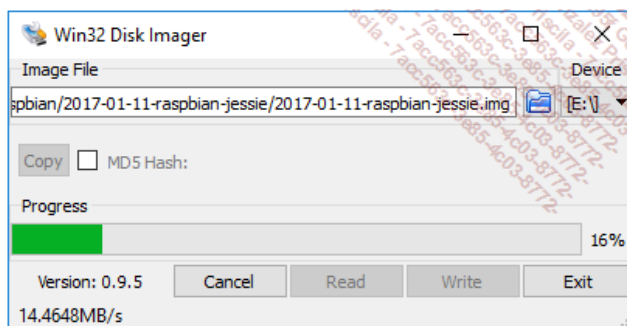
Compruebe cuidadosamente que la tarjeta micro SD detectada por el programa es la que desea preparar. La escritura "física" realizada por *Win32DiskImager* puede hacer que sea inutilizable una memoria de almacenamiento externo, como la tarjeta micro SD. Aquí *Win32DiskImager* indica que va a escribir la imagen en la tarjeta **E:**. Es la letra de lector asociado a la tarjeta micro SD. En la zona **Imagen File**, seleccione la imagen a transferir a la tarjeta micro SD. El icono que representa una carpeta a la derecha de esta zona de texto le permite hacer una búsqueda en su máquina y después validar su selección.

Pulse el botón **Write** para iniciar la operación.

Se abre una ventana de advertencia y tiene que confirmar que desea transferir la imagen a la tarjeta seleccionada, aquí el lector **E:**.

Pulse **Yes** para confirmar su selección.

La transferencia de la imagen en la tarjeta micro SD arranca, la barra **Progress** representa el avance de la copia. La velocidad de escritura se muestra en la parte inferior izquierda de la ventana y da una idea del rendimiento de la tarjeta.



Al final de la transferencia, se muestra un mensaje **Write successful** (Escritura con éxito), que anuncia que la operación se ha desarrollado correctamente.

Pulse el botón **OK** de la ventana de información, y después pulse el botón **Exit** para salir del programa.

La tarjeta micro SD está lista para el inicio de la Raspberry Pi. Puede quitar del lector la tarjeta micro SD.

🔍 Visualice las características de la tarjeta micro SD en la ventana **Administrador de archivos** de Windows 8. Debe ver la tarjeta bajo el nombre de "boot", con un tamaño de solamente 62,9 MB. Esta es solo la primera partición FAT utilizada para el inicio del sistema. La partición Linux que contiene el sistema de archivos raíz es invisible para Windows. Si no ve la tarjeta, es que la operación de escritura no ha tenido éxito.

c. En Debian 8

Para un usuario de Linux, el procedimiento de creación de la tarjeta micro SD es parecido al que se utiliza en Windows. Hay que observar que este procedimiento se puede realizar... en una Raspberry Pi.

Los usuarios de otras distribuciones de Linux adoptarán el procedimiento de su versión.

El uso del modo gráfico o del modo texto permite realizar la preparación de una tarjeta micro SD lista para su uso en la Raspberry Pi.

El procedimiento es el mismo en ambos casos:

- Descarga del archivo.
- Comprobación de su integridad con SHA-1.
- Extracción de la imagen del OS.
- Transferencia a la tarjeta micro SD.

Descarga y extracción de la imagen de Raspbian en modo gráfico

Descarga del archivo almacenado en modo gráfico

Abra un navegador web. En la página www.raspberrypi.org/downloads pulse el enlace **Download ZIP** de Raspbian.

Se abre la página de descarga de Raspbian, seguida de la ventana de descarga unos segundos después. El archivo .zip (alrededor de 1.3 GB) se guarda en su disco duro, dentro de la carpeta *Downloads* (o **Descargas**) si deja la opción por defecto. En caso contrario, seleccione dónde desea guardar el archivo.

Comprobación de la integridad del archivo

Abra un terminal: **Aplicaciones - Accesorios - Terminal administrador**.

Inicie la generación del hash SHA-1 del archivo.

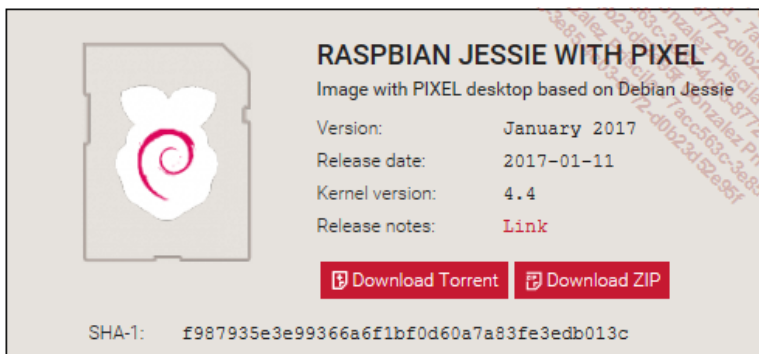
Sintaxis

```
sha1sum [nombre_de_archivo]
```

La operación dura algunos segundos en un PC - mucho más en una Raspberry Pi - y el resultado se muestra en forma de 40 caracteres hexadecimales.

```
root@debian-8:~/raspbian# sha1sum
2017-01-11-raspbian-jessie.zip
f987935e3e99366a6f1bf0d60a7a83fe3edb013c
2017-01-11-raspbian-jessie.zip
```

Compare ahora el hash obtenido con el hash calculado en el archivo de origen, visible en la siguiente captura de pantalla:



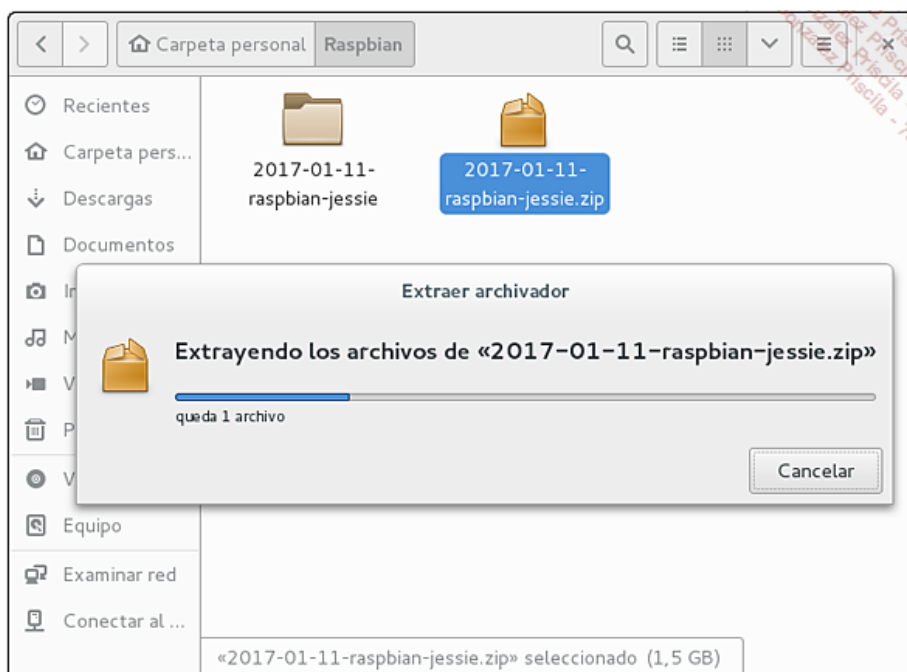
Si ambos valores coinciden, como ocurre aquí, puede pasar a la siguiente etapa. Si ambos valores son diferentes, es necesario descargar de nuevo el archivo almacenado y volver a empezar con la comprobación.

Extracción de la imagen

Vaya a la carpeta donde ha guardado el archivo.

Descomprima el archivo .zip haciendo clic derecho sobre él y seleccionando **Extraer aquí...**, el administrador de archivos va a extraer el archivo y guardar el resultado en la misma carpeta.

La ventana Xarchiver se abre y la operación de extracción empieza.



Al final de la operación, obtendrá una imagen que ocupa 3,7 GB.

Ahora tiene un archivo imagen *2017-01-11-raspbian-jessie.img* (o la versión actual durante su instalación). Solo falta transferirla a la tarjeta micro SD. Diríjase a la sección Transferencia de Raspbian a la tarjeta micro SD, un poco más adelante.

Descarga y extracción de la imagen de Raspbian en modo consola

Descarga del archivo

En un terminal, inicie la descarga del archivo que contiene la imagen de Raspbian:

```
wget http://downloads.raspberrypi.org/raspbian\_latest
```

```
root@debian-8:~/raspbian# wget
http://downloads.raspberrypi.org/raspbian_latest
--2016-06-17 18:16:20--
http://downloads.raspberrypi.org/raspbian_latest
Resolución de downloads.raspberrypi.org (downloads.raspberrypi.org)...
3.93.128.211, 93.93.135.188, 93.93.128.133, ...

Connexión a downloads.raspberrypi.org
(downloads.raspberrypi.org) [93.93.128.211]:80... conectado.
petición HTTP enviada, esperando respuesta... 302 Found
Ubicación: http://downloads.raspberrypi.org/raspbian/images/
raspbian-2016-05-31/2017-01-11-raspbian-jessie.zip [siguiente]

.../...

Tamaño: 1393896178 (1,3G) [application/zip]
Guardado en: « raspbian_latest »
raspbian_latest 100%[=====] 1,30G 446KB/s
ds 57m 15ss
2016-06-17 19:13:35 (396 KB/s) - «raspbian_latest»
guardado [1393896178/1393896178]
```

El archivo que contiene la imagen de Raspbian se ha guardado con el nombre *raspbian_latest* en el directorio actual. Es mejor renombrarlo para identificarlo correctamente.

```
root@debian-7:~# mv raspbian_latest 2017-01-11-raspbian-jessie.zip
```

Comprobación de la integridad del archivo

La comprobación de la integridad del archivo descargado se hace de la misma manera que con Debian 7 en modo gráfico.

En un terminal, inicie la generación del hash SHA-1 del archivo.

Sintaxis

```
sha1sum [nombre_de_archivo]
```

Compare el hash obtenido con el hash visible en la página de descarga, calculado a partir del archivo de origen.

Si ambos valores coinciden, puede pasar a la siguiente etapa. Si ambos valores son diferentes, es necesario descargar de nuevo el archivo de imagen y volver a realizar la comprobación.

Extracción de la imagen

Vaya a la carpeta donde ha guardado el archivo.

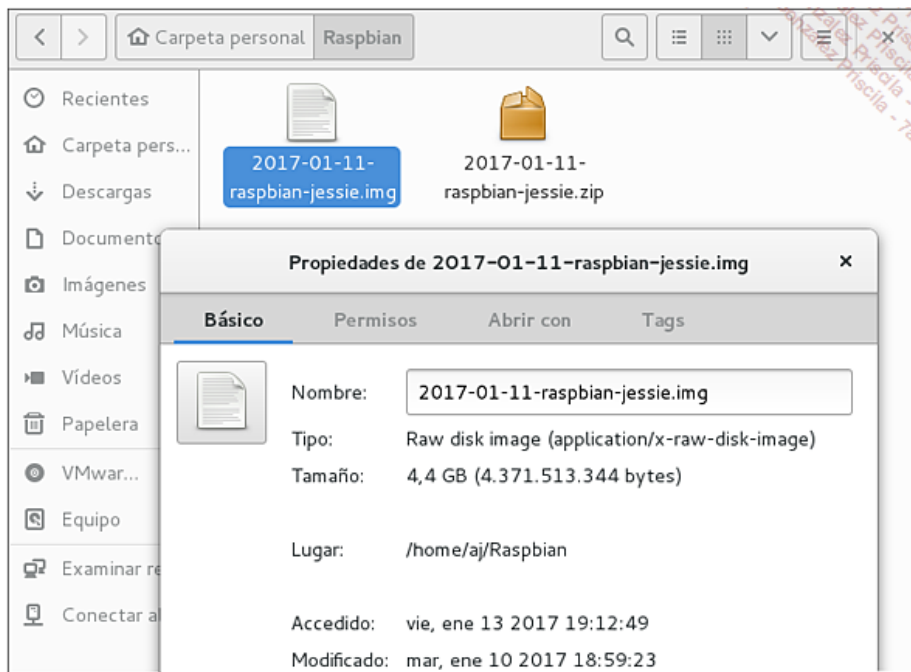
Descomprima el archivo .zip con el comando `unzip`. Si lo necesita, instale el paquete **unzip** en su sistema (`apt-get install unzip`).

Sintaxis

```
unzip [nombre_de_archivo]
```

```
root@debian-8:~/raspbian# unzip 2017-01-11-raspbian-jessie.zip
Archive:  2017-01-11-raspbian-jessie.zip
inflating: 2017-01-11-raspbian-jessie.img
```

Ahora dispone de un archivo imagen `2017-01-11-raspbian-jessie.img`, o la versión actual durante su instalación, como muestra la siguiente captura de pantalla.



Transferencia de Raspbian a la tarjeta micro SD

En primer lugar, es necesario ubicar de manera segura el nombre del periférico asociado por Debian a la tarjeta micro SD. El comando `dmesg` (*Display message*) permite obtener esta información. Es un comando que muestra los mensajes enviados por el núcleo Linux, en particular cuando los periféricos devuelven errores o se conectan en caliente, como la tarjeta micro SD.

Inserte la tarjeta micro SD en el lector. Pasados algunos segundos, ejecute el comando `dmesg`. La información relativa a la tarjeta micro SD se encuentra en las últimas líneas:

```
[ 283.493724] sd 3:0:0:0: Attached scsi generic sg2 type 0
[ 283.494030] sd 3:0:0:1: Attached scsi generic sg3 type 0
[ 283.494338] sd 3:0:0:2: Attached scsi generic sg4 type 0
[ 283.786323] sd 3:0:0:1: [sdc] 30703616 512-byte logical blocks:
(15.7 GB/14.6 GiB)
[ 283.804811] sd 3:0:0: 1: [sdc] Write Protect is off
[ 283.804815] sd 3:0:0: 1: [sdc] Mode Sense: 21 00 00 00
[ 283.823278] sd 3:0:0: 1: [sdc] Write cache: disabled, read cache:
enabled, doesn't support DPO or FUA
[ 283.939196] sdc: sdc1
[ 284.024086] sd 3:0:0:0: [sdb] Attached SCSI removable disk
[ 284.041561] sd 3:0:0:1: [sdc] Attached SCSI removable disk
[ 284.048069] sd 3:0:0:2: [sdd] Attached SCSI removable disk
[ 284.583679] FAT-fs (sdc1): utf8 is not a recommended IO charset
for FAT files
```

La tarjeta micro SD se ha insertado en un lector de tarjetas micro SD, con varios conectores. De ahí la aparición de los discos `sdb`, `sdc` y `sdd`. Vemos que solo se identifica `sdc` como disco adicional:

```
[sdc] 30703616 512-byte logical blocks: (15.7 GB/14.6 GiB)
FAT-fs (sdc1): utf8 is not a recommended IO charset for FAT files
```

El resto de ubicaciones de tarjeta están vacías. La primera línea en la captura anterior confirma el tamaño de la tarjeta micro SD introducida en el lector: 16 GB. Por tanto, la tarjeta micro SD ha sido reconocida por el sistema Debian 8 como `sdc`.

¿Por qué tantas precauciones?

La operación de transferencia de una imagen a una tarjeta micro SD es una escritura física de datos. Aplicado a otro soporte diferente de la tarjeta micro SD puede hacer totalmente inaccesibles los datos del soporte. Por este motivo es necesario identificar con absoluta certeza el destino de la transferencia.

Ahora que la imagen está descomprimida y la tarjeta micro SD identificada, es necesario transferir la imagen del OS a la tarjeta micro SD.

Para esto, usamos el comando `dd`.

Sintaxis

```
dd [if=archivo] [of=archivo] [bs=tamaño de los bloques]
```

if= indica el archivo desde el que se leen los datos (*Input File* = archivo de entrada).

of= indica el archivo en el que se escriben los datos. Linux ve a los periféricos y dispositivos de almacenamiento como archivos (*Output File* = archivo de salida).

bs= indica el tamaño de los bloques de datos que el comando `dd` debe leer y escribir de una vez (*Block Size* = tamaño de los bloques).

Escriba en la línea de comandos `dd if=2013-09-25-wheezy-raspbian.img of=/dev/sdb bs=1M` y valide.

Espere... Por pantalla no aparece ninguna indicación del correcto desarrollo de la operación. Después de algunos minutos de espera, se muestra el resultado. Si no obtiene ningún mensaje de error es que todo ha ido bien.

```
root@debian-8:~/Raspbian# ls
] 2017-01-11-raspbian-jessie.img 2017-01-11-raspbian-jessie.zip

root@debian-8:~/raspbian# dd if=2017-01-11-raspbian-jessie.img
of=/dev/sdc bs=1M
3833+0 líneas leídas
3833+0 líneas escritas
4019191808 bytes (4,0 GB) copiados, 278,565 s, 14,4 MB/s
```

En un Pentium de doble núcleo con 1 GB de RAM, la escritura de la tarjeta micro SD Samsung EVO dura 4mn30s. Estas tarjetas son Class 10 (10 MB/s como mínimo, en modo escritura).

Ahora tiene una tarjeta micro SD lista que puede utilizar para iniciar una Raspberry Pi en Raspbian.

d. Conclusión

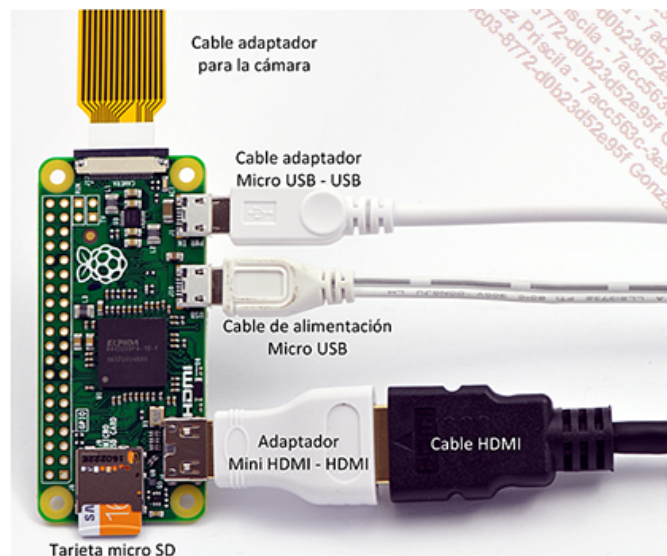
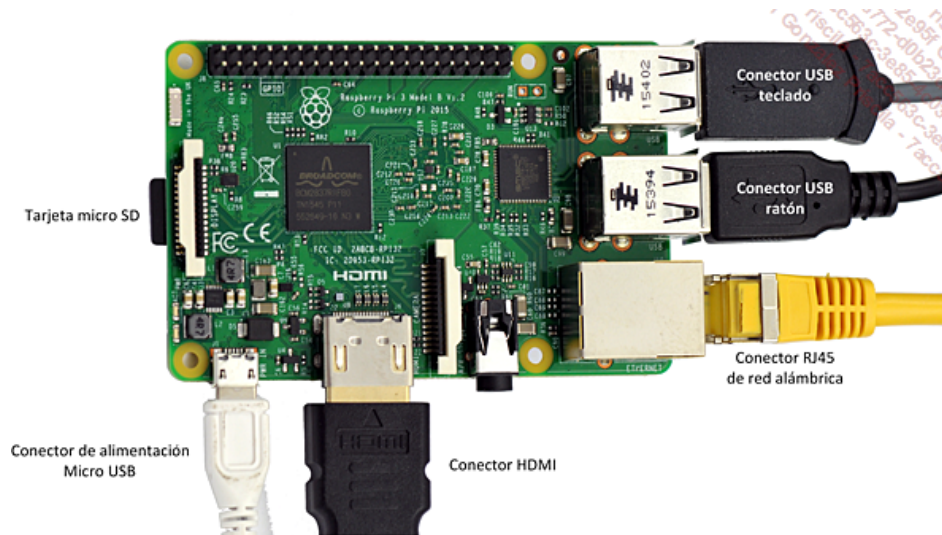
La elección de una distribución varía según los usos previstos para la máquina.

Una Raspberry Pi headless, es decir, configurada para funcionar sin monitor, teclado ni ratón, se equipará con un sistema operativo único.

Por el contrario, NOOBS permite arrancar alternativamente en varios sistemas operativos diferentes. Recibirá la preferencia de los usuarios de Raspberry Pi, que utilizan su nano-ordenador para aprender y experimentar. NOOBS solo sirve para iniciar la Raspberry Pi y para la recuperación rápida del sistema. La capacidad de la tarjeta micro SD no se ve afectada por la presencia de este programa de tamaño reducido.

Preparación de la Raspberry Pi

La tarjeta micro SD está lista y NOOBS o Raspbian instalado, es momento de enchufar los cables en la Raspberry Pi y descubrir todas sus posibilidades.

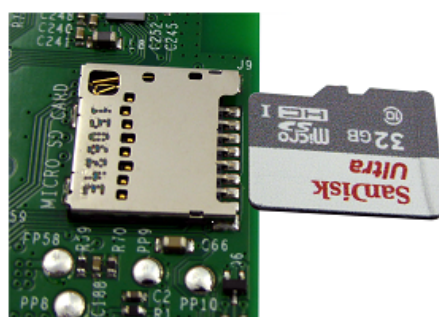


Las imágenes anteriores presentan la Raspberry Pi 3 y la Raspberry Pi Zero listas para su arranque; todos los cables así como la tarjeta micro SD están instalados.

En la Raspberry Pi Zero, las tomas son modelos en miniatura (ahorro de espacio), por lo que es útil prever un juego de adaptadores.

1. Establecimiento de la tarjeta micro SD

En las Raspberry Pi Zero y 3 la tarjeta simplemente se mantiene por fricción, no hay sistema de bloqueo.



Presente la tarjeta micro SD delante del conector situado bajo la tarjeta de la Raspberry Pi 3. El conector está encima de la tarjeta en el caso de la Raspberry Pi Zero. En ambos casos, los contactos de la tarjeta micro SD se deben girar hacia el circuito impreso de la Raspberry Pi.

Inserte la tarjeta micro SD en sus conectores. En los modelos anteriores, había que empujar suavemente la tarjeta en el soporte hasta oír y sentir un "clic" que indicaba el bloqueo de la tarjeta. En la Raspberry Pi Zero y la Raspberry Pi 3, empuje simplemente la tarjeta, que no se bloquea. La tarjeta micro SD se mantiene por fricción. Si la tarjeta se introduce al revés o de forma oblicua, podrá sentir una resistencia importante. No la fuerce para no dañar la tarjeta o el conector.

2. Conexión de los cables

a. Raspberry Pi 3 modelo B

No enchufe el bloque de alimentación inmediatamente.

Conecte la toma micro USB a la toma de alimentación de la Raspberry Pi.

Conecte el teclado y el ratón USB a los puertos USB.

Conecte la toma Ethernet de la Raspberry Pi y de red, según sea necesario (salida en paralelo, switch, box...).

Conecte el cable HDMI a la Raspberry Pi y a la pantalla si esta tiene una entrada HDMI. Por defecto, use un adaptador HDMI/DVI o HDMI/VGA y conecte la salida del adaptador a la toma correspondiente.

Si conecta la Raspberry Pi a un televisor sin toma HDMI:

En una Raspberry Pi presione la toma de cuatro conectores hacia el conector correspondiente de la Raspberry Pi y después conecte las salidas de vídeo compuesto y audio derecho e izquierdo (tomas RCA) del televisor. En un televisor equipado solo con una o varias tomas de Euroconector, prevea un adaptador Euroconector/RCA. Este adaptador puede tener forma de una toma Euroconector, con los conectores RCA integrados o de un cable con un Euroconector en un extremo y conectores RCA en el otro.

b. Raspberry Pi Zero

En la Raspberry Pi Zero se usan tomas en miniatura para el puerto USB así como para el puerto HDMI. Prevea adaptadores si quiere conectar su Raspberry Pi Zero a una pantalla HDMI y a periféricos USB (consulte la sección La Raspberry Pi Zero - Miniaturización de las tomas y consecuencias, en el capítulo Descripción técnica).

No conecte el bloque de alimentación a la red eléctrica. Una la toma micro USB de alimentación a la toma micro USB de la Raspberry Pi.

Enchufe el adaptador micro USB/USB estándar a la toma micro USB de la Raspberry Pi Zero. La toma hembra USB estándar solo permite una conexión entre las siguientes:

- Un teclado
- Un ratón
- Un dongle para teclado/ratón inalámbrico
- Un hub USB
- Una llave Wi-Fi
- ...

Si quiere conectar varios periféricos USB, prevea un hub USB.

Conecte el adaptador mini HDMI/HDMI estándar a la salida HDMI de la Raspberry Pi Zero. Una la pantalla con un cable HDMI a esta toma estándar.

La Raspberry Pi Zero no tiene salida de audio. La salida de vídeo compuesta es accesible desde dos tomas marcadas como TV en la tarjeta. Si desea usar esta salida de vídeo analógico, tendrá que soldar un cable.

3. Enchufar a la corriente eléctrica

Ha llegado el momento que espera todo aquel que compra una tarjeta informática: enchufarla. Conecte el bloque de alimentación a la red eléctrica.

La pantalla conectada a la Raspberry Pi debe mostrar franjas de colores y después el logo de la Raspberry Pi en la parte superior izquierda de la pantalla, aparece texto, etc...

Con Raspbian Jessie, la Fundación ha optado por una gestión gráfica de la configuración.

Encontrará la configuración gráfica con Jessie en el capítulo Utilizar el modo gráfico - Los menús de Raspbian Jessie. Las opciones que se presentan en *raspi-config* son accesibles por medio de botones o casillas de opción en la Raspbian Jessie.

Configurar el sistema con raspi-config

Con Raspbian Jessie Lite el inicio se hace en modo texto. El modo gráfico se ignora porque no existe en esta distribución, quizás destinada a proyectos embebidos.

La herramienta de configuración *raspi-config* está accesible por línea de comandos para una configuración en modo semi-gráfico.

También es posible realizar la configuración avanzada de Raspbian modificando directamente el archivo `/boot/config.txt` (ver más adelante). Permite modificar los argumentos inaccesibles en la interfaz gráfica. Sin embargo, si se añaden modificaciones demasiado importantes de manera manual a la configuración de Raspbian es posible que *raspi-config* no sea capaz de gestionarlas.

1. Arrancar raspi-config

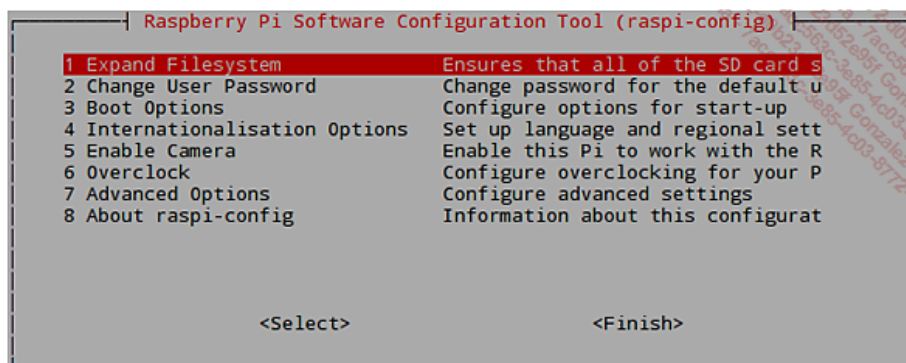
Antes de pasar a la configuración de Raspbian, en una Raspberry Pi conectada a Internet, empiece actualizando el sistema:

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

El inicio de *raspi-config* en modo texto necesita el uso de `sudo` porque el archivo `config.txt` pertenece a *root* y el usuario "normal" *pi* no podría modificarlo.

```
sudo raspi-config
```

2. Usar raspi-config



raspi-config, en la versión actual, ofrece nuevas opciones, algunas de ellas se dividen de nuevo en sub-opciones, como las opciones de internacionalización o las opciones avanzadas. Adapte este capítulo en función de la versión de *raspi-config* disponible en el momento de su instalación.

a. Mover el cursor

Use las teclas de dirección *arriba* y *abajo* para moverse por las opciones. La tecla T fija el foco en los dos botones situados en la parte inferior de la pantalla **<Select>** y **<Finish>** y después de nuevo en la lista de opciones si vuelve a pulsar en la tecla [Tab].

b. Selección

Para validar una selección, fije el foco en **<Select>** y valide con la tecla [Intro]. Para indicar que ha terminado de trabajar con esta herramienta, fije el foco en **<Finish>** y valide.

c. Selección de las opciones

Algunas opciones necesitan marcar una o varias opciones de una lista. En modo texto, las casillas de selección se representan por dos corchetes separados por un espacio: []. Para marcar la casilla, sitúe el cursor entre los corchetes y pulse la tecla [Espacio]. Aparece un asterisco entre los corchetes: [*]. Indica que la casilla está seleccionada. Para desmarcar una casilla, haga lo mismo: el asterisco desaparecerá. Para recorrer las listas use las flechas de dirección arriba y abajo, así como las teclas [Re Pág] y [Av Pág] (retroceder o avanzar una página), lo que aumenta la velocidad de desplazamiento de la lista.

⚠ Algunas opciones que necesitan una conexión Internet no funcionan en una Raspberry Zero desprovista de toma Ethernet como la que muestra el ejemplo.

Las opciones se presentan en el orden en que aparecen en la pantalla de *raspi-config*. Se recomienda empezar actualizando la herramienta *raspi-config* con las opciones **9 Advanced Options** y **A0 Update**.

Atención: si tiene dificultad con el inglés y/o con el uso de un teclado configurado en QWERTY, configure en primer lugar las opciones de internacionalización y reinicie la Raspberry Pi antes de volver al resto de ajustes del sistema.

d. 1 Expand Filesystem

Esta opción se ejecutaba manualmente en las antiguas versiones de Raspbian. Con Jessie, se ejecuta automáticamente durante el primer inicio del sistema. Se sigue de un reinicio automático del sistema operativo, consecuencia del cual toda la tarjeta micro SD queda ocupada por el sistema de archivos. Esto se debe al hecho de que el sistema operativo se entrega como una imagen binaria adaptada a las tarjetas micro SD de 4 GB. Actualmente, con las aplicaciones instaladas de origen, el tamaño del sistema en la tarjeta micro SD es muy próxima a este valor, lo que puede representar un problema en caso de actualización o de instalación de nuevas aplicaciones.

Usted la utiliza si ha transferido la imagen del sistema a otra tarjeta micro SD de mayor capacidad, en este caso, una parte de la tarjeta quedará inutilizada. Para usar la parte restante de la tarjeta micro SD, seleccione esta opción y valide. El *Root File System* (sistema de archivos raíz) se extiende a todo el espacio disponible restante en la tarjeta micro SD. La modificación será efectiva después de un reinicio de la Raspberry Pi.

e. 2 Change User Password

Esta opción permite modificar la contraseña por defecto del usuario pi. Si su máquina está conectada a Internet, es obligatorio cambiar la contraseña original (*raspberrypi*) para cambiarla por una contraseña más segura.

Valide esta opción. Una ventana le advierte de que debe escribir de nuevo la contraseña. Fije el foco en **<Ok>** con la tecla [Tab] y valide.

Aparece un mensaje en la parte inferior de la pantalla:

Escriba la nueva contraseña UNIX: rellene la nueva contraseña una primera vez (como siempre en Linux, durante la escritura de una contraseña no aparece nada en la pantalla, por seguridad) y valide.

Aparece un nuevo mensaje.

Repita de nuevo la contraseña UNIX: rellene de nuevo la contraseña y valide.

Si las dos contraseñas son idénticas, una ventana le informa del éxito de la operación. Valide con **<Ok>** para volver a la pantalla de bienvenida de *raspi-config*.

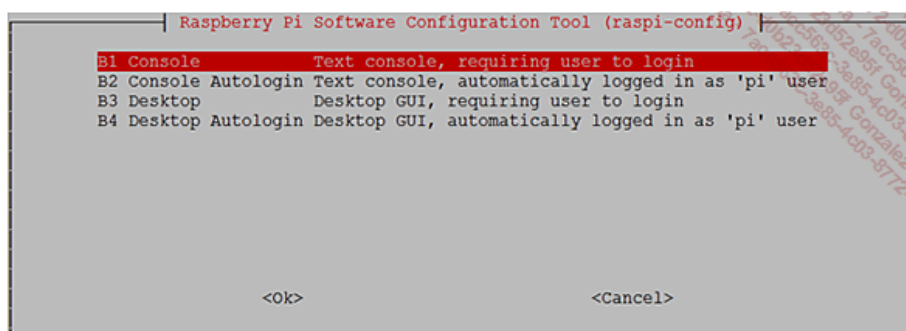
Si el sistema detecta una diferencia entre las dos contraseñas, una ventana le advierte de que se ha producido un error. Valide con **<Ok>** para volver a la pantalla de bienvenida de *raspi-config* y vuelva a comenzar la operación de modificación de la contraseña.

Para que su contraseña sea "sólida":

- Evite ante todo que esta tenga alguna relación con usted: nombre, nombre de sus mascotas, fecha de nacimiento, teléfono, código postal, etc.
- Cuanto más larga sea la contraseña, más difícil será de descubrir. Seleccione contraseñas de 8 a 15 caracteres.
- Mezcle mayúsculas, minúsculas, cifras y caracteres especiales (&, #, €...).
- Memorice la contraseña, no la escriba en un trozo de papel.
- Use frases fácilmente memorizables: "A quién madruga Dios le ayuda", que puede dar como contraseña **1quma\$ileay**, tomando las dos primeras letras de cada palabra, cambiando la A por un 1 y la D por \$...
- ¡No utilice el ejemplo anterior!

f. 3 Boot options

Esta opción permite al usuario seleccionar cómo va a arrancar la distribución.



En Jessie Lite, solo funcionan las opciones **B1** y **B2**. Si selecciona **B3** o **B4**, que necesitan el modo gráfico, *raspi-config* le pedirá instalar *lightdm*.

B1 provocará un arranque en modo consola, solicitando un login/contraseña al usuario.

B2 provocará un arranque en modo consola sin pedir login/contraseña al usuario. El usuario pi se conectará automáticamente sin pedir contraseña.

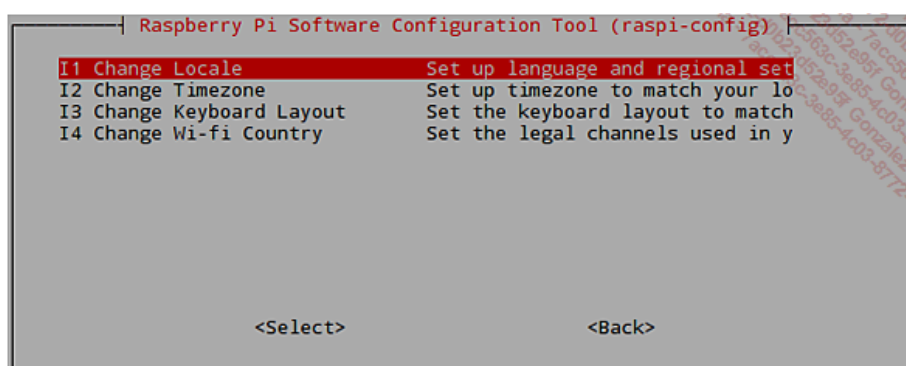
g. 4 Wait for network at boot

Por defecto, el proceso de inicio de Jessie está diseñado para funcionar tan rápido como sea posible. Eventualmente esto puede significar que la secuencia de inicio termina antes de que una conexión de red esté operativa. Esto puede provocar problemas en algunas aplicaciones que empiezan a funcionar tan pronto como se produce el arranque.

Seleccionando esta opción, el arranque no terminará hasta que se haya establecido una conexión de red. Como contrapartida, la secuencia de inicio puede ser más larga. Esta opción está desactivada por defecto. Actívela solo si tiene problemas con las aplicaciones que necesitan una conexión de red.

h. 4 Internationalisation Options

Esta opción abre una ventana con cuatro opciones:



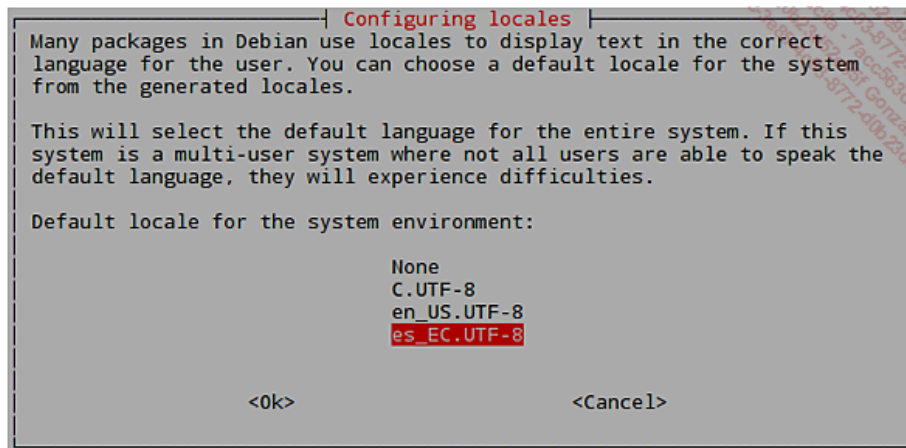
I1 Change Locale

Linux utiliza los argumentos regionales, llamados *Locale* en inglés, para administrar el idioma, representar los caracteres (encoding), las cifras y sus formatos, la fecha y la hora, la moneda, etc.

Seleccione la opción **Change Locale**, sitúe el foco en **<Select>** y valide. En la ventana **Locales to be generated:** descienda (flecha hacia abajo) hasta encontrar **es_ES.UTF-8**. Marque la casilla correspondiente, pulsando la tecla B, para que aparezca un asterisco en la casilla. Observará que la casilla correspondiente a **en_US.UTF-8** está marcada. Déjela marcada si piensa volver al inglés. En caso contrario, borre el asterisco. Varios juegos de argumentos regionales pueden convivir en un sistema, pero solo uno se usará en cada momento.

Fije el foco en **<Ok>** con la tecla [Tab] y valide.

Ahora hay que configurar los argumentos regionales por defecto. Estos son los que Linux usará durante el siguiente inicio.



Fije el foco en **es_EC.UTF-8** y valide.

Un mensaje en la parte inferior de la pantalla indica que el sistema ha generado los locales y que esta operación puede durar un tiempo.

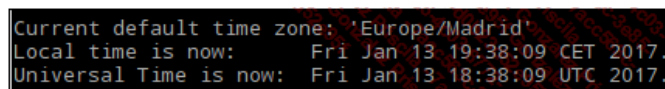
La configuración termina con un mensaje **Generación completa**, y vuelve a aparecer la pantalla de bienvenida de *raspi-config*.

Durante el próximo inicio, los argumentos regionales estarán en español. Y los textos en *raspi-config* también, lo que facilitará la configuración del sistema.

I2 Change Timezone

Para configurar la zona horaria, vuelva a las opciones de internacionalización. Seleccione la opción **Change Timezone**, sitúe el foco en **<Select>** y valide.

En la ventana que se abre, seleccione la zona geográfica en la que está, por ejemplo **Europe** y valide. En la lista de ciudades ofrecidas, seleccione la más cercana, **Madrid** para España, valide.



El sistema guarda su selección. Su Raspberry Pi muestra brevemente la zona geográfica y la ciudad seleccionadas y después la hora local, lo que permite comprobar el ajuste de las opciones seleccionadas.

I3 Change Keyboard Layout

Falta por configurar el teclado. Por defecto el teclado está en QWERTY.

La disposición de las teclas en los teclados varía de un país a otro en función de las letras que se usan con más frecuencia. Esta disposición, heredada de la época de las máquinas de escribir mecánicas, está adaptada a cada idioma. Permitía obtener altas velocidades de pulsación reduciendo los bloqueos mecánicos cuando se tocaban letras cercanas. La aparición de nuevas disposiciones de teclado (Bepo, Dvorak, Marsan) no ha puesto fin al dominio de AZERTY.

Los teclados de PC son todos idénticos. La tecla situada en un lugar del teclado envía siempre el mismo código. Así es como la tecla Q del teclado español se cambia por una A en el teclado francés. Pero solo existe el cambio a nivel del plástico de la tecla. El teclado envía el mismo código en ambos casos.

¿Cómo puede el sistema saber que el usuario ha enviado una A o una Q? Dispone de una tabla en la que el código enviado por el teclado corresponde a una letra. Si se cambia la disposición del teclado, es necesario informar al sistema para que use otra tabla.

Esta opción de configuración permite informar al sistema para que la tabla de correspondencia aplicada sea la que se adapte al tipo de teclado utilizado.

En las opciones de internacionalización, seleccione **Change Keyboard Layout**.

En la lista, intente ver si aparece el teclado que utiliza. Por defecto, seleccione **PC genérico 105-teclas (Intl)**, que se adapta a la mayoría de los teclados de PC. En la siguiente ventana, **Español - Español (variante, Latin-9 solo)**. Puede dejar las opciones por defecto en las siguientes ventanas y simplemente valide las opciones propuestas.

I4 Change Wi-fi Country

Seleccione el país en el que se usará el Wi-Fi. Esta opción permite respetar la ley y los reglamentos del país en lo que respecta a los canales utilizados y la potencia de emisión.

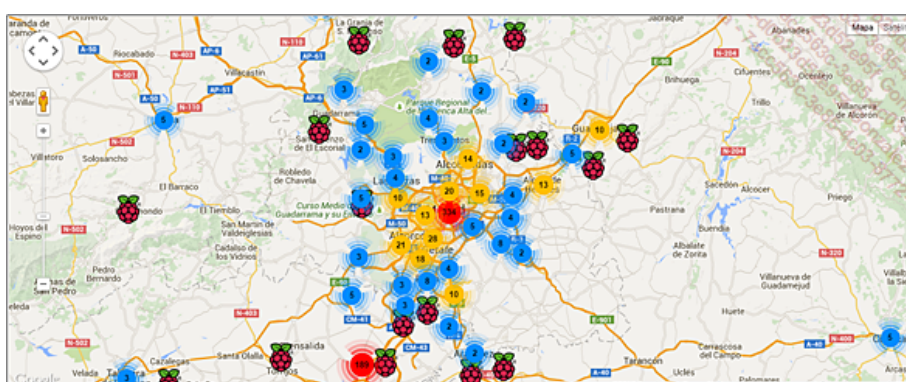
i. 6 Enable Camera

Esta opción permite activar la gestión de la cámara por el sistema. Fije el foco en **6 Enable camera** y valide.

En la siguiente ventana seleccione **<Sí>** y valide. El módulo cámara se tendrá en cuenta en el próximo inicio del sistema.

j. 7 Add to Rastrack

Rastrack es un mapa mundial basado en Google Maps, creado por Ryan Walmsley. Este mapa presenta el reparto mundial de las Raspberry Pi. Se invita a cada propietario de Raspberry Pi a inscribirse y de esta manera su Raspberry Pi se ubicará en el mapa.



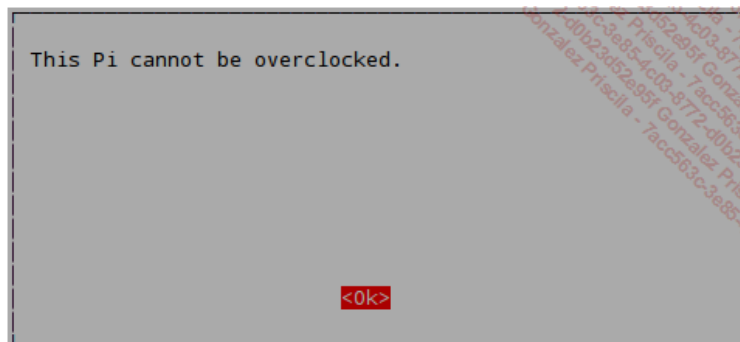
Esta distribución solo refleja la posición de los usuarios inscritos. En junio de 2016 solo había 112.000 Raspberry Pi registradas de los cerca de 8 millones de unidades vendidas. Se puede inscribir directamente en <http://rastrack.co.uk>.

La primera ventana solicita un nombre de usuario o un pseudónimo (**Username/Nickname For Rastrack Addition**), la siguiente ventana le permite indicar su dirección de mail (**Email Address For Rastrack Addition**). Después de validar, recibirá un mensaje de confirmación en su buzón de mail. Contiene un enlace que le permitirá terminar su inscripción en la tarjeta.

k. 8 Overclock

El overclock, también llamado *overclocking*, es la operación que consiste en aumentar la frecuencia de funcionamiento de un circuito electrónico (microprocesador, memoria...), y también algunas veces la tensión que se aplica al circuito (overvoltage).

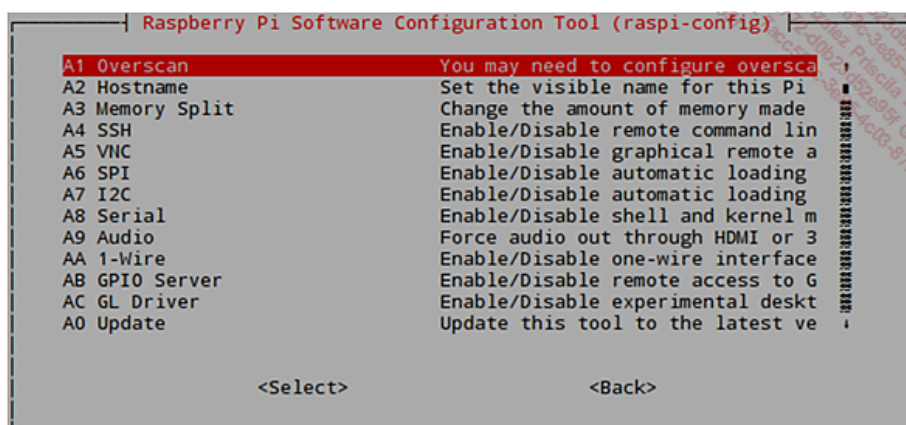
Fije el foco en **8 Overclock** y valide.



En la Raspberry Pi 3 y la Raspberry Pi Zero no es posible el overclock con *raspi-config* y se obtiene el mensaje **This Pi cannot be overclocked**.

l. 9 Advanced Options

Las opciones avanzadas abren una ventana en la que figuran trece ajustes identificados desde A1 hasta A0.



A1 Overscan

La opción **overscan** permite adaptar el tamaño del vídeo que se muestra en pantalla. Si la parte útil del vídeo está enmarcada por franjas negras en las que no se muestra nada o faltan fragmentos de vídeo porque están fuera de la pantalla, es necesario intervenir sobre el overscan.

En algunos monitores o televisores es posible desactivar el overscan usando el menú. Esto normalmente es suficiente para que el vídeo ocupe toda la pantalla. El overscan algunas veces se llama HD Size, pantalla completa o 1:1.

Si hay franjas negras rodeando la parte útil de la imagen, seleccione la opción **A1 Overscan** y valide. La siguiente ventana le invita a seleccionar si desea desactivar **<Disable>** o activar **<Enable>** el overscan. Seleccione **<Disable>**. Durante el próximo reinicio, la imagen ocupará toda la pantalla.

Si es necesario ajustar con más precisión el tamaño del vídeo porque es más pequeño que la pantalla o, por el contrario, se desborda por fuera, es necesario intervenir en el archivo `/boot/config.txt` y configurar manualmente el tamaño de los bordes. En este caso, deje el overscan activo **<Enable>** en la opción **A1 Overscan**. Abra el archivo `config.txt` con *nano* usando el siguiente comando:

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

🔸 El archivo `config.txt` tiene una importancia considerable durante el inicio de Raspbian. Piense en hacer una copia de seguridad antes de modificarlo: **sudo cp /boot/config.txt/boot/config.txt.org**. Esto permitirá volver al archivo original en caso de que exista algún problema después de la modificación. En un PC con Windows, se puede acceder a este archivo porque está en la partición FAT de la tarjeta micro SD.

En el archivo `config.txt`, busque las cuatro líneas siguientes:

```
# uncomment the following to adjust overscan.
# Use positive numbers if console goes off screen,
# and negative if there is too much border
#overscan_left=16
#overscan_right=16
#overscan_top=16
#overscan_bottom=16
```

Modifique las cuatro líneas que empiezan por **#overscan**, eliminando la **#** del comienzo de línea. Estas líneas estaban marcadas como comentadas y por tanto no se tenían en cuenta. Eliminando la **#** quedan activadas.

Si la imagen se sale de la pantalla, escriba números positivos. Si las franjas negras persisten, escriba números negativos. Cuanto más grande sea en número negativo, más fina será la franja. El borde es más pequeño con -30 que con -15. Hay que reiniciar la Raspberry Pi para que las modificaciones se tengan en cuenta y para visualizar su efecto.

Compruebe los valores modificándolos lo suficiente (por 5 o por 10) para ver el cambio en la pantalla. A continuación, afine el ajuste. Los cuatro valores no son necesariamente idénticos. El objetivo es que la imagen ofrecida por la Raspberry Pi esté perfectamente centrada y ocupe la totalidad de la pantalla.

A2 Hostname

La configuración inicial proviene de la imagen que ha servido para la creación del sistema. Esta es la configuración por defecto que se presenta al inicio de la Raspberry Pi. En particular, el nombre del ordenador o *hostname* ya está fijo. Su Raspberry Pi se llama *raspberrypi*. Si ha comprado una única Raspberry Pi, esto no plantea ningún problema, pero si tiene varias tarjetas funcionando simultáneamente corre el riesgo de tener conflictos debido a los nombres idénticos de los sistemas.

Por otra parte, cuando varias máquinas conviven en una red, es interesante dar nombres de acuerdo con su función. De esta manera, *estacionmeteo* o *mediacenter*, permiten identificar de un vistazo a cada máquina mucho más fácilmente que *raspberrypi1* y *raspberrypi2*.

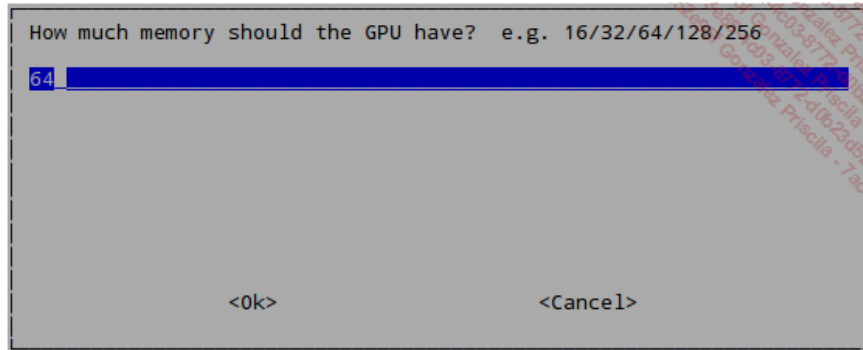
Los archivos */etc/hosts* y */etc/hostame* se ven afectados por la asignación del nombre del host. La opción **A2 Hostname** permite modificar el nombre del ordenador sin preocuparse de las modificaciones que se deben hacer en uno o varios archivos.

Seleccione la opción **A2 Hostname** y valide. Escriba el nuevo nombre del ordenador y valide.

- Atención: solo se aceptan letras de la 'a' a la 'z' (en mayúsculas o minúsculas), las cifras de '0' a '9' y el guión bajo. El nombre de host no puede empezar o terminar por guión. No se permite ningún otro símbolo, carácter de puntuación o espacio en blanco.

A3 Memory Split

La opción **A3 Memory Split** permite repartir la memoria entre los dos procesadores de la Raspberry Pi, la CPU y la GPU. No hay un "mejor valor". El valor de memoria que se debe reservar para cada procesador depende de la aplicación.



Por defecto, se reservan 64 MB de memoria para la GPU. Es posible ajustar este valor a 16, 32, 64, 128 o 256 MB. El resto de la memoria disponible está a disposición de la CPU. Por ejemplo, en una Raspberry Pi Zero equipada con 512 MB de RAM, con el valor por defecto de 64 MB de memoria asignada a la GPU, quedan 448 MB de RAM para la CPU.

Un centro multimedia o una consola de videojuegos necesitan más memoria en la GPU, mientras que un servidor en modo texto, sin pantalla conectada, requiere más recursos de memoria para la CPU. Reservar más memoria para la GPU mejora el rendimiento en modo gráfico, pero limita el número de aplicaciones que se pueden cargar simultáneamente en memoria y puede reducir el rendimiento de otros programas.

Después de haber indicado la cantidad de memoria asignada a la GPU, seleccione **<Ok>** y valide. El reparto de memoria será efectivo durante el próximo reinicio del sistema.

- Si sabe usar la línea de comandos, compruebe con el comando `top` que el reparto de memoria corresponde al valor seleccionado. En la cuarta línea de la pantalla, el valor situado a la derecha de **KiB Mem** debe corresponder a los valores de la memoria total y de la memoria GPU.

También es posible intervenir directamente a nivel del archivo *config.txt* para fijar la cantidad de memoria disponible para la CPU, independientemente del modelo de Raspberry Pi. En este caso es necesario indicar la cantidad de memoria reservada para la GPU para los dos tipos de RAM disponibles:

```
gpu_mem_512=384
gpu_mem_1024=512
```

Con 512 MB de RAM y 384 MB reservados para la GPU, quedan 128 MB para la CPU.

Con 1024 MB de RAM y 512 MB reservados para la GPU, quedan 512 MB para la CPU.

Puede haber diferentes valores de `gpu_mem` en *config.txt*, pero solo se tomará en cuenta el que afecta a la tarjeta en la que funciona el sistema.

En el ejemplo anterior, si el sistema funciona en una Raspberry Pi Zero, será la línea `gpu_mem_512=384` la que se utilizará. En una Raspberry Pi 3, será `gpu_mem_1024=512`.

Los valores máximos admitidos son:

Memoria RAM 512 MB: Memoria GPU maxi = 448 MB

Memoria RAM 1024 MB: Memoria GPU maxi = 944 MB

- Si ajusta el valor de `gpu_mem` a un valor demasiado bajo, algunas funcionalidades del firmware se desactivan automáticamente. El firmware no tendrá suficiente espacio en memoria para ejecutarlos. Esto puede afectar al tratamiento de imágenes, así como al sonido. Si tiene este tipo de problemas, modifique la compartición de la memoria, aumentando la memoria asignada a GPU.

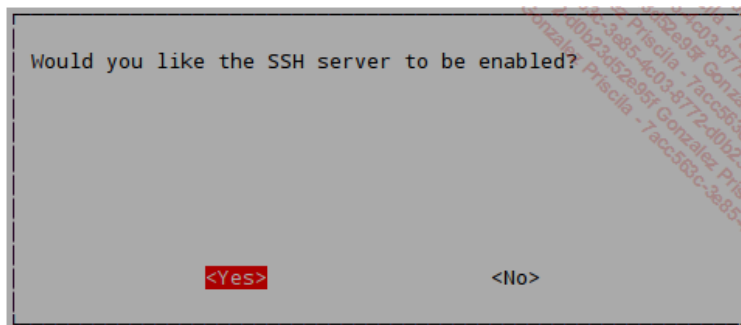
A4 SSH

SSH (*Secure Shell* = conexión segura) permite conectarse en modo texto a un equipo remoto y realizar intercambios de manera cifrada. El servidor dispone de una clave privada que conserva secretamente. Distribuye una clave pública que se usa para cifrar los mensajes. En la recepción, solo la clave privada correspondiente a la clave pública puede descifrar los mensajes. Cuando el servidor envía un mensaje, lo cifra con su clave privada. Solo las máquinas que tienen su clave pública pueden descifrarlo. En este caso se comprueba que el mensaje proviene realmente del servidor, porque solo él dispone de la clave privada que ha servido para crear los mensajes. Actualmente, no existe ningún medio conocido para recrear la clave privada a partir de la clave pública.

Por defecto, el servicio SSH está activo en la distribución Raspbian.

La opción avanzada **A4 SSH** permite activar o prohibir el funcionamiento del servicio SSH.

Para activar el servicio, sitúe el foco en **A4 SSH** y valide.



Se abre una ventana que le pregunta si desea activar **<Yes>** o prohibir **<No>** el servicio SSH. Seleccione **<No>** si desea desactivar el servicio y valide.

Para conectarse a la Raspberry Pi en remoto, consulte el capítulo Conectarse a la Raspberry Pi en remoto.

A5 VNC

Esta opción arranca el servidor VNC para permitir tomar el control en modo remoto a la Raspberry Pi.

A6 SPI

Esta opción valida la carga del módulo de gestión del bus SPI por el núcleo con el arranque del sistema. La gestión del bus SPI permite que la Raspberry Pi tenga en cuenta tarjetas de extensión (como la tarjeta PiFace), que usan este bus para dialogar con la SoC.

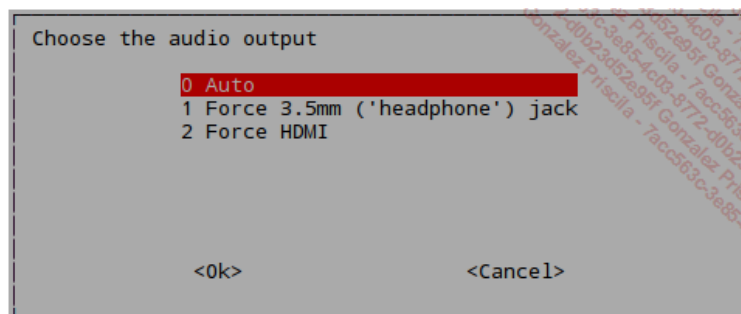
A7 I2C

Esta opción permite cargar el módulo de gestión del bus I2C con el arranque del núcleo. Por defecto, este núcleo no se carga en el arranque. El bus I2C permite la comunicación con las tarjetas de extensión.

A8 Serial

Esta opción permite autorizar el login del usuario a través del puerto en serie de la Raspberry Pi.

A9 Audio



Por defecto, si su pantalla HDMI (TV o monitor) está equipada con altavoces, transmite esta información a la Raspberry Pi y el sonido se envía a la salida HDMI predeterminada. Si no es el caso, la salida por defecto es el jack analógico de 3,5 mm. Esta opción permite forzar el envío del sonido a la salida que prefiera.

En algunos casos muy raros, es preciso forzar el modo HDMI en el archivo `/boot/config.txt`, configurando **hdmi_drive=2**. A continuación, hay que reiniciar la Raspberry Pi para que esta modificación se tenga en cuenta.

AA 1-Wire

Esta opción permite activar la gestión del bus 1-wire (también llamada OneWire). Este bus, diseñado por *Dallas Semiconductor*, permite conectar en paralelo o en estrella componentes con solo dos cables, uno para los datos y otro que hace de masa. Esto es posible gracias a la presencia de un condensador en cada componente 1-drive. Este condensador garantiza la alimentación del circuito durante los periodos de actividad de la línea de datos.

1-drive se utiliza habitualmente en las aplicaciones domóticas para conectar termómetros o sensores de datos meteorológicos. Cada componente se identifica de manera única por una dirección codificada en 64 bits. La dirección tiene un identificador del tipo de componente.

AB GPIO Server

Esta opción permite autorizar o prohibir el acceso a *pigpio* desde el exterior, a través de la red. Esta librería, destinada a la Raspberry Pi, funciona en todos los modelos y proporciona acceso a los puertos de la GPIO en modo lectura y escritura, para generar el PWM (*Pulse Width Modulation* = Modulación de largo impulso)... Si *pigpio* funciona con los permisos de root, permite a un simple usuario que no tenga permisos de administración usar la GPIO.

AC GL Driver

En febrero de 2016, la Fundación publicó un driver Open GL experimental. Este driver usa la aceleración de hardware de la GPU para acelerar la visualización, lo que permite usar algunos juegos, así como software 3D como FreeCAD u OpenSCAD.

A0 Update

Lanza la actualización de la distribución.

m. 9 About raspi-config

No es una opción propiamente dicha. Esta entrada de menú informa del uso de `raspi-config`:

"Esta herramienta ofrece un medio rápido para realizar la configuración inicial de la Raspberry Pi. Aunque se puede ejecutar en cualquier momento, algunas opciones pueden ser difíciles de configurar si ha personalizado mucho manualmente su instalación." (Traducción del mensaje mostrado).

3. Configuración avanzada de Raspbian

La configuración fina de Raspbian utiliza por el archivo `/boot/config.txt`. Este es el archivo que se modifica cuando cambian las opciones ofrecidas por `raspi-config`.

a. El archivo config.txt

El archivo `config.txt` es un archivo de configuración en modo texto. Las líneas que lo componen pueden empezar por el signo `#` si son líneas de comentario. En cualquier caso, todo el contenido de la línea se ignora. Poner un `#` al comienzo de la línea es un sistema para no aplicar la configuración que sigue a continuación. Si desea añadir un comentario, comience la línea con un `#`.

Las líneas que no empiezan con un `#` son las líneas de configuración. Por tanto, contienen información en formato **propiedad=valor**.

Las opciones son numerosas y evolucionan con las versiones de Raspbian. El enlace elinux.org/RPiconfig enumera todas las propiedades y los valores posibles.

La mayor parte de los usuarios pueden configurar la Raspberry Pi a partir de `raspi-config`. Los usos avanzados necesitarán una intervención manual en el archivo `config.txt`.

El archivo contiene una serie de líneas en formato **propiedad=valor**.

La siguiente tabla describe los principales ajustes disponibles (junio de 2016) para personalizar su sistema. Conviene consultar regularmente la página <http://elinux.org/RPiconfig> para conocer la evolución de estos argumentos.

Gestión de la memoria		
Propiedad	Valor por defecto	Descripción
<code>disable_l2_cache</code>	0	Desactiva el acceso de la CPU ARM a la memoria caché de nivel 2 de la GPU. Necesita un núcleo adaptado, sin acceso a la caché L2.
<code>gpu_mem</code>	64	Memoria reservada para la GPU en MB. El resto de memoria se asigna a la CPU ARM. El valor debe ser un múltiplo de 16, con un mínimo de 16. El máximo depende de la RAM de la tarjeta utilizada (ver <code>gpu_mem_xxx</code> más adelante). El valor por defecto es 64. Un valor demasiado bajo puede impedir que funcionen algunas funcionalidades del firmware.
<code>gpu_mem_1024</code>	-	Fija el valor por defecto de la cantidad de memoria asignada a la GPU en las Raspberry Pi que disponen de 1024 MB (1 GB) de memoria. Esta propiedad se ignora en las Raspberry Pi que disponen de 512 o 256 MB de memoria y elimina el valor de <code>gpu_mem</code> . El valor debe ser un múltiplo de 16, con un máximo de 944.
<code>gpu_mem_512</code>	-	Fija el valor por defecto de la cantidad de memoria asignada para la GPU en las Raspberry Pi que disponen de 512 MB de memoria. Esta propiedad se ignora en las Raspberry Pi que disponen de 256 MB de memoria y borra el valor de <code>gpu_mem</code> . El valor debe ser un múltiplo de 16, con un máximo de 448.
<code>gpu_mem_256</code>	-	Fija el valor por defecto de la cantidad de memoria asignada para la GPU en las Raspberry Pi que disponen de 256 MB de memoria. Esta propiedad se ignora en las Raspberry Pi que disponen de 512 MB de memoria y borra el valor de <code>gpu_mem</code> . El valor debe ser un múltiplo de 16, con un máximo de 192.
<code>disable_pvt</code>	0	Si = 1: desactiva el ajuste de la frecuencia de refresco de la RAM cada 500 ms. La GPU monitoriza la temperatura de la RAM y ajusta el refresco (<i>Process Voltage Temperature</i> =proceso tensión temperatura). El valor por defecto es 0.

Gestión de la cámara		
Propiedad	Valor por defecto	Descripción
<code>disable_camera_led</code>	0	Poner esta propiedad a 1 prohíbe el encendido del LED en la cámara durante la grabación de fotografías o vídeos. Atención: si utiliza la cámara, la memoria mínima asignada para la GPU debe ser 128 MB: <code>gpu_mem=128</code> .

Gestión del modo vídeo		
Propiedad	Valor por defecto	Descripción
<code>sdtv_mode</code>	0	0 NTSC normal 1 NTSC japonés 2 PAL normal - Europa - 625 líneas - 50 imágenes/s. 3 PAL brasileño - 525 líneas - 60 imágenes/s.
<code>sdtv_aspect</code>	1	1 imagen 4:3 2 imagen 14:9 3 imagen 16:9
<code>sdtv_disable_colorburst</code>	-	Elimina la información de color de la salida de vídeo compuesta. La imagen es en blanco y negro, pero puede ser de mejor calidad.
<code>hdmi_safe</code>	-	Utiliza los argumentos más seguros para intentar iniciar con la compatibilidad HDMI mejor posible. Asignar el valor 1 a esta propiedad equivale a: <code>hdmi_force_hotplug=1, hdmi_ignore_edid=0xa5000080, config_hdmi_boost=4, hdmi_group=2, hdmi_mode=4, disable_overscan=0, overscan_left=24, overscan_right=24, overscan_top=24, overscan_bottom=24</code> .
<code>hdmi_ignore_edid</code>	-	Si el monitor HDMI conectado a la Raspberry Pi no ofrece datos EDID (<i>Extended Display Identification Data</i> = datos de identificación de la pantalla) para informar al sistema de sus características, esta opción autoriza al sistema para que ignore los datos. Si se usa, es necesario darle el valor <code>0xa5000080</code> .
<code>hdmi_edid_file</code>	-	Si esta propiedad vale 1, el sistema lee los datos en el archivo <code>edid.dat</code> , en lugar de esperarlos por el monitor.
<code>hdmi_force_edid_audio</code>	-	Con valor 1, esta propiedad informa al sistema que todos los formatos de audio se aceptan por pantalla. La DTS/AC3 se envía a la pantalla, incluso aunque no indique que lo acepta.
<code>hdmi_ignore_edid_audio</code>	-	Con valor 1, esta propiedad informa al sistema que no se acepta por la pantalla ningún formato de audio. Por defecto, ALSA (<i>Advanced Linux Sound Architecture</i> = arquitectura sonora avanzada para Linux) utiliza la salida de sonido analógico (jack 3,5 mm).
<code>hdmi_force_edid_3d</code>	-	Con valor 1, esta propiedad informa al sistema que todos los modos 3D del CEA (<i>Consumer Electronics Association</i> = asociación de usuarios de electrónica) están soportados por la pantalla, incluso aunque no lo indiquen los datos EDID (<i>Extended Display IDentification</i> = identificación mejorada de la pantalla).

avoid_edid_fuzzy_match	-	Con valor 1, esta propiedad informa al sistema que haga búsquedas aproximadas en los modos de vídeo indicados en los datos EDID. El sistema selecciona el modo estándar con la resolución exacta y la frecuencia de refresco más cercana, incluso aunque el borrado no sea correcto.
hdmi_ignore_cec_init	-	Con valor 1, esta propiedad indica al sistema que no envíe el mensaje inicial que indica que está activo. Esto evita que el televisor salga de su estado de reposo cuando el sistema arranca de nuevo por la CEC (<i>Consumer Electronics Control</i> = control por el mando a distancia del televisor).
hdmi_ignore_cec	-	Con valor 1, esta propiedad indica al sistema que el televisor no soporta la CEC. Ninguna de las funciones CEC será reconocida.
hdmi_force_hotplug	-	Con valor 1, esta propiedad hace creer al sistema que un monitor HDMI está conectado, incluso aunque no sea cierto. La salida de vídeo se fuerza a HDMI.
hdmi_ignore_hotplug	-	Con valor 1, esta propiedad hace creer al sistema que no hay ningún monitor. Permite forzar la salida de vídeo a la salida de vídeo compuesta.
hdmi_pixel_encoding	0	Fuerza el modo de codificación de los píxeles. Por defecto a 0. El modo DMT (<i>Display Monitor Timings</i> = frecuencias de los monitores informáticos) utilizado en los monitores informáticos no tiene los modos entrelazados, mientras que los modos CEA utilizados en televisión incluyen los modos de escaneo entrelazado. 0 limitado en CEA, completo en DMT 1 RVB limitado (16 a 235) 2 RVB completo (0 a 255) 3 YCbCr limitado (16 a 235) 4 YCbCr completo (0 a 255)
hdmi_drive	-	Permite seleccionar entre los modos HDMI y DVI. 1 modo DVI normal, sin sonido 2 modo HDMI normal, el sonido se envía al monitor si no está invalidado y si el monitor lo acepta.
hdmi_group	0	Si el grupo no se especifica o se pone a 0, el sistema utiliza el grupo transmitido por EDID. 1 CEA = formatos de escaneo de televisión. 2 DMT = formatos de escaneo de monitores informáticos.
hdmi_mode		Define la resolución de pantalla que se debe usar para los modos CEA y DMT. Usa un enlace con <code>hdmi_group</code> anterior. Ver las tablas del anexo.
overscan_left		Número de píxeles que se ignoran a la izquierda de la pantalla.
overscan_right		Número de píxeles que se ignoran a la derecha de la pantalla.
overscan_top		Número de píxeles que se ignoran en la parte superior de la pantalla.
overscan_bottom		Número de píxeles que se ignoran en la parte inferior de la pantalla.
framebuffer_width	L - overscan	
framebuffer_height	H - overscan	
framebuffer_depth	16	Profundidad de la memoria imagen de la consola. Corresponde al número de bits por píxel. El valor por defecto es 16. Se aceptan 8 bits de profundidad, pero la paleta RGB hace que la pantalla no sea legible. 24 bits parecen mejor, pero presentan defectos. Con 32 bits no hay defectos, pero necesitan que <code>framebuffer_ignore_alpha</code> valga 1 indican y muestra colores falsos.
framebuffer_ignore_alpha		Poner a 1 esta propiedad desactiva el canal alpha. Puede ser útil con una profundidad de 32 bits (<code>framebuffer_depth=32</code>).
test_mode		Permite el vídeo y el sonido durante el boot con fines de prueba en producción (solo en la salida de vídeo compuesto).
disable_overscan		Si esta propiedad vale 1, el overscan se desactiva.
config_hdmi_boost	0	Configura el nivel de la señal en la salida HDMI. El valor por defecto es 0, puede intentar pasar a 4 si tiene problemas de interferencias durante el uso de la salida HDMI. El valor máximo es 7.
display_rotate	0	Rotación de la visualización en el sentido de las agujas del reloj en la pantalla. El valor por defecto es 0. También puede invertir la visualización. 0 normal 1 rotación de 90 grados 2 rotación de 180 grados 3 rotación de 270 grados 0x10000 invierte la pantalla horizontalmente 0x20000 invierte la pantalla verticalmente Las rotaciones de 90 y 270 grados necesitan más memoria en la GPU. 16 MB no son suficientes y la Raspberry Pi no puede arrancar.
overscan_scale		Con el valor 1, esta propiedad fuerza a openFrameworks a que respete los argumentos de overscan (izquierda, derecha, arriba y abajo) en caso de utilizar una pantalla LCD conectada a la salida de vídeo compuesta. Se usa con XBMC (OpenELEC) para que respete los valores de overscan.
lcd_rotate	0	Rotación simultánea de la visualización y de la pantalla táctil. Las opciones son idénticas a las de <code>display_rotate</code> , pero la más usada es <code>lcd_rotate=2</code> cuando el vídeo se utiliza en la pantalla oficial Raspberry Pi.

Configuración del modo vídeo

Es posible forzar el modo vídeo en el archivo `config.txt` durante el arranque de Raspbian. Esto puede ser útil con un monitor antiguo que no es capaz de mostrar la resolución por defecto de la Raspberry Pi.

`hdmi_group` determina si la pantalla es un monitor informático (grupo 2) o un televisor (grupo 1).

`hdmi_mode` indica la resolución de la imagen y la frecuencia de imagen.

Las tablas de configuración del vídeo se encuentran en el Anexo.

b. Configuración del vídeo

Un monitor con una entrada HDMI soporta únicamente, por lo general, una parte de los formatos anteriores. Cuando inicia la Raspberry Pi, es

posible que el monitor no muestre nada. Entonces hay que configurar la salida en VGA, que es el modo más básico que debería ser reconocido por el monitor. Para esto ajuste las propiedades `hdmi_group=1` y `hdmi_mode=1` en el archivo `config.txt`. En ausencia de visualización en la Raspberry Pi esta modificación se puede hacer en un PC, accediendo a la tarjeta micro SD con un lector de tarjetas.

Cuando se ha obtenido una visualización, el comando `/opt/vc/bin/tvservice` permite recuperar cierta cantidad de información relativa al monitor conectado al puerto HDMI. El conocimiento de los modos soportados por el monitor permite configurar manualmente el vídeo.

Sintaxis

`tvservice [OPTION]`

[OPTION]	Ejecutado sin opciones, el comando <code>tvservice</code> muestra la lista completa de opciones disponibles.
-m MODE	Enumera los modos soportados por el monitor en el modo indicado. <code>/opt/vc/bin/tvservice -m CEA</code> Enumera los modos soportados en CEA. <code>/opt/vc/bin/tvservice -m DMT</code> Enumera los modos soportados en DMT.
-d edid.dat	Muestra la información detallada proporcionada por el monitor.
-s	Muestra el estado actual de su conexión.

Después de haber recuperado la información relativa a su monitor, configure los valores `hdmi_group` y `hdmi_mode` para que la salida de vídeo HDMI envíe al monitor una imagen en un formato que reconozca.

c. Formatos soportados por un monitor

Un monitor HDMI solo acepta un número limitado de formatos. Para saber cuáles se soportan, use los siguientes comandos:

¿Cuáles son los modos soportados en CEA (TV)?

```
/opt/vc/bin/tvservice -m CEA
```

¿Cuáles son los modos soportados en DMT (monitor informático)?

```
/opt/vc/bin/tvservice -m DMT
```

¿Cuál es la configuración utilizada actualmente?

```
/opt/vc/bin/tvservice -s
```

¿Cómo mostrar más información?

```
/opt/vc/bin/tvservice -d edid.dat; /opt/vc/bin/edidparser edid.dat
```

Resultados obtenidos con una Raspberry Pi Zero conectada a una pantalla iiyama ProliteB2274HDS

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -m CEA
Group CEA has 12 modes:
  mode 1: 640x480 @ 60Hz 4:3, clock:25MHz progressive
  mode 2: 720x480 @ 60Hz 4:3, clock:27MHz progressive
  mode 3: 720x480 @ 60Hz 16:9, clock:27MHz progressive
  mode 4: 1280x720 @ 60Hz 16:9, clock:74MHz progressive
  mode 5: 1920x1080 @ 60Hz 16:9, clock:74MHz interlaced
  (prefer) mode 16: 1920x1080 @ 60Hz 16:9, clock:148MHz progressive
  mode 17: 720x576 @ 50Hz 4:3, clock:27MHz progressive
  mode 18: 720x576 @ 50Hz 16:9, clock:27MHz progressive
  mode 19: 1280x720 @ 50Hz 16:9, clock:74MHz progressive
  mode 20: 1920x1080 @ 50Hz 16:9, clock:74MHz interlaced
  mode 30: 720x576 @ 50Hz 16:9, clock:54MHz x2 progressive
  mode 31: 1920x1080 @ 50Hz 16:9, clock:148MHz progressive
```

El modo usado entre los 12 disponibles es el CEA 31 1920x1080 píxel @50Hz

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -m DMT
Group DMT has 19 modes:
  mode 4: 640x480 @ 60Hz 4:3, clock:25MHz progressive
  mode 5: 640x480 @ 72Hz 4:3, clock:31MHz progressive
  mode 6: 640x480 @ 75Hz 4:3, clock:31MHz progressive
  mode 8: 800x600 @ 56Hz 4:3, clock:36MHz progressive
  mode 9: 800x600 @ 60Hz 4:3, clock:40MHz progressive
  mode 10: 800x600 @ 72Hz 4:3, clock:50MHz progressive
  mode 11: 800x600 @ 75Hz 4:3, clock:49MHz progressive
  mode 16: 1024x768 @ 60Hz 4:3, clock:65MHz progressive
  mode 17: 1024x768 @ 70Hz 4:3, clock:75MHz progressive
  mode 18: 1024x768 @ 75Hz 4:3, clock:78MHz progressive
  mode 21: 1152x864 @ 75Hz 4:3, clock:108MHz progressive
  mode 32: 1280x960 @ 60Hz 4:3, clock:108MHz progressive
  mode 35: 1280x1024 @ 60Hz 5:4, clock:108MHz progressive
  mode 36: 1280x1024 @ 75Hz 5:4, clock:135MHz progressive
  mode 47: 1440x900 @ 60Hz 16:10, clock:106MHz progressive
  mode 48: 1440x900 @ 75Hz 16:10, clock:136MHz progressive
  mode 51: 1600x1200 @ 60Hz 4:3, clock:162MHz progressive
  mode 58: 1680x1050 @ 60Hz 16:10, clock:146MHz progressive
  mode 82: 1920x1080 @ 60Hz 16:9, clock:148MHz progressive
```

El monitor dispone de 19 modos en DMT.

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -s
state 0x12000a [HDMI CEA (16) RGB lim 16:9], 1920x1080 @ 60.00Hz, progressive
```

Este comando muestra el modo actual utilizado en el puerto HDMI.

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -d edid.dat; /opt/vc/bin/edidparser edid.dat
```

```
Written 256 bytes to edid.dat
Enabling fuzzy format match...
Parsing edid.dat...
HDMI:EDID version 1.3, 1 extensions, screen size 48x27 cm
.../...
HDMI:EDID DMT mode (82) 1920x1080p @ 60 Hz with pixel clock 148 MHz
has a score of 149416
HDMI:EDID preferred mode remained as CEA (16) 1920x1080p @ 60 Hz
with pixel clock 148 MHz
HDMI:EDID has HDMI support and audio support
edid_parser exited with code 0
```

En este breve extracto de información obtenida con este comando se puede ver que el monitor tiene un tamaño de 48x27 cm y que soporta el audio en HDMI.

Esta información se puede usar cuando se solucionan problemas, para entender el origen de un comportamiento incorrecto.

Resultados obtenidos con una Raspberry Pi 3 conectada en una pantalla táctil de 7 conexiones oficial

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -m CEA
Group CEA has 0 modes:
```

La interfaz utilizada para la adaptación HDMI/pantalla táctil no soporta ningún modo CEA (TV).

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -m DMT
Group DMT has 0 modes:
```

La interfaz utilizada para la adaptación HDMI/pantalla táctil, no soporta ningún modo DMT (monitor informático).

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -s
state 0x400000 [LCD], 800x480 @ 60.00Hz, progressive
```

El modo usado es 800 x 480 píxeles @60Hz, lo que corresponde a la pantalla táctil conectada a la Raspberry Pi 3.

```
pi@raspberrypi:~ $ /opt/vc/bin/tvservice -d edid.dat;
/opt/vc/bin/edidparser edid.dat
Nothing written!
Enabling fuzzy format match...
Failed to open edid.dat
edid_parser exited with code -1
```

El circuito utilizado en la pantalla táctil no devuelve ningún archivo edid.dat.

Añadir las licencias

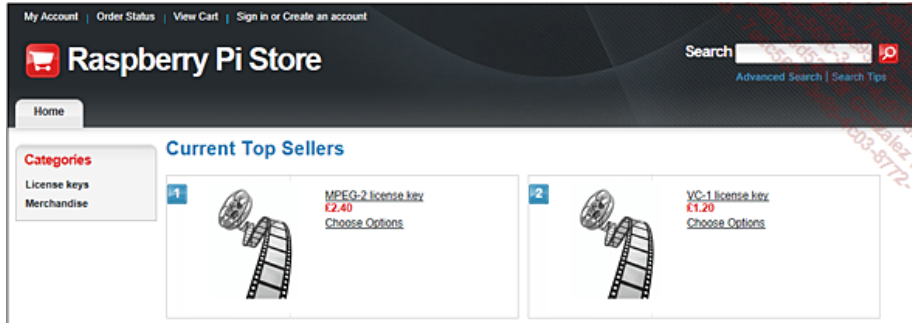
Para mantener un precio inferior al límite fijado, la Fundación Raspberry Pi ha decidido no incluir en su producto las licencias que permiten decodificar los flujos MPEG-2 y VC-1.

Es posible solicitar las licencias en línea.

La licencia MPEG-2 cuesta alrededor de 3 €. Permite visualizar los flujos de vídeo codificados según esta norma. MPEG2 es un formato cerrado debido a la presencia de patentes de software. Define la compresión que se aplica a la imagen y al sonido, así como su transporte en las redes. MPEG-2 se aplica en televisión digital (cable, TNT, satélite), y también en los DVD de vídeo y en los SVCD (*Super Video CD*).

La licencia VC-1 se corresponde con un códec competencia del H.264. Se ha creado por la SMTPE (*Society of Motion Picture and Television Engineers*= Sociedad de Ingenieros de Cine y Televisión). Esta asociación desarrolla los estándares de vídeo utilizados tanto para la televisión como para el cine digital. La versión HD de VC-1 se usa en los HD DVD de NEC y los Blu-ray de Sony. La licencia se vende por unos 1,50 €.

Puede comprar las licencias en Raspberry Pi Store (<http://www.raspberrypi.com/>).



Después de hacer clic en la licencia, es necesario introducir el número de serie de su SoC. Este número no es el número de serie que figura en la tarjeta de la Raspberry Pi, sino el número interno del procesador.

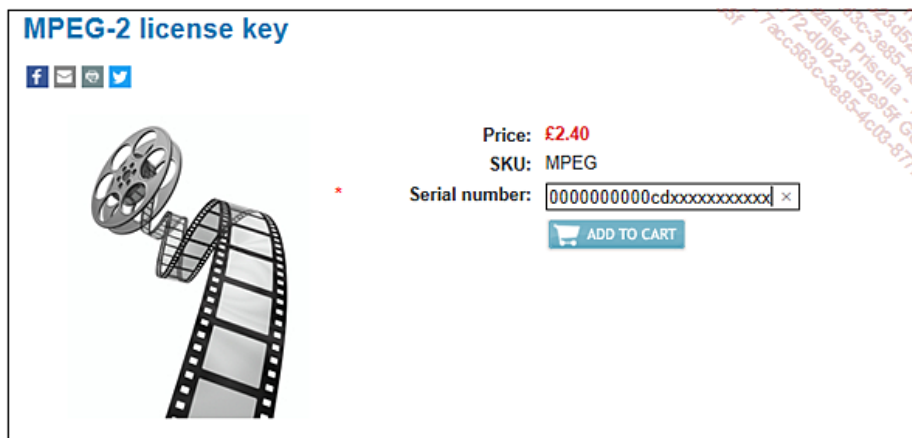
De hecho, la licencia que compra está vinculada al procesador de su tarjeta y solo será válida para esta tarjeta Raspberry Pi. Si tiene varias tarjetas, deberá adquirir varias licencias.

Para obtener el número de serie de su procesador, consulte el contenido del archivo `/proc/cpuinfo` (las últimas cifras del número de serie se han sustituido por x por razones de confidencialidad).

```
pi@raspberrypi ~ $ cat /proc/cpuinfo
processor       : 0
model name     : ARMv6-compatible processor rev 7 (v6l)
Features       : swp half thumb fastmult vfp edsp java tls
CPU implementer : 0x41
CPU architecture: 7
CPU variant    : 0x0
CPU part       : 0xb76
CPU revision   : 7

Hardware       : BCM2708
Revision       : 0010
Serial         : 00000000xxxxxxxxxx
```

Copie el número de serie que figura en la última línea del archivo y vuelva a copiarlo en la zona de texto de la tienda Raspberry Pi.



Añada su compra al carrito pulsando **ADD TO CART**. Compre también la licencia VC-1 si lo necesita. Después del pago, sus licencias le llegarán en 72 horas por e-mail.

Cuando las haya recibido, las tendrá que añadir a su archivo de configuración `/boot/config.txt`. Este archivo es accesible y modificable desde una máquina en Windows, usando un lector de tarjeta micro SD.

➤ Piense en realizar una copia del archivo original antes de realizar cualquier modificación.

Añada estas líneas al final del archivo en función de las licencias adquiridas:

```
decode_MPG2=0x12345678
decode_WVC1=0x90123456
```

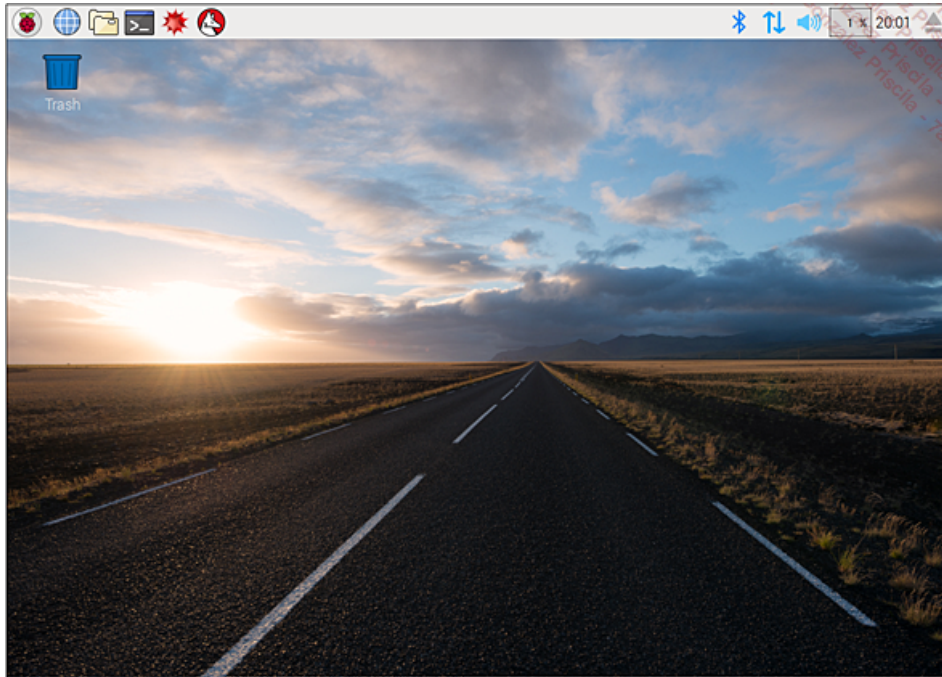
Sustituya los valores a la derecha del signo igual por las claves de licencia que haya recibido. Tras el próximo inicio de su Raspberry Pi, los flujos codificados en MPEG-2 y VC-1 se leerán con normalidad.

Si tiene varias Raspberry Pi B+ y utiliza una misma micro SD de Raspberry Pi en Raspberry Pi, puede guardar hasta 8 licencias en esta tarjeta micro SD:

```
decode_XXXX=0x12345678, 0x90123456, 0xabcdabcd, ...
```

Utilizar el sistema

Si opta por un arranque en modo gráfico (es el modo por defecto), un fondo de pantalla que representa una carretera debería decorar su pantalla. Vaya directamente al capítulo Utilizar el modo gráfico.



Si ha seleccionado arrancar en modo texto, la pantalla de la Raspberry Pi se ilumina y una cantidad de información aparece debajo del logo. A continuación, todo se detiene. En la parte inferior de la pantalla, a la derecha del texto `raspberrypi login:`, un cursor parpadea. El capítulo Usar la línea de comandos le permitirá descubrir la línea de comandos.

```

[... ] Configuring network interfaces...eth0: link up
done.
[ ok ] Cleaning up temporary files...
[ ok ] Setting up ALSA...done (none loaded).
[info] Setting console screen modes.
[info] Skipping font and keymap setup (handled by console-setup).
[ ok ] Setting up console font and keymap...done.
[ ok ] Setting kernel variables ...done.
[ ok ] Setting up X socket directories... /tmp/.X11-unix /tmp/.ICE-unix.
INIT: Entering runlevel: 2
[info] Using makefile-style concurrent boot in runlevel 2.
[ ok ] Network Interface Plugging Daemon...skip eth0...done.
[ ok ] Starting enhanced syslogd: rsyslogd.
[ ok ] Starting periodic command scheduler: cron.
[ ok ] Starting system message bus: dbus.
Starting dphys-swapfile swapfile setup ...
want /var/swap=100MByte, checking existing: keeping it
done.
[ ok ] Starting NTP server: ntpd.
[ ok ] Starting OpenBSD Secure Shell server: sshd.
My IP address is 192.168.1.115

Debian GNU/Linux 7 raspberrypi tty1

raspberrypi login: _
```

En la Raspberry Pi Zero (imagen anterior) aparece una sola frambuesa (un solo núcleo en la CPU), mientras que en la Raspberry Pi 3 (siguiente imagen) se mostrarán cuatro frambuesas (4 núcleos en la CPU).

```

[ 3.454635] hub 1-1:1.0: 5 ports detected
Welcome to Raspbian GNU/Linux 8 (Jessie)!

[ 3.613261] NET: Registered protocol family 10
[ 3.624908] systemd[1]: Inserted module 'ip6'
[ 3.630620] systemd[1]: Set hostname to 'raspberrypi'.
[ 3.721227] usb 1-1:1: new high-speed USB device number 3 using duc_otg
[ 3.753296] uart-pl011 3f201000.uart: no DMA platform data
[ 3.853534] usb 1-1.1: New USB device found, idVendor=0424, idProduct=ec00
[ 3.865578] usb 1-1.1: New USB device strings: bfr=0, Product=0, SerialNumber=0
[ 3.881180] smsc95xx v1.0.4
[ 3.947824] smsc95xx 1-1:1.0 eth0: register 'smsc95xx' at usb-3f900000.usb-1.1, smsc95xx USB 2.0 Ethernet, 80:27:eb:f4:de:06
[ 4.043069] usb 1-1.2: new high-speed USB device number 4 using duc_otg
[ 4.136623] systemd[1]: Expecting device dev-ttyAMA0.device...
Expecting device dev-ttyAMA0.device...
[ 4.154694] usb 1-1.2: New USB device found, idVendor=7392, idProduct=7811
[ 4.154707] usb 1-1.2: New USB device strings: bfr=1, Product=2, SerialNumber=3
[ 4.154717] usb 1-1.2: Product: BQ2.11n WLAN Adapter
[ 4.154726] usb 1-1.2: Manufacturer: Realtek
[ 4.154735] usb 1-1.2: SerialNumber: 00e04c000001
[ 4.209969] systemd[1]: Starting Forward Password Requests to Wall Directory Watch.
[ 4.221977] systemd[1]: Started Forward Password Requests to Wall Directory Watch.
[ 4.226410] systemd[1]: Starting Remote File Systems (Pre).
[ OK ] Reached target Remote File Systems (Pre).
[ 4.253662] systemd[1]: Reached target Remote File Systems (Pre).
[ 4.263049] usb 1-1.4: new low-speed USB device number 5 using duc_otg
[ 4.271311] systemd[1]: Starting Encrypted Volumes.
[ OK ] Reached target Encrypted Volumes.
[ 4.293703] systemd[1]: Reached target Encrypted Volumes.
[ 4.304678] systemd[1]: Starting Arbitrary Executable File Formats File System Automount Point.
[ OK ] Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 4.325770] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 4.340781] systemd[1]: Starting Swap.
```

Introducción

El modo texto es un poco austero, pero ofrece muchas posibilidades. La distribución Raspbian Jessie permite iniciar en modo gráfico para un manejo más usable.

En Windows, se ofrece una interfaz gráfica única, aunque es personalizable. Las distribuciones Linux se basan todas en un núcleo que dialoga con el mundo exterior en modo texto. Los creadores de una distribución seleccionan los intérpretes de comandos que van a permitir al usuario dialogar en modo texto. Seleccionan también, entre la larga lista de interfaces disponibles, la que se ajuste mejor a sus objetivos.

Las posibilidades son amplias; a continuación se presentan algunas interfaces gráficas, escogidas entre las más utilizadas:

- *Enlightment* es un entorno gráfico y un entorno de escritorio. Funciona en los PC y notebooks y en los dispositivos portátiles, tabletas y smartphones. Este entorno es personalizable, de una extrema sencillez, y permite el uso de funciones más avanzadas. La apariencia de las ventanas está muy trabajada y recuerda a la interfaz Mac OS X.
- *GNOME* es uno de los entornos de trabajo más extendido. Pero no es solo un entorno gráfico: se instala con una cantidad importante de aplicaciones incluidas. Inicialmente, *GNOME* tiene un software de correo electrónico, un visor de PDF, un softphone para la telefonía IP, juegos, un terminal, un administrador de archivos, un lector multimedia, etc.
- *KDE* se incluye en la mayor parte de las distribuciones de Linux, como *GNOME*. Existen numerosas aplicaciones desarrolladas especialmente para *KDE* que ofrecen al usuario el acceso a una gama importante de programas. Las aplicaciones desarrolladas para *KDE* tienen un nombre que contiene una K. *Kmail* es un administrador de correo, *KOrganizer* una agenda, *Konqueror* un administrador de archivos, *AmaroK* un lector multimedia, *KWrite* un editor de texto, *Konsole* un emulador de terminal...
- *LXDE* es un entorno de escritorio ligero. Está diseñado para consumir los menos recursos posibles y para ser rápido. Inicialmente, *LXDE* tiene pocas aplicaciones. *LXPanel* gestiona la apariencia del escritorio, las barras de herramientas y los applets que muestran. *LXSession* gestiona el inicio de sesión en el puesto y *LXTask* permite controlar la actividad de la máquina (carga del procesador, ocupación de memoria, programas que se están ejecutando). La gestión de las ventanas y escritorios virtuales se hace con *Openbox*.
- *XFCE* también es un entorno de escritorio ligero. Su filosofía se basa en tres ejes: sencillo de usar, rápido y con bajo consumo de recursos de sistema. *XFCE* integra algunas aplicaciones: un emulador de terminal, un visor de imágenes, una herramienta de captura de pantalla y un lector multimedia.

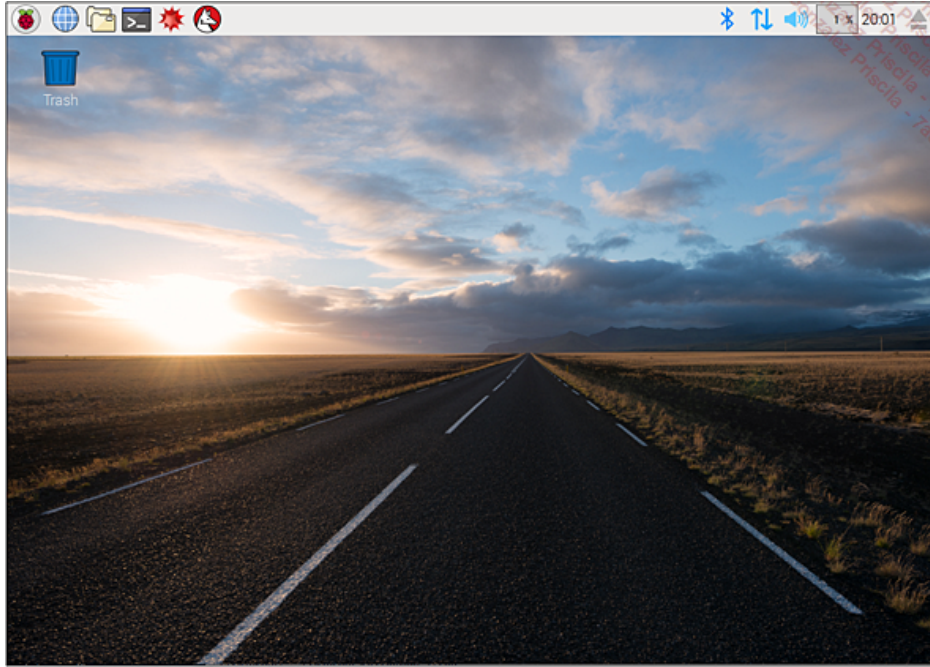
Los diseñadores de Raspbian han seleccionado *LXDE* para vestir su distribución. La decisión se basa en su bajo impacto en memoria y también bajo consumo de recursos. *LXDE* da muy buenos resultados en un Pentium III a 600 MHz, equipado con 512 MB de memoria RAM. En una Raspberry Pi 3, los resultados también son válidos, incluso si se retrasa algunas veces algunos segundos entre un clic y la apertura de la ventana correspondiente.

La aparición de Jessie y el aumento de potencia de la Raspberry Pi han modificado el arranque de la Raspberry Pi. Mientras que las versiones anteriores arrancaban en modo texto y necesitaban una modificación de la configuración para arrancar en modo gráfico, Jessie arranca en modo gráfico y dispone de un equivalente de la herramienta de configuración en modo texto, *raspi-config*.

Este capítulo le permitirá descubrir bastantes entornos de escritorio de la Raspberry Pi en Raspbian Jessie.

Descubriendo la pantalla de Raspbian

1. Iconos del escritorio de Jessie



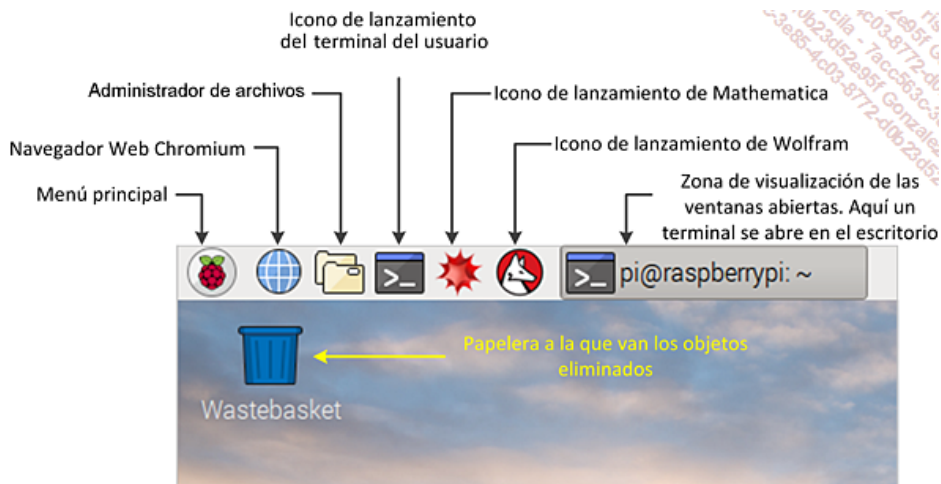
El escritorio se ha reducido a la más mínima expresión. Los muchos iconos que adornaban el escritorio en las primeras versiones de Raspbian han desaparecido en favor de un acceso a través del **Menú**. La única concesión es el mantenimiento de la papelera. La modernización llevada a cabo por PIXEL, además de la nueva apariencia de los iconos, es la introducción de fondos de pantalla. Estas evoluciones acercan Raspbian PIXEL a los sistemas operativos actuales.

2. La barra de tareas

a. Presentación

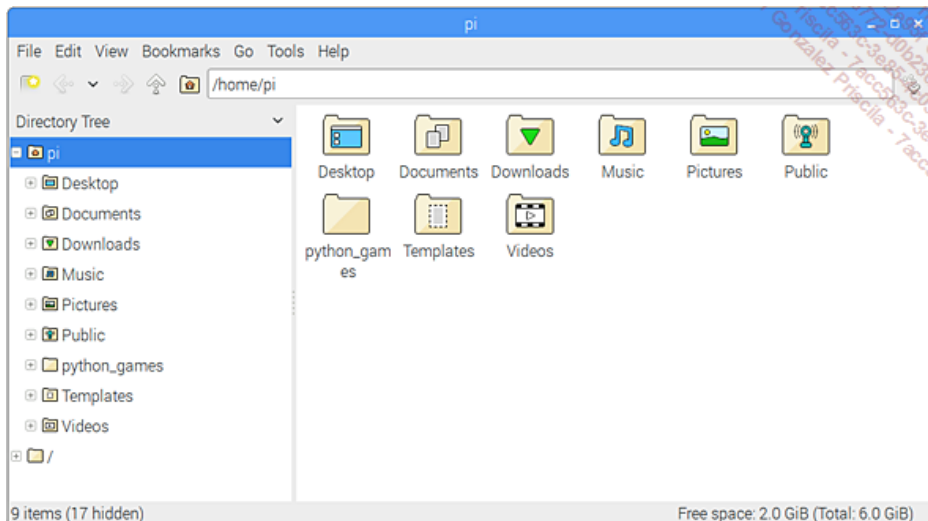
La barra de tareas de LXDE tiene una parte izquierda (siguiente imagen), una zona central que muestra las ventanas abiertas en el escritorio o minimizadas y una parte derecha que da información acerca de la carga del microprocesador e incluso la hora.

Zona izquierda de la barra de tareas



Un botón redondo con el logo de la Raspberry Pi, da acceso al menú principal de Raspbian. Es el botón **Menú**.

A la derecha del botón **Menú**, un globo del mundo permite abrir el navegador web Chromium. Después viene un icono en forma de carpeta que abre el administrador de archivos en la carpeta personal del usuario pi:

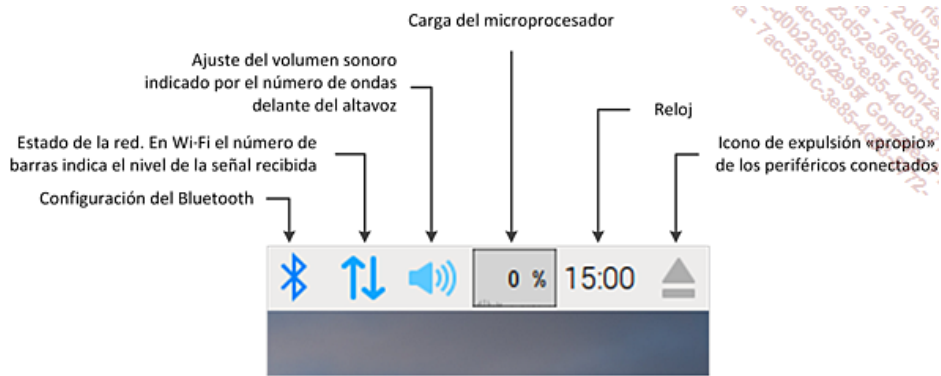


También tenemos iconos que permiten iniciar un terminal de usuario, de Mathematica y de Wolfram.

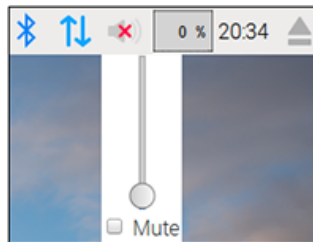
- Estos iconos inician las aplicaciones presentes en el menú. Vaya a la descripción del menú principal de Raspbian para obtener más detalles.

La zona central de la barra de tareas muestra una lista de ventanas abiertas en el escritorio o minimizadas.

Zona derecha de la barra de tareas



La zona derecha de la barra de tareas muestra, de izquierda a derecha, el icono de configuración Bluetooth y el estado de la red. El altavoz permite acceder a los ajustes del volumen, en forma de un cursor. El número de ondas sonoras junto al altavoz da una idea del ajuste actual del volumen. Después tenemos un monitor con la carga del procesador, un reloj digital y el icono de ejecución de los periféricos USB.



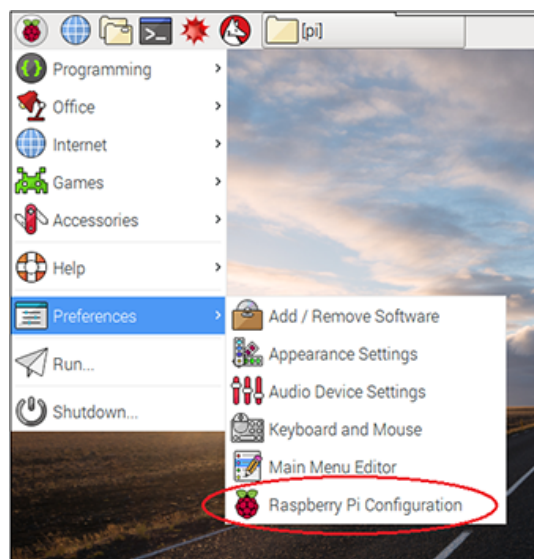
En la captura de pantalla anterior, el icono de red que representa dos ordenadores se transforma en un icono Wi-Fi que indica que la Raspberry Pi está conectada por este medio. El altavoz se adorna con una cruz cuando el volumen está ajustado al mínimo o está marcada la opción **Mute** (silenciar).

- El uso de estos iconos figura más adelante en este capítulo.

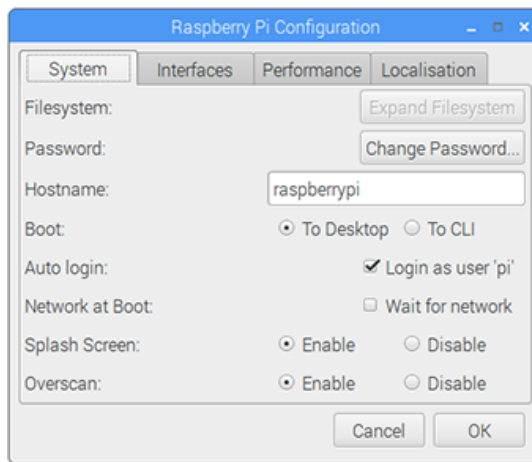
3. Configuración del escritorio en español

Durante el arranque de Raspbian PIXEL el idioma por defecto es el inglés. Es en este idioma en el que se muestran todos los menús y mensajes del sistema. Para utilizar el escritorio de Raspbian en español la primera operación a realizar es la configuración de las diferentes opciones relativas a la localización del sistema (idioma, huso horario, tipo de teclado, etc...).

a. Mostrar el escritorio de Raspbian en español



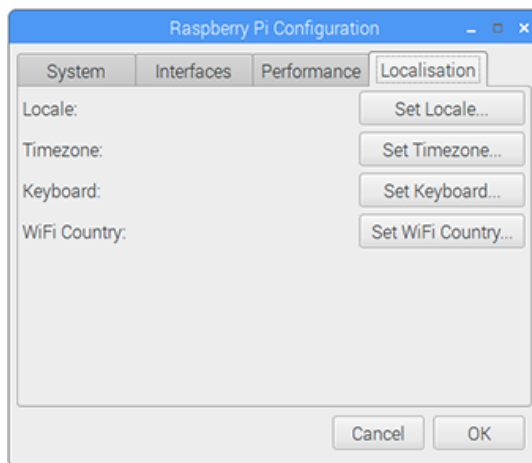
Pulse en el botón **Menú** situado en la parte superior izquierda del escritorio. Se abre el menú principal de LXDE. Pulse en **Preferences** y después en **Raspberry Pi Configuration**.



Se abre la ventana **Raspberry Pi Configuration**. Volveremos sobre todas las opciones disponibles a lo largo de este capítulo. Por el momento, el objetivo es configurar el sistema para que use el español.

➤ La traducción de todos los elementos de menú, etiquetas en las ventanas, mensajes, etc. la hace la comunidad. Durante las evoluciones del sistema, es posible que algunas partes aparezcan todavía en inglés. Esto se corregirá en futuras versiones.

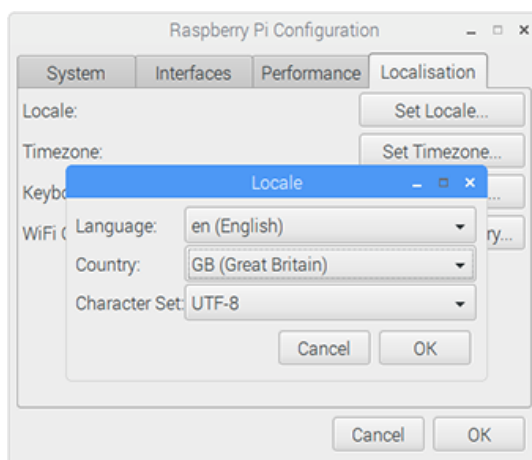
Pulse en la pestaña **Localisation**, que le va a dar acceso a las opciones disponibles para adaptar el sistema al idioma español.



La pestaña **Localisation** tiene cuatro ajustes:

- **Locale:** define el idioma a utilizar por el sistema
- **Timezone:** indica el huso horario del usuario
- **Keyboard:** especifica el tipo de teclado a utilizar (QWERTY, AZERTY...)
- **WiFi Country:** permite ajustar el Wi-Fi con la legislación del país.

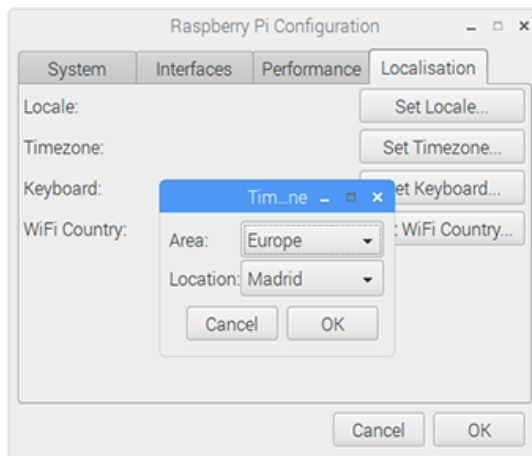
b. Configuración de las opciones Locales



Pulse en el botón **Set Locale**. Se abre la ventana **Locale**. Pulse en el botón de selección del idioma (**Language**) que indica por defecto **en (English)**. En la lista que aparece pulse en **es (Spanish)**. Seleccione ahora el país (**Country**) en la lista: pulse en **ES (Spain)**. Para terminar, para **Character Set** (juego de caracteres) deje la opción por defecto (**UTF-8**), que es la más universal. Valide sus selecciones pulsando en el botón **OK** de la ventana **Locale**.

c. Configuración del huso horario

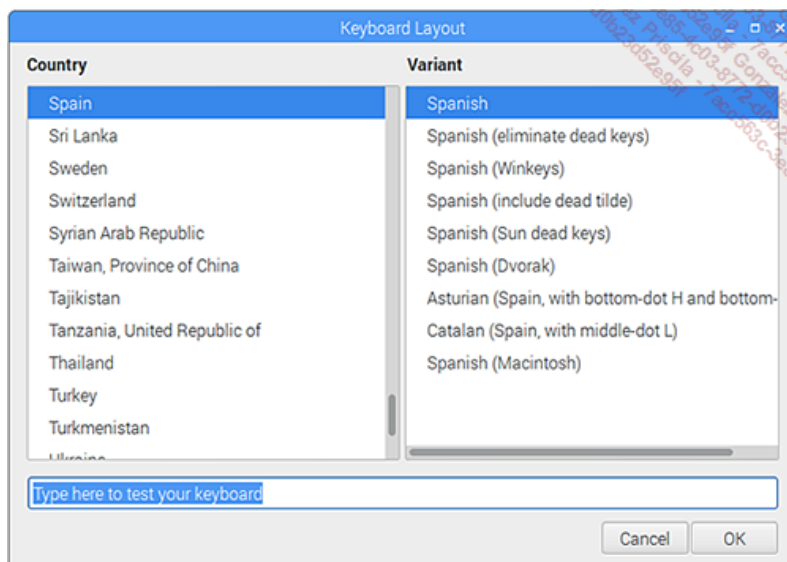
En una Raspberry Pi 3 conectada a Internet, Raspbian recupera la hora de un servidor de tiempo. Con una Raspberry Pi Zero, esta puesta en hora necesitará la implantación de un adaptador de Internet (alámbrico o Wi-Fi). Esta hora, llamada universal, es la hora UTC (Tiempo Universal Coordinado). El ajuste del huso horario informa al sistema del decalaje que existe entre la hora universal y el huso horario en el que se encuentre, teniendo en cuenta la hora de verano/invierno. Para España, se utiliza el huso horario de Madrid. A partir de esta información, Raspbian puede determinar la hora actual del lugar donde habita.



Pulse en el botón **Set Timezone**. En la lista desplegable **Area**, pulse en **Etc** y seleccione **Europe** en la lista que se muestra. Seleccione **Madrid** en la lista **Location**. Valide su selección pulsando el botón **OK** de la ventana **Timezone**.

d. Configuración del teclado

Pulse en el botón **Set Keyboard**.



En la ventana **Keyboard Layout** seleccione en la parte izquierda (**Country**) el país: **Spain**. En la parte derecha (**Variante**) seleccione **Spanish**. Valide pulsando en el botón **OK** en la parte inferior de la ventana.

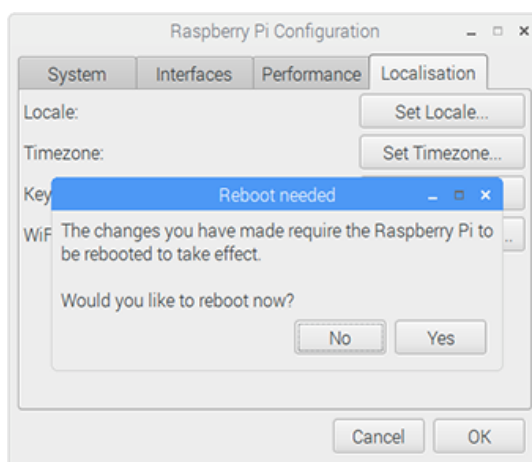
e. Configuración del país para el Wi-Fi

Por último, seleccione el país en el que se utilizará el Wi-Fi. Esto es necesario para la conformidad con las leyes o reglamentos del país en lo que respecta a los canales utilizados y la potencia de emisión.

Pulse en el botón **WiFi Country**. En la ventana **WiFi Country Code** que se abre, despliegue la lista y pulse en **ES Spain**. Una vez realizada haga la selección, válidela pulsando en el botón **OK** de la ventana **WiFi Country Code**.

f. Reinicio de la Raspberry Pi

Se han rellenado los cuatro argumentos de la pestaña **Localisation**. Pulse en el botón **OK** situado en la parte inferior derecha de esta pestaña, para validar su selección.



El sistema abre una ventana **Reboot needed** (Reinicio necesario) para indicar que las modificaciones se tendrán en cuenta después de un reinicio.

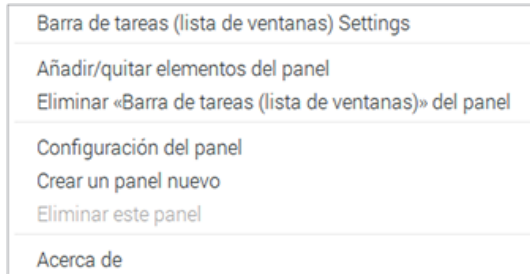
Pulse en el botón **Yes** para autorizar el reinicio de la Raspberry Pi.

Cuando la Raspberry Pi ha arrancado, los mensajes y la mayor parte de los elementos de los menús están en español. Ahora es posible explorar la interfaz gráfica de Raspbian.

4. Configuración de la barra de tareas

La interfaz gráfica de LXDE es configurable en función de las necesidades del usuario. La barra de tareas es configurable a partir de un menú

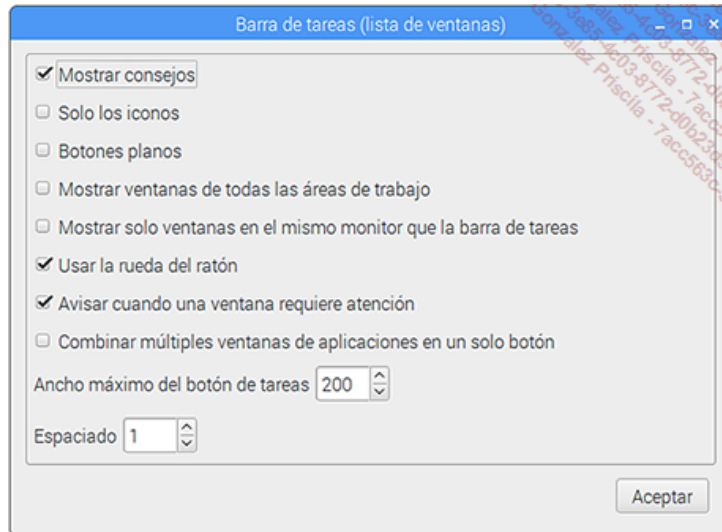
accesible con un clic derecho en la barra de tareas.



A continuación se muestra la función de cada uno de los elementos de este menú.

a. Barra de tareas (lista de ventanas) Settings

Los ajustes accesibles en esta ventana permiten personalizar el comportamiento de la barra de tareas y de las indicaciones que ofrece al usuario.



Cuando las opciones están marcadas, las acciones que figuran a derecha de cada opción se ejecutan. Haga un clic izquierdo con el ratón para marcar/desmarcar una opción.

Mostrar consejos: cuando está marcada, esta opción autoriza la visualización de un tooltip de información sobre un elemento de la barra de tareas. El cursor debe permanecer sobre el elemento más de un segundo para que el tooltip aparezca.

Solo los iconos: las ventanas abiertas aparecen en la barra de tareas en forma de un icono a la derecha, en la que figura el nombre de la ventana. Esta opción permite restringir la visualización a un solo icono.

Botones planos: la visualización de las ventanas abiertas se hace sin que el icono ni el nombre de la ventana aparezcan en un botón.

Mostrar ventanas de todas las áreas de trabajo: cuando el usuario ha creado varios espacios de trabajo, todas las ventanas abiertas en estos diferentes espacios son las que se muestran.

Mostrar solo ventanas en el mismo monitor que la barra de tareas: cuando esta opción está marcada, solo se muestran las ventanas abiertas en la misma pantalla que la barra de tareas.

Usar la rueda del ratón: la rueda del ratón activa sucesivamente los diferentes escritorios virtuales.

Avisar cuando una ventana requiere atención: cuando una ventana espera la intervención del usuario (clic, validación, etc.), el botón situado en la barra de tareas parpadea.

Combinar múltiples ventanas de aplicaciones en un solo botón: si hay abiertas varias instancias de una aplicación, se muestra un único botón para el conjunto de las ventanas, si esta opción está marcada. El número de ventanas abiertas se indica a la derecha del icono, entre paréntesis.

Los dos campos situados en la parte inferior de la ventana dan acceso al ajuste de la longitud máxima ocupada por cada botón y el espacio entre los botones. Estos ajustes permiten adaptar la presentación de la barra de tareas a sus gustos y a la definición de su pantalla.

b. Agregar/Eliminar elementos al panel

Un clic en este elemento del menú abre la ventana **Preferencias del panel**. Se selecciona la pestaña **Miniaplicaciones del panel**. Contiene la lista de todo lo que se muestra en la línea superior del escritorio de Raspbian. Esta pestaña da acceso a estos "añadidos", que puede eliminar o agregar a partir de la lista. Los botones **Subir** y **Bajar** le permiten organizar su panel en función de sus gustos o de sus necesidades.

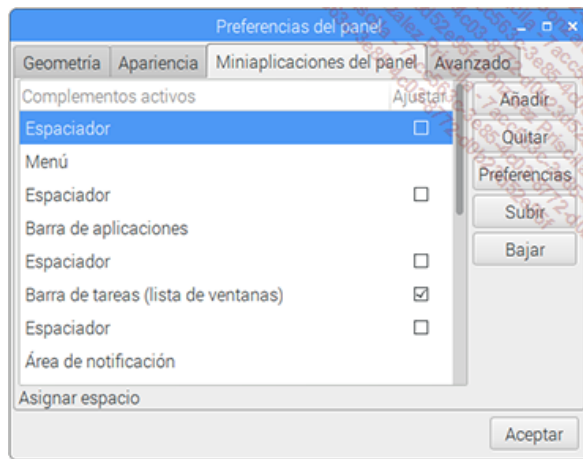
- Cada applet añadido al panel consume recursos de la Raspberry Pi (memoria, tiempo CPU, etc...). Reduzca al mínimo necesario el número de applets presentes en el panel.

Como ejemplo, veamos paso a paso cómo agregar al panel un indicador de la temperatura del SoC. Este indicador se deberá posicionar a la izquierda del icono Red/Wi-Fi y se separará con una línea vertical:

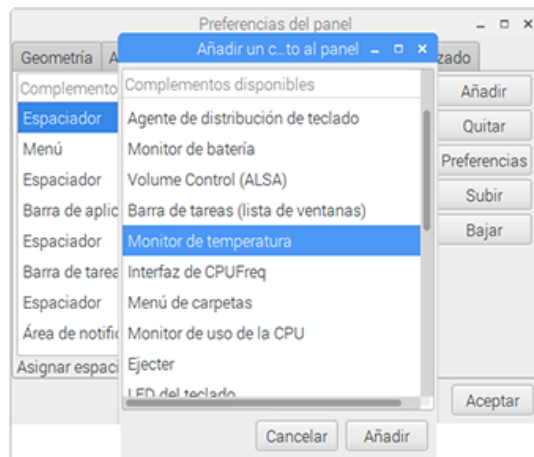


Empiece con un clic derecho en la barra de tareas.

En el menú que se abre, pulse en **Añadir/Eliminar elementos al panel**.

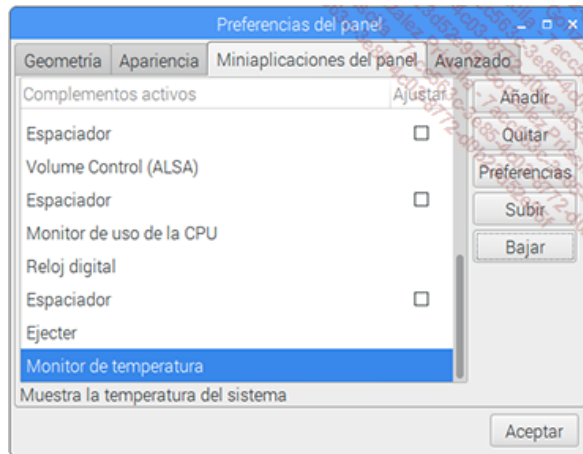


Aparecen los applets existentes en el panel. Pulse en el botón **Añadir**:

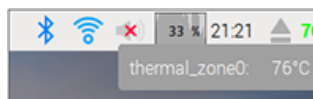


Se abre una ventana que contiene todos los "complementos" que se pueden agregar al panel.

Seleccione la línea **Monitor de temperatura** y pulse en el botón **Añadir** en la parte inferior de la ventana **Añadir un complemento al panel**.

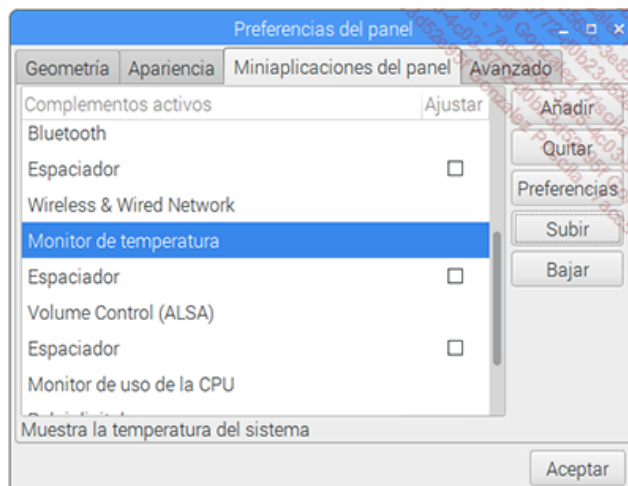


El monitor de temperatura se ha añadido en la parte inferior de la lista de complementos. Por tanto, se sitúa en el extremo derecho del panel:



Pasando el ratón por este monitor, se abre el cuadro de diálogo correspondiente.

Ahora hay que poner el monitor de temperatura en su lugar definitivo.



Pulse varias veces en el botón **Subir** de la ventana **Preferencias del panel**. Observe el movimiento del monitor de temperatura

después de cada clic:

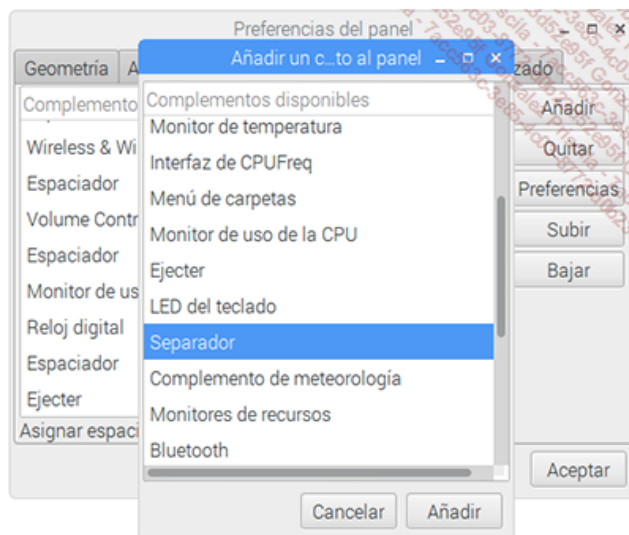


Lleve el monitor a la izquierda del icono Red/Wi-Fi, pulsando el botón **Subir**.

Ahora debería obtener el siguiente panel:



Cuando el monitor de temperatura esté en su posición definitiva, pulse de nuevo en el botón **Añadir** y seleccione en la lista de los "complementos" el **Separador** (trazo vertical).



Agréguelo al panel y muévelo a la derecha del monitor de temperatura.

- Algunos complementos figuran en la lista pero no funcionan. Por ejemplo, es el caso del Monitor de batería, del Agente de distribución de teclado o del Complemento de meteorología.

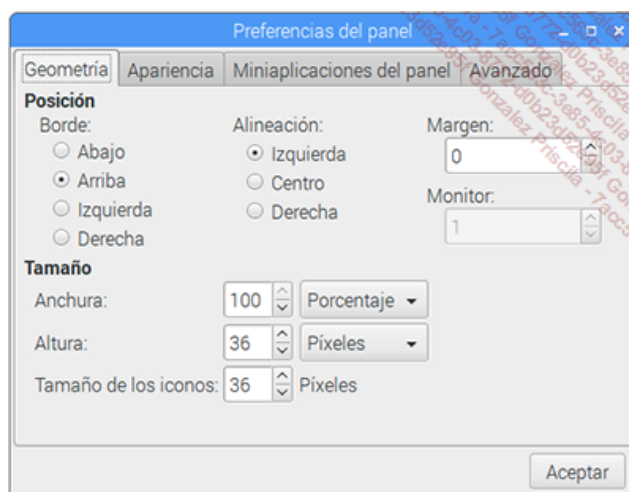
Como ejercicio, agregue un administrador de escritorio virtual a la izquierda del monitor de temperatura. Debería obtener el panel siguiente. La miniatura del escritorio actual muestra la o las ventanas abiertas en el escritorio.



c. Argumentos del panel

Este elemento del menú abre la ventana **Preferencias del panel**, en la pestaña **Geometría**.

- Observe atentamente los valores de cada uno de los ajustes antes de modificarlos, porque después le podría llevar algún tiempo volver a la presentación estándar.

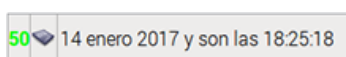


Puede mover el panel por los cuatro lados de la pantalla, cambiar la alineación, el margen e incluso las dimensiones del panel (anchura, altura, dimensiones de los iconos).

d. Crear un nuevo panel

Este elemento del menú abre una ventana **Preferencias del panel** virgen. No contiene ningún complemento y el panel está vacío. El panel está situado a la izquierda de la pantalla.

Como ejercicio, cree un panel situado en la parte inferior de la pantalla y que se presente de la siguiente manera:



De izquierda a derecha tiene un monitor de temperatura, un monitor de frecuencia CPU y un reloj digital personalizado.

La solución de este ejercicio aparece en el párrafo Creación de un nuevo panel del capítulo Anexos.

e. Eliminar un panel

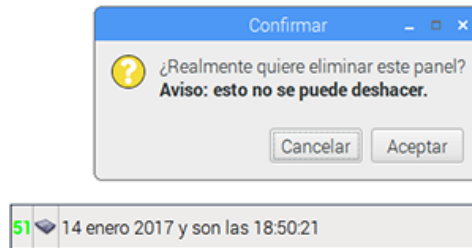
Vamos a comprobar la eliminación del panel que acabamos de crear en la parte inferior de la página.

- Si elimina el panel original (en la parte superior de la página) no podrá volver atrás. Esto será definitivo y debe volver a crearlo desde el principio. Es mejor probar la eliminación en un panel de prueba.

Pulse con el botón derecho del ratón en el panel que ha creado en la parte inferior de la página.



Pulse en **Eliminar este panel**. Una ventana le avisa de que la operación es irreversible.



Confirme la eliminación del panel pulsando el botón **Aceptar** en la parte inferior de esta ventana.

f. Acerca de

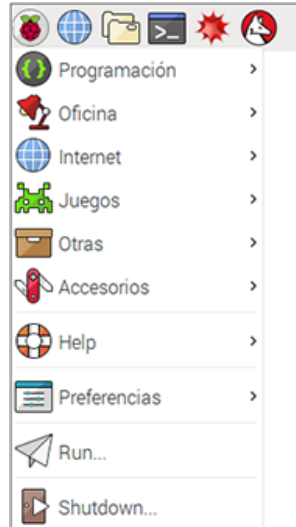
La ventana **Acerca de**, informa al usuario de la versión del software LXPanel.



El botón **Créditos** muestra los nombres y la dirección de correo electrónico de los participantes en el proyecto, así como el nombre del traductor. El botón **Licencia** muestra un texto que indica que el software está protegido por la GNU General Public Licence.

Los menús de Raspbian Jessie

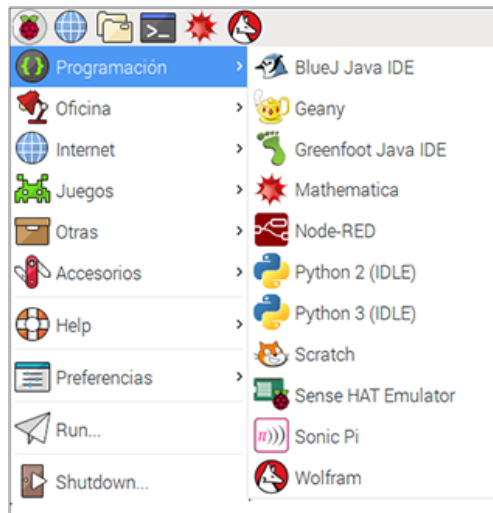
En esta versión aparece una modificación. Todos los iconos han desaparecido del escritorio. Solo queda la papelera conservando su lugar en el escritorio. El resto de aplicaciones se han agrupado en el menú principal, accesible pulsando el botón **Menú** situado en la parte superior izquierda de la pantalla, con la instalación por defecto.



Las aplicaciones se agrupan por categorías. Cada ítem del menú da acceso a una serie de programas.

1. Programación

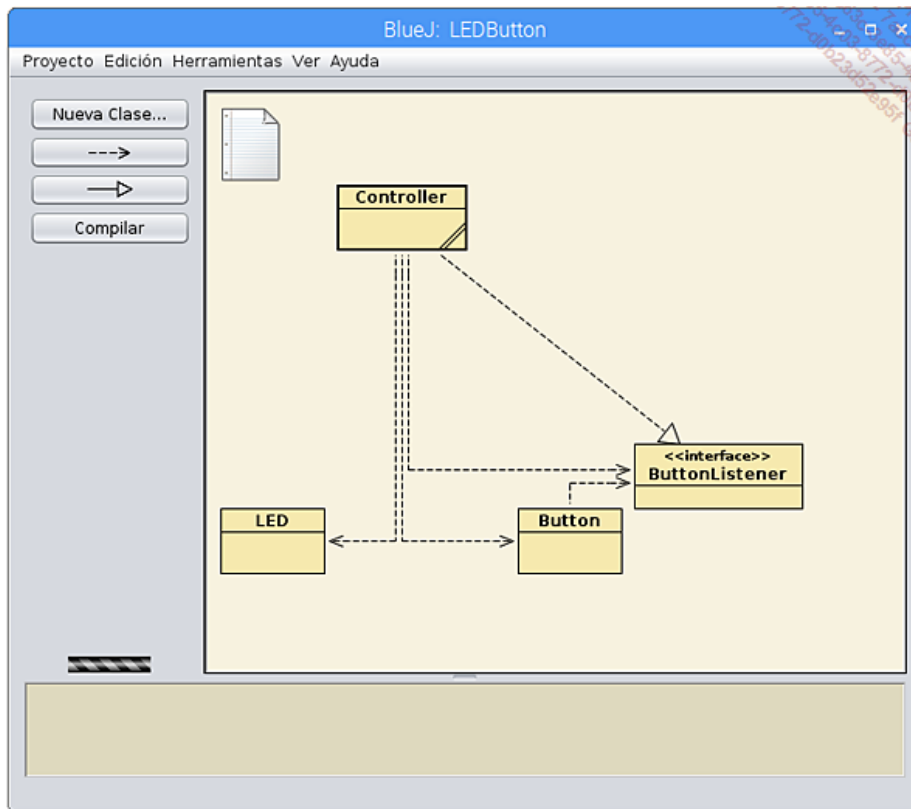
En esta sección figuran las herramientas de desarrollo que la Fundación Raspberry Pi piensa que son útiles para los usuarios de la Raspberry Pi. Algunas se pueden usar por principiantes, mientras que otras obligan a tener más experiencia.



En esta versión de Raspbian existen al menos nueve herramientas de desarrollo a disposición del usuario. Por supuesto, es posible añadir cualquier otro entorno de desarrollo si lo desea.

a. BlueJ Java IDE

BlueJ es un IDE (*Integrated Development Environment* = entorno de desarrollo integrado) destinado a facilitar la programación en Java, orientada a objetos. Se creó por la Universidad de Kent.



En este IDE, las clases se representan mediante rectángulos y es posible organizarlos como se quiera en el plan de trabajo. En la Raspberry Pi, BlueJ puede acceder a los puertos de la GPIO e interactuar con el mundo real. Acceda a la dirección <http://www.bluej.org/raspberrypi/> para encontrar ejemplos de estas funcionalidades (lectura de un botón o encendido de un LED).

b. Geany

Geany es un IDE (*Integrated Development Environment* = entorno de desarrollo integrado) ligero. Se ha desarrollado para ofrecer un IDE sencillo y rápido, que tuviera poca dependencia de otros paquetes. Geany solo necesita las bibliotecas GTK2.

Sus principales características son:

- la coloración sintáctica,
- la indentación del código,
- el auto completado,
- el cierre automático de las etiquetas XML y HTML,
- el soporte de numerosos tipos de archivos: C, Java, PHP, HTML, Python, Perl, Pascal...
- la gestión sencilla de proyectos,
- la disponibilidad de una interfaz para agregar plugins (ver Plugins).

```

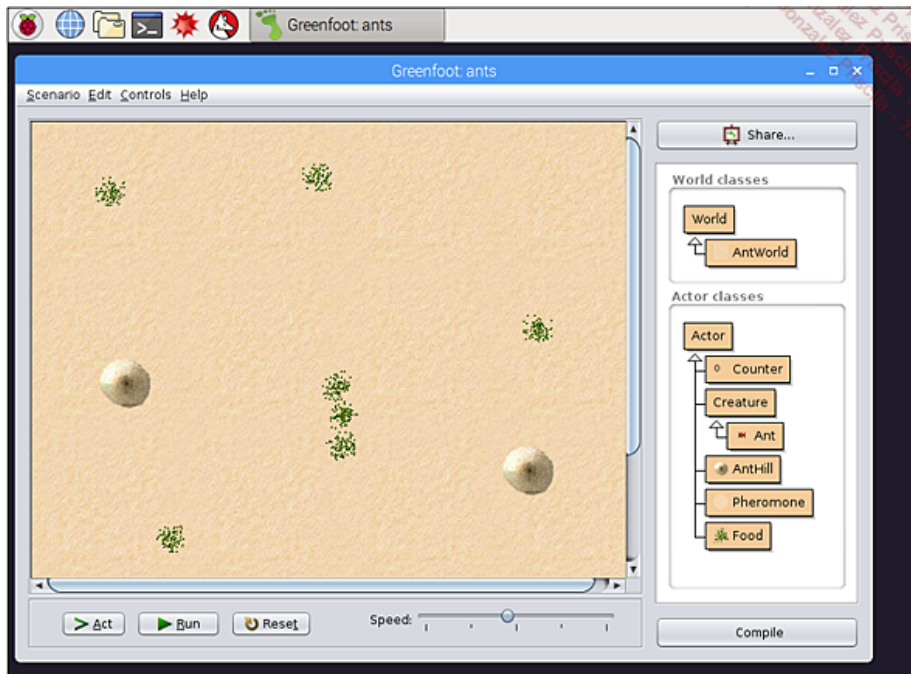
1  import com.pi4j.io.gpio.*;
2  import com.pi4j.io.gpio.event.*;
3  import java.util.*;
4
5  /**
6   * This class represents a Button connected to the Raspberry pi on a given pin.
7   * Connect the button to 3.3v power and to a GPIO.
8   *
9   * @author Fabio Hedayioglu
10  * @author Ian Utting
11  * @author Michael Kolling
12  * @version 1.0
13  */
14
15  public class Button
16  {
17      // the button instance is an input pin.
18      private GpioPinDigitalInput button;
19
20      //this array maps the GPIO pin numbers to integers
21      private static final Pin[] pinMap = new Pin[] {RaspiPin.GPIO_00, RaspiPin.G
  
```

Geany es un IDE multi-pestañas que pone a disposición de los desarrolladores que usan la Raspberry Pi un entorno de desarrollo muy práctico. La coloración sintáctica y la indentación facilitan la organización de los programas y la búsqueda de errores. La parte inferior del IDE contiene pestañas donde están los mensajes de estado de Geany (Statut), los mensajes de salida del compilador o incluso un terminal en el que el desarrollador directamente puede interactuar con el sistema operativo.

La información y el manual de uso están disponibles en <http://www.geany.org>.

c. Greenfoot Java IDE

También creado por la Universidad de Kent, Greenfoot se usa para enseñar la programación orientada a objetos con Java. Usa "actores" que evolucionan en "mundos", podrá crear juegos, simulaciones y otros programas gráficos.

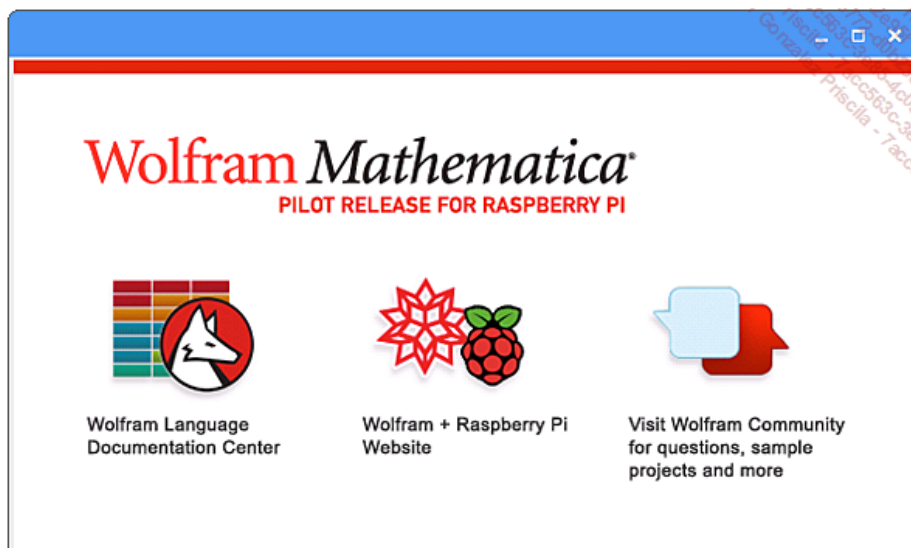


Las herramientas permiten gestionar las interacciones y visualizar el resultado de los programas, integrado en el IDE. Los programas Java se escriben en forma de texto. Greenfoot combina la redacción de código clásica con un entorno de ejecución visual.

d. Mathematica

El software Mathematica se usa en los entornos científicos desde hace más de 25 años. Se integra en Raspberry Pi y se pone a disposición de los usuarios gratuitamente, sin embargo el software permanece cerrado y restringido a una licencia restrictiva (<http://www.wolfram.com/legal/agreements/wolfram-mathematica-raspberry-pi.html>).

Al hacer doble clic en el icono de Mathematica se abre la ventana de la aplicación. La llegada de la Raspberry Pi ha tenido un efecto beneficioso para la velocidad de ejecución de Mathematica, ya que el tiempo necesario para abrir el programa ha pasado de varios minutos a menos de 10 segundos.

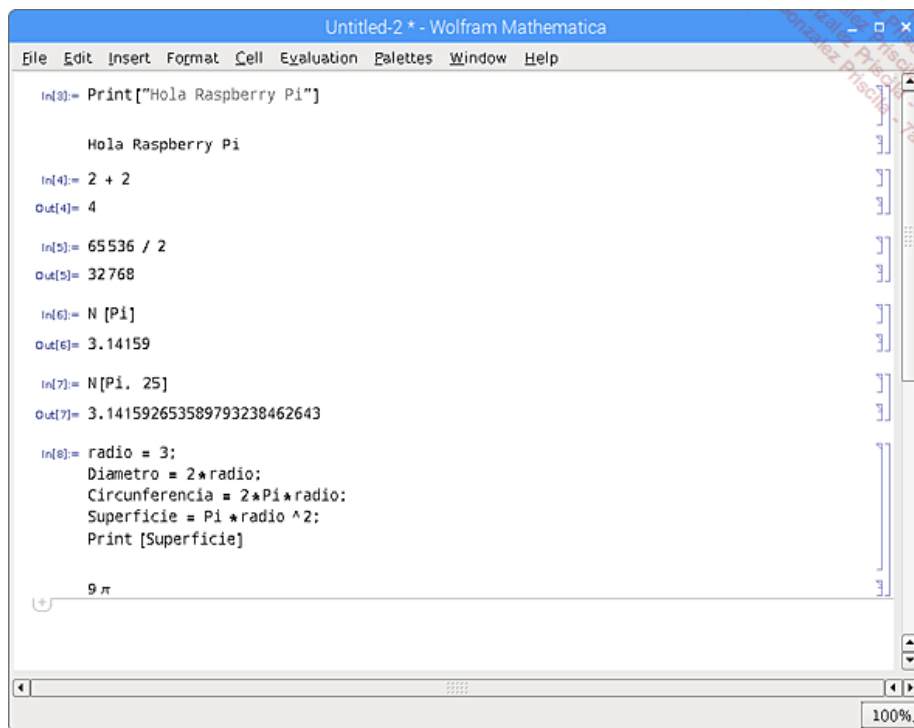


Se abre la ventana **Wolfram Mathematica** y una pantalla de bienvenida le permite acceder a:

- **Wolfram Language Documentation Center:** la documentación en línea del lenguaje Wolfram, en inglés.
- **Wolfram + Raspberry Pi Website:** ofrece ejemplos de operaciones que combinan una Raspberry Pi y el lenguaje Wolfram. Encontrará un registro GPS, una estación meteorológica, el uso de la GPIO e incluso la captura de imágenes con la cámara de la Raspberry Pi.
- **Visit Wolfram Community:** es el foro de la comunidad Wolfram. Está en inglés. Si tiene problemas con su versión de Mathematica, este es el lugar donde encontrará una solución.

Si no desea acceder a estos recursos, cierre la pantalla de bienvenida pulsando en el aspa situada en la parte superior derecha.

Acceda a la ventana de **Wolfram Mathematica**. A través de un terminal va a poder introducir comandos en lenguaje Wolfram.



Empiece escribiendo:

```
Print ["Hola Raspberry Pi"]
```

Después valide pulsando simultáneamente en las teclas [Shift][Intro].

Mathematica añade **In[1]:=** delante de su entrada y pasados algunos segundos muestra el resultado de la operación, escribiendo el texto *Hola Raspberry Pi* en la pantalla.

Pruebe a introducir otras operaciones. Con Mathematica dispone de una calculadora perfeccionada.

Si rellena:

```
N [Pi]
```

Mathematica devuelve 3.14159 (6 cifras, por defecto).

¿Necesita más decimales? Intente con:

```
N [Pi, 25]
```

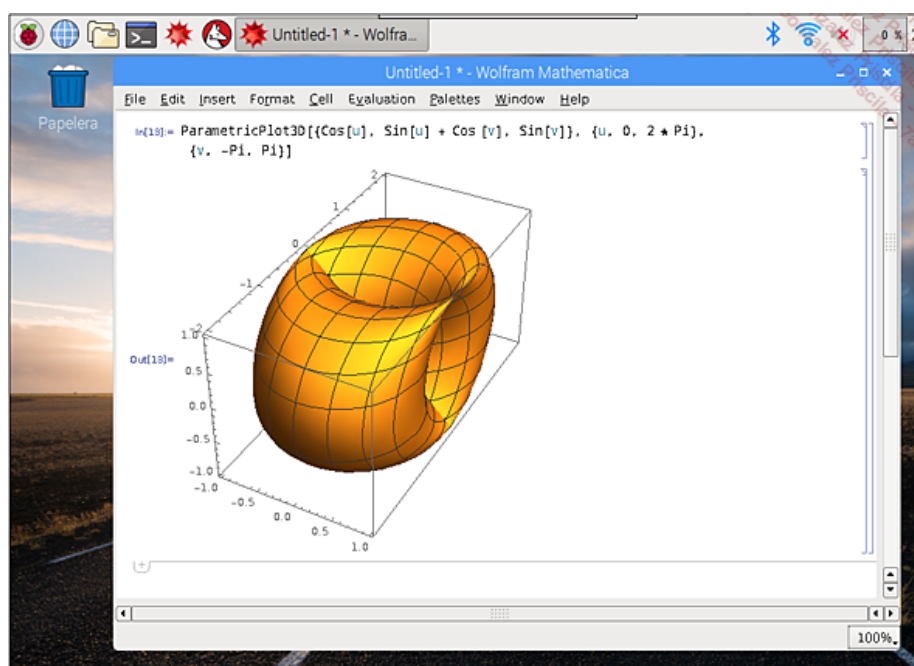
También es posible guardar el resultado de cálculos en variables:

```
radio = 3;
diametro = 2 * radio;
circunferencia = 2 * Pi * radio;
superficie = Pi * radio^2;
```

El punto y coma al final de la línea evita la visualización de los resultados en la pantalla. Por tanto, es posible usar variables en los cálculos:

```
Print [superficie]
9 pi
```

Mathematica también puede representar funciones en un plano o en tres dimensiones, como se muestra en el siguiente ejemplo.



Puede consultar la documentación del lenguaje Wolfram en línea para la Raspberry pi en la dirección <http://reference.wolfram.com/language/guide/RaspberryPi.html>.

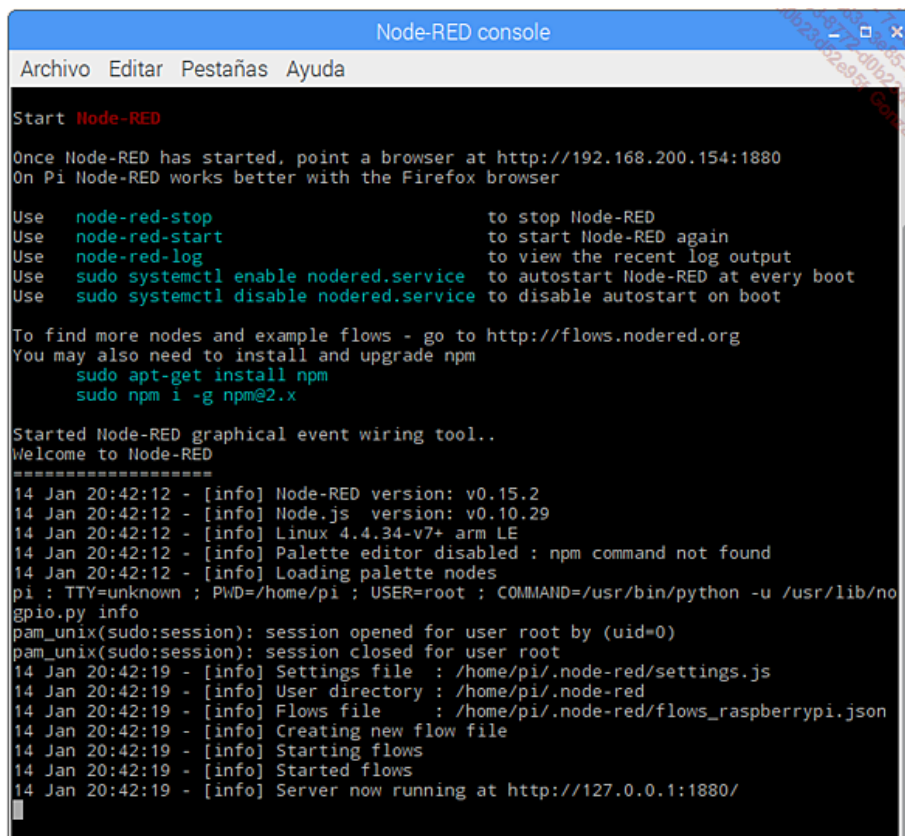
e. Node-RED

Node-RED es un proyecto open source de IBM Emerging Technology. Es una herramienta visual construida en Node js, un motor muy ligero y bien adaptado a la Raspberry Pi. Node js usa el lenguaje Javascript. Normalmente ejecutado en el navegador del puesto cliente, Javascript se utiliza aquí en el lado servidor.

Node-RED administra los flujos. Permite unir periféricos, aplicaciones a través de una API (*Application Programming Interface* = Interfaz de programación de aplicaciones) y de servicios en línea.

Es una herramienta extraordinariamente rápida para desarrollar aplicaciones domóticas. Node-RED permite iniciarse en el mundo del IoT (*Internet of Things* = Internet de los objetos).

Cuando arranca Node-RED, solo será una ventana de terminal.



```
Node-RED console
Archivo Editar Pestañas Ayuda

Start Node-RED

Once Node-RED has started, point a browser at http://192.168.200.154:1880
On Pi Node-RED works better with the Firefox browser

Use node-red-stop to stop Node-RED
Use node-red-start to start Node-RED again
Use node-red-log to view the recent log output
Use sudo systemctl enable nodered.service to autostart Node-RED at every boot
Use sudo systemctl disable nodered.service to disable autostart on boot

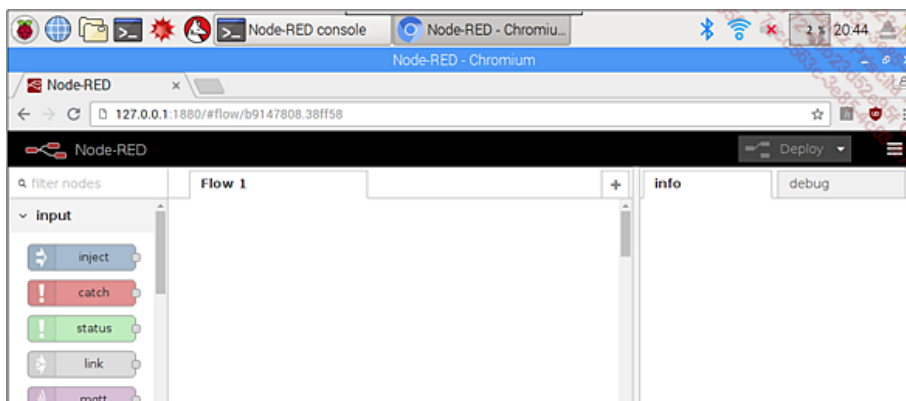
To find more nodes and example flows - go to http://flows.nodered.org
You may also need to install and upgrade npm
sudo apt-get install npm
sudo npm i -g npm@2.x

Started Node-RED graphical event wiring tool..
Welcome to Node-RED
=====
14 Jan 20:42:12 - [info] Node-RED version: v0.15.2
14 Jan 20:42:12 - [info] Node.js version: v0.10.29
14 Jan 20:42:12 - [info] Linux 4.4.34-v7+ arm LE
14 Jan 20:42:12 - [info] Palette editor disabled : npm command not found
14 Jan 20:42:12 - [info] Loading palette nodes
pi : TTY=unknown ; PWD=/home/pi ; USER=root ; COMMAND=/usr/bin/python -u /usr/lib/no
gpio.py info
pam_unix(sudo:session): session opened for user root by (uid=0)
pam_unix(sudo:session): session closed for user root
14 Jan 20:42:19 - [info] Settings file : /home/pi/.node-red/settings.js
14 Jan 20:42:19 - [info] User directory : /home/pi/.node-red
14 Jan 20:42:19 - [info] Flows file : /home/pi/.node-red/flows_raspberrypi.json
14 Jan 20:42:19 - [info] Creating new flow file
14 Jan 20:42:19 - [info] Starting flows
14 Jan 20:42:19 - [info] Started flows
14 Jan 20:42:19 - [info] Server now running at http://127.0.0.1:1880/
```

Las primeras líneas de la pantalla le invitan a instalar *npm*. Es el administrador de paquetes de Node js que ayuda a los desarrolladores a compartir fácilmente sus creaciones.

La última línea aporta una información esencial: el servidor es accesible en la dirección local 127.0.0.1 en el puerto 1880.

Abra el navegador web de su Raspberry Pi 3 e introduzca la dirección local: <http://127.0.0.1:1880>. Accederá al entorno de desarrollo Node-RED. También es posible conectarse a través de la red. En la Raspberry Pi, Node Red está accesible en la dirección 127.0.0.1:1880. En una máquina conectada a la red (PC, Mac, smartphone, tableta...) abra un navegador (PC, Mac, smartphone, tableta...) e introduzca la dirección IP de la Raspberry Pi 3 (por ejemplo: <http://192.168.1.230:1880>). Accederá también a la interfaz de desarrollo Node-RED).



La ventana de Node-RED muestra, de izquierda a derecha:

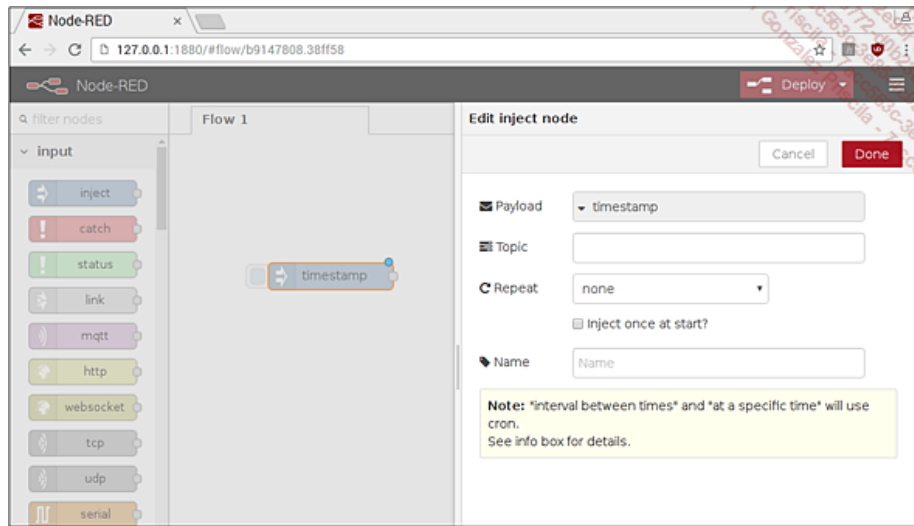
- los nodos de inserción alineados verticalmente en la paleta,
- la ventana de trabajo, en la que se conectarán los nodos,
- una pestaña **Info** que ofrece ayuda del nodo seleccionado,
- una pestaña **Debug** que muestra los flujos,
- un botón **Deploy** situado encima de la pestaña **Debug**, que se utiliza después de la modificación de los nodos o uniones en la ventana de trabajo. Cuando está en rojo, indica que hay una modificación que no se ha tenido en cuenta. Pulse este botón para que el sistema tenga en cuenta sus modificaciones.

Un primer proyecto con Node-RED

Para probar el uso y el funcionamiento de Node-RED vamos a realizar una aplicación sencilla con el objetivo de mostrar la hora a intervalos regulares en la pestaña **Debug**.

Pulse en el primer nodo de la lista llamada **Inject**. Muévelo hasta la ventana de trabajo.

Por defecto, cuando lo deja en la ventana central, indica **Timestamp**. Esta es la función que nos interesa, el nodo va a enviar la fecha y hora en el flujo de salida.

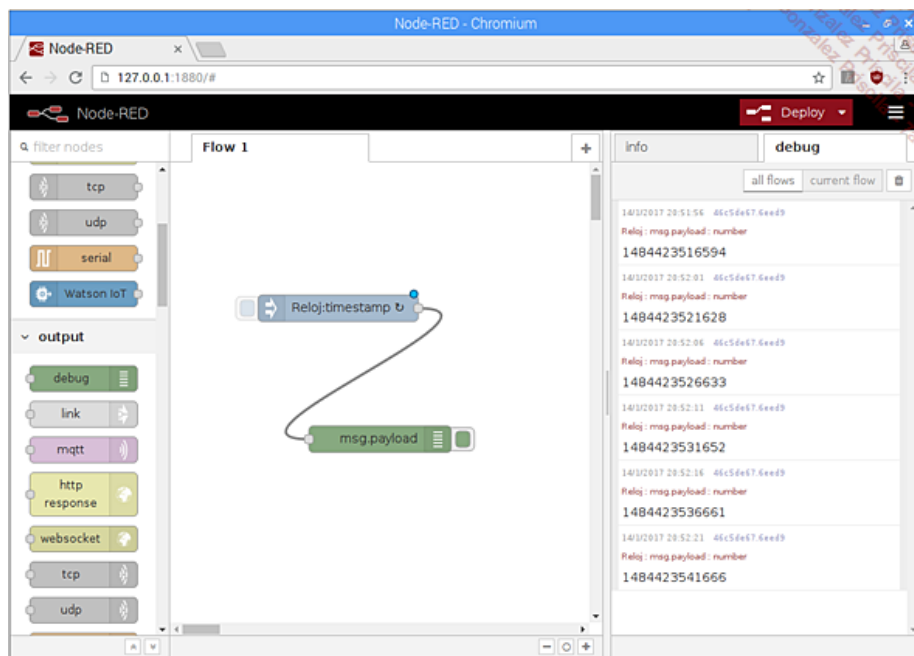


Pulse dos veces en el nodo. Se abre una ventana de configuración. Puede seleccionar el contenido del flujo (Payload). Por ejemplo, podría ser una cadena de caracteres. **Topic** permite introducir una cadena de caracteres que se incluirá en el mensaje. En la lista **Repeat** seleccione **Intervalle** y fije el intervalo de repetición con el valor que desee (aquí 5 segundos). Para terminar, asigne un nombre al nodo (es la cadena que se mostrará). Valide pulsando en **OK**.

Añada el nodo **Debug**, el primer nodo de la sección **output**.

Pulse en el cuadrado claro a la derecha del nodo **reloj** (Timestamp). Manteniendo el botón del ratón pulsado, muévase hasta el cuadrado claro situado a la izquierda del nodo **Debug**. Suelte el botón del ratón.

Los dos nodos se deben relacionar de la siguiente manera.



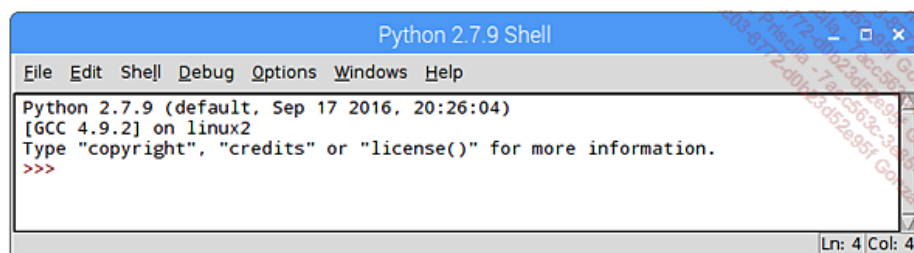
También puede renombrar el nodo de salida si lo desea. Pulse el botón **Deploy** para informar al sistema de las modificaciones que se han producido. Las tiene en cuenta y la pestaña **Debug** muestra la fecha y la hora siguiendo el intervalo de tiempo seleccionado. La papelera visible en el ángulo de la pestaña **Debug** vacía su contenido.

- Cuando modifique el contenido de la ventana de trabajo, no se sorprenda al ver la pestaña **Debug** continuar rellenándose. La versión anterior de su flujo permanece activo. Se modificará cuando pulse de nuevo el botón **Deploy**.

f. Python 2 (IDLE)

IDLE (*Integrated Development Environment* = entorno de desarrollo integrado) es el IDE de Python 2. Agrupa las herramientas necesarias para la escritura de programas en Python y su puesta a punto. El capítulo Programar en Python volverá en detalle sobre este IDE.

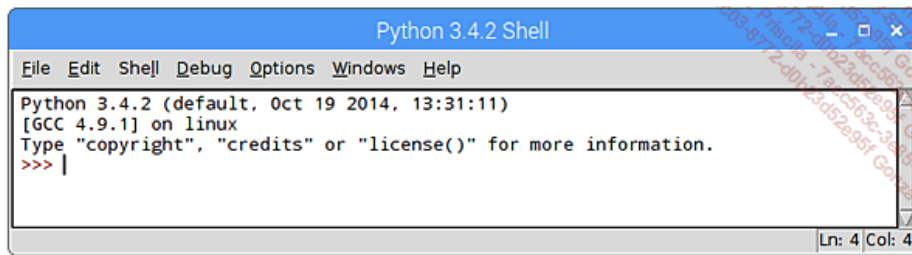
Cuando se abre IDLE, la versión de Python disponible en este entorno es la 2.7.9.



g. Python 3 (IDLE)

IDLE 3 es el mismo entorno de desarrollo que IDLE. Da acceso a Python 3.4.2. La versión 3 de Python es la versión más reciente y, por tanto, debería ser la que se use. Sin embargo, los muchos desarrollos y librerías realizados en Python 2 obligan a mantener las dos versiones. La versión 2 todavía estará en servicio algunos años. Para un debutante en Python, lo más sencillo es iniciar con la versión 3.

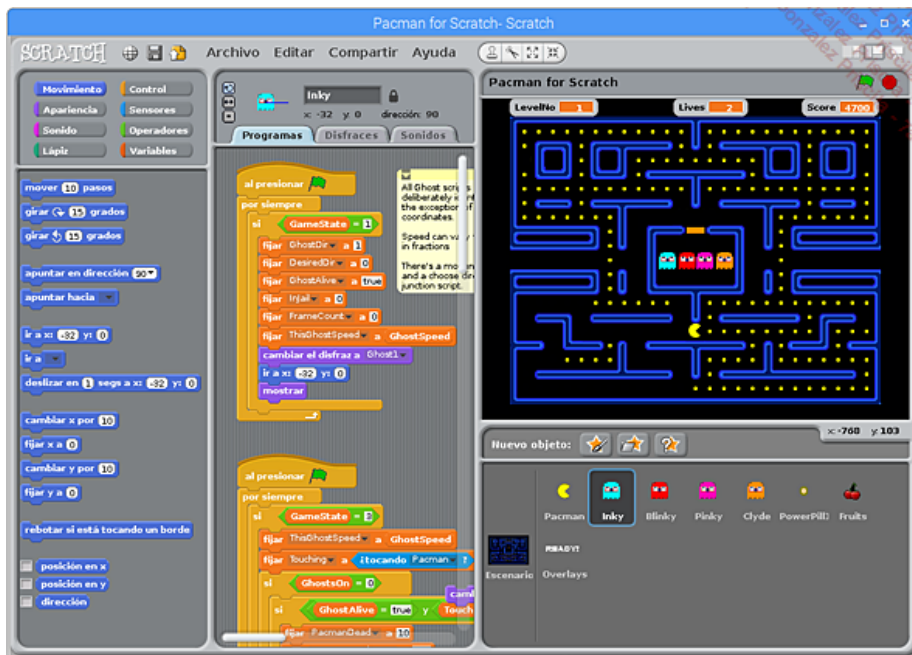
Para los usuarios profesionales, la práctica de las dos versiones es obligatoria porque es necesario mantener las aplicaciones escritas en versión 2 y migrar las que sea necesario a la versión 3.



h. Scratch

Scratch es un lenguaje de programación desarrollado por el MIT (*Massachusetts Institute of Technology* = Instituto de tecnología de Massachusetts). Su primera versión es de 2003. Permite realizar programas de manera visual y muy lúdica, que ensambla bloques de código. Es posible configurar el idioma entre una veintena de idiomas disponibles. Hay accesibles más de tres millones de proyectos compartidos. Es posible descargarlos, modificarlos, etc...

El enfoque gráfico de Scratch permite a los niños desde los ocho años, realizar programas. Pero no hay límite de edad y Scratch permite a todo el mundo iniciarse en la programación informática.



Scratch está disponible en línea (captura de pantalla anterior), y también en varias plataformas: Mac, Windows y algunas versiones de Linux (32 bits). En una Raspberry Pi que usa la distribución Raspbian está disponible la versión 1.4 de Scratch.

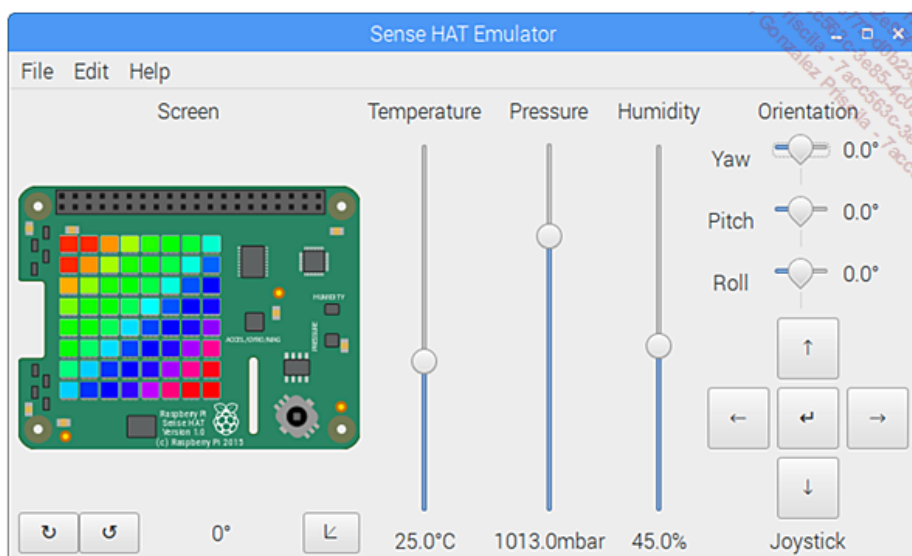
La ventaja de Scratch reside en la implementación de bloques estructurados que son una aproximación intuitiva de la algoritmia. El capítulo Programar en Scratch, aporta más detalles sobre este lenguaje de programación.

i. Sense HAT

Con la aparición de la tarjeta Sense Hat (antiguamente Astro Pi), la Fundación ha dado un paso hacia las estrellas, porque hay dos ejemplares de esta tarjeta montados en la actualidad en las Raspberry Pi utilizadas en la ISS (estación Espacial Internacional). La parte más visible de esta tarjeta es una matriz de 64 LED RVB que permite ver los mensajes, símbolos coloreados o niveles de temperatura, presión, etc... La Sense HAT también tiene sensores: giróscopo, acelerómetro, magnómetro, sensor de temperatura de presión y de humedad.

El astronauta Tim PEAKE fue el encargado de implementar en el espacio los programas escolares ingleses y enviar los resultados a la Tierra. Estas experiencias se desarrollaron desde diciembre de 2015 hasta junio de 2016.

Un nuevo concurso lanzado por la ESA (Agencia Espacial Europea) permitirá esta vez a los estudiantes europeos, proponer proyectos que el astronauta Francés Thomas PESQUET instalará en las Raspberry Pi del ISS a partir de mediados de noviembre de 2016.

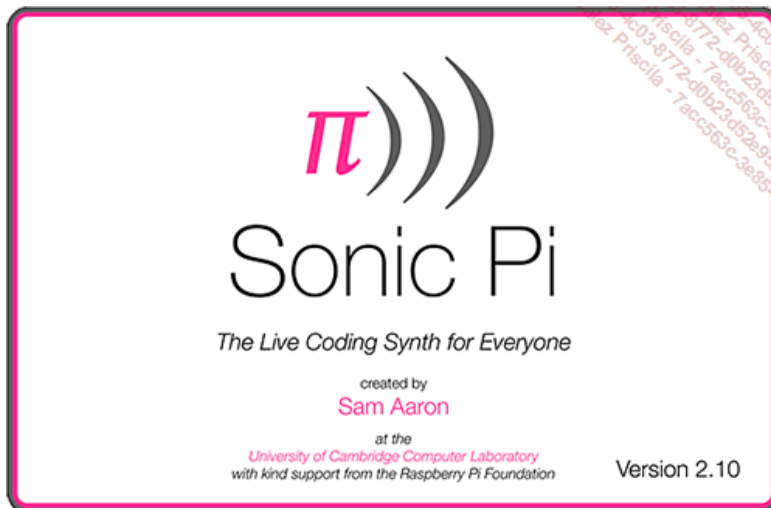


Este simulador de Sense HAT permite preparar programas sin disponer de Sense HAT. Los cursores y botones simulan los sensores y el visualizador funciona exactamente de la misma manera que el visualizador real.

j. Sonic Pi

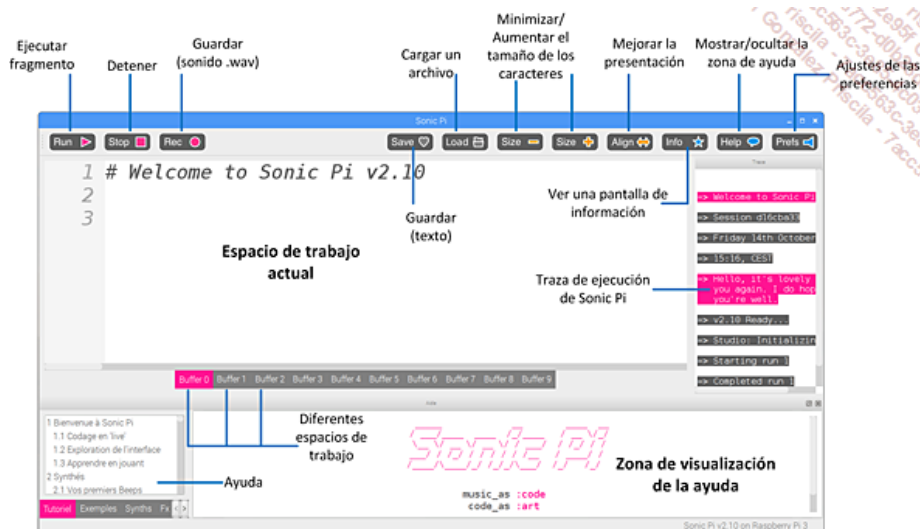
Sonic Pi es un entorno de programación open source diseñado para explorar y aprender los conceptos de programación a través del proceso de creación de nuevos sonidos. Sonic Pi se entrega con un entorno de trabajo que pone el acento en la importancia de la creatividad en el

proceso de aprendizaje, y da a los usuarios la posibilidad de transformar sus ideas en sonidos (<http://www.cl.cam.ac.uk/projects/raspberrypi/sonicpi/>).



Cuando arranca la aplicación, una pantalla de bienvenida informa de la versión de Sonic Pi, así como de los autores del programa.

Después, se abre la pantalla de Sonic Pi. Hay disponibles varias zonas de trabajo y botones que dan acceso a las funciones de la aplicación:



La zona de ayuda también permite el acceso a ejemplos de programas.

Prefs (en la parte superior derecha de la ventana) da acceso a diferentes ajustes: orientar la salida de sonido al puerto de su elección, personalizar la visualización de Sonic Pi o comprobar la existencia de actualizaciones.

Tocar una nota con Sonic Pi

Para tocar una nota, es necesario usar el comando **play** seguido del número de la tecla (en el orden correspondiente en un teclado de piano).

Escriba la siguiente línea en un espacio de trabajo (después de haber eliminado `# Welcome to Sonic Pi v2.0`). Respete las minúsculas ya que, en caso contrario, puede provocar un error.

```
play 60
```

Pulse a continuación el botón **Run**. Si hay altavoces conectados a la salida a la que se envía el sonido debería oír la nota correspondiente.

La equivalencia entre las notas y la cifra figura en la siguiente tabla.

Do	Re	Mi	Fa	Sol	La	Si
60	62	64	65	67	69	71

Modifique el programa para que se convierta:

```
play 60
play 64
play 68
```

Pulse de nuevo el botón **Run**. Esta vez son tres las notas que se tocan, pero tan rápido que las oímos al mismo tiempo. Para separar las notas, es necesario añadir un intervalo (`sleep 1 = un segundo`), durante el que el programa va a esperar:

```
play 60
sleep 1
play 64
sleep 1
play 68
```

Esta vez, el botón **Run** permite oír las tres notas de manera separada, para una conocida canción francesa.



Es posible tocar melodías, por ejemplo Frère Jacques, cuyas dos primeras frases son:

```

play 60
sleep 0.5
play 62
sleep 0.5
play 64
sleep 0.5
play 60
sleep 0.5
play 60
sleep 0.5
play 62
sleep 0.5
play 64
sleep 0.5
play 60
sleep 0.5

```

Las dos primeras frases se repiten de manera idéntica. Sonic Pi puede gestionar estas repeticiones. Basta con indicarle el número de repeticiones y poner las notas entre **do** y **end**, lo que resulta para estas dos primeras frases:

```

2.times do
  play 60
  sleep 0.5
  play 62
  sleep 0.5
  play 64
  sleep 0.5
  play 60
  sleep 0.5
end

```

➔ En el anexo puede encontrar la melodía completa.

Las posibilidades de Sonic Pi son muy amplias. Puede usar múltiples sintetizadores, sincronizar varias particiones o añadir reverberaciones, como en este ejemplo:

```

with_fx:reverb do |r|
  play 50
  sleep 0.5
  control r, mix: 0.7
  play 55
  sleep 1
  control r, mix: 0.9
  sleep 1
  play 62
end

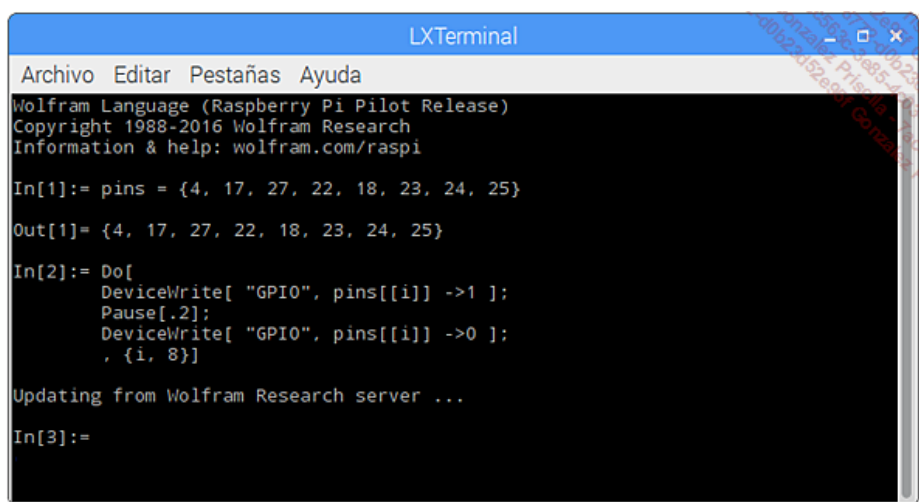
```

La lectura de la ayuda (en inglés) y el uso de los ejemplos proporcionados como base para el inicio son un buen sistema para descubrir las posibilidades de este programa. Sin embargo, hay que observar que cuando la Raspberry Pi debe tocar fragmentos complejos, la carga del procesador sube rápidamente al 100 % y se pueden oír defectos.

k. Wolfram

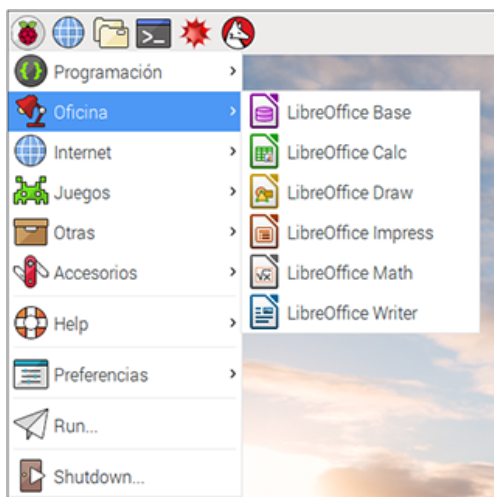
Este elemento del menú abre un terminal en el que puede escribir directamente en lenguaje Wolfram, el lenguaje utilizado en Mathematica.

Wolfram es un lenguaje muy generalista, que se puede utilizar en múltiples dominios. En la Raspberry Pi es posible controlar la GPIO (siguiente ejemplo), así como tratar la información que proviene de un GPS, o de una amplia gama de sensores. Wolfram puede trazar gráficos en 2D o 3D, tratar en tiempo real las imágenes que provienen de la cámara, identificar caras o enfocarlas en un vídeo, por ejemplo.



2. Herramientas de oficina

Con el aumento de la capacidad de cálculo que aporta la Raspberry Pi 3, la Fundación ha decidido integrar la suite LibreOffice en la distribución Raspbian.



LibreOffice está disponible en Raspbian en versión 4.3 (data de finales de 2015). Es probable que a medida que Raspbian evolucione, LibreOffice se actualice para adaptarse a las versiones más recientes.

Esta suite de escritorio se utiliza en muchas empresas. Pone a disposición de sus usuarios herramientas que permiten producir documentos de calidad profesional. LibreOffice usa el formato abierto OpenDocument para registrar documentos, pero es capaz de importar/exportar documentos de la suite Microsoft Office. LibreOffice también gestiona los documentos PDF y HTML.

La comunidad española LibreOffice traduce la documentación de la suite de escritorio y la publica en <https://es.libreoffice.org/>.

La suite de escritorio se compone de seis módulos.

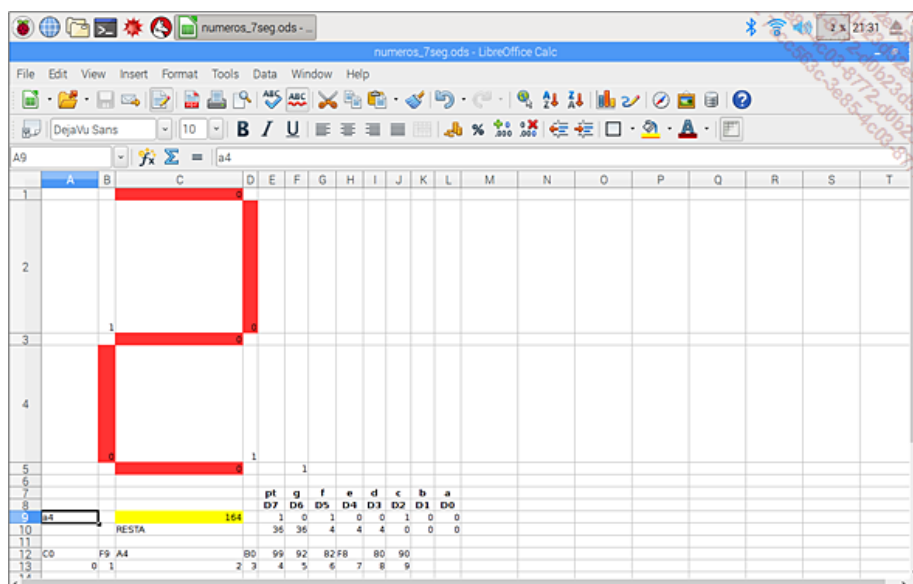
➤ Si desea utilizar LibreOffice en español ejecute el siguiente control en un terminal: `sudo apt-get install libreoffice-l10n-es`.

a. LibreOffice Base

Este módulo puede crear y gestionar bases de datos. Soporta un gran número de bases de datos. Las más utilizadas son: MariaDB, MySQL, PostgreSQL y Microsoft Access. El módulo HSQLDB integrado en LibreOffice Base está desarrollado en Java y pone a disposición del usuario un motor de base de datos interno, evitando recurrir a una base de datos externa.

b. LibreOffice Calc

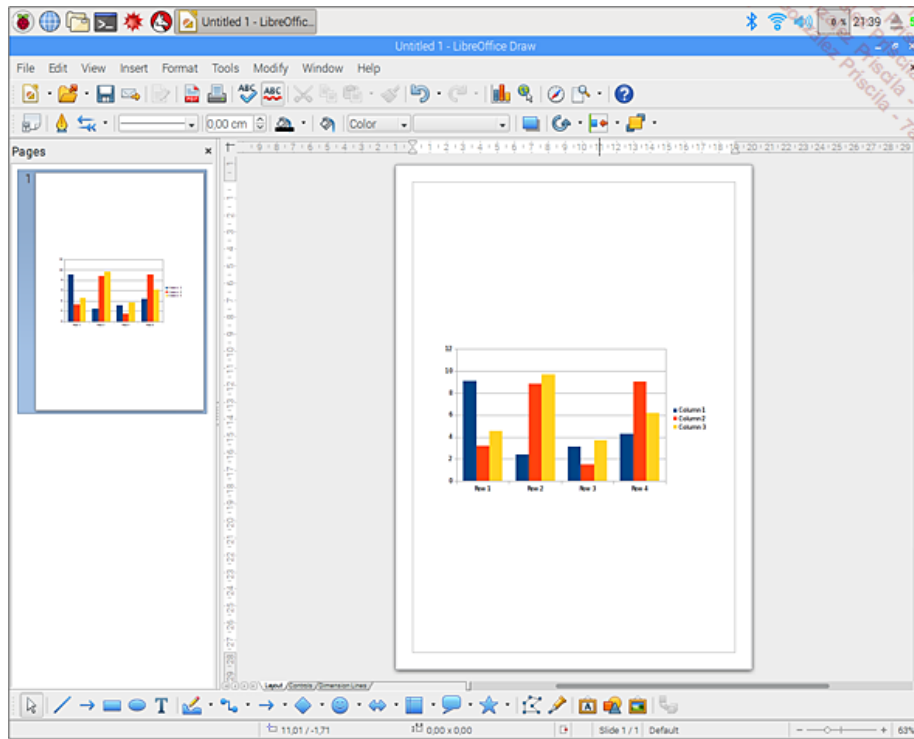
Calc es la hoja de cálculo de la suite de escritorio LibreOffice. Dispone de todas las funciones necesarias para un uso profesional. Calc es compatible con Microsoft Excel.



En el ejemplo anterior, utilizado para una formación Arduino, Calc se utiliza para convertir en binario un byte expresado en hexadecimal. Los bits a 0 activan segmentos que simulan una muestra de siete segmentos controlado por las salidas de la tarjeta Arduino. La modificación del valor del byte implica la modificación de la cifra que se muestra por los segmentos.

c. LibreOffice Draw

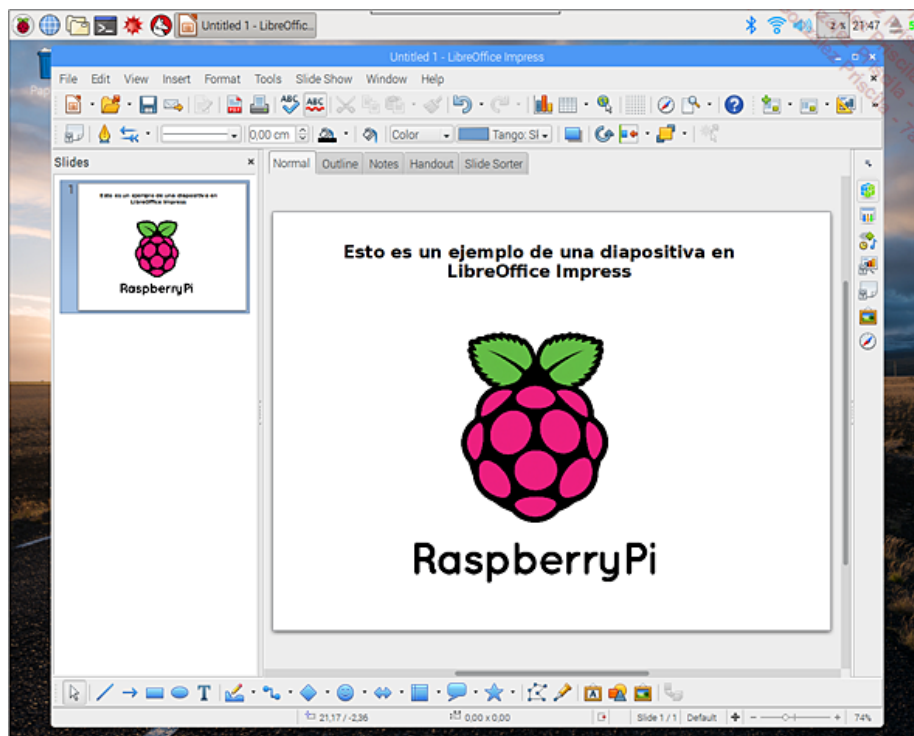
Draw es el módulo de diseño vectorial. Pone a disposición del usuario un medio sencillo y accesible para diseñar esquemas, ilustraciones u organigramas. Draw tiene herramientas para extrudir una forma diseñada en 2D a un volumen 3D.



Hay disponibles muchas galerías que contienen iconos o diseños. El diseño de ilustraciones sencillas o diapositivas para el software de presentación Impress es muy sencillo usando Draw.

d. LibreOffice Impress

En la suite LibreOffice, Impress se encarga de la creación de presentaciones multimedia. Tiene herramientas de edición para los diferentes niveles de realización. El modo **Normal** permite preparar sus diapositivas. El modo **Outline** presenta el contenido de las diapositivas en forma de plano. **Notes** es la pestaña en la que puede agregar notas a cada diapositiva. **Handout** junta varias diapositivas en una misma página, que se podrá imprimir y enviar a los participantes de una reunión, por ejemplo. El **Slide sorter**, interviene en la clasificación de las imágenes, la adición o eliminación de diapositivas.

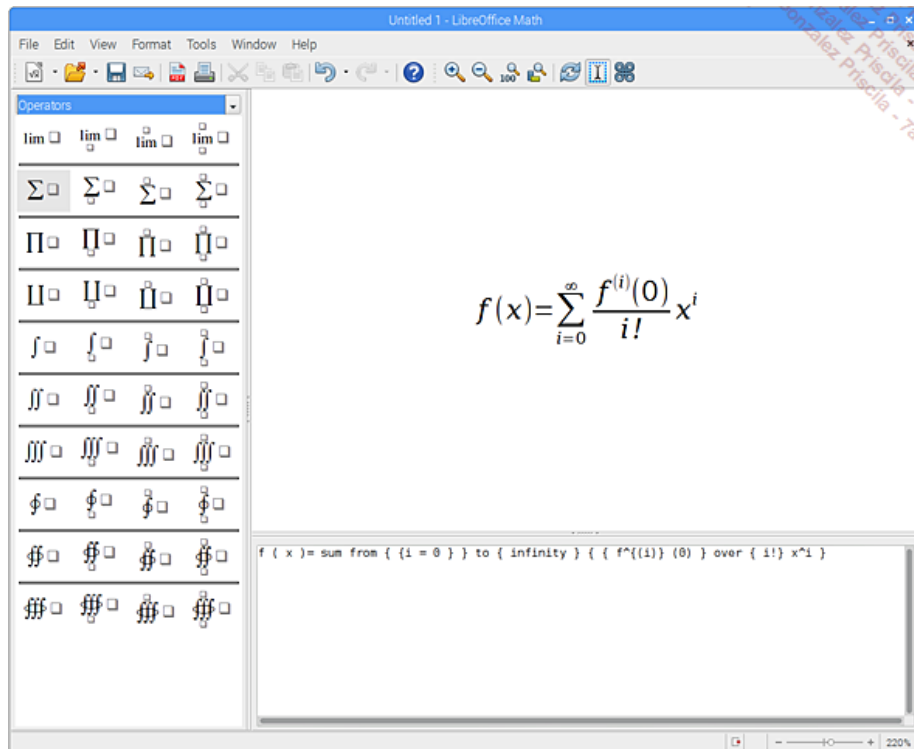


La captura de pantalla anterior muestra la pantalla de Impress con una imagen de ejemplo.

El diseño de las diapositivas no presenta problemas en particular, ni de lentitud en la realización. Por el contrario, algunas diapositivas "pesadas" pueden implicar algunos ligeros saltos de imágenes en la visualización del diaporama. Es el caso en concreto cuando una diapositiva tiene muchas imágenes y animaciones. Vista la potencia de cálculo disponible en la Raspberry Pi, es mejor jugar con la simplicidad.

e. LibreOffice Math

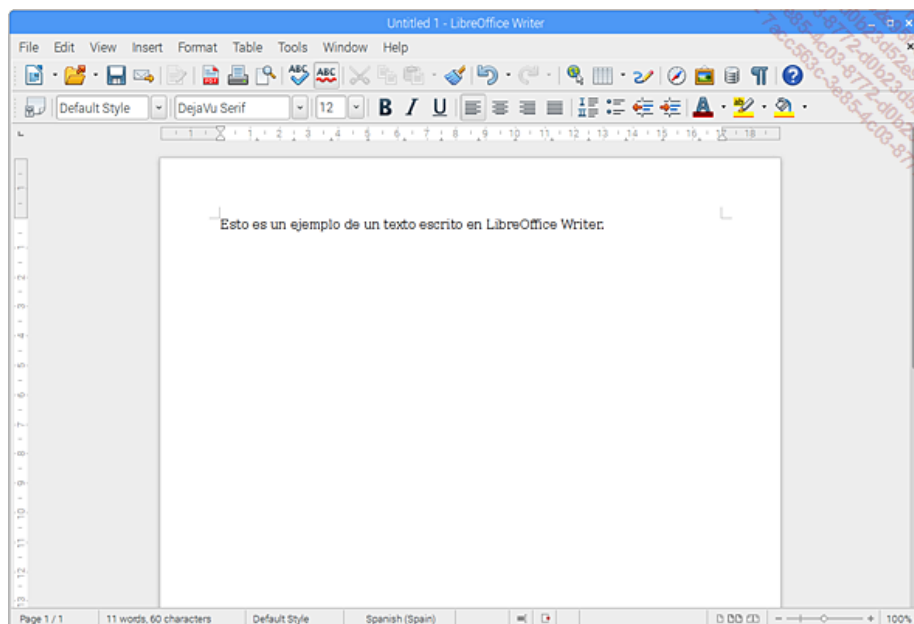
Math no está diseñado para hacer cálculos. Su única función es facilitar la escritura de fórmulas matemáticas, químicas, et... El módulo Writer no dispone de la posibilidad de crear fórmulas que utilicen una alineación especial (imagen siguiente). El módulo Math se puede ejecutar de manera autónoma o llamándolo desde otra aplicación LibreOffice, usando el menú **Edit - Object - Formule**. Esto abre el módulo Math dentro de la aplicación que lo llama. A partir de ese momento, es posible crear una fórmula que utilice las funcionalidades de Math. Un clic fuera de la fórmula permite volver a la edición del documento original.



En modo autónomo, la parte izquierda de la ventana ofrece algunas categorías (operadores, funciones, conjuntos, etc...) que permiten construir la fórmula. La zona inferior está reservada a la escritura de las fórmulas que usan un lenguaje dedicado. La fórmula aparece en la parte superior. Cuando Math se llama desde otra aplicación LibreOffice, se abren las ventanas izquierda e inferior. La fórmula aparece directamente en el documento original.

f. LibreOffice Writer

Writer es el procesador de textos de la suite de escritorio LibreOffice. En él encontramos las funciones clásicas de cualquier procesador de textos. Permite la producción de documentos profesionales, directamente exportables a formato PDF.

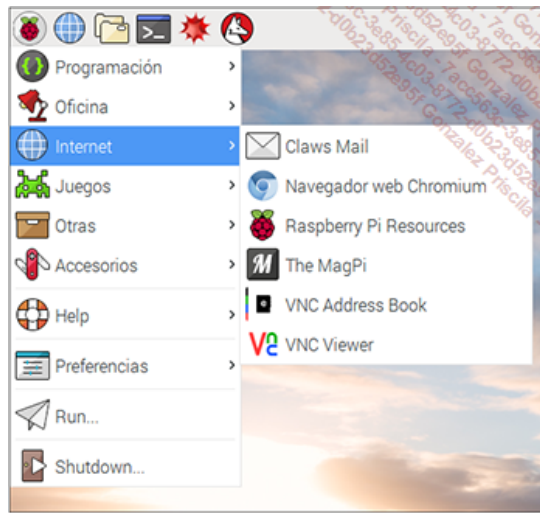


Los asistentes ofrecen modelos de documentos para facilitar la redacción. La captura de pantalla anterior muestra un texto de ejemplo.

Conclusión

LibreOffice ya funcionaba en la Raspberry Pi 2, en ocasiones con algunos problemas, aunque había que instalarlo manualmente. Al pasar a la Raspberry Pi 3 y la integración de LibreOffice en Raspbian, ahora ofrece la posibilidad de utilizar directamente las herramientas de oficina de manera relativamente fluida, para una máquina de este precio. Esto debería permitir a los establecimientos escolares poner a disposición de los alumnos máquinas poco costosas, con un acceso a las herramientas de oficina.

3. Internet



La sección **Internet** del menú principal de Jessie permite el acceso a las aplicaciones relacionadas con Internet (correo electrónico y navegador web) o a sitios web en línea.

a. Claws Mail

Claws Mail es un cliente de correo electrónico muy completo y parecido a los clientes de correo electrónico utilizados habitualmente. Ligero y fácil de configurar, pone a disposición de los usuarios de la Raspberry Pi un medio sencillo para acceder a sus diferentes cuentas de correo electrónico, independientemente de que protocolo POP (*Post Office Protocol*) descargue los mensajes en la máquina local o bien a través de IMAP (*Internet Message Access Protocol*), mediante el cual se consultan los mensajes en el servidor de mensajería. IMAP garantiza la sincronización entre las diferentes máquinas que intervienen en el buzón de mensajes.

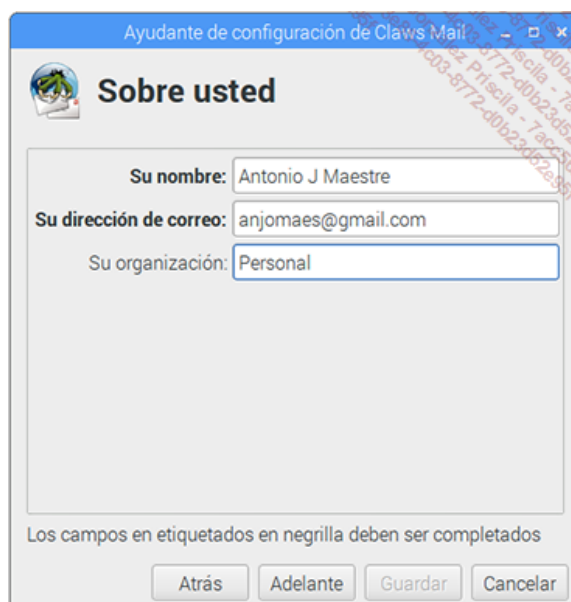
- Las instrucciones dadas en este capítulo son válidas cuando se están redactando estas líneas. Es posible que también haya modificaciones que se deban realizar en el cliente de correo electrónico y en el servicio remoto. Tendrá que adaptar las instrucciones en función de estas evoluciones.

Inicio de Claws Mail

Arranque el programa Claws Mail pulsando en el nombre del software, en la sección **Internet**, del menú principal de Raspbian.



Durante el primer arranque de Claws Mail es necesario configurar el o las cuentas de correo electrónico que se van a utilizar. A continuación, este mensaje de bienvenida no se mostrará más. Pulse el botón **Adelante**.



Rellene la información necesaria en las zonas de texto previstas para ello y después pulse el botón **Adelante**.

Para guardar los campos en las siguientes ventanas, necesita la información proporcionada por su servicio de mensajería. Para Google Mail, la página <https://support.google.com/mail/troubleshooter/1668960?hl=es> proporciona las siguientes indicaciones para la configuración del cliente

de correo electrónico. Vaya a la página de ayuda de su programa de correo electrónico para obtener esta información.

Servidor de correo entrante (IMAP)	imap.gmail.com
	Requiere SSL: Sí
	Puerto: 993
Servidor de correo saliente (SMTP)	smtp.gmail.com
	Requiere SSL: Sí
	Requiere TLS: Sí (si está disponible)
	Requiere autenticación: Sí
	Puerto para SSL: 465
	Puerto para TLS/STARTTLS: 587
Nombre completo o nombre mostrado	Tu nombre
Nombre de la cuenta, nombre de usuario o dirección de correo electrónico	Tu dirección de correo electrónico completa
Contraseña	Tu contraseña de Gmail

Complete la información necesaria para la recepción y el envío de correo. El buzón de mensajes utilizado aquí es un buzón gmail sobre IMAP. Será necesario adaptar las respuestas a la configuración de su propio servidor.

Ayudante de configuración de Claws Mail

Recibiendo correo

Configuración automática

Tipo de servidor: IMAP

Dirección del servidor: imap.gmail.com

Nombre de usuario: anjomaes@gmail.com

Contraseña:

Usar SSL para conectar al servidor de recepción

Usar SSL a través de STARTTLS

Certificado SSL de cliente (opcional)

Fichero: [] Explorar

Contraseña: []

Directorio del servidor IMAP: []

Mostrar sólo las carpetas suscritas

Los campos en etiquetados en negrilla deben ser completados

Atrás Adelante Guardar Cancelar

Cuando la recepción de correo esté configurada, pulse el botón **Adelante**.

Ayudante de configuración de Claws Mail

Enviando correo

Dirección del servidor SMTP: smtp.gmail.com

Usar autenticación (vacío para usar el mismo que en la recepción)

Usuario SMTP: anjomaes@gmail.com

Contraseña SMTP:

Usar SSL para conectar al servidor SMTP

Usar SSL a través de STARTTLS

Certificado SSL de cliente (opcional)

Fichero: [] Explorar

Contraseña: []

Los campos en etiquetados en negrilla deben ser completados

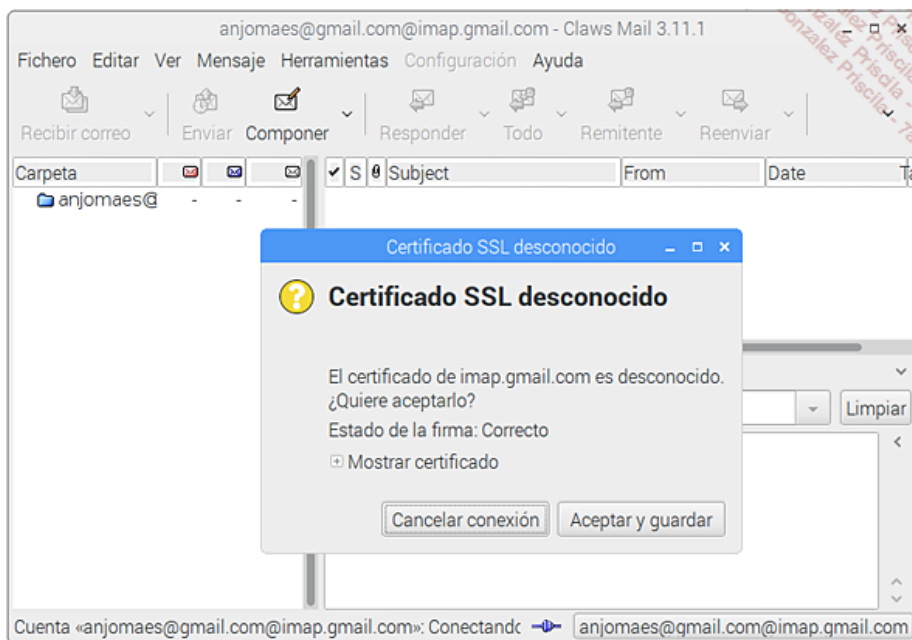
Atrás Adelante Guardar Cancelar

Rellene ahora la información para el envío del correo y pulse el botón **Adelante**.

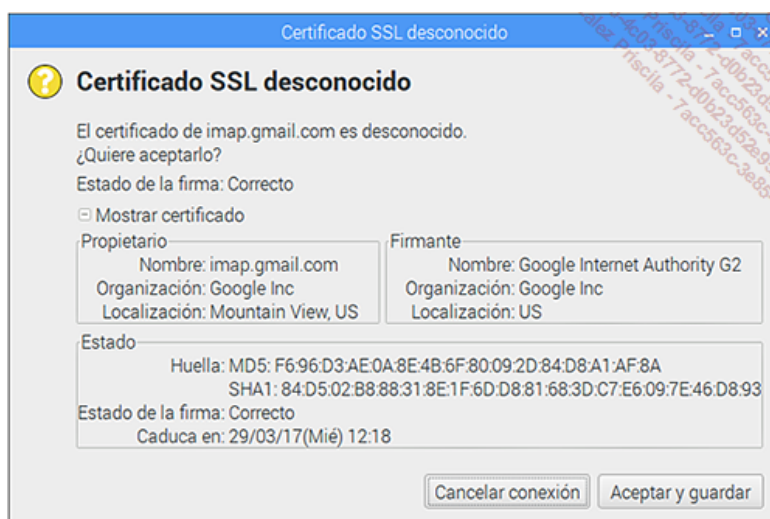


Claws Mail le informa que se ha configurado su primera cuenta de correo electrónico. Pulse el botón **Guardar** para memorizar la información.

Durante el primer uso del correo electrónico, el certificado proporcionado por Gmail es desconocido para la aplicación. Este certificado sirve para proteger los intercambios entre el servidor de correo y la Raspberry Pi, cifrando los mensajes.



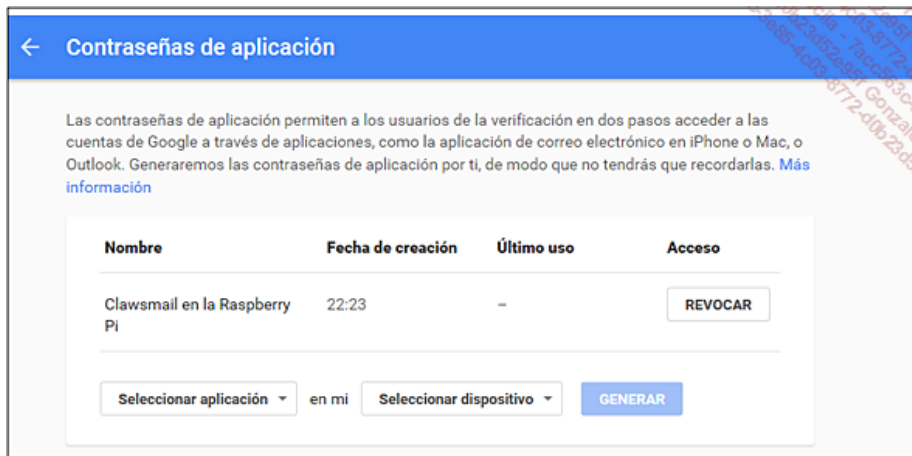
Haciendo clic en **Mostrar certificado** puede comprobar si los campos presentes en el certificado son los esperados (zonas **Propietario** y **Firmante**).



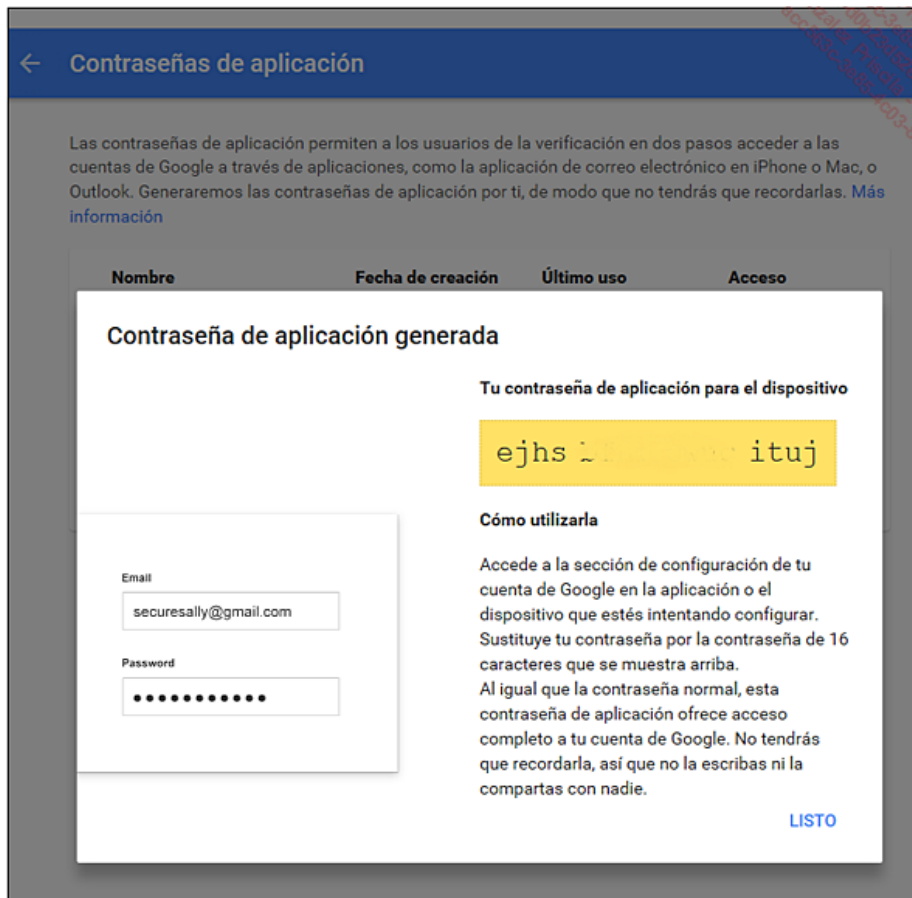
Después de comprobar que esta información se corresponde con el servicio al que se conecta, pulse el botón **Aceptar y guardar**. El certificado se almacena y esta comprobación ya no se pedirá más.

Con algunos servicios de mensajería, como Gmail, es posible que la conexión sea rechazada desde una aplicación como Claws. Incluso si introduce correctamente la información del login y la contraseña, habría que generar una contraseña para la aplicación. En Gmail, es una sucesión de 16 caracteres que habría que introducir para conectarse desde Claws.

Para Gmail, vaya a la página <https://security.google.com/settings/security/apppasswords>

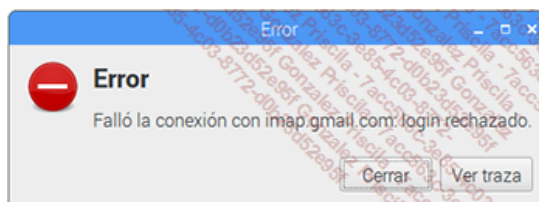


Pulse el botón **GENERAR** para obtener una contraseña para la aplicación Claws Mail.



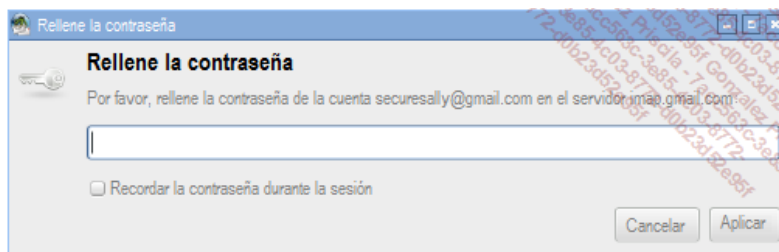
Guarde esta contraseña, porque se le pedirá más adelante para conectarse desde la aplicación.

Durante las siguientes conexiones a su correo electrónico con su dirección de mail y su contraseña, obtendrá el siguiente cuadro de diálogo:

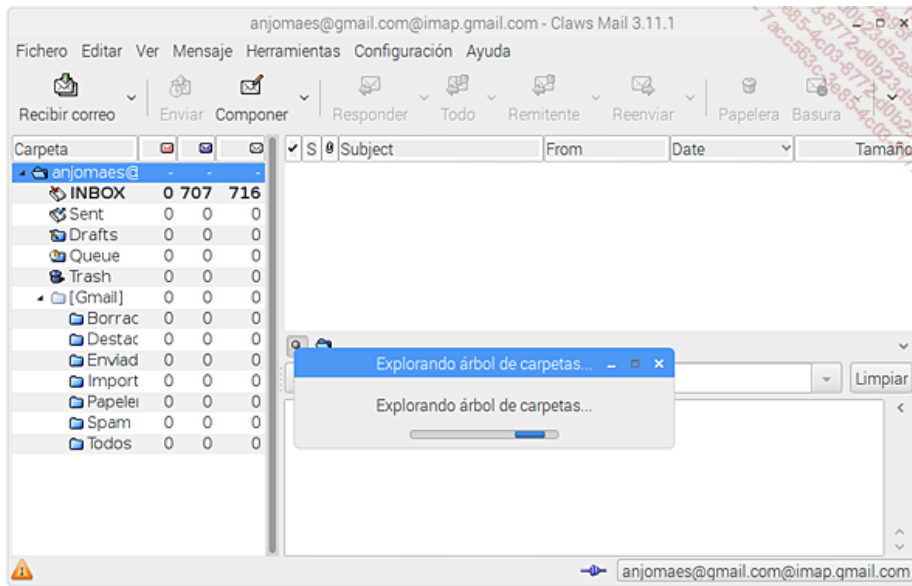


Es la advertencia que indica que Gmail ha rechazado la conexión.

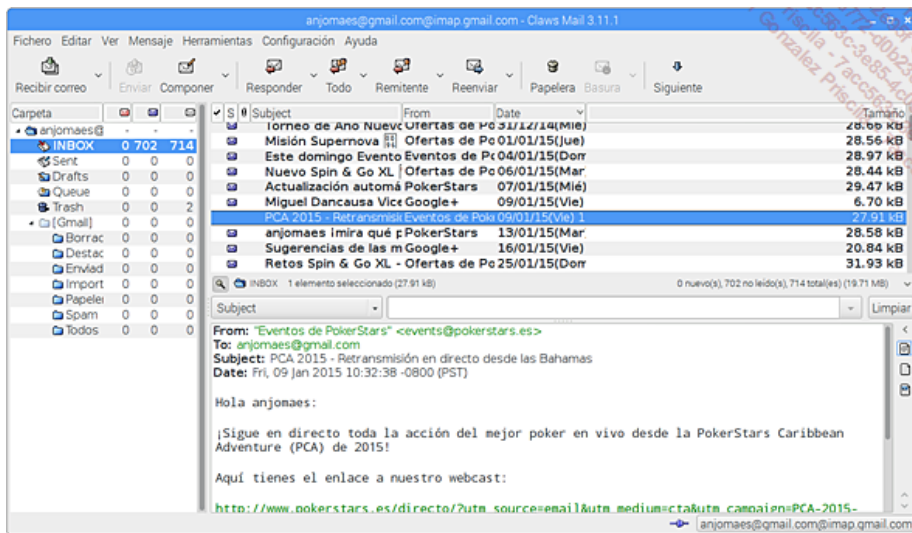
Pulse el botón **Cerrar** e introduzca la contraseña de la aplicación (de 16 caracteres) en la zona de texto de la siguiente ventana:



Durante la primera conexión, Claws Mail va a reconstruir el árbol de su buzón de mensajes preguntando al servidor remoto.



Ahora tendrá a su disposición su buzón de mensajes en la Raspberry Pi. Los botones de la barra superior dan acceso a las funciones habituales de los clientes de correo electrónico: escribir, responder, reenviar, etc...



Haciendo clic en el menú **Configuración - Crear nueva cuenta** puede configurar varias cuentas de correo electrónico en Claws Mail.

Conclusión

Con la introducción de la suite LibreOffice y de Claws Mail, el objetivo de la Fundación es transformar la Raspberry Pi en un puesto de oficina. Con la potencia de la Raspberry Pi 3, este objetivo se consigue.

b. Navegador web Chromium

Con PIXEL, es el navegador Web Chromium se integra por defecto en Raspbian. Chromium es un navegador Web libre que sirve de base al navegador Web propietario de Google Chrome. Algunas veces los usuarios prefieren Chromium a Google Chrome, pues consideran poco útil e incluso nefasto lo que aporta Google Chrome respecto a Chromium, en particular las funcionalidades de rastreo (fuente Wikipedia).



Epiphany añade mejoras respecto a las versiones anteriores de Epiphany, en particular un mejor soporte de HTML5.

En una Raspberry Pi 3, el uso de Chromium no puede competir con la experiencia de un navegador experimentado como Chrome o Firefox que se ejecute por un procesador a 3 GHz. Sin embargo, en el ámbito de un uso ocasional de la Raspberry Pi para acceder a Internet, el uso de este navegador es muy aceptable. Los vídeos de YouTube se pueden ver cómodamente, y apenas se notan algunas pequeñas paradas del

todo son aceptables.

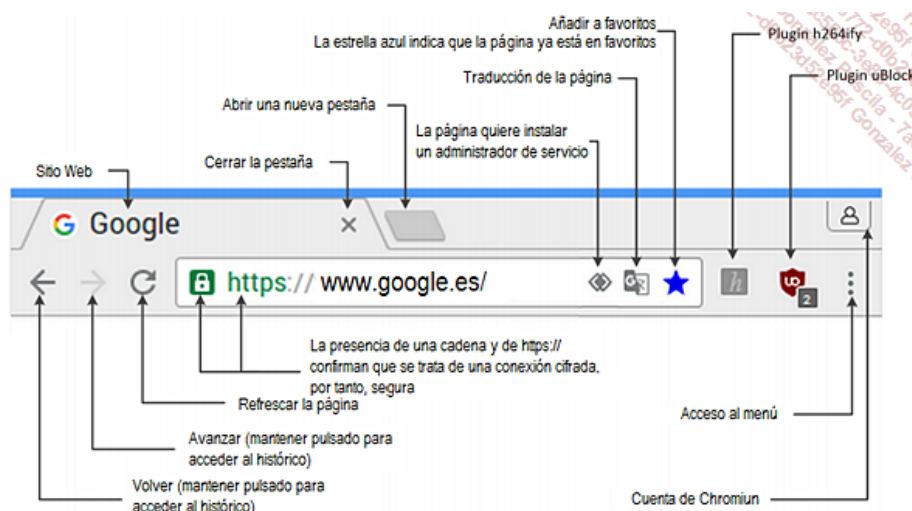
Barra de título de Chromium

La barra de título azul indica el nombre del sitio web que se está mostrando.

Barra de navegación de Chromium

A la derecha están los iconos habituales en las ventanas gráficas, de izquierda a derecha:

- **Minimizar**, que hace desaparecer la ventana de la pantalla sin cerrarla, porque permanece visible en la barra de tareas. Bastará con pulsar en el nombre de la ventana en la barra de tareas para que se abra de nuevo.
- **Maximizar/Restaurar**, que cambia la visualización de la ventana de su tamaño actual a pantalla completa y hace que vuelva a su tamaño reducido pulsando de nuevo en el icono.
- **Cerrar**, representado por una cruz que cierra la ventana y el programa.



La barra de navegación tiene las pestañas y el acceso a la cuenta Chromium en el extremo derecho.

A la izquierda de la barra de navegación figuran los botones para navegar a la página anterior o siguiente, opciones representadas por una flecha a la izquierda o a la derecha. Una flecha circular permite refrescar la página.

A continuación viene la barra de dirección. Es aquí donde puede introducir la dirección del sitio web que desea visitar. Observe a la izquierda de la barra de dirección la presencia de una cadena cerrada, que confirma el uso del protocolo de seguridad HTTPS, con un certificado de autenticación emitido por una autoridad reputada fiable. Si el certificado es auto firmado, la cadena está cruzada por un aspa y se muestra sobre fondo rojo. https está tachado y se muestra en rojo.

Si introduce una palabra o una sucesión de palabras en la barra de dirección, el navegador realiza una búsqueda usando **Duck Duck Go**. Este motor de búsqueda garantiza que no guarda ninguna información personal de sus visitas, ya sea su dirección IP o las búsquedas realizadas. Esto evita ser "perseguido" a través de sus movimientos en la web con anuncios dirigidos.

A la derecha de la barra de dirección existen iconos que señalan, cuando están presentes, la petición de la página para instalar un administrador de servicio o la posibilidad de traducir la página. El icono en forma de estrella permite agregar el sitio Web a los favoritos.

Los iconos **h264ify** y **uBlock Origin** están situados en el exterior y a la derecha de la barra de dirección. La fundación ha añadido estos dos plugins a Chromium.

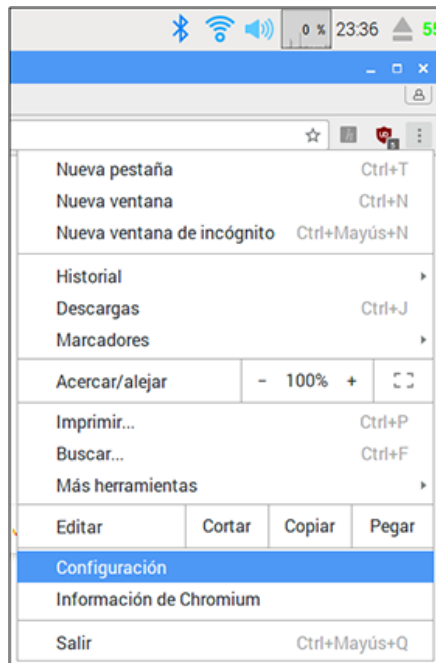
h264ify es una extensión de Chromium que pide a YouTube enviar vídeos en formato H.264 en lugar de enviarlos en los formatos VP8/VP9, que no usan generalmente aceleración basada en hardware. Esta extensión es útil cuando los vídeos saltan o consumen muchos recursos de CPU. Dejando los vídeos H.264 con la aceleración basada en hardware por la GPU, h264ify mejora la fluidez y minimiza el uso del procesador. El uso en una Raspberry Pi Zero no ofrece el mismo rendimiento. Un clic derecho en el icono h264ify abre un menú gracias al cual es posible ajustar las opciones o eliminar el plugin.

uBlock Origin es una extensión poco exigente en términos de memoria y de uso del procesador. Bloquea la publicidad, así como la recogida de los datos de navegación.

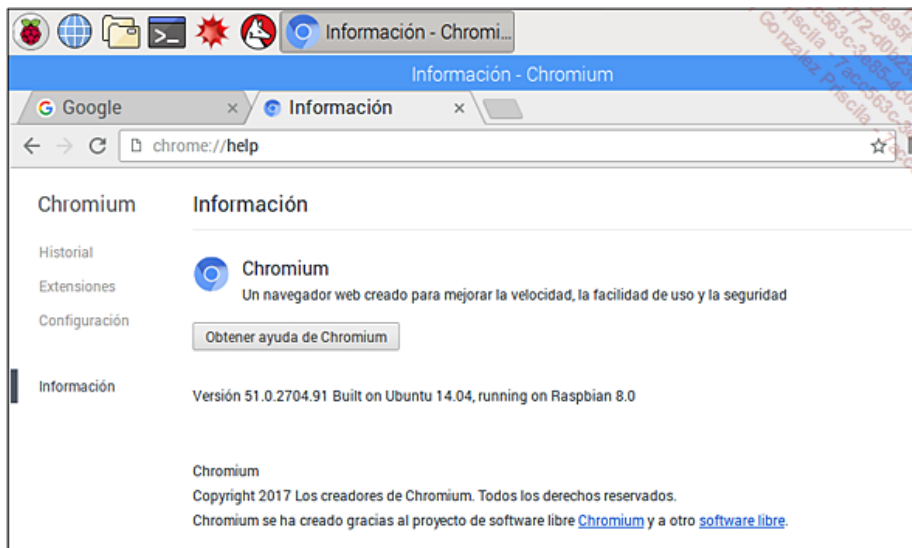


Un clic izquierdo en el icono de uBlock muestra información de los bloqueos realizados por el plugin (imagen anterior). Los iconos en la parte inferior de la ventana autorizan el bloqueo de las ventanas pop-up, de recursos multimedia de gran tamaño, el filtrado estético o los tipos de letra en remoto.

En el extremo de la derecha, el icono representa tres puntos de acceso al menú del navegador.



Más allá de las clásicas: **Nueva pestaña**, **Nueva ventana**, encontramos el elemento **Nueva ventana de incógnito**, que abre una nueva ventana del navegador web que usa los argumentos de la ventana principal. Esta "ventana privada" está especialmente configurada para proteger la vida privada, impidiendo que los cambios se guarden. Por ejemplo, las cookies solo serán válidas durante el ciclo de vida de esta ventana web. Después, desaparecerán. El resto de elementos del menú llaman a pocos complementos. **Configuración** abre una página web (<chrome://settings>), que da acceso a la configuración avanzada de Chromium.



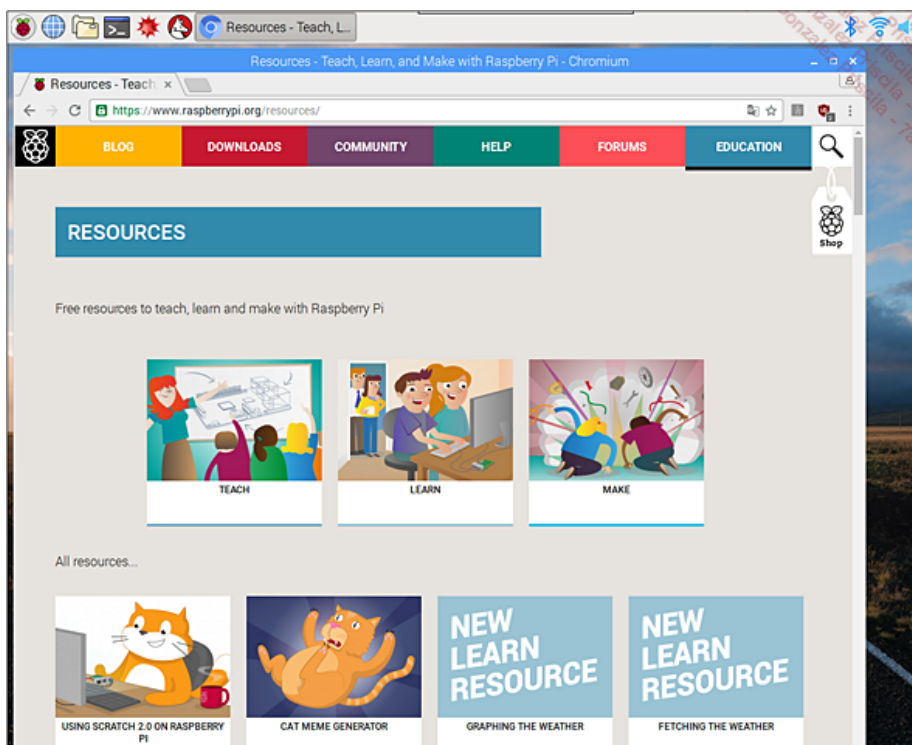
Acerca de Chromium informa de la versión de Web, el equipo de creadores y la licencia del navegador.

Utilizar Chromium

Cuando Chromium está configurado según las necesidades del usuario, se usa tan fácilmente como otro navegador. Abre el acceso a Internet desde una sencilla Raspberry Pi 3. Por supuesto, el rendimiento es menor en una Raspberry Pi Zero, más limitada en recursos.

c. Raspberry Pi Resources

La Fundación Raspberry Pi se creó con el objetivo de facilitar el aprendizaje de la informática por parte de los niños, poniendo recursos a disposición de los usuarios.



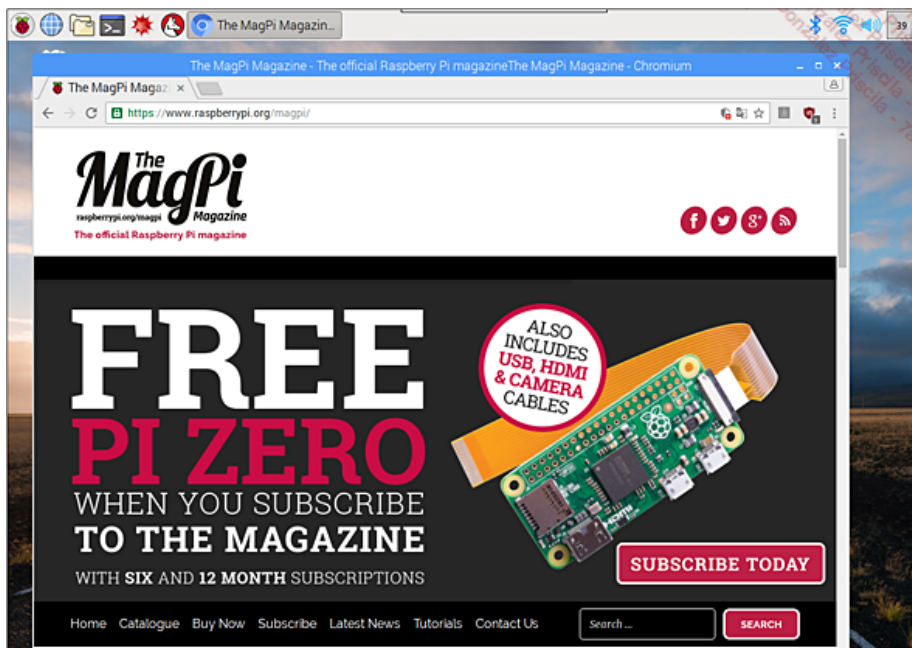
Hay tres secciones:

- Fichas con objetivo docente
- Fichas para los alumnos
- Proyectos a realizar con la Raspberry Pi

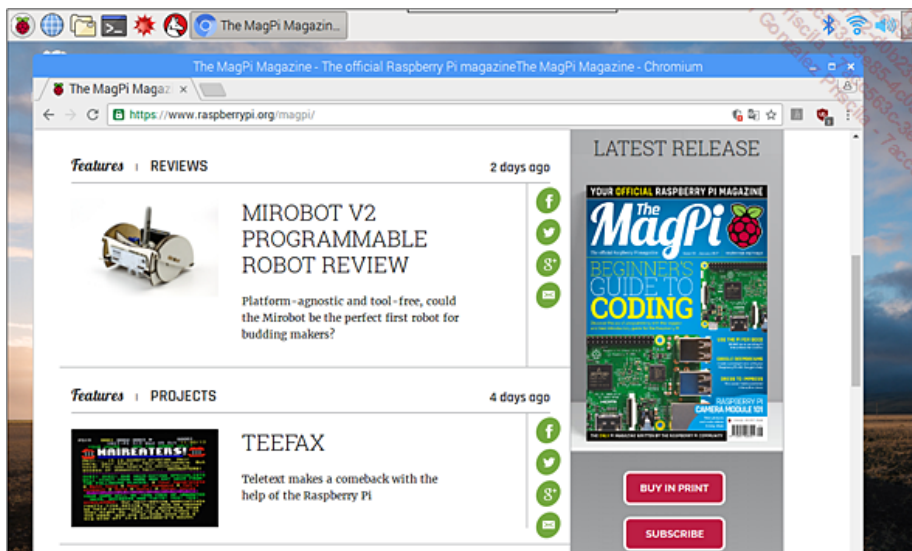
Estos recursos están en inglés, pero la comunidad traduce estos documentos a diferentes idiomas, entre ellos el español.

d. The MagPi

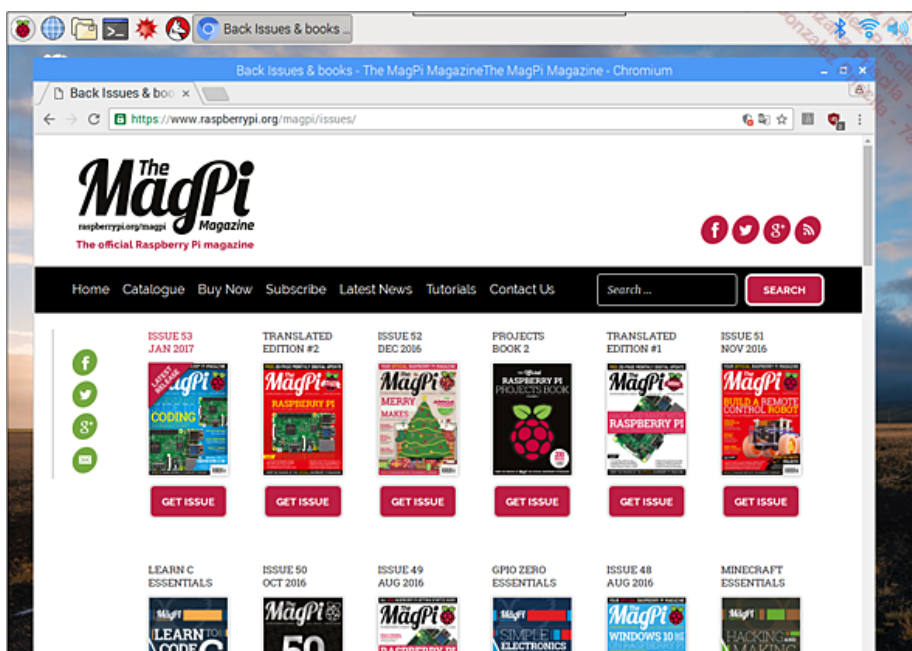
El elemento **The MagPi** del menú **Internet** le envía directamente a la página de la revista oficial de la Raspberry Pi. Esta revista mensual en inglés ofrece al lector artículos de presentación de los productos, ejemplos de proyectos, programas, etc...



La barra de menús en la parte inferior de la ventana da acceso a diferentes secciones, donde encontramos un conjunto de tutoriales dedicados a la puesta en marcha de la Raspberry Pi y sus accesorios.



La revista es gratuita y está disponible para su descarga en formato PDF. También es posible suscribirse a la revista en formato papel.

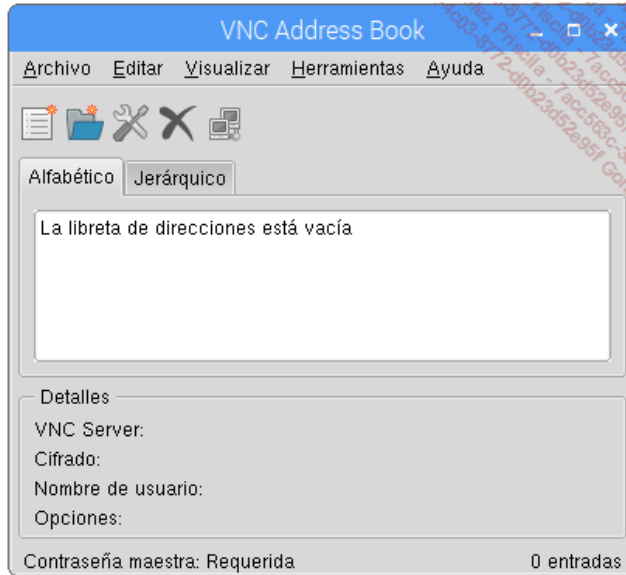


Para acceder a todos los números de la revista The Maggi, pulse en **Catalogue** y seleccione un número de la revista. En la página que se abre pulse el botón **DOWNLOAD PDF** para leer el número de The MagPi que haya seleccionado.

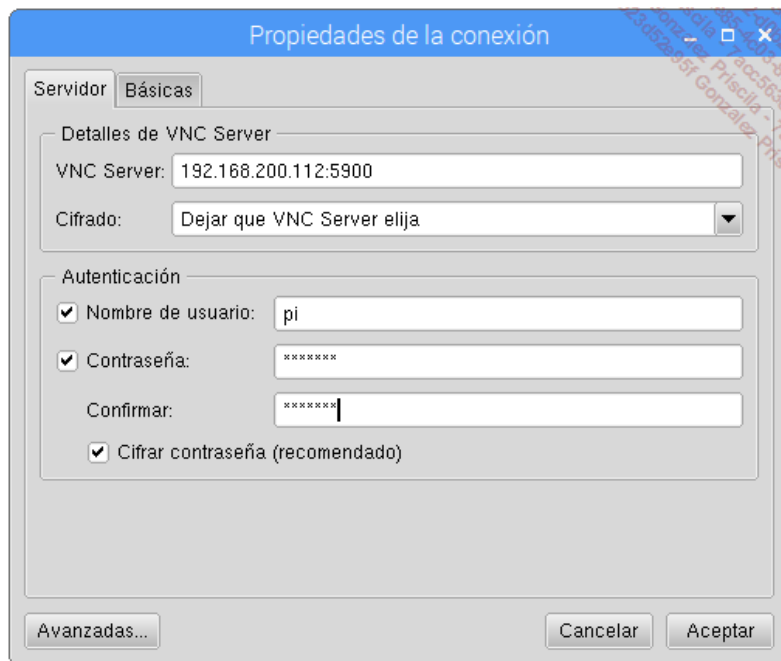
e. VNC Address Book

Vnc Address Book es una herramienta que centraliza todas sus conexiones VNC. Es particularmente útil si debe conectarse a muchas máquinas remotas desde su Raspberry Pi. Permite gestionar la organización de las conexiones en un árbol de carpetas.

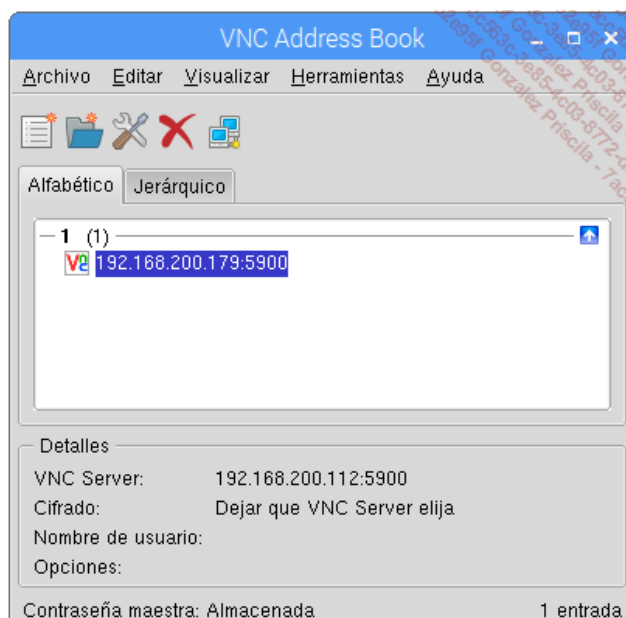
En la primera conexión, VNC Address Book le solicita una contraseña maestra, que protegerá el acceso a sus datos de conexión.



Para crear una nueva conexión, haga clic en **Archivo - Nueva entrada** o en el icono más a la izquierda, que representa un documento.



Rellene la información relativa a la conexión a guardar. La zona **Autenticación** debe informarse con los identificadores utilizados para conectarse a la Raspberry Pi local.

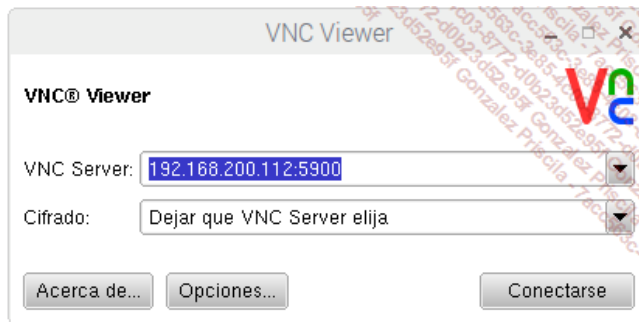


Después de validar, la línea correspondiente aparece en la ventana de VNC Address Book.

Para conectarse a una máquina remota, bastará con hacer doble clic en la línea y después introducir la contraseña maestra. La conexión con VNC Viewer se lanza automáticamente.

f. VNC Viewer

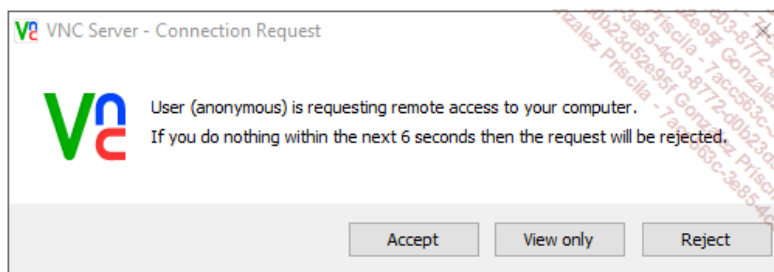
Con la llegada de las herramientas proporcionadas por VNC, la Fundación ha integrado el servidor VNC, que permite conectarse a la Raspberry Pi desde un puesto conectado a la red local. También ha optado por agregar VNC Viewer, que permite tomar el control remoto de un puesto de trabajo que ejecuta el servidor VNC. Este puesto puede ser un PC (en Windows/Linux) o un Mac.



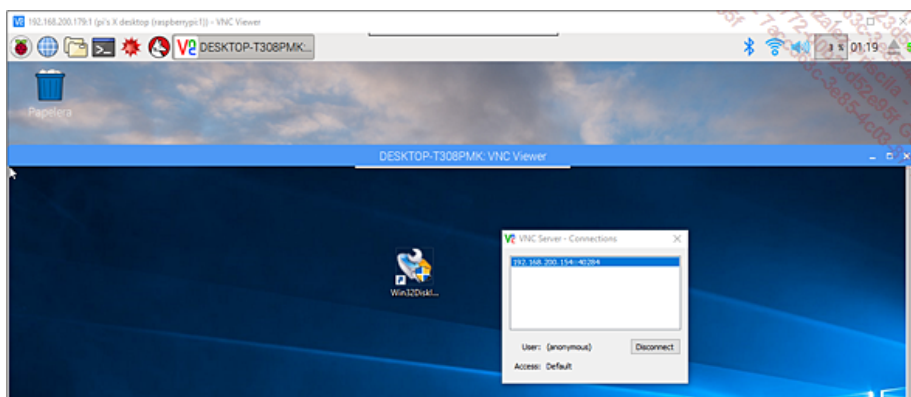
Cuando lance VNC Viewer, debe indicar la dirección del servidor al que desea conectarse en la ventana anterior. Aquí la dirección corresponde a un PC con Windows 8.



Escriba la contraseña que ha introducido durante la instalación del servidor.



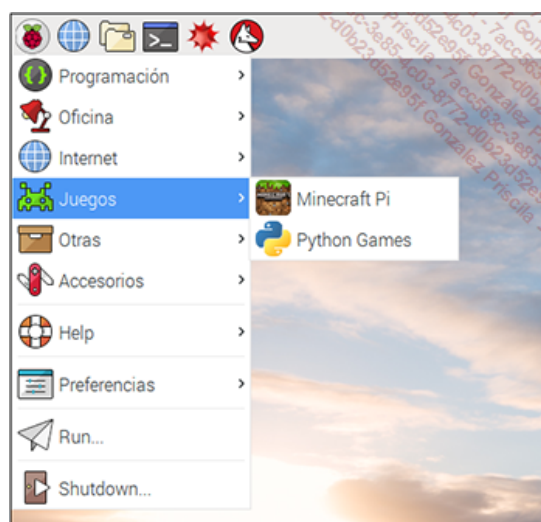
En la máquina remota (aquí el PC) se abre una ventana para indicarle que un usuario desea conectarse a su máquina. El tiempo es limitado y cuando el contador llega a 0 ya no es posible establecer la conexión. Es necesario recomenzar con la conexión desde la Raspberry Pi. Si acepta la conexión dentro del plazo, la ventana de VNC Viewer se abre en la Raspberry Pi y tiene acceso al PC remoto.



En la captura de pantalla anterior, realizada en la Raspberry Pi, se ve la ventana del PC en Windows y se puede utilizar en modo remoto. La ventana VNC Server abierta en el PC indica que un usuario está conectado. La dirección que se muestra es la de la Raspberry Pi.

Añadiendo VNC a la lista de herramientas disponibles en la Raspberry Pi, la Fundación permite gestionar la Raspberry Pi en modo remoto, pero también tomar el control desde la Raspberry Pi de otras máquinas, lo que hace que se trate de un terminal ligero, barato y en definitiva, una buena opción.

4. Juegos



La sección **Juegos** de Raspbian Jessie contiene inicialmente dos elementos. El primero permite lanzar Minecraft y el segundo ofrece juegos escritos en Python, el lenguaje original de la Raspberry Pi.

a. Minecraft Pi

Minecraft es un videojuego independiente de tipo caja de herramientas (construcción completamente libre), desarrollado por el sueco Markus Person (Wikipedia - página dedicada a Minecraft).

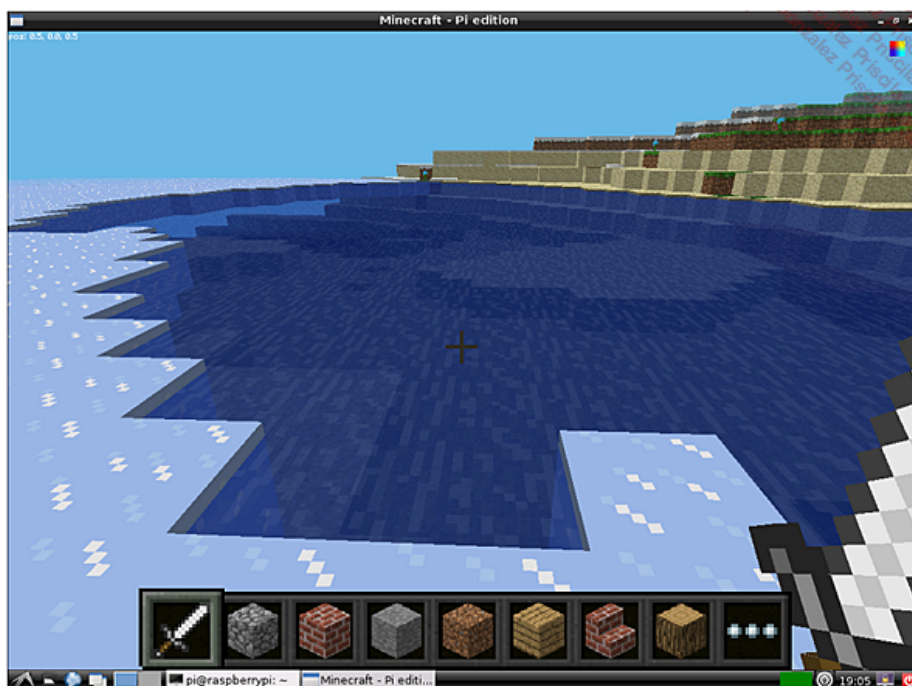
Inicialmente ejecutado en un navegador web, el juego se ha adaptado a los PC, Mac y Linux gracias a Java. A continuación, el juego se migró a los smartphones (Android e iOS), así como a la Xbox y PlayStation de Sony.

La versión Raspberry Pi, que salió a finales de diciembre de 2012, es gratuita y está integrada en la versión de Raspbian ofrecida por la Fundación Raspberry Pi. Minecraft introduce al jugador en un mundo compuesto de bloques que representan la tierra, la arena, el agua, las piedras, los árboles, los animales, etc. El jugador es libre de modificar este mundo como desee, añadiendo y retirando los cubos, intentando sobrevivir.

Para ejecutar el juego, pulse en la opción **Minecraft Pi** en el menú **Juegos**.



Para iniciar un juego, pulse en **Start Game**. Minecraft construye un universo y muestra el resultado en la ventana de juego.



La rueda del ratón le permite seleccionar un objeto del panel situado en la parte inferior de la pantalla. Si pulsa la tecla **E** se abre la tabla completa de los objetos a su disposición. El movimiento del ratón permite volver la mirada en todas direcciones. Las siguientes teclas mueven al jugador:

- **W** hacia adelante
- **S** hacia atrás
- **A** a la izquierda
- **D** a la derecha

Las otras teclas :

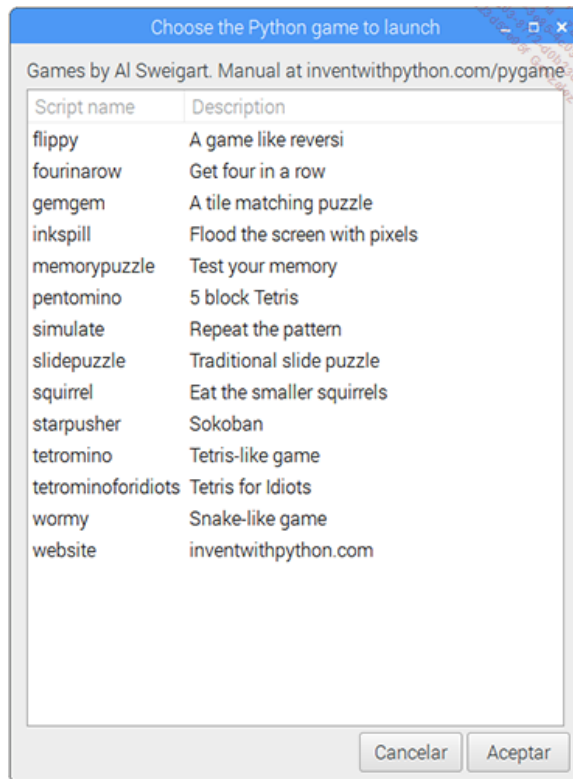
- **E** muestra el inventario de los objetos
- **Espacio** permite saltar
- **Dos veces Espacio** permite volar. La altitud aumenta todo el tiempo que la tecla Espacio está pulsada. Un doble clic en Espacio permite aterrizar.
- **ESC** pausa el juego y vuelve al menú.
- **Tabulación** permite al sistema operativo retomar el control del ratón.

Cuando tenga la espada en la mano, pulse para eliminar los bloques que se encuentran frente a usted o para excavar. Cuando tenga un objeto, un clic en el botón derecho del ratón lo deja en la escena y un clic en el botón izquierdo del ratón lo elimina.

También es posible jugar en red o introducir comandos en el entorno Python IDLE (<http://www.raspberrypi.org/learning/getting-started-with-minecraft-pi/worksheet.md>).

b. Python Games

Este icono abre una primera ventana que le invita a configurar el sonido de la Raspberry Pi. Puede dejarlo como está, dejar que el sistema configure el audio de manera automática, forzar la salida por el jack 3,5 mm o forzar la salida de sonido por HDMI. Después, se abre una ventana que contiene los juegos y podrá seleccionar para jugar el que desee.



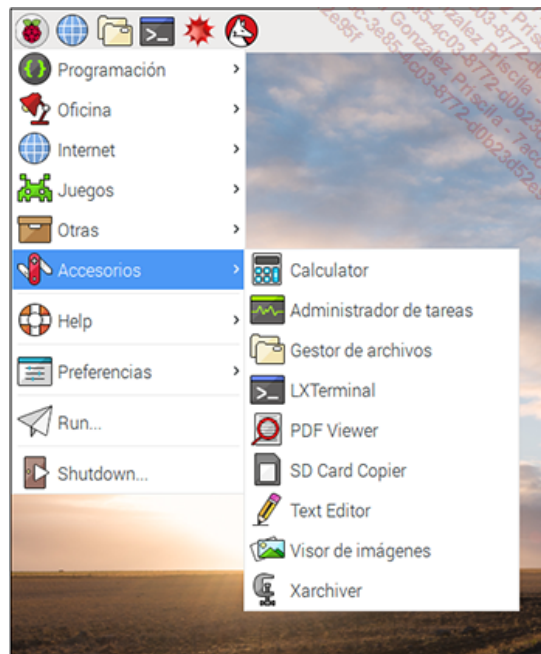
Hay trece juegos escritos en Python. Seleccione el que desee probar y después pulse en **Aceptar**.

La calidad de los juegos no es la que puede encontrar en una consola de videojuegos moderna. Pero la última línea de la ventana le envía al sitio web del autor, inventwithpython.com/pygame/, que le permite leer en línea el libro Making Games with Python & Pygame (Construir juegos con Python y Pygame). Este libro tiene licencia de Creative Commons y contiene los programas de algunos juegos.

El aprendizaje de la programación pasa por el estudio de programas escritos por otros. El dominio de estos programas y su modificación es una buena manera de entender los principios de programación en Python.

El capítulo Programar en Python le dará las bases para entender los aspectos principales y empezar la programación en Python.

5. Accesorios



Accesorios, del menú principal de Raspbian, da acceso a una barra de herramientas que permiten realizar las tareas más habituales. Para arrancar una de estas herramientas solo tiene que pulsar el icono correspondiente.

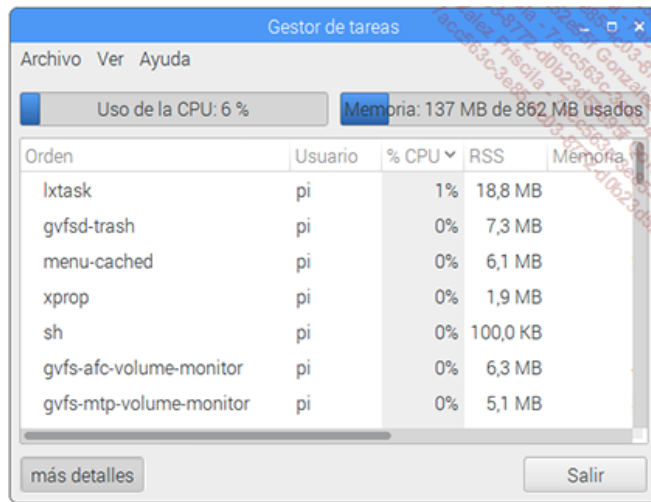
a. Calculator

Calculator es una calculadora que tiene tres modos de funcionamiento. El menú **Visualización** de la calculadora permite seleccionar entre los tres modos:

- **Modo Básico**, muestra por pantalla una calculadora básica para las cuatro operaciones, el cálculo de raíz cuadrada y de porcentaje. También está equipada con una función de memoria para memorizar los resultados (**MS** y después **guardar aquí**), recordarlos (**MR**) o añadirlos (**M+**).
- **Modo Científico**, ofrece una calculadora científica que dispone de muchas funciones (potencias, factoriales, logaritmos, trigonometría, etc...). Las constantes π y e están presentes y es posible definir por defecto la base de numeración y las unidades angulares. Para los habituados a las calculadoras HP que usan este modo de trabajo, existe un modo Notación Polaca Inversa.
- **Paper Mode** funciona línea a línea, como si el usuario contara en una hoja. Este modo acepta todas las operaciones y gestiona los niveles de paréntesis.

b. Administrador de tareas

El administrador de tareas es una herramienta destinada a informar al usuario de los sistemas operativos.



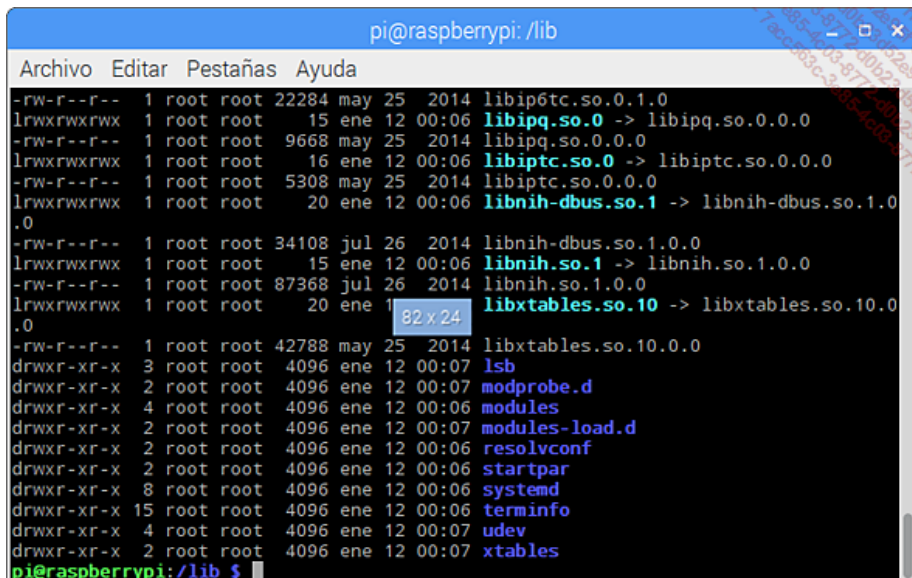
En la parte superior de la ventana, dos barras resumen el uso del procesador y de la memoria. En la parte inferior, el administrador de tareas muestra los procesos activos (eventualmente con la línea de control que los ha lanzado), el usuario que ha ejecutado los procesos, el PID (*Process Identifier* = identificador del proceso) y la tasa de consumo de procesador (%CPU) para cada proceso, así como la memoria asignada RSS (*Resident Set Size* = Memoria RAM ocupada).

Esta visualización de los recursos y del consumo de procesador y de memoria permite entender alguna ralentización de la Raspberry Pi, debido a procesos que monopolizan la CPU.

c. LXTerminal

Esta herramienta es un emulador de terminal. Frente al sistema es como un terminal en modo texto y muestra los caracteres que recibe en una ventana. Es posible lanzar varias instancias de LXTerminal, es decir, mostrar varias ventanas de LXTerminal al mismo tiempo en la pantalla, cada una de ellas pudiendo ejecutar una o varias tareas diferentes. Sin embargo, salvo si el usuario cambia de identidad con el control `su` (consulte el capítulo Usar la línea de comandos), se conecta con su propia identidad en cada ventana.

Al abrirse, el tamaño del terminal se ajusta a los estándares informáticos de los terminales y muestra 24 líneas de 80 caracteres. La modificación de las dimensiones de la ventana se hace llevando el ratón a uno de los ángulos: el cursor representa un ángulo y una flecha. Pulse, mantenga pulsado y mueva el ratón. En el centro de la ventana del terminal aparece la dimensión de la ventana en forma líneas x columnas. Las dimensiones se actualizan cuando el ratón se desplaza. Cuando se alcanza la dimensión deseada, suelte el ratón.



La ventana del terminal tiene, en su parte superior, una barra de título que recuerda el nombre del usuario, seguido por una @, el nombre del ordenador seguido del signo dos-puntos (:), por último, el directorio de trabajo.

LXTerminal tiene una barra de menús debajo de la barra de título. Los menús disponibles son **Archivo**, **Edición**, **Pestañas** y **Ayuda**. Pulsando la tecla [Alt] se subrayan las letras del menú. Estas letras subrayadas indican los accesos directos de teclado para acceder a los menús: pulsando al mismo tiempo en las teclas [Alt] y **F** se abre el menú **Archivo**.

El menú **Archivo** permite abrir una nueva ventana de LXTerminal o una nueva pestaña en el terminal. También es posible cerrar la pestaña activa o salir completamente de LXTerminal.

El menú **Edición** ofrece la posibilidad de copiar y pegar. Permite acceder a la modificación de la apariencia del terminal: tipo y tamaño de letra utilizada, color del fondo y color de los caracteres, tipo de cursor, etc... La opción **Visualización** de la ventana de configuración ofrece la posibilidad de seleccionar la posición de las pestañas (en la parte superior, en la parte inferior, a la derecha o a la izquierda), la cantidad de líneas de texto memorizadas (1.000 por defecto) y la visualización de las diferentes barras y botones.

La opción **Avanzado** permite modificar la selección de los caracteres para la selección por palabra, el uso de las teclas [F10] y [Alt] **N**. Normalmente no hay que hacer ninguna modificación en esta pestaña.

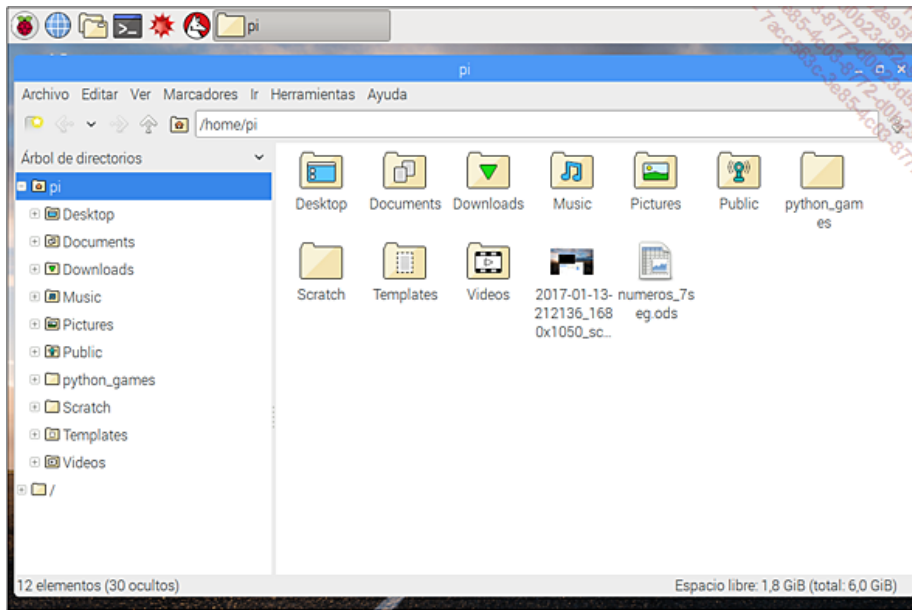
Para terminar, la opción **Shorcuts** define y permite modificar los accesos directos utilizados en LXTerminal.

El menú **Pestañas** permite dar un nombre a una pestaña. Esto puede ser práctico cuando hay abiertas varias pestañas. En este menú también es posible moverse entre las pestañas (a la izquierda o a la derecha) y mover las pestañas para organizar el espacio de trabajo.

El menú **Ayuda** solo tiene una opción: **Sobre**, que indica el número de versión de LXTerminal, un enlace a lxde.org, los nombres y direcciones de correo de los autores y del traductor, así como el texto de la licencia del terminal.

d. Administrador de archivos

La opción **Abrir del administrador de archivos** abre PCManFM (PCMan File Manager = administrador de archivos PCMan). Es el administrador que se ofrece por defecto en LXDE.



El uso de *PCManFM* es intuitivo y muy parecido al del resto de administradores de archivos. El menú **Edición - Preferencias** permite ajustar los parámetros de la aplicación en función de los gustos o necesidades del usuario. Por ejemplo, las imágenes presentes en la pantalla anterior aparecen en forma de vista preliminar si no superan los 2 MB (2017-01-03-212136_1680x1050_sc.png). Las imágenes que superan este tamaño, se representan con un icono estándar (numero_7seg.ods) para no ralentizar la visualización. Esta propiedad se ajusta en **Edición - Preferencias - Mostrar**.

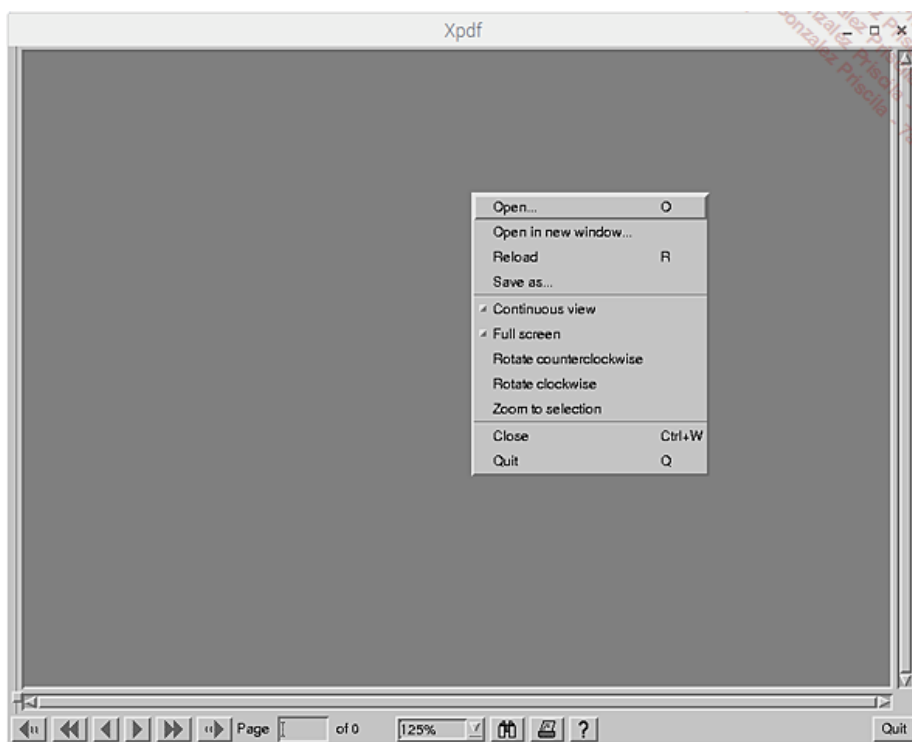
La visualización de los archivos cacheados se activa en el menú **Ver**, marcando la opción correspondiente.

Cuando una llave USB se conecta al sistema, un mensaje de advertencia indica que se ha reconocido. Se invita al usuario a abrir el periférico en el administrador de archivos.

El uso actual de *PCManFM* y la exploración de los menús bastan para adquirir un control rápido de esta herramienta.

e. Xpdf

Para abrir Xpdf pulse en **PDF Viewer** en el menú **Accesorios**. También es posible abrir directamente un documento PDF en Xpdf haciendo doble clic en el icono del documento.



El menú de Xpdf se abre haciendo clic derecho en la ventana. Permite abrir un documento (**Open**) y escribirlo en la tarjeta micro SD (**Save as...**). La opción **Continuous view** hace que aparezca el documento de manera continua, sin tener en cuenta las páginas. Las siguientes opciones permiten la visualización del documento completo en pantalla o su rotación si es necesario.



La captura de pantalla anterior muestra una revista The MagPi en formato PDF, que se muestra en Xpdf. La zona a la izquierda de la pantalla indica los números de las páginas, que permiten una navegación rápida por la revista. Esta información es interesante para los documentos que tienen un número de páginas importante.

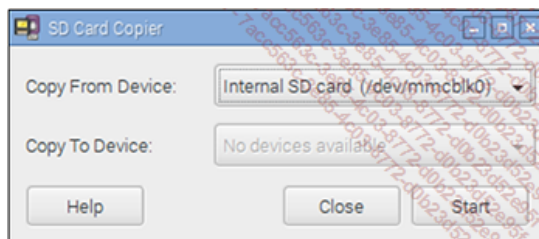
f. SD Card Copier

Una de las herramientas que faltaba entre las herramientas disponibles en Raspbian era un copiador de tarjetas SD. Para hacer una copia de seguridad de un sistema funcional era necesario pararlo, sacar la tarjeta SD y copiar esta tarjeta con un lector de tarjetas SD conectado a un PC o Mac. Con **SD Card Copier**, Raspbian se dota de una herramienta que permite realizar una copia de un sistema activo en otra tarjeta SD. Las siguientes capturas de pantalla le guiarán para realizar, paso a paso, la copia de su tarjeta SD.

Para copiar la tarjeta micro SD de su Raspberry, primero debe conectar un lector de tarjetas SD en uno de los puertos USB.

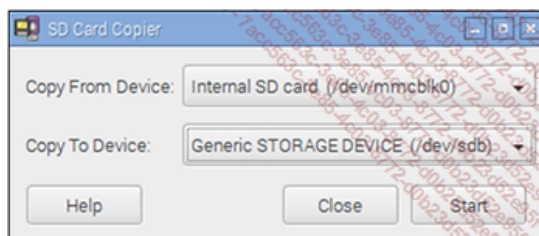
Inserte una tarjeta micro SD con una capacidad, al menos, igual a la que se encuentra en la Raspberry Pi.

En la sección **Accesorios** del menú de Raspbian, seleccione **Sd Card Copier**. Se abre la ventana de la herramienta.



La tarjeta desde la que se va a realizar la copia se indica en **Copy From Device**. Se trata de la tarjeta interna de la Raspberry Pi, la que ha servido para arrancar el sistema. La tarjeta de destino se indica en **Copy To Device**.

Pulse en la zona vacía (**No devices available**) y seleccione **Generic Storage Device (/dev/sdb)** para seleccionar la tarjeta insertada en el lector externo, como destino de la copia.

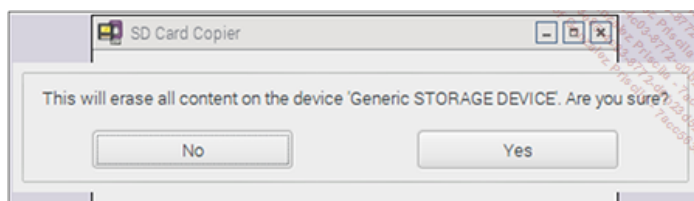


La tarjeta micro SD insertada en el lector externo de la tarjeta SD aparece ahora como destino de la copia.

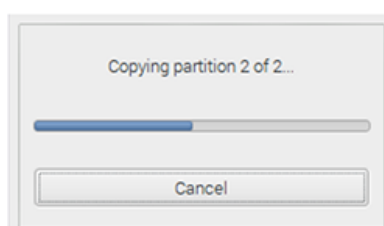
Pulse el botón **Start**.

El programa le informa de la peligrosidad de la operación. Todos los datos presentes en la tarjeta de destino se perderán. Si está seguro de que no se va a arrepentir de eliminar el contenido de la tarjeta SD:

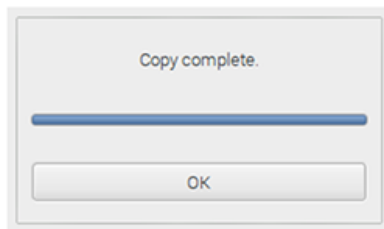
Pulse el botón **Yes** de la siguiente ventana.



El software prepara la tarjeta de destino creando las particiones y después copia el contenido de la tarjeta de la Raspberry Pi.



La copia de una tarjeta micro SD de 16 GB en una Raspberry Pi 3 dura unos diez minutos. Una barra permite seguir el progreso de la operación.



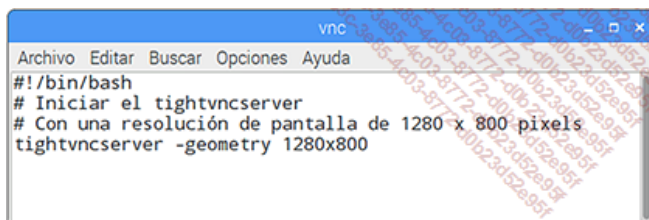
Cuando la operación termina, la ventana anterior indica el final de la copia.

Pulse el botón **OK**.

A continuación es posible retirar del lector de tarjetas la tarjeta micro SD en la que se ha copiado la tarjeta original. Pruébela insertándola en una Raspberry Pi, para comprobar que sigue funcionando. Puede utilizarla en otra Raspberry Pi o guardarla en un lugar seguro para poder volver a arrancar rápidamente una Raspberry Pi cuya tarjeta haya quedado corrupta.

g. Editor de texto (Text Editor)

LeafPad es la herramienta para editar texto en modo Gráfico. Esta herramienta, poco exigente en recursos, ofrece características interesantes.



La captura de pantalla anterior muestra el archivo vnc abierto en Text Editor. La barra superior indica el nombre del archivo que se está editando. El número de operaciones Anular/Repetir está limitado y el menú **Opciones** permite seleccionar el tipo de letra utilizada e indicar si la indentación automática y la numeración de línea deben estar activadas.

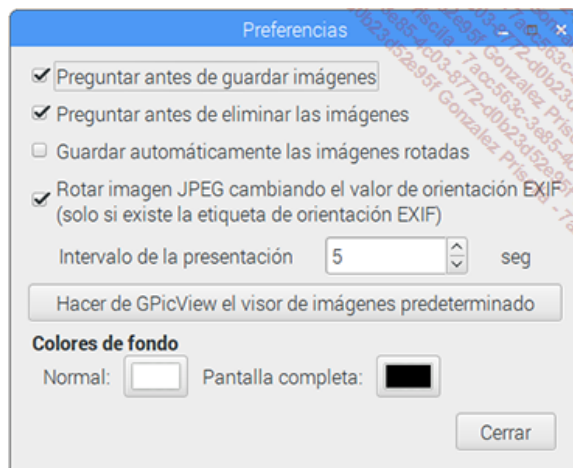
h. Visor de imágenes

GPicView es el programa seleccionado para mostrar las imágenes. Cuando arranca, muestra la primera imagen que encuentra en la carpeta actual. Su funcionamiento es parecido al del Visor de Windows.



La captura de pantalla anterior muestra la visualización en pantalla de una fotografía de 18 Mpx de tamaño. El nombre de la imagen aparece en la línea superior. Los botones de la línea inferior permiten moverse entre las imágenes, iniciar un pase de diapositivas, adaptar el tamaño de la imagen e incluso rotarla.

Un clic derecho en la ventana de GPicView, da acceso al menú de la aplicación. Encontramos las opciones propuestas por los botones de la línea inferior, que es posible ocultar.



El elemento **Preferencias** abre la siguiente ventana. Por ejemplo, es posible seleccionar el color de fondo, el intervalo de tiempo entre las diapositivas para el pase de diapositivas e incluso definir GPicView como visor de imágenes por defecto.

i. Xarchiver

Como la mayoría de programas presentes en Raspbian, se ha seleccionado el administrador de archivos comprimidos **Xarchiver** por su ligereza y pocas dependencias necesarias para su funcionamiento. Xarchiver se ha diseñado para ser rápido y fácil de utilizar. Su interfaz de usuario es

clara e intuitiva y no incluye las muchas opciones confusas o inútiles que hay en herramientas similares.

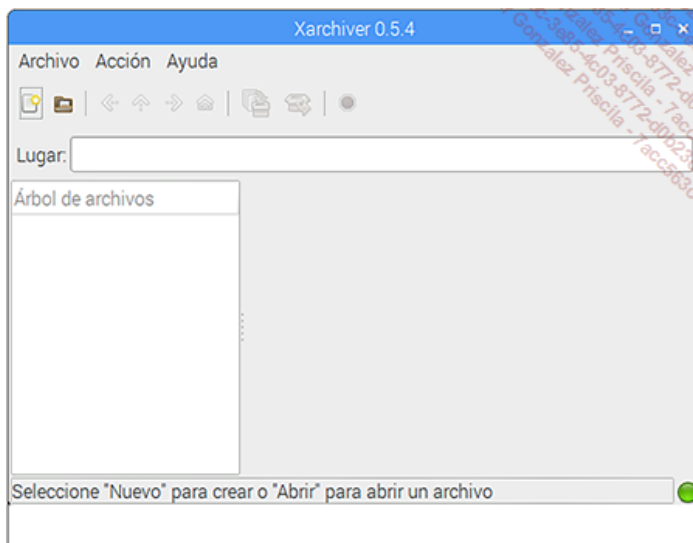
Xarchiver soporta los tipos de archivo 7-zip, arj, bzip2, gzip, rar, lha, lзма, lzop, deb, rpm, tar y zip. Gestiona todas las funciones habituales como la navegación por los archivos comprimidos, las contraseñas y el cifrado de los archivos. También puede gestionar archivos binarios auto-extraíbles en zip, 7-zip y rar. Por último, se muestra una barra de progreso cuando se usa Xarchiver por línea de comandos.


Esta herramienta permite agrupar y comprimir varios archivos en uno solo. Esto puede facilitar el almacenamiento de datos o la transferencia de archivos múltiples por e-mail, por ejemplo.

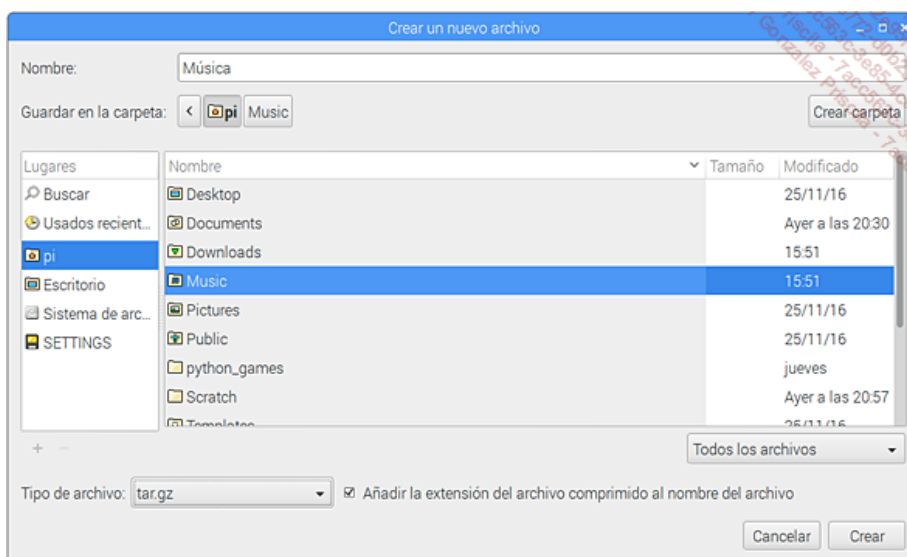
Creación de un archivo con Xarchiver

Pulse en la opción del menú **Accesorios** para abrir Xarchiver.

Aparece la siguiente ventana.



Cree un nuevo archivo pulsando en **Archivo - Nuevo** o pulsando en el icono **Crear un archivo**  situado en la parte superior izquierda de la ventana.

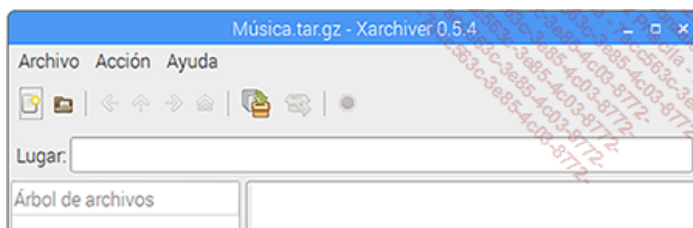


Rellene el nombre que desee dar al archivo en la línea superior de la ventana (aquí: **Música**).

En la lista desplegable situada en la parte inferior de la ventana, seleccione el tipo de archivo deseado (aquí: **tar.gz**).

Seleccione la carpeta en la que se guardará el archivo.

Pulse el botón **Crear** para validar su selección y crear el archivo comprimido.

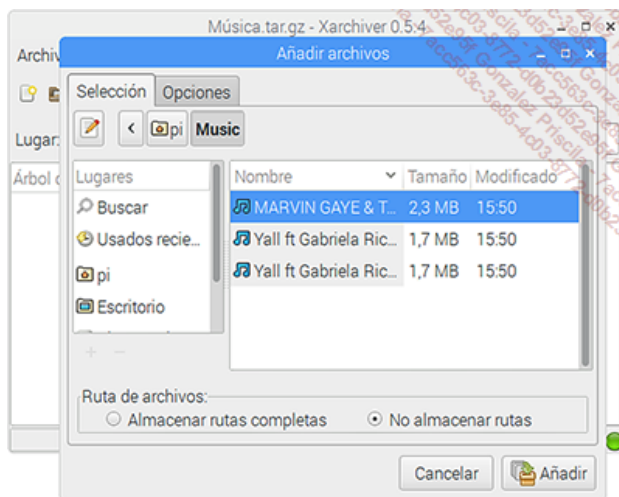


En la captura de pantalla anterior, el archivo se acaba de crear. El nombre del archivo Música.tar.gz se muestra en la barra de título, antes del nombre del programa.

Agregar archivos al archivo comprimido

Pulse en **Acción - Añadir** en la barra de menús.

Se abre la ventana **Añadir archivos**. Permite agregar los archivos que desee al archivo comprimido.

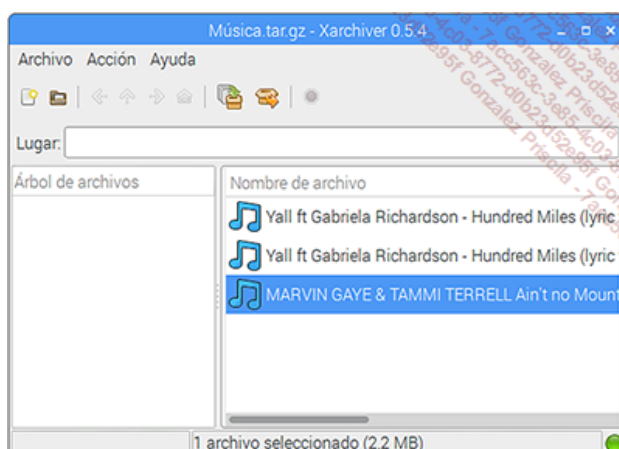


Seleccione la carpeta en el que se encuentran los archivos que desea agregar.

Seleccione los archivos pulsando en su nombre.

➤ Use las teclas [Shift] (selección individual) y [Ctrl] (selección de todos los archivos comprendidos entre dos líneas seleccionadas).

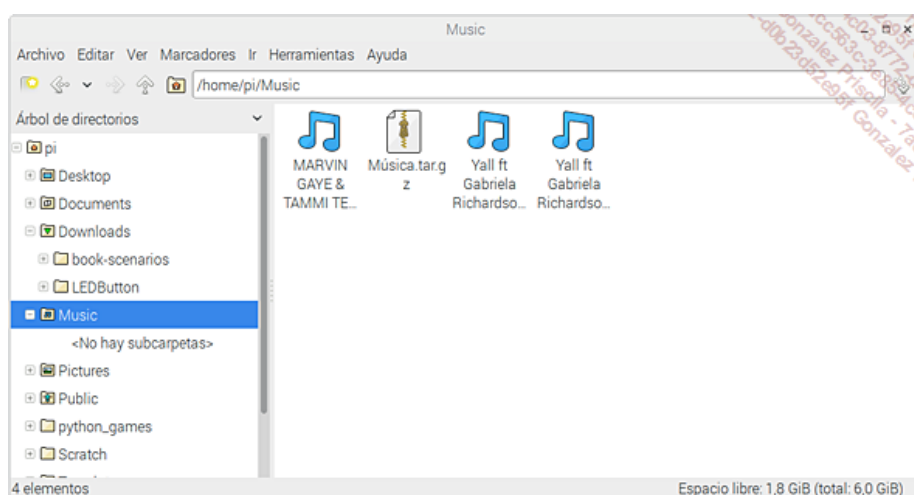
Cuando la selección ha terminado, pulse el botón **Añadir** en la parte inferior derecha de la ventana. Repita la operación si debe agregar otros archivos.



La luz que simula un LED (en la parte inferior derecha de la ventana anterior) parpadea durante la adición de archivos al archivo comprimido. Cuando se han añadido los archivos, aparecen en la parte central de la ventana.

Cierre la ventana pulsando el botón **Cerrar** representado por un aspa, en la parte superior derecha de la ventana o seleccionando **Archivo - Salir** en la barra de menús.

La siguiente ventana muestra el archivo creado en la carpeta /home/pi/Music.



Salga de la herramienta Xarchiver pulsando en el aspa situada en el ángulo superior derecho de la ventana.

Extracción de los archivos del archivo comprimido

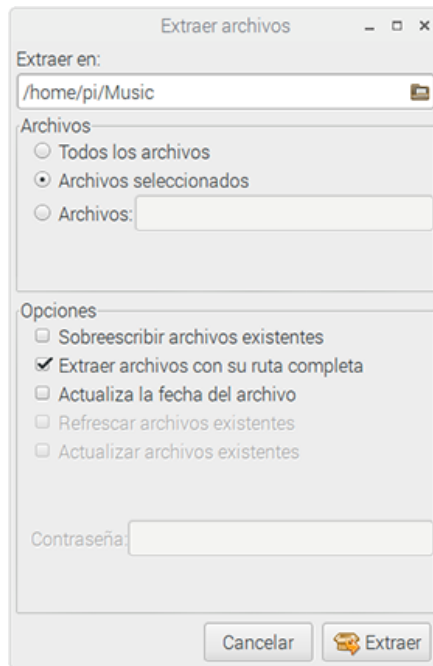
En la carpeta /home/pi/Music, pulse dos veces en el archivo comprimido para abrirlo.



Encontrará la ventana de Xarchiver.

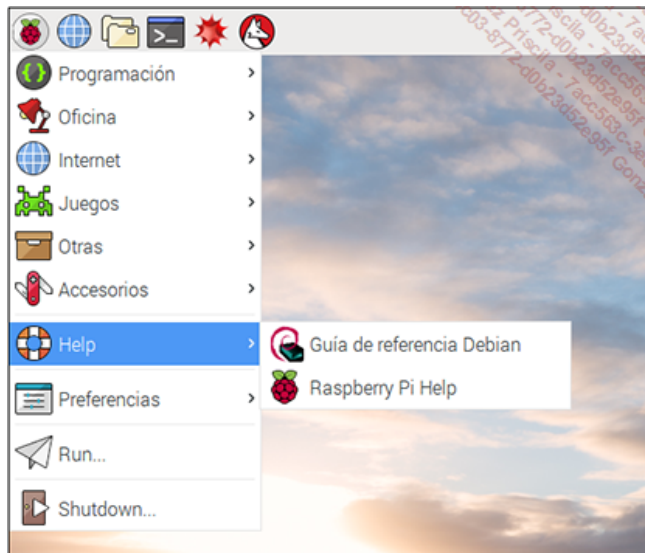
Seleccione el o los archivos que quiere extraer. Pulse en el icono de extracción, representado por una flecha que sale de una caja (el cuadro de diálogo muestra **Extraer archivos** cuando pasa el ratón sobre el icono).

Verá la siguiente ventana:



Ajuste los parámetros en función de sus necesidades y después pulse el botón **Extraer** para que Xarchiver descomprima los archivos seleccionados y los guarde en la ubicación deseada. Esta operación no modifica el archivo comprimido y siempre será posible repetir la operación en caso necesario.

6. Help



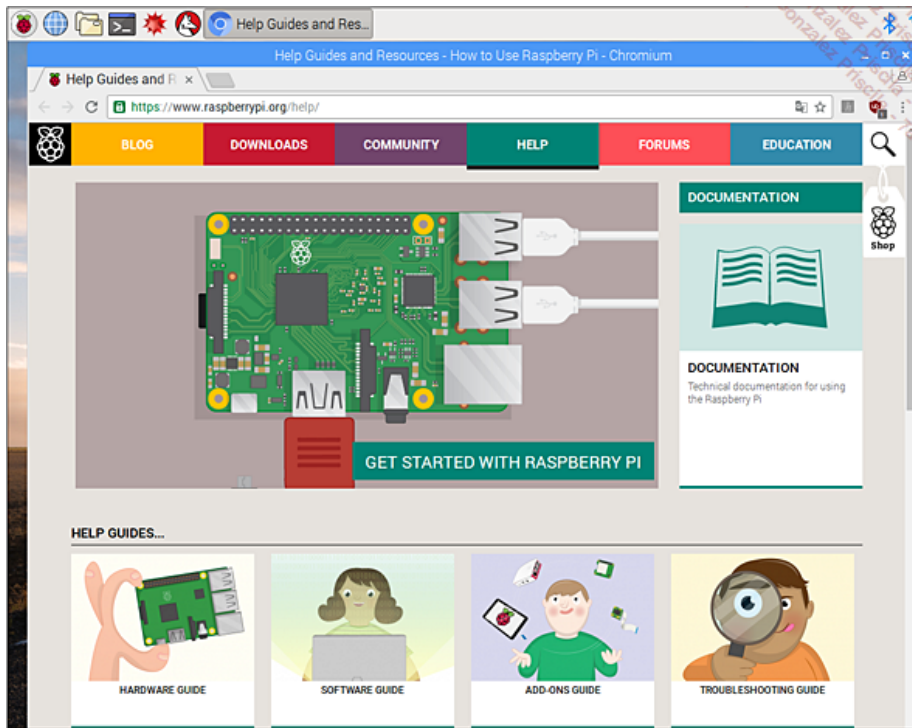
Esta sección está destinada a ofrecer ayuda (en inglés) a los usuarios de Raspbian.

a. Guía de referencia para Debian

La guía de referencia de Debian versión 2 de Osamu Aoki está disponible en su versión inglesa en la Raspberry Pi. La guía está disponible en español en la dirección www.debian.org/doc/manuals/debian-reference/. Es un documento que se distribuye con licencia GNU GPL, que cualquiera puede redistribuir y modificar a su gusto, bajo licencia.

Es una guía interesante para profundizar en el conocimiento de Debian y, por tanto, de Raspbian.

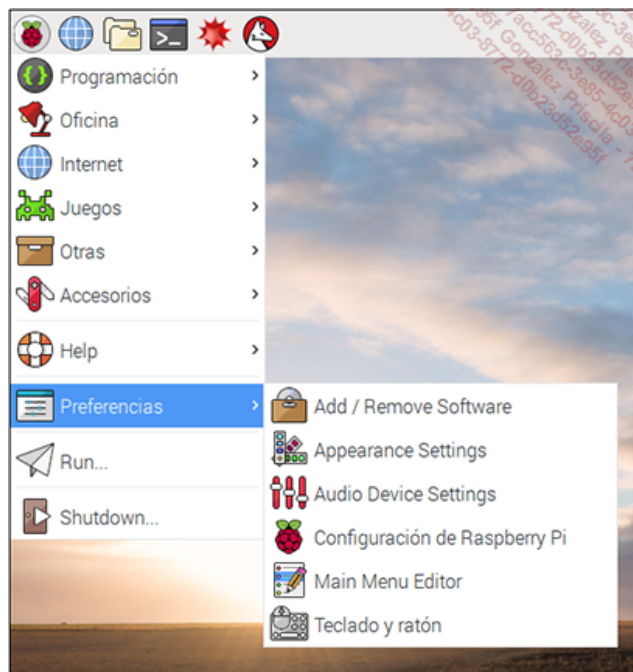
b. Raspberry Pi Help



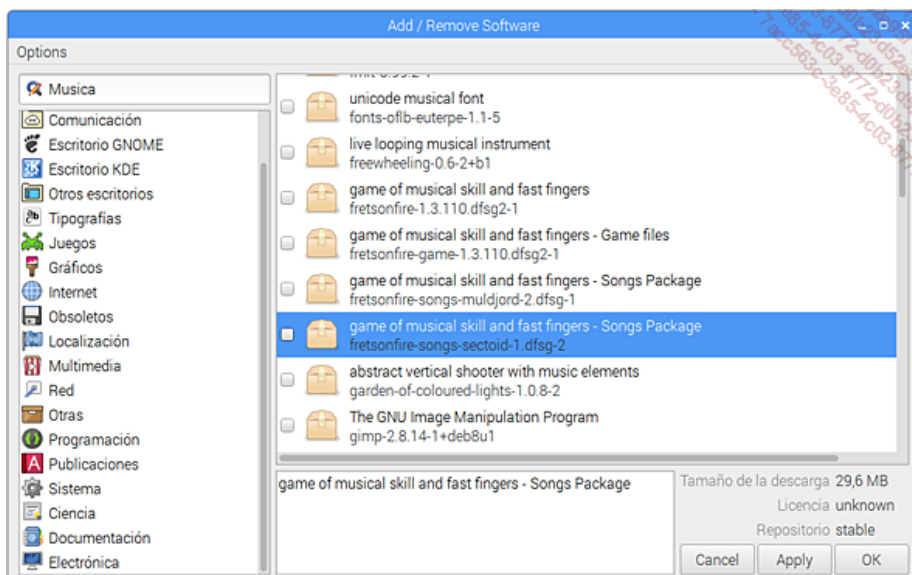
Un clic en esta línea abre el navegador web en la página <http://www.raspberrypi.org/help>. Hay disponible una cantidad de recursos, así como guías, tutoriales y vídeos. Una parte se está traduciendo al español en este momento.

7. Preferencias

El menú **Preferencias** permite al usuario adaptar el sistema operativo a sus necesidades o costumbres. Los parámetros accesibles permiten personalizar la presentación del escritorio, los programas instalados, el uso del teclado y del ratón, los dispositivos de audio, etc...



a. Add/Remove Software

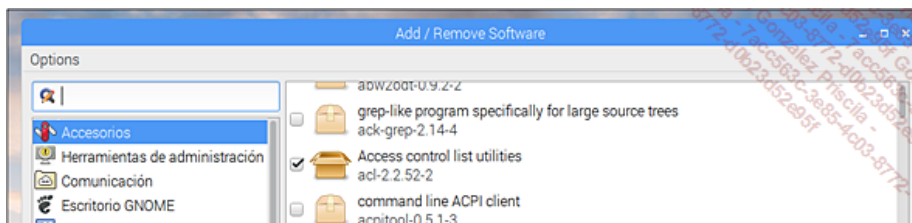


PiPackages es el administrador de programas utilizado. Se basa en PackageKit. Permite al usuario gestionar los paquetes instalados en su máquina. En la parte izquierda de la ventana, figura la lista de las secciones en las que se clasifican los paquetes disponibles. Encima de esta lista, una zona de texto permite introducir una palabra clave para buscar programas relacionados con esta palabra clave.

La parte central de la ventana muestra los programas disponibles en función de la búsqueda realizada o de la sección seleccionada. Haciendo clic en la línea de un programa aparece una breve descripción en la ventana inferior.

- El paquete Debian no es más que un archivo comprimido destinado a la instalación. Describe las relaciones con otros paquetes Debian (dependencias, conflictos, sugerencias). También ofrece scripts que permiten ejecutar controles durante las diferentes etapas de la vida del paquete (instalación, eliminación, actualización) (debian-handbook.info).

Eliminar un programa



En la captura de pantalla anterior, el paquete **Access control list utilities** se muestra a través de un icono que representa una caja abierta y la opción marcada al inicio de la línea. Por tanto, se instala en la Raspberry Pi.

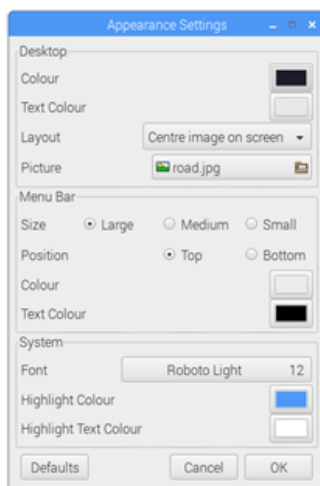
Para desinstalarlo, es necesario pulsar en la casilla para que desaparezca la marca. Pulse después en el botón **Apply** en la parte inferior derecha de la ventana. El programa solicita su contraseña para autorizarle a realizar la desinstalación.

Agregar un programa

Ayudándose de secciones o palabras clave en la zona de búsqueda, seleccione los programas que le interesen. Seleccione el que desee instalar pulsando en la opción situada al inicio de la línea. Pulse después el botón **Apply** en la parte inferior derecha de la ventana. El programa solicita su contraseña para autorizarle a realizar la instalación.

b. Appearance Settings

En esta parte del menú tiene acceso a los ajustes que permiten personalizar los diferentes elementos del escritorio: color, tipo de letra, imagen de fondo, etc...



Desktop

Determina el color del fondo, del texto, la imagen que se muestra, así como la manera en la que aparece: centrada, repetida, etc... Estos ajustes afectan al escritorio de Jessie.

Menu Bar

Ajusta los parámetros de la barra de menús: su tamaño, su posición (en la parte superior o inferior de la pantalla), el color de la barra y el de los caracteres que se muestran.

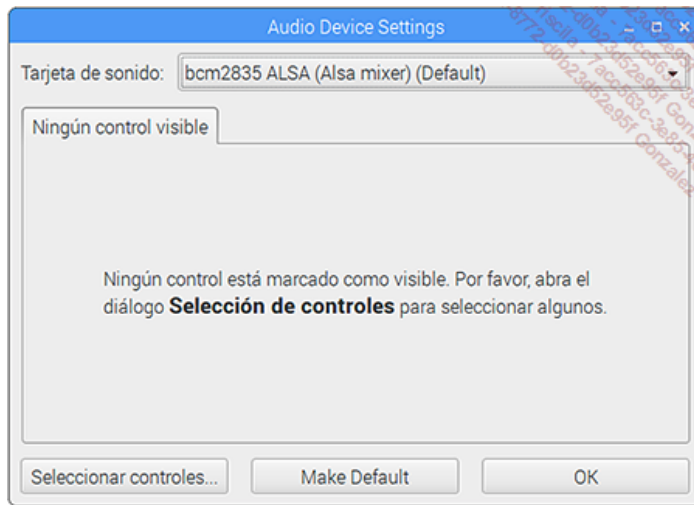
System

Permite seleccionar el tipo de letra utilizado, el color del texto en modo normal y en modo destacado.

El botón **OK** valida las selecciones que haya realizado. El botón **Defaults** vuelve a poner los valores originales en los argumentos.

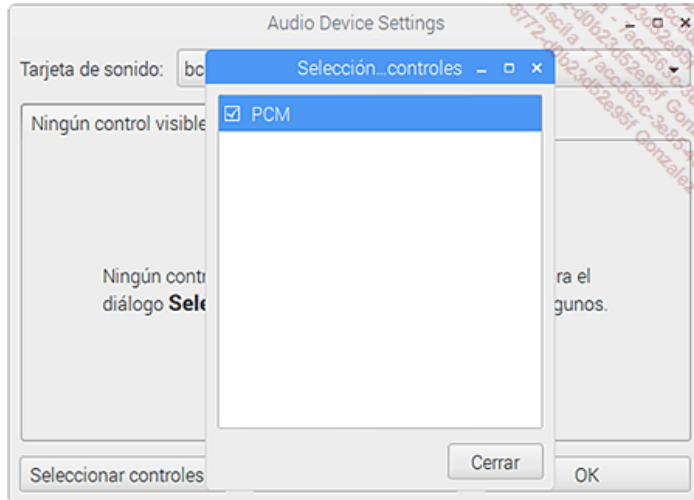
c. Audio Device Settings

Esta sección da acceso a los ajustes del equipamiento de audio conectado a la Raspberry Pi.

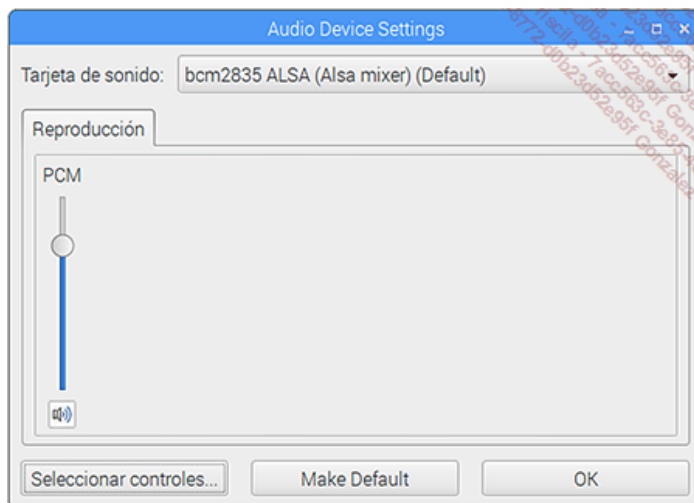


Por defecto, la visualización es mínima y no se ve ningún control (potenciómetro o interruptor).

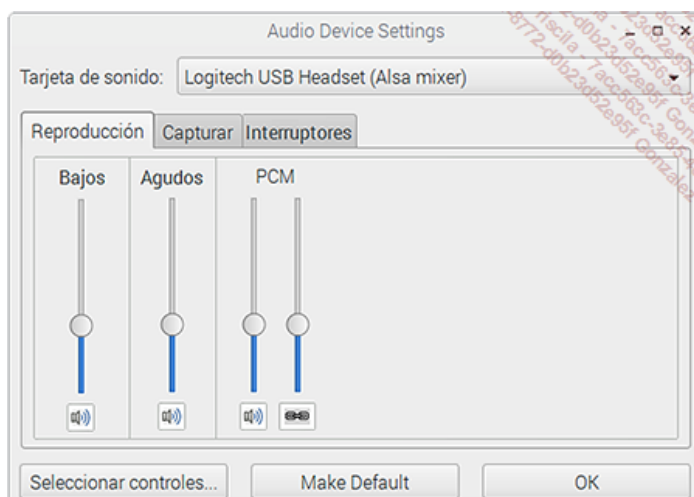
Pulse el botón **Seleccionar controles** para activar el o los controles correspondientes al periférico seleccionado.



En la ventana de selección de los controles, marque la casilla del único control disponible, **PCM**. Después pulse el botón **Cerrar** para validar su elección.



Los auriculares USB Logitech, equipados con un DSP (*Digital Signal Processor* = Procesador de señal digital), están conectados a la Raspberry utilizada para realizar estas capturas de pantalla. Este equipamiento ofrece posibilidades de ajuste mucho más extensas que la simple salida analógica de la Raspberry Pi.



Los auriculares Logitech se han seleccionado en la lista desplegable **Tarjeta de sonido** situada en la parte superior de la ventana. Después de un clic en el botón **Seleccionar controles**, todos los controles disponibles se han seleccionado. La captura de pantalla anterior muestra tres pestañas.

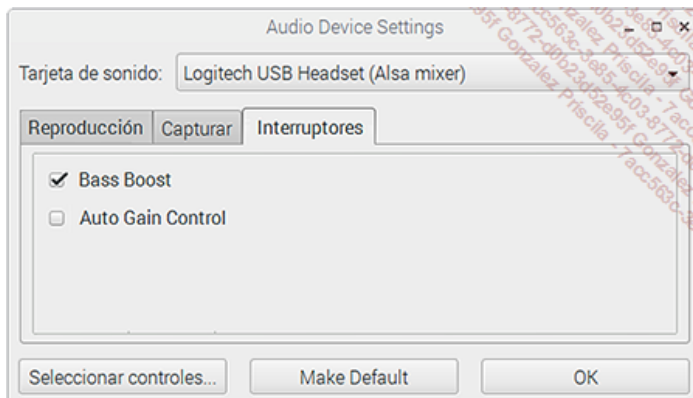
Reproducción

Muestra cuatro potenciómetros asociados a los ajustes de los graves y agudos para los dos primeros. Las demás marcas PCM (*Pulse Code Modulation* = Modulación por impulsos codificados) ajustan el nivel de la salida. La cadena situada debajo del potenciómetro derecho indica que los dos potenciómetros están unidos, el ajuste de uno implica al otro. Haciendo clic en la cadena, se desacoplan los potenciómetros, que vuelven a ser autónomos. A continuación es posible ajustar por separado el volumen para el canal derecho y el canal izquierdo.

Capturar

Esta pestaña muestra un potenciómetro de ajuste del nivel de entrada del micro que equipa a los auriculares.

Interruptores



La última pestaña ofrece la posibilidad de activar los dos interruptores de software de los que están provistos los auriculares. Uno activa el aumento de los bajos, el otro el control automático de la ganancia.

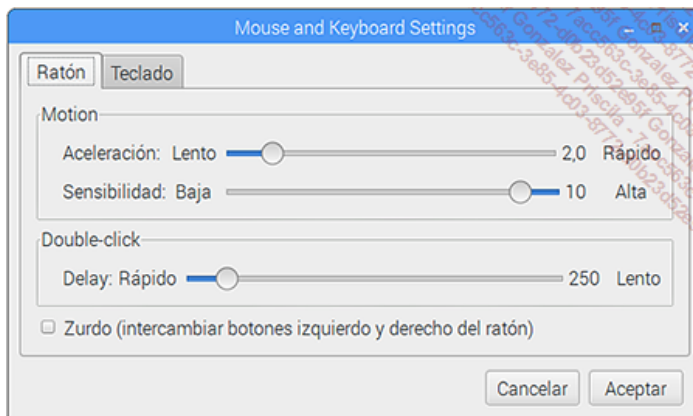
Conclusión

Aunque este programa puede parecer simple y poco útil para un uso básico de la Raspberry Pi, este último ejemplo muestra que permite explotar las características de equipamiento más avanzado que unos simples auriculares cuando se conecta a la Raspberry Pi.

d. Teclado y ratón

Este programa le permite adaptar los ajustes del ratón y del teclado a sus costumbres.

La pestaña Ratón



El grupo **Motion** de la pestaña **Ratón** le da la posibilidad de ajustar la aceleración y sensibilidad del ratón. La aceleración disminuye la distancia a recorrer con el ratón para mover el cursor desde un punto a otro de la pantalla. Cuanto más rápido sea el desplazamiento, más se reduce la distancia que hay que recorrer. La sensibilidad del ratón corresponde a la relación entre la distancia real recorrida por el ratón y el movimiento del cursor en la pantalla. Si la sensibilidad es baja, la precisión es mejor, porque un movimiento pequeño del ratón posiciona de manera precisa el cursor en la pantalla. Los dos argumentos se combinan y un movimiento rápido del ratón (mucha aceleración) lleva rápidamente el cursor a la zona deseada de la pantalla, y un movimiento lento del ratón permite posicionar de manera exacta el cursor en el lugar seleccionado.

El cerebro humano se adapta muy rápidamente a este modo de funcionamiento, que facilita el uso de los movimientos rápidos y muy lentos del ratón.

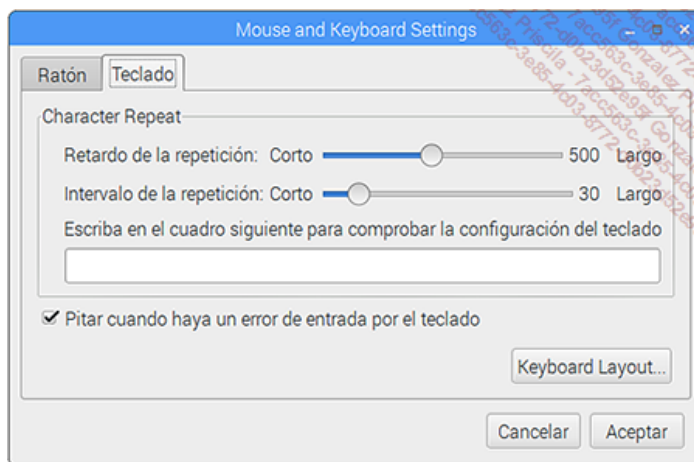
El grupo **Double-click** ajusta la sensibilidad del ratón cuando se hace doble clic. Si está acostumbrado al ratón, opte por una velocidad rápida.

Existe una opción que permite intercambiar los botones derecho e izquierdo del ratón para un mejor uso por parte de los zurdos.

Una vez ajustados todos los parámetros, pulse el botón **Aceptar** en la parte inferior derecha de la ventana para validar su selección.

- No dude en volver a estos ajustes si es necesario para encontrarse más a gusto con el ratón. La comodidad de uso de un ordenador forma parte de los parámetros importantes que hay que adaptar a sus costumbres.

La pestaña Teclado



En esta pestaña están los parámetros del teclado. El grupo **Character Repeat** permite ajustar el tiempo antes de que la pulsación se repita cuando mantiene el dedo pulsando la tecla (**Retardo de la repetición**). También es posible modificar la velocidad a la que la tecla se repetirá (**Intervalo de la repetición**). Una zona de texto permite probar estos ajustes.

Con el botón **Keyboard Layout** accede a la lista de los teclados nacionales disponibles. Puede seleccionar el que mejor se adapte al alfabeto de su país y probar su ajuste en la zona de texto.

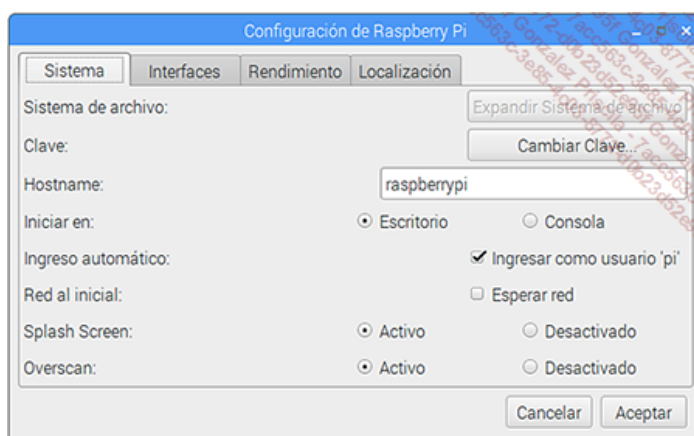
Cuando esté satisfecho con el resultado, pulse el botón **Aceptar** para guardar las modificaciones.

e. Configuración de la Raspberry Pi

Si ha leído este capítulo desde el inicio, habrá tenido la ocasión de utilizar esta herramienta de configuración para localizar su distribución (huso horario, idioma, tipo de teclado, etc...). La configuración de Raspbian en la Raspberry Pi es más larga que esta localización.

- Algunas modificaciones necesitan reiniciar la Raspberry Pi para que se tengan en cuenta. Un mensaje le advierte de ello. Puede optar por ignorar este aviso. En ese caso, las modificaciones solo serán efectivas cuando se produzca el siguiente inicio.

Pestaña Sistema



Los principales parámetros del sistema están accesibles a través de esta pestaña. Una imagen descargada ocupa un espacio reducido en la tarjeta micro SD. Esto limita el tiempo de descarga del archivo comprimido. El botón **Expandir Sistema de archivo** ejecuta una herramienta encargada de recuperar toda el espacio disponible en la tarjeta micro SD, en beneficio del sistema de archivos.

Cambiar Clave permite modificar fácilmente la contraseña del usuario por defecto (*pi*). Hay que rellenar dos veces la nueva contraseña para evitar errores y validar su modificación para que se tenga en cuenta.

La zona **Hostname** permite cambiar el nombre de la máquina. Esto es particularmente útil cuando varias Raspberry Pi están conectadas a la red. Es interesante dar a cada una un nombre diferente para facilitar el acceso a los diferentes sistemas.

Con las versiones actuales de Raspbian Jessie, el arranque del sistema abre automáticamente el escritorio. La opción **Iniciar en** permite seleccionar entre un arranque en modo gráfico o en modo texto (**Consola**).

Durante el inicio del sistema, es el usuario pi el que se conecta por defecto. Es posible no permitir esta conexión automática desmarcando la opción **Ingreso automático**.

Normalmente el proceso de arranque de Raspbian Jessie es el más rápido posible. Algunas veces el sistema ha terminado de arrancar antes de que la conexión de red esté operativa. Marcando la opción **Red al inicial**, la secuencia de arranque no continuará hasta que se haya establecido la conexión de red. Por defecto, esta opción no está seleccionada. Marque esta opción únicamente si tiene problemas con aplicaciones que necesitan la red durante el arranque del sistema. La validación de esta opción se traduce, en general, en un aumento del tiempo de arranque del sistema.

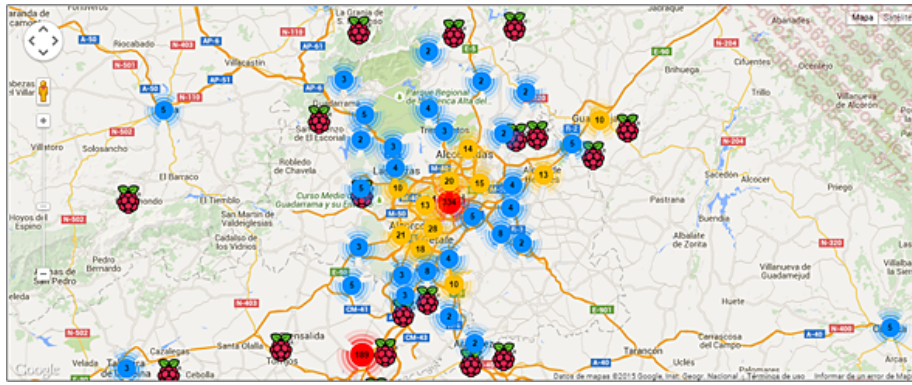
La opción **Overscan** permite adaptar el tamaño del vídeo que se muestra en la pantalla. Si la parte útil del vídeo está rodeada por bandas negras en las que no se muestra nada o si faltan porciones de vídeo porque están fuera de la pantalla, es necesario intervenir en el overscan.

En algunos monitores o televisores es posible desconectar el overscan en el menú del dispositivo. Normalmente esto es suficiente para que el vídeo ocupe la totalidad de la pantalla. Algunas veces, el overscan se llama HD Size, pantalla completa o 1:1.

Si aparecen bandas negras que rodean la parte útil de la imagen, marque la opción **Desactivado**. Durante el siguiente reinicio, la imagen ocupará toda la pantalla.

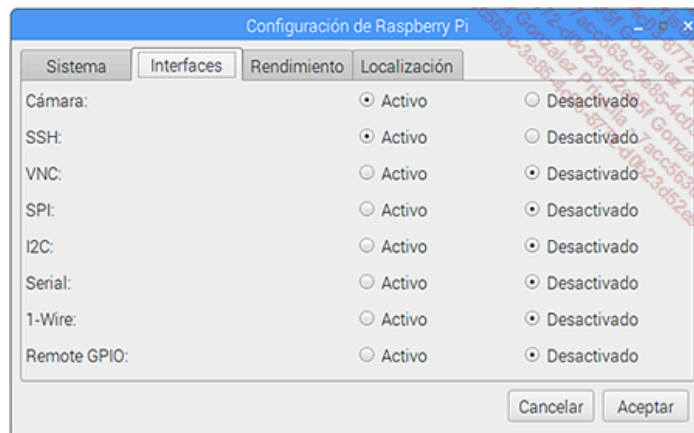
Si es necesario ajustar de manera más fina el tamaño del vídeo porque sigue siendo más pequeño que la pantalla, o al contrario, desborda el tamaño, es necesario modificar el archivo `/boot/config.txt` y configurar manualmente el tamaño de los bordes. En este caso, deje marcada la opción **Activado**.

Rastrack es un mapa mundial basado en Google Maps, creada por Ryan Walmsley. Este mapa representa el reparto mundial de las Raspberry Pi. Cada propietario de Raspberry Pi puede inscribirse y su Raspberry Pi se ubicará en el mapa.



Este reparto solo refleja la posición de los usuarios inscritos. En junio de 2016, solo se han registrado 112.000 Raspberry Pi, de los más de ocho millones de unidades vendidas.

Pestaña Interfaces

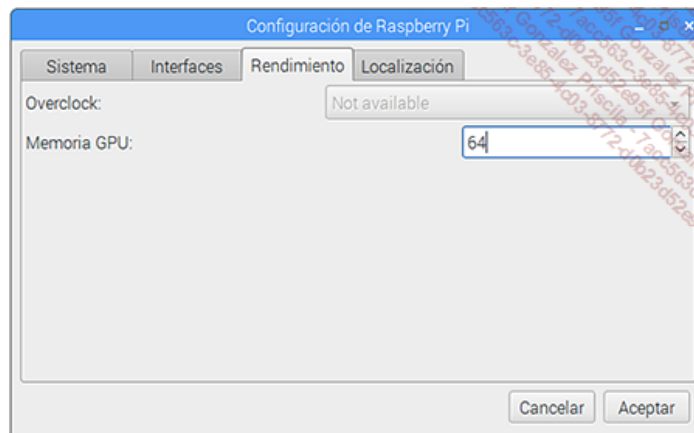


La pestaña **Interfaces** permite activar o desactivar fácilmente las interfaces cuando lo necesite.

Con PIXEL, aparece la opción **VNC**. Permite tener el control en modo remoto de la Raspberry Pi (vaya al capítulo Conectarse a la Raspberry Pi en remoto).

Basta con marcar la opción correspondiente y después pulsar el botón **Aceptar**. Algunas opciones necesitan un reinicio de la Raspberry Pi para aplicarse. En este caso, una ventana le advertirá.

Pestaña Rendimiento



El ajuste del rendimiento de la Raspberry Pi se realiza a través de esta ventana. Puede modificar el **Overclock** cuando esté disponible (según el modelo de Raspberry Pi utilizado). En la Raspberry Pi 2, por ejemplo, puede seleccionar un funcionamiento a 900 MHz o a 1000 MHz. En la Raspberry Pi 3 (imagen anterior), el **Overclock** está **Not available**.

El overlock, también llamado overlocking, es la operación que consiste en aumentar la frecuencia de funcionamiento de un circuito electrónico (microprocesador, memoria, etc...), y algunas veces también la tensión aplicada al circuito (sobre-voltaje). Para más detalles, consulte el capítulo Descripción Técnica: La CPU de la Raspberry Pi Zero - Overclocking.

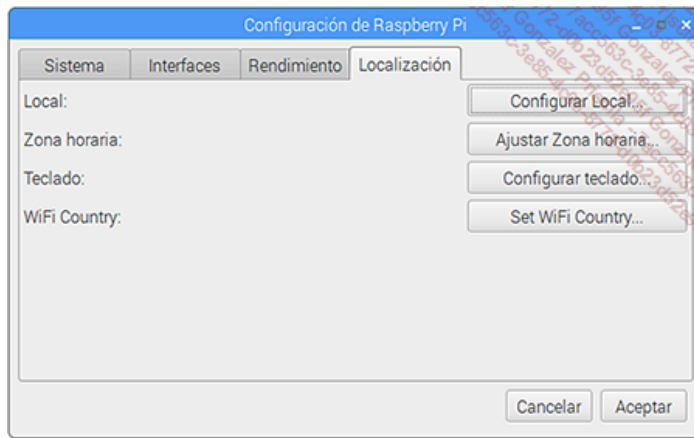
- Este aumento de la velocidad y/o de la tensión se traduce generalmente en un calentamiento más importante del procesador, un envejecimiento más rápido del componente y una disminución del tiempo de vida de la Raspberry Pi. Por tanto, se debe utilizar con conocimiento de causa.

Memoria GPU ajusta el reparto de la memoria entre el microprocesador y el procesador gráfico (*GPU = Graphical Processor Unit = Procesador gráfico*). El valor por defecto es de 64 MB. Para la Raspberry Pi 3 puede variar de 16 MB a 880 MB, en pasos de 16 MB.

Si pone en marcha un centro multimedia o usa la cámara de la Raspberry Pi, prevea más memoria para la GPU (al menos 128 MB). Para un servidor en modo texto sin pantalla conectada, será suficiente con 64 MB de memoria GPU (incluso menos), pero requerirá más recursos de memoria para la CPU. Reservar más memoria para la GPU mejora el rendimiento del modo gráfico, pero limita el número de aplicaciones que se pueden cargar al mismo tiempo en memoria y puede reducir el rendimiento de otros programas. La opción correcta es, ante todo, tener un compromiso.

Localización

Esta pestaña da acceso a los parámetros de la localización del sistema operativo.



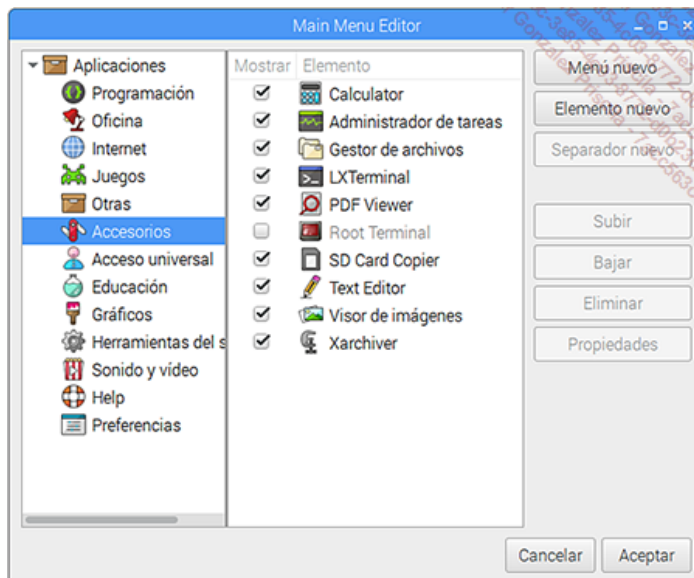
La pestaña **Localización** tiene cuatro ajustes:

- **Local:** define el idioma que el sistema debe utilizar.
- **Zona horaria:** indica el huso horario del usuario.
- **Teclado:** especifica el tipo de teclado que se debe utilizar (QWERTY, AZERTY...).
- **WiFi Country:** permite cumplir la legislación del país respecto al Wi-Fi.

El uso de estos ajustes se ha descrito al inicio de este capítulo. Vaya a esta sección si desea añadir modificaciones a la localización del sistema.

f. Main Menu Editor

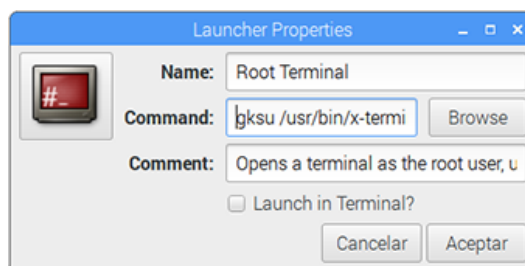
Esta herramienta permite personalizar el menú principal del escritorio de Raspbian.



La parte izquierda de la ventana presenta el contenido del menú principal. En la captura de pantalla anterior aparece seleccionada la sección **Accesorios**. La parte central de la ventana muestra el contenido de esta sección, es decir, los elementos que se muestran cuando pulsa en **Menú - Accesorios**. La opción situada junto a **Root Terminal** no está marcada y el nombre del accesorio está deshabilitado. Este accesorio está presente en el menú, pero no aparece cuando accede a la sección **Accesorios**. El elemento presente en la ventana central se llama lanzador. Se encarga de ejecutar la línea de comando que tiene asociada. Para mostrar las propiedades del lanzador **Root Terminal**:

Seleccione la línea pulsando en **Root Terminal**.

Pulse el botón **Propiedades** en la parte derecha de la ventana.



La ventana anterior muestra las características del lanzador. Encontramos el nombre del lanzador, la línea de comando que ejecuta, así como un comentario que describe lo que hace.

Los botones de la parte derecha de la ventana permiten agregar un **Nuevo menú** o un **Nuevo elemento** al menú. **Nuevo separador** añade una línea gris entre dos elementos de menú, para generar bloques de elementos que desee agrupar. Para agregar un separador:

Pulse en el último elemento del bloque a crear.

Pulse el botón **Nuevo separador**.

El separador (representado por tres líneas) se sitúa debajo el elemento seleccionado.

Para eliminar un elemento de menú o un separador, pulse en el elemento a eliminar para seleccionarlo y a continuación pulse el botón **Eliminar**.

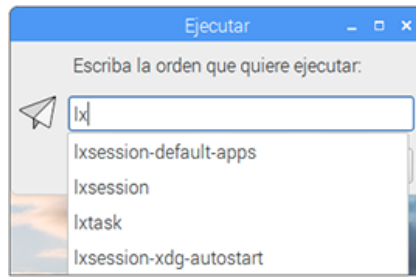
Los botones **Subir** y **Bajar** mueven un elemento seleccionado hacia arriba o hacia abajo para organizar el menú según le convenga.

Conclusión

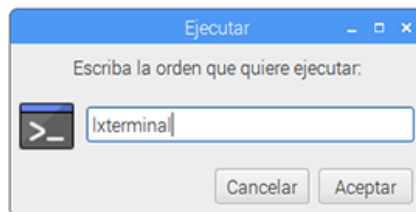
Esta herramienta le permitirá adaptar el menú principal de Raspbian para responder a sus necesidades. La configuración por defecto responde a las restricciones de la mayor parte de los usuarios. La modificación de la línea de comando de un lanzador puede impedir el uso del accesorio correspondiente. Solo intervenga sobre estos lanzadores si domina el tema. Si desea hacer pruebas, realice una copia de seguridad de la tarjeta SD antes de empezar las modificaciones.

8. Run

La sección **Run** del menú principal del escritorio de Raspbian abre una ventana que tiene una zona de introducción de texto. Puede indicar un comando en esta zona de texto.



Indique un control y haga una pausa. La captura de pantalla anterior muestra la entrada de un comando que empieza por **lx**. Después de un instante, la ventana **Ejecutar** muestra la lista de todos los comandos disponibles que empiezan por las letras **lx**. Pulse en el comando que desee ejecutar.

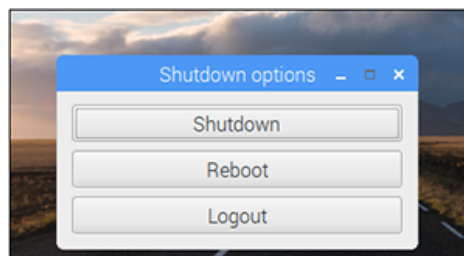


Aquí, un clic en **lxterminal** completa automáticamente el comando y el icono que representa unas ruedas dentadas reemplaza al icono del terminal. Pulse el botón **Aceptar** para ejecutar el comando. Se abre un nuevo terminal en el escritorio.

9. Shutdown

Para detener un sistema operativo, se debe hacer bajo las condiciones correctas. Linux no es una excepción a esta regla. El último elemento del menú principal da acceso a los mecanismos que permiten salir de la sesión actual o parar el sistema.

- La Raspberry Pi no tiene interruptor manual o automático. La parada del sistema no interrumpe la alimentación eléctrica de la tarjeta. Es responsabilidad del usuario hacer lo necesario para interrumpir la alimentación.



Shutdown detiene el sistema propiamente dicho. Después de la parada, la tensión no se interrumpe. Para reiniciar el sistema, el usuario debe desenchufar y volver a enchufar la alimentación de la Raspberry Pi. También es posible pulsar el botón **Reset** si la Raspberry Pi lo tiene. El botón debe estar conectado a los bornes Run, situados entre el extremo del conector GPIO y los puertos USB.

Reboot detiene el sistema propiamente dicho, pero lo reinicia inmediatamente.

Logout termina la sesión actual. Una ventana permite al usuario introducir su login y contraseña para iniciar una nueva sesión.

Iconos a la derecha de la barra de tareas

Después de una presentación resumida del papel que juegan estos iconos al inicio de este capítulo, a continuación se detalla el uso de cada uno de ellos.



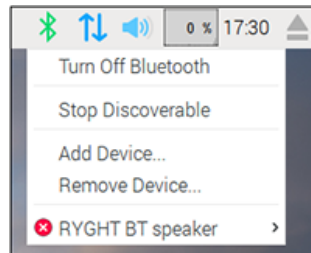
- 1 - Bluetooth
- 2 - Red
- 3 - Audio
- 4- Carga CPU
- 5 - Reloj
- 6 - Expulsión

1. Bluetooth

a. El Bluetooth en la Raspberry Pi

Después de la aparición del Bluetooth en la Raspberry Pi 3, la primera versión de Raspbian Jessie disponible obligaba a instalar manualmente el Bluetooth. La salida de la versión 2016-05-10 ha puesto fin a esta obligación. Raspbian Jessie integra todos los componentes para la gestión del Bluetooth y el icono 1 da acceso a estas herramientas.

Bluetooth permite una unión hasta como máximo diez metros de la máquina. En general, los dos dispositivos están separados por algunos metros solamente (tres o cuatro metros), para garantizar una buena calidad de las transmisiones.



Cuando pulsa el icono Bluetooth del menú se abre la gestión de periféricos Bluetooth. En la captura de pantalla anterior aparece, en la parte inferior del menú, un altavoz Bluetooth RYGH BT utilizado en la Raspberry Pi 3. El aspa que figura junto al icono de este altavoz indica que no está disponible (apagado o demasiado lejos de la Raspberry Pi). La primera línea del menú interrumpe la alimentación del periférico Bluetooth. Esto permite ahorrar energía o utilizar otro periférico Bluetooth, en lugar de aquel que integra la tarjeta.

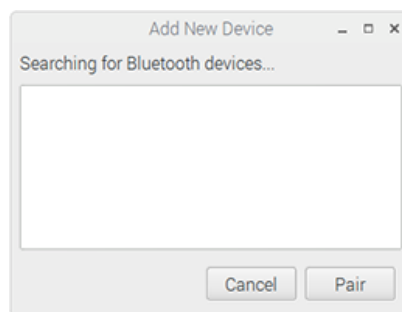
b. Hacer visible a la Raspberry Pi

Si pulsa en **Make Discoverable** (Hacer visible) la Raspberry Pi se podrá detectar por un dispositivo que use Bluetooth. La línea del menú se convierte en **Stop Discoverable** (Hacer invisible). Si pulsa en esta línea, la Raspberry Pi dejará de estar visible a través de Bluetooth.

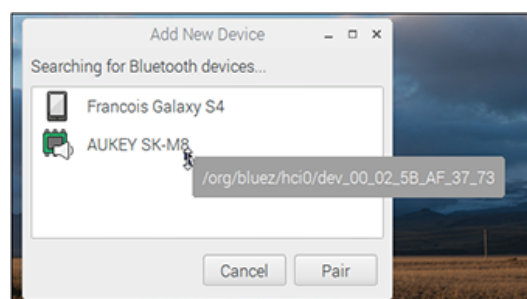
Para eliminar un dispositivo de la lista, pulse en **Remove Device**. La Raspberry Pi muestra una ventana que contiene los dispositivos conocidos. Seleccione el que desee eliminar y pulse a continuación en el botón **Remove**. El dispositivo se retirará de la lista.

c. Agregar un dispositivo Bluetooth

Un clic en la opción **Add Device** del menú abre una ventana y lanza la búsqueda de dispositivos Bluetooth desconocidos por la Raspberry Pi. Antes de lanzar esta búsqueda, enchufe el dispositivo que desea conectar a la Raspberry Pi. Pásele a modo emparejamiento (lea el aviso del dispositivo para saber la manera de proceder, propia de cada equipamiento).



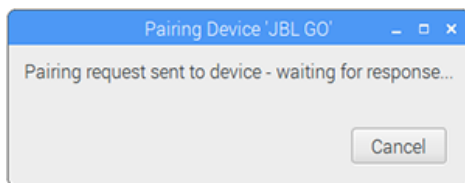
La búsqueda de los dispositivos puede durar algunos segundos. Como resultado de la búsqueda, los dispositivos detectados por la Raspberry Pi aparecen en la ventana.



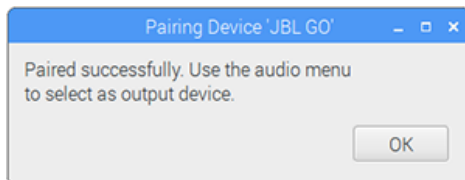
La Raspberry ha detectado un altavoz AUKEY SK-M8 y una tableta Galaxy Tab S4. Si mueve el cursor del ratón por encima de alguno de los dispositivos, se muestra un cuadro de diálogo con la dirección MAC (dirección de hardware) de dicho dispositivo.

Por razones de seguridad, el intercambio de datos entre dos dispositivos unidos a través de Bluetooth necesita un emparejamiento entre

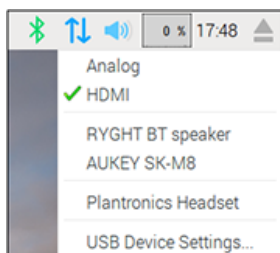
ambos dispositivos. En la Raspberry Pi el emparejamiento se realiza seleccionando el dispositivo a emparejar (aquí el altavoz AUKEY) y pulsando el botón **Pair**. Los dos dispositivos intercambian un código y quedan unidos. En algunos casos, el código es libre, otras veces es necesario rellenar manualmente un código.



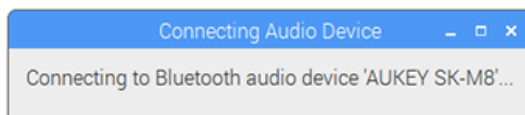
Se abre la ventana anterior, indicando que la operación de emparejamiento se está realizando y que la Raspberry Pi espera la respuesta del dispositivo correspondiente.



El dispositivo se ha emparejado con la Raspberry Pi. Esta operación se realiza una única vez. A partir de ese momento, la Raspberry Pi reconocerá automáticamente este altavoz cuando se encuentre cerca. Ambos dispositivos se reconocerán y estarán listos para conectarse. Pulse ahora con el botón derecho del ratón sobre el icono que representa un altavoz.



Puede seleccionar el destino del sonido en el menú que se abre. Es posible dirigir el sonido a la salida Jack 3,5 mm (Analog), a la pantalla (HDMI) o incluso a uno de los periféricos Bluetooth emparejados. Pulse en el dispositivo que desea utilizar, aquí AUKEY SK-M8. En la imagen anterior se ve que también hay un altavoz RYGHT BT y unos auriculares Plantronics emparejados con la Raspberry Pi.



Se abre una ventana durante algunos instantes. Indica que se está procediendo a la conexión con el dispositivo. Cuando la conexión se establece, la ventana desaparece. En el altavoz AUKEY la conexión se traduce por la emisión de un bip y por el parpadeo lento de un LED azul. Consulte las instrucciones de su dispositivo para saber cómo indica que la conexión se ha realizado.

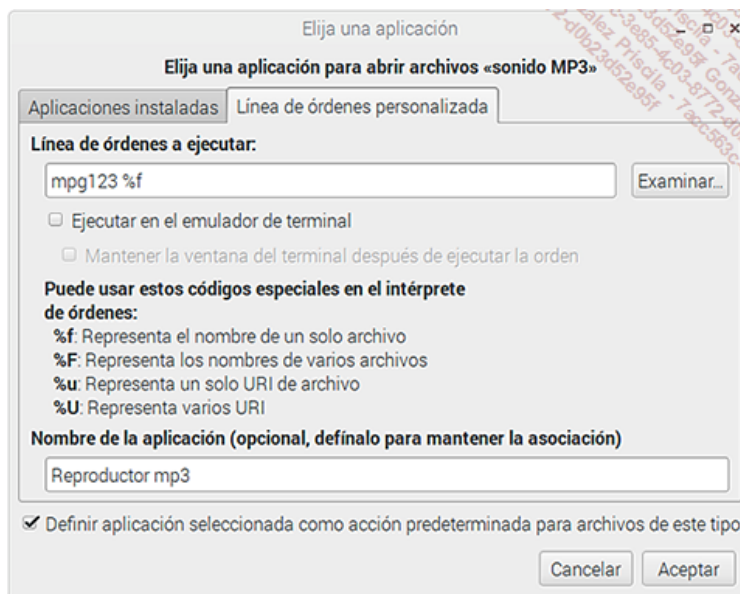
d. Distribuir la música en un altavoz Bluetooth

La difusión de música a un dispositivo Bluetooth necesita el uso de un software capaz de reproducir un archivo musical (mp3, por ejemplo) y enviar la señal al altavoz.

- Algunas aplicaciones envían directamente el sonido al equipamiento sin utilizar los modos normales de difusión del sonido. Si no transmite ningún sonido al altavoz o a los auriculares Bluetooth conectados a al Raspberry Pi, consulte la documentación de la aplicación para comprobar esta información.

Para enviar música al altavoz Bluetooth he seleccionado mpg123 (puede instalarlo a partir de **Menú - Preferencias - Add/Remove Software**).

Pulse dos veces sobre algún fragmento de música almacenada en la tarjeta micro SD de su Raspberry Pi.



Abra la pestaña **Línea de órdenes personalizada** e introduzca:

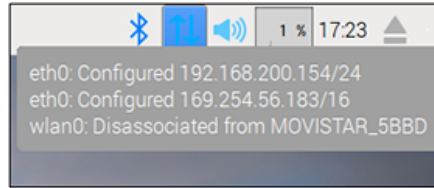
```
mpg123 %f
```

%f indica el archivo en el que ha pulsado. Es el que se reproducirá y enviará al altavoz Bluetooth.

Si desea que siempre sea mpg123 la aplicación que lea los archivos mp3 marque la opción **Definir aplicación seleccionada como acción predeterminada para archivos de este tipo** y después pulse el botón **Aceptar** en la parte inferior derecha de la ventana. La música suena por el altavoz Bluetooth.

2. Red

El segundo icono situado a la derecha de la barra de tareas representa dos flechas que simbolizan la emisión y la recepción de los datos por la red. Cuando la Raspberry Pi no está conectada a la red, las flechas aparecen con un aspa roja. Cuando mueve el ratón sobre este icono, se muestra el cuadro de diálogo **eth0: Link is down** (el enlace está interrumpido). Si la Raspberry Pi está conectada a la red por un cable y hay una dirección IP (configurada manualmente o proporcionada por DHCP) el cuadro de diálogo muestra la dirección y aparece la máscara como en la siguiente captura de pantalla.



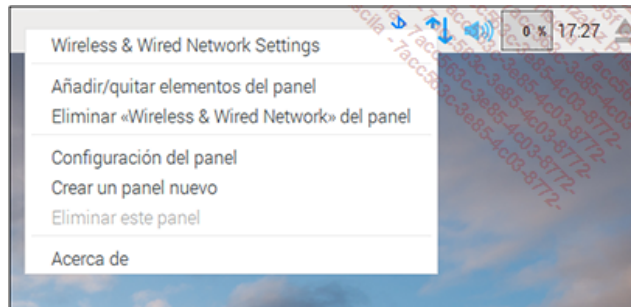
a. Visualización de la configuración de red

eth0 representa la interfaz de red alámbrica (toma RJ45). La red está configurada y la interfaz tiene la dirección IP 192.168.200.154. La máscara de sub-red es 24, lo que significa que 24 bits de la máscara de sub-red valen 1 (255.255.255.0).

wlan0 representa la interfaz Wi-Fi de la Raspberry Pi. Utilizada anteriormente, esta interfaz se había conectado al punto de acceso Wi-Fi llamado MOVISTAR_5BBD. Esta interfaz está desconectada (**Disassociated**) del punto de acceso.

b. Configuración de la red

Haga un clic derecho en el icono de red. Se muestra un menú (siguiente captura de pantalla). La primera línea del menú menciona: **Wireless & Wired Networks Settings** (Configuración de redes y Wi-Fi). Haga clic con el botón izquierdo en esta línea.



La ventana que se abre da acceso a la configuración de la red. Seleccione la interfaz que desea configurar. Pulse en la lista desplegable situada a la derecha del botón **interface**. Seleccione de la lista la interfaz a configurar: **eth0** o **wlan0**.



Configuración automática

Si su red está equipada con DHCP (*Dynamic Host Configuration Protocol* = Protocolo de configuración dinámica del host), como ocurre por defecto en los paquetes ofrecidos por proveedores de acceso a Internet, marque la opción **Automatically configure empty options** (Configuración automática de las opciones vacías). Los parámetros correspondientes:

- a la dirección IP de la máquina,
- al router (puerta de enlace),
- al DNS (*Dynamic Name Server* = Servidor dinámico de nombres),
- al dominio de búsqueda DNS (**DNS search**),

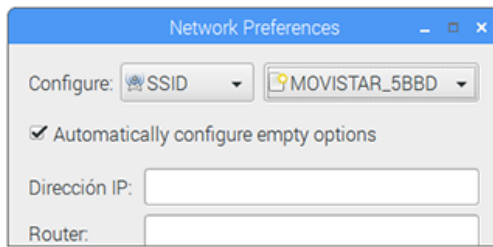
se enviarán automáticamente a la Raspberry Pi.

Configuración manual

También puede decidir configurar manualmente su hardware de red. En este caso, no marque la opción **Automatically configure empty options**. Indique manualmente la información necesaria en las zonas correspondientes. Utilice esta configuración únicamente si domina los parámetros a configurar. En caso de funcionamiento incorrecto, desmarque la opción para volver al modo automático.

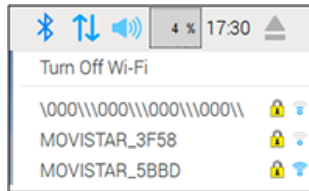
Configuración de acceso al Wi-Fi

Haga clic en el icono que representa la red. Pulse en la primera línea del menú que se abre (**Wireless & Wired Networks Settings**). Se abre la siguiente ventana.



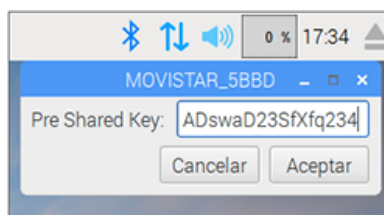
En la sección **Configure**, seleccione **SSID** (*Service Set Identifier*=Nombre de la red Wi-Fi) en la lista desplegable más a la izquierda. Una lista de los puntos de acceso Wi-Fi accesibles aparece cuando abre la lista desplegable de la derecha. Seleccione el punto de acceso que le conviene (aquí **MOVISTAR_5BBD**). Valide su opción pulsando el botón **Cerrar**.

Un clic izquierdo en el icono de red muestra la lista de los puntos de acceso disponibles.

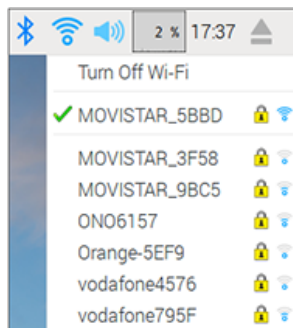


La primera línea del menú permite detener la interfaz Wi-Fi para ahorrar energía o utilizar otro periférico diferente al que se integra en la tarjeta.

Pulse en el nombre del punto de acceso. Se abre una ventana de configuración (siguiente imagen).



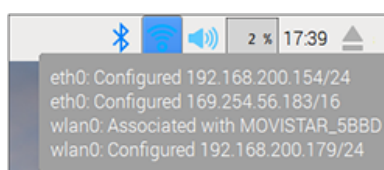
En la zona **Pre Shared Key**, introduzca la llave de acceso al paquete o a su router Wi-Fi. Pulse el botón **Aceptar**. Pasados algunos segundos, el icono que representa dos ordenadores se sustituye por un icono que representa al Wi-Fi. El número de barras activas da una idea de la potencia con la que se recibe el punto de acceso.



Un clic izquierdo en el icono Wi-Fi muestra los identificadores (SSID). La red Wi-Fi a la que está conectada la Raspberry Pi está identificada con una marca verde.

➤ Para desconectar la red Wi-Fi del punto de acceso al que está conectada haga clic con el botón izquierdo en el icono Wi-Fi, y después en el nombre de la red a desconectar. Escriba caracteres al azar (al menos ocho) en la zona de texto **Pre Shared Key**. Pulse el botón **Aceptar**. El icono Wi-Fi desaparece, se sustituye por el icono de red que muestra dos ordenadores.

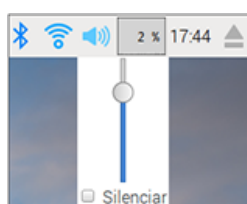
Para conocer la configuración de red de su Raspberry Pi, pase el ratón por el icono red/Wi-Fi. Pasados unos instantes se abre un cuadro de diálogo con la información de configuración de la red.



En la captura de pantalla anterior, la interfaz alámbrica eth0 de la Raspberry Pi está conectada y tiene la dirección IP 192.168.200.154. La interfaz Wi-Fi wlan0 está asociada al punto de acceso llamado MOVISTAR_5BBD, que tiene la dirección IP 192.168.200.179.

3. Volumen sonoro

El altavoz abre una ventana que tiene un potenciómetro. Pulse en el círculo que representa el botón de ajuste para moverse con el ratón.

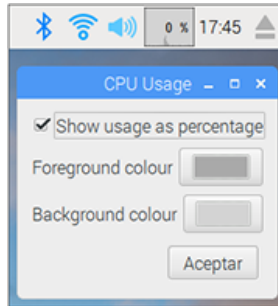


El valor del volumen se ajusta de 0 a 100. Para poner el sonido al mínimo, puede llevar el ajuste del potenciómetro a la parte inferior de la zona de ajuste o pulsar en la opción **Silenciar**. Si pulsa en la opción **Silenciar**, el cambio del volumen al mínimo es temporal. Haciendo clic de nuevo en esta opción, la marca desaparece y el volumen vuelve a su nivel anterior.

➤ Para conocer el nivel al que el está ajustado el volumen, lleve el ratón al icono que representa un altavoz. Pasados unos instantes aparece un cuadro de diálogo. Contiene la información necesaria en forma: **Control del volumen 78**.

4. Indicador de actividad

El medidor (cuarto icono) es el indicador de actividad de la CPU. Una barra de color da una estimación instantánea de la carga de la CPU entre 0 y 100 %. El movimiento permite seguir la evolución a lo largo del tiempo. La ventana mostrada representa el último minuto de actividad del procesador.



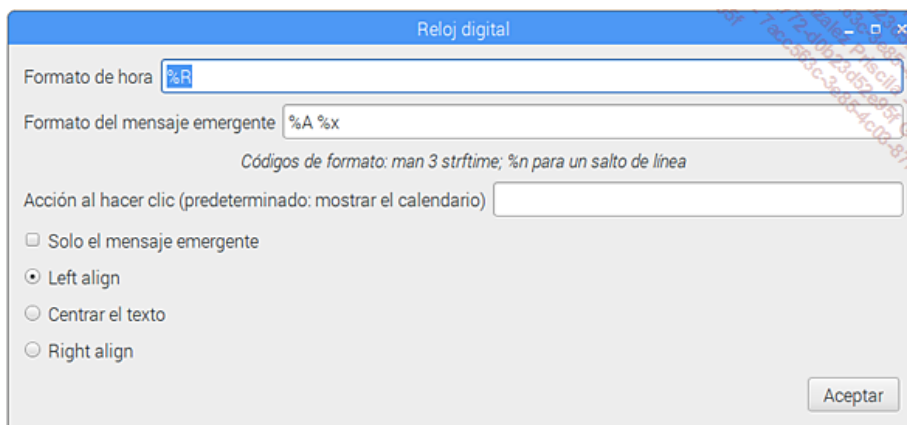
Para ajustar el color del monitor haga clic con el botón derecho del ratón sobre el monitor de actividad. Pulse en la primera línea del menú que se abre: **CPU Usage**. Aparece la pantalla anterior. Si está marcada la opción **Show usage as percentage**, permite visualizar el porcentaje de uso del microprocesador en el monitor de actividad. **Foreground color** (color del primer plano) y **Background color** (color de fondo) le permiten ajustar los colores del monitor según sus preferencias.

5. Reloj

El reloj digital indica la hora, como cualquier reloj. Al pasar el ratón por el reloj, después de un breve periodo de tiempo se lanza la visualización de un tooltip que contiene la fecha.

Haciendo clic con el botón izquierdo del ratón sobre el reloj se muestra un calendario que se vuelve a cerrar cuando se hace un nuevo clic con el botón izquierdo del ratón en el reloj.

Un clic con el botón derecho del ratón da acceso a la configuración del reloj digital (siguiente captura de pantalla). Es posible cambiar el formato del reloj o del tooltip, cambiar la alineación del texto, modificar la acción del clic que muestra el calendario por defecto, etc.



El formato de visualización del reloj aquí es **%R**. Equivale a %H:%M. Existen muchas opciones. Se aplican tanto al reloj como a los tooltips. Aquí se muestra un extracto (<http://php.net/manual/es/function.strftime.php>):

%A nombre completo del día de la semana (sábado)

%a nombre abreviado del día de la semana (sáb.)

%B nombre completo del mes (diciembre)

%b nombre abreviado del mes (dic.)

%c fecha y hora en formato nacional (en función de la configuración local)

%d día del mes en formato decimal (01 a 31)

%e día del mes formato decimal (1-31); las cifras únicas están precedidas de un espacio

%R es equivalente a %H:%M. Es el valor original del reloj

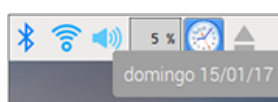
%x se sustituye por la fecha, que se muestra en formato nacional (en función de la configuración local).

Existen más de 40 parámetros. Vaya al enlace indicado para tener todos los detalles en español. Por ejemplo %A %e %B %Y mostrará sábado 17 de diciembre de 2016 cuando se pase el cursor del ratón sobre el reloj.

➤ El reloj de la Raspberry Pi no se basa en un reloj RTC que funciona con la batería. La Raspberry Pi se conecta a un servidor de tiempo (NTP para *Network Time Protocol* = protocolo de actualización horaria en la red). Si no se ha establecido ninguna conexión a Internet, la hora que se indica no guarda relación con la hora real.

Acción con un clic: si esta zona está vacía, se muestra el calendario cuando se hace clic en el reloj. Si desea ejecutar un comando cuando hace clic en el reloj, introduzca este comando en la zona de texto a la derecha de **Acción con un clic**. Para probar, introduzca **pcmanfm** y valide. Cuando pulse en el reloj, el administrador de archivos se abrirá.

Solo tooltips: esta opción tiene una denominación ambigua. Si está marcada, la visualización del reloj digital se cambia por un reloj analógico, pero está parado y no indica la hora. Para leer la hora, habría que pasar el ratón por el reloj para hacer aparecer el tooltip.



Aquí el tooltip muestra el resultado del formato %A %x, la configuración por defecto.

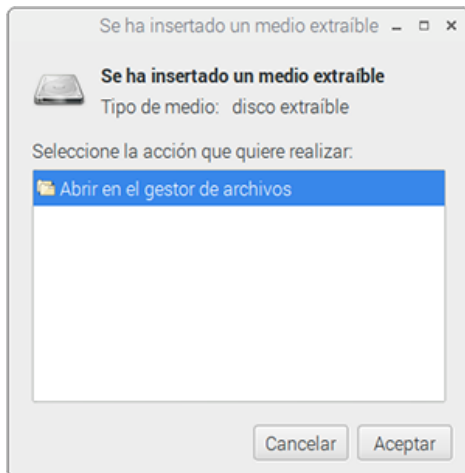
Las opciones de alineación del texto (izquierda/centrada/derecha), no tienen efecto visible.

6. Expulsar periférico USB

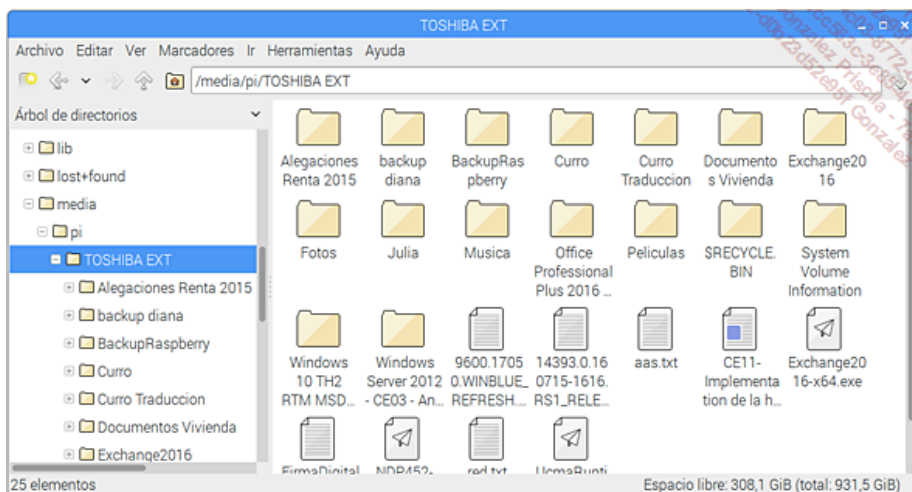
El icono de ejecución (último icono) se ha añadido al escritorio para facilitar la retirada de llaves USB en condiciones correctas, sin poner en peligro los datos presentes en el periférico.

a. Inserción de periféricos USB

Durante la inserción de un periférico USB, una llave por ejemplo, esta se detecta por el sistema y se monta automáticamente. Se abre una ventana para señalar la conexión de un periférico (siguiente captura de pantalla).



En este caso, la llave USB se ha identificado correctamente y aparece en el sistema de archivos en `/dev/sda1`. En una Raspberry Pi configurada por defecto, si se introduce la contraseña `raspberrypi` en la zona **Password**, seguida de un clic en el botón **Aceptar**, se monta la llave USB en el sistema de archivos. Es posible seleccionar otro usuario en la lista desplegable **Identity**. Será necesario informar la contraseña de este usuario. A partir de ese momento, el acceso al contenido del periférico de almacenamiento es posible.



A continuación, el sistema permite abrir el periférico desde el administrador de archivos (como se ve a la izquierda en la imagen anterior). Pulse el botón **Aceptar** para acceder al contenido de la llave USB en el administrador de archivos. La llave USB (aquí FM_16GB) está accesible a través de `/media/pi/TOSHIBA_EXT` (como se ve a la derecha en la imagen anterior).

b. Eyección de un periférico USB

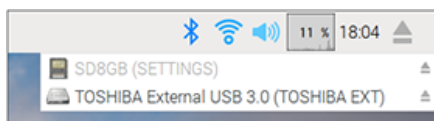
Cuando un periférico de almacenamiento USB (llave o disco) se utiliza en Linux, se puede utilizar por uno o varios procesos (programas que funcionan en memoria). Linux no escribe directamente los datos en un periférico. Se almacenan temporalmente en una memoria intermedia llamada *buffer*, a la espera de que el sistema termine de transferirlos al periférico USB.



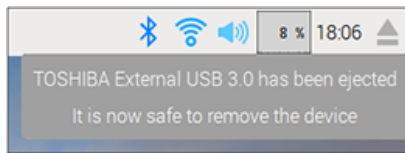
Si se retira bruscamente un periférico USB, se desconecta del sistema y se desenchufa antes de que el sistema haya podido terminar de transferir los datos contenidos en el *buffer* a la llave USB. Algunos archivos pueden permanecer abiertos y las tablas que permiten al sistema gestionar los archivos, no están actualizadas. Algunas veces, esto puede provocar un deterioro de los datos de uno o varios archivos. En los casos más graves, el periférico USB deja de ser accesible y se debe formatear.

- En algunos casos, la herramienta `fsck` es capaz de reparar los daños. La llave no se debe desmontar en el sistema de archivos durante su uso.

Para desconectar sin dañar un periférico de almacenamiento USB conectado a la Raspberry Pi, pulse en el icono **Eyección**. Si no hay ningún periférico conectado, se muestra el mensaje **No ejecutable devices**. Se abre una lista (siguiente captura de pantalla) si hay conectado algún periférico. Seleccione de la lista el periférico a ejecutar y pulse en su nombre.



El sistema va a desmontar "correctamente" el periférico USB.



Si desea utilizar la llave USB de nuevo, habría que desconectarla físicamente de la Raspberry Pi, y volverla a conectar para que se detecte de nuevo.

Conclusión

Con la Raspberry Pi 3 y Jessie, la Fundación ofrece un conjunto coherente. La presencia de un navegador web que se pueda utilizar de manera fluida, de la suite de escritorio LibreOffice y de un cliente para gestionar el correo electrónico permiten considerar el uso de la Raspberry Pi como puesto de oficina.

Las escuelas y los espacios públicos digitales, también pueden apostar por esta máquina de bajo precio para equipar las clases o las áreas de recepción con ordenadores de escritorio.

Sin embargo, la domótica o los drones que usan una Raspberry Pi no necesitan la interfaz gráfica para funcionar. Ocurre lo mismo con los servidores. Para estas aplicaciones, el uso de la línea de comandos evita consumir potencia de cálculo del procesador para gestionar la visualización gráfica.

Es el objetivo del siguiente capítulo.

Introducción

Después de haber preparado la tarjeta micro SD, conectado todos los cables en la Raspberry Pi, enchufarla a la corriente y configurado el sistema, la pantalla muestra sus primeras palabras.

Pero ¿por qué usar este modo texto arcaico, cuando es posible trabajar con maravillosas pantallas gráficas con el ratón? Varias razones explican esta elección.

El modo texto está disponible en todas las distribuciones

Históricamente, el modo texto es el modo nativo de Linux. Las interfaces gráficas son un recubrimiento. En todas las distribuciones, el modo texto está accesible a través de las consolas o *tty* (*TeLeType* = máquina de escribir controlada en modo remoto). Las combinaciones de teclas desde la [Ctrl][Alt][F1] a la [Ctrl][Alt][F6] dan acceso a seis consolas, en las que de los diferentes usuarios pueden ejecutar tareas diferentes. La combinación [Ctrl][Alt][F7] permite pasar al modo gráfico (si existe).

El modo texto consume pocos recursos

En todo ordenador, el microprocesador comparte su tiempo entre las diferentes tareas que debe ejecutar. La creación y gestión de un conjunto de ventanas, así como de su contenido, consume potencia de cálculo. En ciertas máquinas, algunas veces hay una decena de ventanas abiertas correspondientes a varias aplicaciones, un navegador web en el que hay pestañas abiertas... Todo esto se gestiona con un único procesador y cada tarea ocupa espacio en memoria.

En modo texto, una página solo representa algunos kilobytes y el microprocesador solo envía texto a la consola cuando es necesario. La ocupación de memoria se reduce al mínimo estricto. El procesador se puede dedicar por completo a la ejecución del trabajo que le corresponde.

El procesador ARM de la Raspberry Pi 3 no es de última generación. Funciona a 1200 MHz y dispone solamente de 1 Go de memoria, que comparte con la GPU. En la Raspberry Pi Zero, la frecuencia de funcionamiento es de 1 GHz y la memoria está limitada a 512 MB. El uso del modo texto permite optimizar los recursos disponibles.

El modo texto normalmente es más rápido

Con la costumbre, el uso del modo texto normalmente es más rápido que el modo gráfico. Mientras que con el modo gráfico es necesario algunas veces navegar en varios niveles de ventanas para alcanzar la opción que se quiere modificar, la misma modificación en modo texto se reduce a una línea de comandos. Además, si debemos repetir la operación, la función de recuerdo de comandos anteriores acelera todavía más la operación.

El modo texto para que la comunidad te ayude

En general, en los foros, la asistencia que se da por los participantes se hace en modo texto. Las numerosas interfaces gráficas disponibles en Linux hacen complicada la asistencia en modo gráfico. El modo texto es válido en todas las distribuciones, independientemente del entorno gráfico. Además, para enviar el resultado de los comandos al foro, basta con copiar-pegar, sin necesidad de realizar capturas de pantalla y añadir imágenes voluminosas.

Este capítulo ofrece información de los comandos necesarios para empezar con Linux en modo texto. Este manual de supervivencia de comandos es el mínimo básico que permite a un principiante:

- Arrancar y detener el sistema operativo.
- Navegar por el sistema de archivos y modificarlo.
- Administrar el sistema.

Convenciones utilizadas para la sintaxis

```
comando [opciones opcionales] argumento_obligatorio
```

La primera palabra de la línea es el comando objeto de la descripción.

Entre corchetes figuran las opciones o argumentos opcionales. Su ausencia no impide que el comando se ejecute. En algunos casos, hay comandos que utilizan argumentos por defecto.

El o los argumentos obligatorios deben figurar en la línea de comandos. Su ausencia provoca un mensaje de error.

Para saber más acerca de los comandos

Cada comando puede tener una gran cantidad de opciones. A continuación, solo se enumeran las utilizadas con más frecuencia. Para saber más acerca de cada comando, existe un comando universal en todas las distribuciones Linux: `man`!

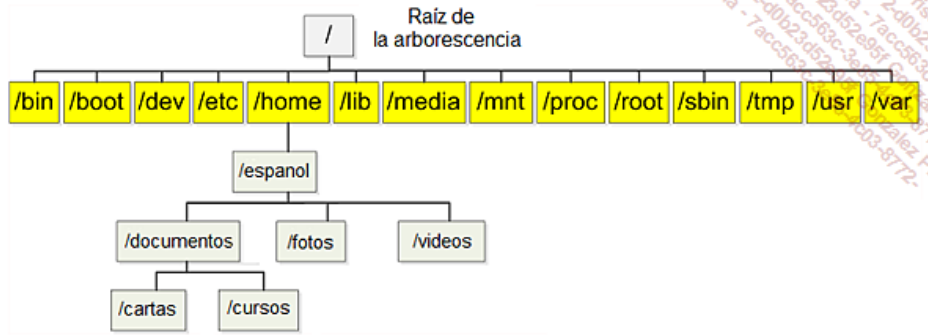
Sintaxis

```
man [nombre_de_comando]
```

Este comando muestra una página de ayuda de cada comando. Pulse la tecla [Espacio] o [Intro] para moverse, [q] para volver al intérprete de comandos, [h] para mostrar la ayuda y todas las opciones. Además, [/] permite hacer búsquedas en el texto. Algunas veces esas páginas de ayuda son un poco áridas para los principiantes, pero pronto se convierten en recursos indispensables para cualquier usuario de Linux.

La arborescencia de Linux

El sistema de archivos de Linux se organiza a partir de un punto de inicio, llamado root, raíz e incluso /. En esta raíz se despliegan los directorios que contienen los archivos y programas necesarios para el sistema operativo. La forma de árbol inverso de esta estructura hace que se haya llamado arborescencia.



Cada usuario puede crear en su directorio la arborescencia que le convenga. No es necesario el conocimiento completo de la arborescencia de Linux para usar Linux. La siguiente descripción puede ayudarle a encontrar el archivo que le interesa.

Principales directorios de la arborescencia Linux	
/	Raíz o root, contiene los directorios de la arborescencia Linux.
bin	Ejecutables binarios del sistema <i>cp</i> , <i>ls</i> , <i>mount</i> , <i>rm</i> , etc.
boot	Archivos de inicio de Linux.
dev	Archivos especiales que garantizan el enlace con los periféricos.
etc	Archivos de configuración del sistema, servicios, etc.
home	Directorio personal de los usuarios.
lib	Librerías del sistema compartidas.
media	Punto de conexión de las llaves USB, CD-ROM, etc.
mnt	Punto de conexión temporal de particiones y periféricos.
proc	Información de los procesos y el núcleo Linux.
root	Directorio personal del super-usuario.
sbin	Binarios de sistema y herramientas como <i>fsck</i> .
tmp	Archivos temporales.
usr	Archivos binarios y comandos de usuario.
var	Sistema de archivos "variables" (modificables); aquí encontramos el contenido web (directorio <i>www</i>) y los logs (archivos de traza).

Estos directorios contienen otros directorios que conforman la arborescencia de Linux. Puede haber algunas diferencias entre las distribuciones, pero la mayor parte de la arborescencia es común.

La línea de comandos

Después del inicio, Linux le da la bienvenida, solicitándole el *login*. Es el primer contacto con la interfaz por línea de comandos.

1. Conexión a Raspbian

En los sistemas Linux hay dos tipos de usuarios. El usuario normal, que tiene permisos limitados, lo que le impide cometer errores de configuración o destruir por error el sistema. Limita la propagación de virus en GNU/Linux. Puede ejecutar aplicaciones y modificar archivos a los que tiene acceso, pero no puede intervenir totalmente en el sistema.

El super-usuario tiene permisos completos sobre el sistema. Puede realizar las actualizaciones y eliminar o modificar cualquier archivo. El super-usuario se llama también *root*. No hay que confundir *root*, raíz del sistema de archivos, con *root*, el super-usuario e incluso */root*, el directorio del super-usuario.

En ciertas distribuciones, *root* no tiene permisos para conectarse al sistema por razones de seguridad. Esto evita que un intruso o un virus puedan usar los privilegios de *root* para sabotear el sistema o introducir programas dañinos. En Raspbian, *root* no tiene permisos para conectarse por defecto.



```
[ 3.454510] hub 1-1:1.0: USB hub found
[ 3.454625] hub 1-1:1.0: 5 ports detected
Welcome to Raspbian GNU/Linux 8 (Jessie)!

[ 3.613261] NET: Registered protocol family 10
[ 3.624901] systemd[1]: Inserted module 'ipovf'
[ 3.630620] systemd[1]: Set hostname to raspberrypi.
[ 3.733127] usb 1-1.1: new high-speed USB device number 3 using duc_otg
[ 3.753291] uart-pl011 3f201000.uart: no DMA platform data
[ 3.853534] usb 1-1.1: New USB device found, idVendor=0424, idProduct=ec00
[ 3.863570] usb 1-1.1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
[ 3.881180] smsc95xx v1.0.4
[ 3.947824] smsc95xx 1-1.1:0 eth0: register 'smc95xx' at usb-3f900000.usb-1.1, smc95xx USB 2.0 Ethernet, 10:27:eb:f4:de:06
[ 4.043069] usb 1-1.2: new high-speed USB device number 4 using duc_otg
[ 4.136623] systemd[1]: Expecting device dev-tty000.device...
[ 4.154694] usb 1-1.2: New USB device found, idVendor=7392, idProduct=7811
[ 4.154707] usb 1-1.2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 4.154717] usb 1-1.2: Product: 002.lin WLAN Adapter
[ 4.154726] usb 1-1.2: Manufacturer: Realtek
[ 4.154735] usb 1-1.2: SerialNumber: 00e94c000001
[ 4.209591] systemd[1]: Starting Forward Password Requests to Wall Directory Watch.
[ 4.223397] systemd[1]: Started Forward Password Requests to Wall Directory Watch.
[ 4.236418] systemd[1]: Starting Remote File Systems (Pre).
[ OK ] Reached target Remote File Systems (Pre).
[ 4.253621] systemd[1]: Reached target Remote File Systems (Pre).
[ 4.263049] usb 1-1.4: new low-speed USB device number 5 using duc_otg
[ 4.277131] systemd[1]: Starting Encrypted Volumes.
[ OK ] Reached target Encrypted Volumes.
[ 4.297031] systemd[1]: Reached target Encrypted Volumes.
[ 4.304670] systemd[1]: Starting Arbitrary Executable File Formats File System Automount Point.
[ OK ] Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 4.325770] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 4.340701] systemd[1]: Starting Swap.
```

Durante el inicio en modo texto, se muestra el logo de la Raspberry Pi en la parte superior izquierda de la pantalla (cuatro frambuesas para la Raspberry Pi 3, una sola para la Raspberry Pi Zero). Los mensajes enviados por el sistema durante su fase de inicio desfilan debajo del logo. Cuando el sistema ha arrancado, indica la dirección IP que utiliza, informa del usuario que trabaja en la consola *tty1* y se pone en modo espera hasta que se conecte un usuario.

a. Conexión como usuario normal

Atención: a diferencia de otros sistemas operativos, Linux es sensible a las mayúsculas y minúsculas. Considera que las mayúsculas y las minúsculas son caracteres diferentes. Esto es así porque sus códigos ASCII son diferentes. Conviene respetar las mayúsculas/minúsculas en la escritura de los comandos.

Al inicio, la Raspberry Pi muestra el nombre del ordenador *raspberrypi* y la palabra *login*: que invita el usuario a identificarse introduciendo su nombre.

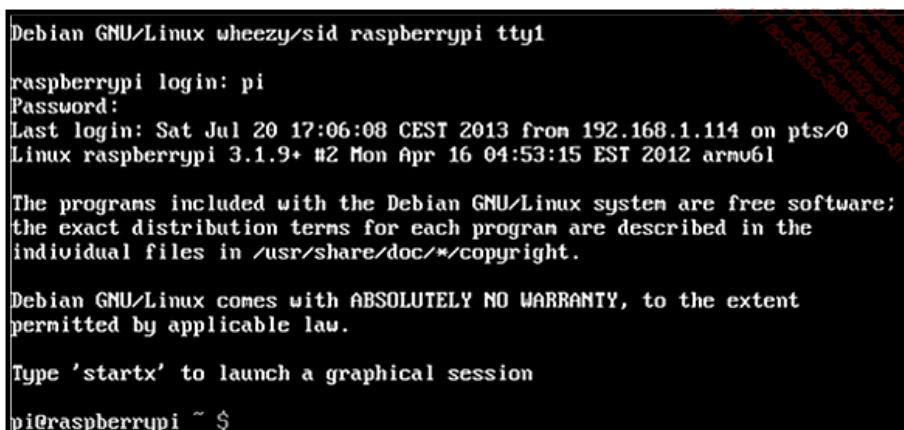
Ya hay creado un usuario por defecto en la distribución Raspbian. Se trata del usuario *pi*, con contraseña *raspberry*.

Escriba el login *pi* y valide con la tecla [Intro].

El sistema solicita una contraseña para autenticar al usuario.

Escriba *raspberry* y valide con la tecla [Intro].

➤ La contraseña introducida por el usuario no provoca ninguna acción en pantalla. Es una medida de seguridad para evitar que una persona que esté observando pueda adivinar el número de caracteres de la contraseña.



```
Debian GNU/Linux wheezy/sid raspberrypi tty1
raspberrypi login: pi
Password:
Last login: Sat Jul 20 17:06:08 CEST 2013 from 192.168.1.114 on pts/0
Linux raspberrypi 3.1.9+ #2 Mon Apr 16 04:53:15 EST 2012 armv6l

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Type 'startx' to launch a graphical session
pi@raspberrypi ~ $
```

El sistema muestra información como: la fecha de la última conexión del usuario *pi*, la versión del núcleo Linux... y, en la última línea, la línea de comandos compuesta aquí de 16 caracteres (sin contar los espacios): *pi@raspberrypi ~ \$*

Esta línea de comandos se interpreta de la siguiente manera: el usuario *pi* se ha conectado al ordenador *raspberrypi*. Trabaja en su directorio personal *~ (/home/pi)*. Solo puede ejecutar un número limitado de acciones, porque solo es un usuario normal. Esto se indica con el símbolo *\$* al final de la línea de comandos. Todo esto se expresa con 16 caracteres, lo que muestra cómo de conciso es el modo texto.

Si el usuario *pi* intenta parar el sistema:

```
pi@raspberrypi ~ $ shutdown -h now
shutdown: you must be root to do that!
Usage: shutdown [-akrhPHFnc] [-t sec] time [warning message]
```

Un mensaje le advierte que solo *root* puede parar el sistema.

El usuario tiene dos posibilidades:

- Desconectarse y conectarse como *root*.
- Convertirse momentáneamente en super-usuario.

Desconexión de un usuario

Sintaxis

```
logout
```

Para cerrar una sesión Linux, escriba `logout` seguido de la tecla [Intro].

La sesión termina y el sistema sigue funcionando. La consola muestra de nuevo `raspberrypi login:` y espera la conexión de un nuevo usuario.

El usuario ahora puede reconectarse como *root* para detener la máquina.

Convertirse momentáneamente en super-usuario

Un usuario normal puede convertirse momentáneamente en super-usuario, el tiempo de ejecutar un comando que únicamente *root* puede ejecutar. Existen dos maneras de alcanzar este resultado.

La primera es `sudo` (*Substitute User Do* = ejecutar como otro usuario). Para usar `sudo`, este usuario debe figurar en un archivo llamado *sudoers*, que enumera los usuarios autorizados a realizar esta operación. Es el administrador el que decide los miembros de la lista *desudoers*.

Sintaxis

```
sudo [comando]
```

Para probar `sudo`, el uso de `ls` simplemente muestra el contenido del directorio (consulte la sección *Moverse en la arborescencia* de este capítulo) sin peligro.

```
pi@raspberrypi ~ $ sudo ls
Desktop python_games
```

La contraseña solicitada por `sudo` es la del usuario que desea convertirse en super-usuario. Por defecto, se memoriza durante 15 minutos. El usuario puede volver a utilizar el comando `sudo` sin necesidad de introducir de nuevo la contraseña, durante estos 15 minutos.

El comando `ls` se ha ejecutado como super-usuario. Si un usuario no autorizado (*user*) intenta realizar la misma operación:

```
user@raspberrypi ~ $ sudo ls

We trust you have received the usual lecture from the local
System Administrator.
It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for user:
user is not in the sudoers file. This incident will be reported.
user@raspberrypi ~ $
```

El usuario *user* no forma parte de los usuarios autorizados para usar el comando `sudo`. Si lo intenta, será rechazado.

`sudo` permitirá al usuario detener el sistema de la misma manera:

```
pi@raspberrypi ~ $ sudo shutdown -h now

Broadcast message from root@raspberrypi (pts/0) (Sat Jul 20
17:43:56 2013):
The system is going down for system halt NOW!
pi@raspberrypi ~ $
```

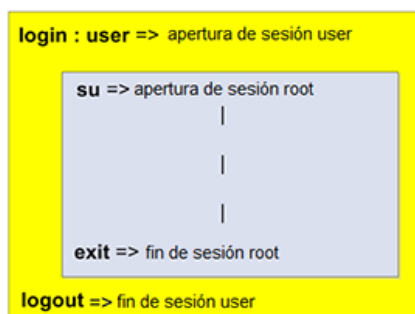
La otra posibilidad para convertirse en super-usuario es iniciar una sesión temporal con otra identidad. Esto autoriza el comando `su`. Si ningún nombre de usuario sigue al comando `su`, es el super-usuario *root* el que se usa por defecto.

Sintaxis

```
su [nombre_de_usuario]
```

<code>nombre_de_usuario</code>	Argumento opcional. Si se omite <code>nombre_de_usuario</code> , es el usuario <i>root</i> el que se selecciona por defecto.
<code>- nombre_de_usuario</code>	La presencia del guión después del comando <code>su</code> permite ejecutar un shell recuperando las variables del usuario correspondiente.
<code>-</code>	El guión únicamente ejecuta un shell con el entorno por defecto de <i>root</i> . Por tanto, <code>su -</code> es mejor que <code>su</code> .

El funcionamiento de `su` se resume a continuación. Un usuario (*user*) inicia una sesión. Durante esta sesión, se conecta momentáneamente como *root* utilizando el comando `su`. Cuando sale de esta sesión *root*, vuelve a su sesión anterior.



La contraseña solicitada por su es la del usuario cuya identidad se corresponde con la que el usuario actual desea conectarse. En el siguiente ejemplo, se espera la contraseña de `root`.

```
user@raspberrypi ~ $ su
Contraseña:
root@raspberrypi:/home/user# ls
archivol.txt
root@raspberrypi:/home/user# exit
exit
user@raspberrypi ~ $
```

Esto implica que el administrador comunicó su contraseña a, al menos, una persona. En materia de seguridad informática, está totalmente prohibido. Por el contrario, en su propia Raspberry Pi, nada le impide usar `su` para convertirse momentáneamente en super-usuario.

Para finalizar una sesión como super-administrador abierta con `su`, debe usar el comando `exit`, que provoca una vuelta a la línea de comandos de inicio (ver la captura de pantalla anterior).

El comando `su` también puede servir para convertirse momentáneamente en otro usuario:

```
pi@raspberrypi ~ $ su user
Contraseña:
user@raspberrypi /home/pi $ ls
Desktop python_games
user@raspberrypi /home/pi $ exit
exit
pi@raspberrypi ~ $
```

Esta vez, el usuario `pi` recibe momentáneamente la identidad del usuario `user`, por tanto, conoce su contraseña. Sin embargo aunque tiene la identidad de `user`, permanece en su carpeta personal propia: `/home/pi` salvo si utiliza el comando `su - user`. El comando `exit` termina la sesión abierta como `user` y vuelve a la línea de comandos `pi@raspberrypi ~ $`.

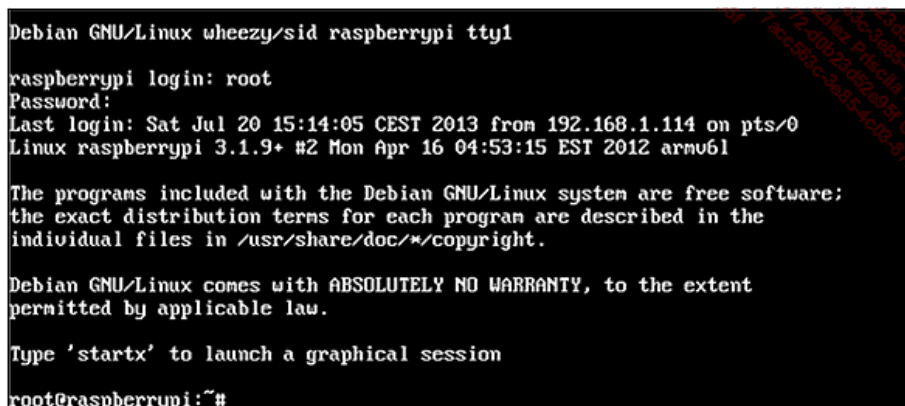
b. Conexión como root

Para permitir la conexión del super-usuario `root` a Raspbian, en primer lugar es necesario activar la cuenta del super-usuario. Con la cuenta del usuario `pi`, es necesario pasar por el comando `sudo` y el comando `passwd`:

```
pi@raspberrypi ~ $ sudo passwd root
Indique la nueva contraseña UNIX:
Repita la nueva contraseña UNIX:
passwd: la contraseña se ha actualizado con éxito
pi@raspberrypi ~ $
```

Observe la contraseña que acaba de asignar a `root`. Cierre la sesión con el comando `logout` y conéctese de nuevo.

Escriba el login `root` y valide con la tecla [Intro].



```
Debian GNU/Linux wheezy/sid raspberrypi tty1
raspberrypi login: root
Password:
Last login: Sat Jul 20 15:14:05 CEST 2013 from 192.168.1.114 on pts/0
Linux raspberrypi 3.1.9+ #2 Mon Apr 16 04:53:15 EST 2012 armv6l

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Type 'startx' to launch a graphical session
root@raspberrypi:~#
```

El sistema solicita la contraseña del usuario para autenticarlo.

Después de escribir la contraseña y validarla, el sistema envía una serie de información a la pantalla de la consola que se está utilizando (tty1), antes de mostrar: `root@raspberrypi:~#`.

Esta línea significa que el usuario `root` está conectado en la máquina `raspberrypi` y que su directorio de trabajo es `~` (abreviación de «directorio personal», aquí `/root`). El signo `#` al final de la línea significa que los comandos se ejecutan a nivel de super-usuario, por tanto sin restricción.

Una vez conectado, `root` tiene acceso al conjunto del sistema. Puede ejecutar todos los comandos sin que el sistema le rechace. La peligrosidad de ciertos comandos hace que la cuenta `root` se use el menor tiempo posible en las máquinas Linux.

Detener correctamente el sistema

Solo el super-usuario está autorizado a detener el sistema.

Linux utiliza un sistema de caché para la escritura en los dispositivos de almacenamiento (tarjeta SD, disco duro, llave USB, etc.). Almacena el mayor tiempo posible los datos en memoria. La escritura de datos al final de un archivo de traza modifica con frecuencia el tamaño del archivo. Si el archivo está en memoria, su modificación es muy rápida, solo durante la parada del sistema se escriben en la tarjeta SD o el disco duro, tanto el archivo como su tamaño.

En caso de parada accidental, los archivos permanecen abiertos, los datos se pierden, los tamaños de archivos ya no corresponden, etc. Un reset o una desconexión de la alimentación pueden dejar el sistema de archivos en un estado tal que el sistema no pueda arrancar de nuevo. Es una de las razones del uso de un convertidor encargado de mantener una alimentación permanente en los puestos de trabajo o servidores sensibles.

Durante una parada iniciada por el administrador con el comando `shutdown`, en primer lugar el sistema solicita a los procesos terminar su actividad. Fuerza la parada de los que siguen funcionando, se escriben las cachés en los dispositivos de almacenamiento y las particiones se desmontan correctamente. Solo cuando se ha hecho esto el sistema se detiene realmente.

Sintaxis

```
shutdown [-t segundos] [-hfFr] hora [mensaje a los usuarios]
```

<code>-t segundos</code>	Argumento opcional. Indica el tiempo de espera en segundos entre el envío del mensaje de advertencia a los usuarios y la parada de los procesos.
--------------------------	--

-h	Argumento opcional. Indica que la máquina se debe detener después de la parada del sistema.
-f	Argumento opcional. Indica que no hay que realizar comprobación del sistema de archivos después de reiniciar.
-F	Argumento opcional. Indica que es necesario obligatoriamente realizar una comprobación del sistema de archivos después de reiniciar.
-r	Argumento opcional. Indica que es necesario reiniciar la máquina después de su parada (reboot).
Hora	Argumento obligatorio. Indica la hora a la que el sistema se debe detener. hh:mm: hora en formato hora:minuto. +m: número de minutos antes de la parada. now: detener ahora, equivalente a +0.
mensaje a los usuarios	Mensaje enviado a todos los usuarios conectados para indicar la hora de parada de la máquina y permitir su desconexión.

Para detener la Raspberry Pi correctamente de manera inmediata utilice el siguiente comando:

```
root@raspberrypi:~# shutdown -h now
```

Espera la parada completa del sistema. No es suficiente con que la pantalla se apague para garantizar la parada del sistema. Hay que observar los LED, en particular el LED ACT, que indica el acceso del sistema a la tarjeta SD. La parada del sistema se traduce en que se apagan todos los LED, excepto el LED de alimentación PWR.

🔴 El sistema está detenido, pero la Raspberry Pi no tiene sistema de apagado. Al final de la secuencia de parada (el LED rojo PWR encendido en la Raspberry Pi 3 y el LED rojo ACT apagado en la Raspberry Pi Zero), es necesario desenchufarla manualmente de la alimentación.

2. Moverse en la arborescencia

Una vez conectado a un sistema operativo, es necesario conocer el directorio de trabajo (o directorio actual) y ser capaz de moverse en la arborescencia.

a. Identificar el directorio actual

Cuando un usuario se conecta, el sistema se sitúa automáticamente en su directorio personal, lo que se indica mostrando ~, que es la abreviatura correspondiente:

```
user@raspberrypi ~ $
```

El comando `pwd` (*print name of working directory* = mostrar el nombre del directorio de trabajo) permite conocer el directorio actual de manera más detallada:

```
user@raspberrypi ~ $ pwd
/home/user
```

Este comando confirma que el directorio de trabajo es `/home/user`, que es el directorio personal del usuario `user`.

b. Listar el contenido de un directorio

Para poder moverse en la arborescencia, es necesario conocer el nombre de los directorios a los que desea acceder. El comando `ls` (*list sort* = lista ordenada), muestra el contenido de un directorio: archivos y sub-directorios. Este comando soporta numerosas opciones, en particular las opciones de ordenación para la presentación del resultado. Únicamente se enumeran a continuación las más útiles.

Sintaxis

```
ls [opciones] [nombre_de_archivo]
```

La siguiente tabla presenta algunas opciones útiles para el uso del comando `ls`. Consulte el manual de este comando para descubrir todas sus posibilidades (`man ls`).

-F	Añade una / al final del nombre de los directorios.
-R	Muestra recursivamente el contenido de los sub-directorios. Si un subdirectorio existe, se muestra su contenido.
-l	Muestra información muy completa de los archivos y directorios (tipo, permisos de acceso, nombre del propietario...).
-a	Muestra todos los archivos, incluidos los que empiezan por un . y que son archivos cacheados.
nombre_de_archivo	Nombre de archivo a mostrar, acepta los "comodines" * y ?.

Ejemplo de comando ls

```
user@raspberrypi /etc $ ls -al n*
-rw-r--r-- 1 root root 8453 marzo 30 2012 nanorc
-rw-r--r-- 1 root root 767 mayo 2 2011 netconfig
-rw-r--r-- 1 root root 60 julio 15 2012 networks
-rw-r--r-- 1 root root 475 agosto 28 2006 nswitch.conf
-rw-r--r-- 1 root root 1988 mayo 18 2012 ntp.conf

network:
total 28
drwxr-xr-x 6 root root 4096 julio 15 2012 .
drwxr-xr-x 87 root root 4096 julio 20 19:52 ..
drwxr-xr-x 2 root root 4096 julio 15 2012 if-down.d
drwxr-xr-x 2 root root 4096 julio 15 2012 if-post-down.d
drwxr-xr-x 2 root root 4096 julio 15 2012 if-pre-up.d
drwxr-xr-x 2 root root 4096 julio 15 2012 if-up.d
-rw-r--r-- 1 root root 53 julio 15 2012 interfaces
lrwxrwxrwx 1 root root 12 julio 15 2012 run -> /run/network
user@raspberrypi /etc $
```

El directorio actual es `/etc`. El usuario introduce el comando `ls -al n*`, lo que significa:

`ls` = mostrar la lista de los archivos y carpetas.

-al = incluso los archivos cacheados, con todos los detalles.

n* = solamente los archivos que empiezan por la letra n.

El sistema muestra cinco archivos y el contenido del directorio *network*, antes de volver a la línea de comandos.

El comando `ls` permite fijarse en los directorios: la primera letra de la línea es una `d` si se trata del nombre de un directorio. Un `-` indica que el nombre corresponde a un archivo normal.

```
user@raspberrypi /etc $ ls -al | more
total 804
drwxr-xr-x 87 root root 4096 julio 20 19:52 .
drwxr-xr-x 22 root root 4096 julio 15 2012 ..
-rw-r--r-- 1 root root 2981 julio 15 2012 adduser.conf
drwxr-xr-x 2 root root 4096 julio 15 2012 alternatives
drwxr-xr-x 7 root root 4096 julio 15 2012 apm
drwxr-xr-x 2 root root 4096 julio 15 2012 apparmor.d
drwxr-xr-x 6 root root 4096 julio 15 2012 apt
drwxr-xr-x 3 root root 4096 julio 15 2012 avahi
-rw-r--r-- 1 root root 1762 abril 30 2012 bash.bashrc
```

La barra después del comando `ls -al` se obtiene pulsando simultáneamente las teclas [Alt Gr] 1. Esta barra se llama *pipe*. La salida del comando `ls`, que se habría enviado a la pantalla, se envía al comando `more`, encargado de gestionar la visualización. Para avanzar pulse en [Espacio] (avanza una página) o [Intro] (avanza una línea), para salir pulse en [q].

➤ Cualquier comando seguido de `| more` provocará que su salida se envíe a la pantalla, página a página.

c. Cambiar de directorio

El comando `cd` (*change working directory* = cambiar de directorio de trabajo) permite cambiar el directorio actual.

Sintaxis

```
cd [nuevo_directorio]
```

Si se omite `nuevo_directorio`, el usuario permanece en su directorio personal. Hay que escribir obligatoriamente un espacio entre el comando `cd` y el siguiente argumento.

Ejemplo del uso del comando cd

```
1 - user@raspberrypi ~ $ cd /
2 - user@raspberrypi / $ ls
3 - bin dev home lost+found mnt proc run selinux sys
4 - usr
5 - boot etc lib media opt root sbin srv tmp
6 - var
7 - user@raspberrypi / $ cd usr
8 - user@raspberrypi /usr $ ls -al
9 - total 64
10 - drwxr-xr-x 10 root root 4096 julio 15 2012 .
11 - drwxr-xr-x 22 root root 4096 julio 15 2012 ..
12 - drwxr-xr-x 2 root root 20480 julio 20 14:50 bin
13 - drwxr-xr-x 2 root root 4096 junio 2 2012 games
14 - drwxr-xr-x 56 root root 4096 julio 15 2012 include
15 - drwxr-xr-x 64 root root 12288 julio 20 14:50 lib
16 - drwxrwsr-x 10 root staff 4096 julio 15 2012 local
17 - drwxr-xr-x 2 root root 4096 julio 15 2012 sbin
18 - drwxr-xr-x 144 root root 4096 julio 20 14:50 share
19 - drwxr-xr-x 2 root root 4096 junio 2 2012 src
20 - user@raspberrypi /usr $ cd bin
21 - user@raspberrypi /usr/bin $ cd
22 - user@raspberrypi ~ $
```

Línea 1: el usuario está en su carpeta personal y solicita ir a la raíz.

Línea 2: el usuario está en la raíz y solicita la lista de archivos y carpetas.

Líneas 3 a 6: el sistema muestra la lista.

Línea 7: el usuario está en la raíz y solicita ir al directorio *usr*.

Línea 8: el usuario está en el directorio *usr* y solicita la lista de archivos y carpetas con detalles.

Líneas 9 a 19: el sistema muestra el resultado. Todas las líneas empiezan por una `d`, por tanto solo hay directorios.

Línea 20: el usuario está en el directorio *usr* y solicita ir al directorio *bin*.

Línea 21: el directorio de trabajo es la carpeta */usr/bin*. El usuario introduce `cd` sin argumento.

Línea 22: el directorio de trabajo es la carpeta personal de *user*: `~`.

Observe las líneas 10 y 11. Hay dos directorios que tienen nombres extraños: `.` y `..`.

- El primero, llamado `.`, es el directorio actual. Es aquí donde el sistema guarda el directorio de trabajo o carpeta activa.
- El segundo, llamado `..`, es el directorio padre, es decir, el que se encuentra inmediatamente antes de la carpeta actual en la arborescencia.

Estos nombres particulares se pueden utilizar en los comandos.

Ejemplo de uso de ..

```
1 - user@raspberrypi ~ $ cd /usr/bin
2 - user@raspberrypi /usr/bin $ cd ..
3 - user@raspberrypi /usr $
```

Línea 1: el usuario trabaja en su directorio personal y solicita el cambio de directorio a */usr/bin*.

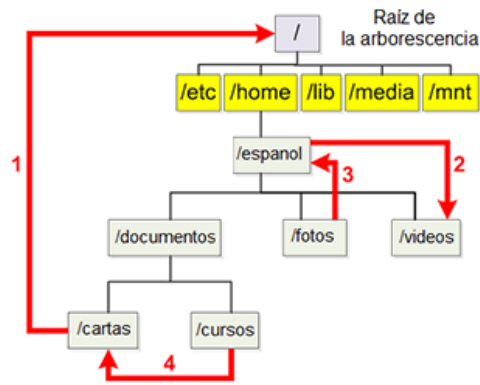
Línea 2: el directorio activo es */usr/bin*. El usuario solicita al sistema volver al directorio padre del directorio de trabajo.

Línea 3: el directorio de trabajo es ahora */usr*.

d. Ruta relativa y ruta absoluta

Existen dos maneras de moverse en la arborescencia: el modo absoluto, que vuelve siempre a la raíz, y el modo relativo, que parte del directorio de trabajo activo.

El siguiente esquema muestra un ejemplo de arborescencia, en la que el usuario desea moverse.



La siguiente tabla enumera los comandos que el usuario debe usar si utiliza el direccionamiento absoluto.

Utilización del direccionamiento absoluto		
	Punto de inicio	Direccionamiento absoluto
1	/home/espanol/documentos/cartas	cd /
2	cd /home/espanol/documentos	cd /home/espanol/videos
3	cd /home/espanol/fotos	cd /home/espanol
4	cd /home/espanol/documentos/cursos	cd /home/espanol/documentos/ cartas

La siguiente tabla enumera los comandos que el usuario debe usar si utiliza el direccionamiento relativo.

Utilización del direccionamiento relativo		
	Punto de inicio	Direccionamiento relativo
1	/home/espanol/documentos/cartas	cd ../../../../
2	cd /home/espanol/documentos	cd videos
3	cd /home/espanol/fotos	cd ..
4	cd /home/espanol/documentos/cursos	cd ../cartas

Las dos formas de moverse en la arborescencia tienen usos diferentes. Nunca estamos seguros de llegar a la dirección correcta. El direccionamiento relativo se utiliza normalmente en los sitios web, lo que permite mover una parte de la arborescencia en la que se ha utilizado el direccionamiento relativo, sin consecuencia para los enlaces entre las páginas.

Para entender correctamente el mecanismo de los movimientos absolutos y relativos, construya una arborescencia (consulte la siguiente sección: Modificar la arborescencia) en su carpeta personal, como en el ejemplo anterior, y muévase en la arborescencia utilizando ambos métodos.

e. Modificar la arborescencia

El usuario puede unir a la arborescencia origen dispositivos de almacenamiento externos (disco duro USB, llave USB, etc.). Esta "conexión" de periféricos externos se aborda en el capítulo Utilizar una memoria de almacenamiento externa.

El usuario también puede necesitar modificar la arborescencia origen de Linux. Esto puede servir para crear o eliminar sub-directorios en su directorio personal, clasificar documentos, fotos, videos...

Crear un directorio

La creación de un directorio se hace con el comando `mkdir` (*make directory* = crear un directorio). Si se pasa como argumento un nombre de directorio, se crea en el directorio de trabajo actual. Para crearlo en otro sitio diferente al directorio activo es necesario informar la ruta en modo absoluto.

Sintaxis

```
mkdir nombre_del_directorio
```

Ejemplo de uso del comando mkdir

```
1 - espanol@raspberrypi ~ $ ls
2 - documentos fotos vacios
3 - espanol@raspberrypi ~ $ mkdir musica
4 - espanol@raspberrypi ~ $ ls
5 - documentos fotos musica vacios
6 - espanol@raspberrypi ~ $ mkdir musica
7 - mkdir: imposible crear el directorio "musica":
8 - El archivo ya existe
9 - espanol@raspberrypi ~ $
```

Línea 1: el usuario lista los archivos y directorios presentes en su directorio personal.

Línea 2: el sistema muestra los archivos y directorios.

Línea 3: el usuario crea un directorio *musica*.

No hay mensaje de error, por tanto el comando se ha ejecutado sin problema.

Línea 4: el usuario solicita la lista de archivos y directorios.

Línea 5: el sistema muestra los archivos y directorios. Un directorio *musica* se ha creado correctamente.

Línea 6: el usuario crea de nuevo un directorio *musica*.

Líneas 7 y 8: el sistema rechaza el comando porque ya existe un directorio con este nombre. Si hubiera existido un archivo *musica*, el sistema

también hubiera rechazado ejecutar el comando.

- Para crear los directorios padres si no existen, es necesario usar la opción `-p`. Por ejemplo, si el directorio `/data` no existe, el comando `mkdir -p /data/copia_seguridad`, creará ambos directorios.

Eliminar un directorio

La eliminación de un directorio se hace con el comando `rmdir` (*remove directory* = eliminar un directorio) o con el comando `rm` si el directorio no está vacío.

Sintaxis

```
rmdir nombre_del_directorio
```

Ejemplo de uso del comando `rmdir`

```
espanol@raspberrypi ~ $ ls
documentos fotos musica vacios
espanol@raspberrypi ~ $ rmdir musica
rmdir: error eliminando «musica»: El directorio no está vacío
```

El directorio `musica` contiene un archivo. Es imposible borrarlo con `rmdir`. Hay que usar `rm -r`:

```
espanol@raspberrypi ~ $ rm -r musica
espanol@raspberrypi ~ $ ls
documentos fotos vacios
```

Esta vez, el archivo se ha borrado correctamente. El comando `rm` normalmente se utiliza para eliminar un archivo. Con la opción `-r` se elimina un directorio no vacío.

```
espanol@raspberrypi ~ $ rmdir -p musica/clasica/mozart
espanol@raspberrypi ~ $ ls
documentos fotos vacios
```

El usuario quiere eliminar el directorio `mozart` y sus padres `musica` y `clasica`. La opción `-p` permite esta eliminación, con la condición de que el directorio esté vacío. Si un directorio padre no está vacío, la eliminación de los padres se detiene a este nivel.

3. Administrar los archivos

La intervención en la estructura de la arborescencia permite preparar directorios para recibir los archivos de los usuarios. Los comandos de copiado, movimiento, eliminación, etc., sirven para poblar los directorios.

a. Copiar los archivos

La copia de archivos y directorios en Linux se hace con el comando `cp`. Este comando acepta varias sintaxis para:

- Copiar un archivo en otro archivo. Si el archivo no existe, el archivo se copia. Si el destino existe, el archivo se borra (ver la opción `-i` a continuación).
- Copiar varios archivos a un directorio.
- Copiar un directorio y sus sub-directorios a un directorio.

Sintaxis 1

```
cp [-iu] archivo destino
```

archivo	Nombre del archivo a copiar, también llamado archivo fuente.
destino	Si <code>destino</code> es un nombre de archivo, el archivo fuente se copia de nuevo en el archivo destino. Si <code>destino</code> es un nombre de directorio, el archivo fuente se copia en este directorio.
<code>-i</code>	Esta opción indica al sistema que debe preguntar al usuario antes de eliminar un archivo existente.
<code>-u</code>	El sistema no hace la copia si el archivo destino tiene una fecha de última modificación igual o más reciente que el archivo fuente.

Ejemplo de uso de la sintaxis 1

El archivo `bolero.wav` (archivo fuente) se copia al archivo `alcyone.wav` (archivo destino).

```
espanol@raspberrypi ~/musica/clasica $ ls
bolero.wav espejos.wav ravel
espanol@raspberrypi ~/musica/clasica $ cp bolero.wav
alcyone.wav
espanol@raspberrypi ~/musica/clasica $ ls
alcyone.wav bolero.wav espejos.wav ravel
espanol@raspberrypi ~/musica/clasica $
```

Sintaxis 2

```
cp [-iu] archivo_1 archivo_2, etc archivo_n directorio
```

archivo_n	Los archivos 1 a n son los archivos a copiar.
directorio	El directorio en el que se copiarán los archivos.
<code>-i</code> y <code>-u</code>	Estas opciones tienen el mismo efecto que en la sintaxis 1.

Ejemplo de uso de la sintaxis 2

Dos archivos, `bolero.wav` y `espejos.wav`, se copian al directorio `ravel`.

```
espanol@raspberrypi ~/musica/clasica $ ls
bolero.wav espejos.wav ravel
espanol@raspberrypi ~/musica/clasica $ cp bolero.wav
espejos.wav ravel
```

```
espanol@raspberrypi ~/musica/clasica $ cd ravel
espanol@raspberrypi ~/musica/clasica/ravel $ ls
bolero.wav espejos.wav
```

Sintaxis 3

```
cp [-iuR] directorio_fuente directorio_destino
```

directorio_fuente	Directorio a copiar.
directorio_destino	Directorio en el que se copiará el directorio fuente y sus sub-directorios.
-i y -u	Estas opciones tienen el mismo efecto que en la sintaxis 1.
-R	Copia recursivamente los sub-directorios del directorio fuente. La presencia de -R durante la copia de directorios evita la visualización de un mensaje de error.

Ejemplo de uso de la sintaxis 3

Un directorio se copia a otro directorio que ya contiene los archivos. Con la opción `-i` el sistema solicita confirmación antes de eliminar los archivos existentes.

```
espanol@raspberrypi ~ $ cp -R -i musica copia_seguridad
cp: ¿Desea eliminar
«copia_seguridad/musica/clasica/bolero.wav»? o
cp: ¿Desea eliminar
«copia_seguridad/musica/clasica/espejos.wav»? o
cp: ¿Desea eliminar
«copia_seguridad/musica/clasica/alcyone.wav»? o
cp: ¿Desea eliminar
«copia_seguridad/musica/clasica/ravel/bolero.wav»? o
```

b. Mover y renombrar los archivos

El comando `mv` (*move* = mover) permite mover y renombrar archivos y directorios.

Sintaxis

```
mv [-fiuv] fuente destino
```

fuentes	Nombre del archivo o del directorio fuente.
destino	Nombre del archivo o del directorio destino. Si destino es un nombre de directorio, mv mueve los archivos fuente a este directorio. Si fuente y destino son archivos en el mismo sistema de archivos, mv renombra el archivo. Si fuente y destino son archivos en sistemas de archivos diferentes, mv mueve el archivo. Si fuente es un nombre de directorio, mv renombra este directorio.
-f	Fuerza el borrado de los archivos existentes.
-i	Pregunta al usuario antes de eliminar un archivo. Si -f y -i existen en la línea de comandos, es la última la que se tiene en cuenta.
-u	El sistema no hace la copia si el archivo destino tiene una fecha de última modificación igual o más reciente que la del archivo fuente.
-v	Indica por pantalla el nombre de los archivos que se han movido.

Ejemplo de uso del comando mv

El usuario renombra el archivo `bolero.wav`, que se convierte en `bolero_de_ravel.wav`.

```
espanol@raspberrypi ~/musica/clasica $ ls
alcyone.wav bolero.wav espejos.wav ravel
espanol@raspberrypi ~/musica/clasica $ mv bolero.wav
bolero_de_ravel.wav
espanol@raspberrypi ~/musica/clasica $ ls
alcyone.wav bolero_de_ravel.wav espejos.wav ravel
```

c. Eliminar archivos

La eliminación de archivos se hace con el comando `rm` (*remove* = eliminar).

Sintaxis

```
rm [-fiRv] nombre_de_archivo
```

nombre_de_archivo	Nombre del archivo o del directorio a eliminar.
-f	No advierte al usuario si un archivo que se debe eliminar no existe. Fuerza la eliminación sin solicitar confirmación si el archivo existe.
-i	Pregunta al usuario antes de eliminar un archivo. Si -f y -i existen en la línea de comandos, es la última la que se tiene en cuenta.
-R	Elimina recursivamente los subdirectorios del directorio fuente y del propio directorio.
-v	Indica por pantalla el nombre de los archivos eliminados.

Ejemplo de uso del comando rm

El usuario decide eliminar todos los archivos `.wav` del directorio actual. El asterisco `*` sustituye al nombre de todos los archivos. La opción `-v` vindica al sistema que muestre por pantalla los nombres de los archivos eliminados.

```
espanol@raspberrypi ~/musica/clasica $ rm -v *.wav
«bolero_de_ravel.wav» eliminado
«espejos.wav» eliminado
```

d. Mostrar el contenido de un archivo

La visualización del contenido de un archivo se hace con el comando `cat`.

Sintaxis

```
cat [-n] nombre_de_archivo_1 nombre_de_archivo_2
```

nombre_de_archivo_1	Nombre del archivo del que el usuario quiere visualizar el contenido. Si hay varios nombres de archivos en la línea de comandos, <code>cat</code> muestra su contenido en la pantalla siguiente.
-n	Numera las líneas mostradas en la pantalla. Es útil cuando es necesario encontrar una línea que tiene problemas.

Ejemplo de uso del comando cat

El usuario muestra el contenido de `/etc/timezone`, en primer lugar sin ningún argumento y después con numeración de las líneas.

```
espanol@raspberrypi /etc $ cat timezone
Europa/Madrid
espanol@raspberrypi /etc $ cat -n timezone
 1 Europa/Madrid
```

➤ En caso de un archivo con un número importante de líneas, la visualización que proporciona `cat` desfila demasiado rápido. Piense en usar `less` para facilitar la lectura. La separación entre el comando `cat passwd` y `less` es el signo pipe, que se obtiene pulsando simultáneamente [Alt Gr] 1.

```
espanol@raspberrypi /etc $ cat passwd | less
```

e. Modificar el contenido de un archivo

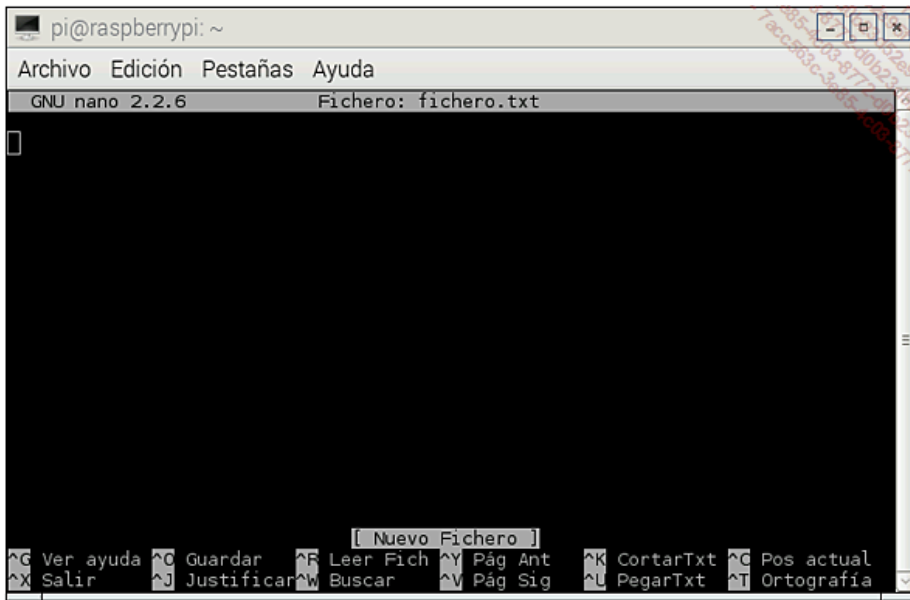
Desde los orígenes de Linux existen editores de archivos. Más o menos prácticos, más o menos complicados de usar, se han sustituido en el año 2000 por *nano*, un editor libre disponible en todas las distribuciones.

Los editores como *vi* y *emacs* se continúan utilizando puntualmente, fundamentalmente para las búsquedas y las sustituciones.

Sintaxis

```
nano nombre_de_archivo
```

Si el archivo existe, *nano* lo abre y muestra su contenido. El movimiento del cursor se realiza con las fechas de dirección. Es posible añadir o eliminar caracteres.



En la parte superior izquierda, *nano* muestra su número de versión. En el centro figura el nombre del archivo.

En la parte inferior, en la mitad de la pantalla, la indicación **[Nuevo Fichero]** muestra que el archivo *fichero.txt* no existe y que *nano* lo creará.

Las dos líneas inferiores de la pantalla recuerdan los comandos disponibles precedidos por el signo ^, que hace referencia a la tecla [Ctrl]. Por ejemplo, para buscar una palabra, pulse [Ctrl] W para que aparezca una zona llamada **Buscar**. Si escribe una palabra y pulsa [Intro], el cursor va a esta palabra. La siguiente búsqueda propondrá por defecto la misma palabra. Pulse en [Intro] para buscar la siguiente ocurrencia de la palabra en el texto. Si escribe otra palabra seguida de [Intro], provoca una nueva búsqueda.

La escritura del archivo se hace con [Ctrl] O. La combinación de teclas [Ctrl] X sale de *nano*. Si el archivo se ha modificado, pregunta si desea guardarlo.

La combinación de teclas [Ctrl] G es la más importante de *nano*. Permite acceder a la ayuda del software. La correspondencia entre las combinaciones de teclas y las teclas de función figura en la ayuda, así como otra información útil.

Para entretenerse, cree un archivo con *nano*. Introduzca texto y guarde el archivo. Vuelva a abrirlo con *nano* y explore las funciones disponibles.

En poco tiempo dominará este editor de texto. Su manejo permite ahorrar mucho tiempo durante la intervención sobre los archivos de configuración, por ejemplo.

f. Comprimir y descomprimir un archivo

El almacenamiento de numerosos archivos poco utilizados se puede optimizar usando *gzip*. Este comando sustituye un archivo por su versión comprimida, añadiendo `.gz` después del nombre de origen. También descomprime los archivos con la opción `-d`.

Sintaxis

```
gzip nombre_de_archivo_1 nombre_de_archivo_2... nombre_de_directorio
```

nombre_de_archivo_n	Se comprimirán todos los archivos cuyo nombre figura en la línea de comandos.
nombre_de_directorio	Si uno o varios nombres son nombres de directorios, se ignoran, salvo si se indica la opción <code>-r</code> .
-d	Descomprime un archivo comprimido.
-l	Proporciona información de los archivos comprimidos (tamaño del archivo comprimido, tamaño del archivo original, tasa de compresión y nombre del archivo original).
-r	Si en la lista hay uno o varios nombres de directorios, recorre los directorios recursivamente comprimiendo individualmente todos los archivos que encuentra. El directorio en sí mismo no

	se comprime.
-t	Comprueba la integridad de un archivo comprimido.
-v	Modo conversación, que ofrece información de cada archivo comprimido o descomprimido.

Ejemplo de uso del comando gzip

El usuario desea comprimir el archivo *archivo1.txt*, que ocupa 35 KB. El comando `gzip` genera un archivo *archivo1.txt.gz* que sustituye al archivo original, pero solo ocupa 12 KB.

```
espanol@raspberrypi ~/documentos $ ls -al
total 44
drwxr-xr-x 2 espanol espanol 4096 julio 22 18:13 .
drwxr-xr-x 7 espanol espanol 4096 julio 21 17:32 ..
-rw-r--r-- 1 espanol espanol 35379 julio 22 17:48 archivo1.txt
espanol@raspberrypi ~/documentos $ gzip archivo1.txt
espanol@raspberrypi ~/documentos $ ls -al
total 24
drwxr-xr-x 2 espanol espanol 4096 julio 22 18:13 .
drwxr-xr-x 7 espanol espanol 4096 julio 21 17:32 ..
-rw-r--r-- 1 espanol espanol 12703 julio 22 17:48
archivo1.txt.gz
```

El usuario quiere visualizar la información de todos los archivos *.gz* del directorio de trabajo.

```
espanol@raspberrypi ~/documentos $ gzip -l *.gz
compressed      uncompressed ratio uncompressed_name
12703           35379 64.2% archivo1.txt
```

El usuario lanza la descompresión de todos los archivos comprimidos del directorio activo.

```
espanol@raspberrypi ~/documentos $ gzip -d *.gz
```

La compresión de *archivo1.txt* se ejecuta en modo *detallado* (conversación). Se muestra la información relativa a la compresión.

```
espanol@raspberrypi ~/documentos $ gzip -v archivo1.txt
archivo1.txt: 64.2% -- replaced with archivo1.txt.gz
```

El usuario vuelve a su carpeta personal y quiere comprimir todos los archivos contenidos en los directorios y subdirectorios, con una visualización por pantalla de las operaciones realizadas por `gzip`.

```
espanol@raspberrypi ~/documentos $ cd ..
espanol@raspberrypi ~ $ ls
documentos copia_seguridad fotos musica video217
espanol@raspberrypi ~ $ gzip -rv *
gzip: documentos/archivo1.txt.gz already has .gz suffix --
unchanged
musica/clasica/ravel/bolero.wav: 22.2% -- replaced with
musica/clasica/ravel/bolero.wav.gz
musica/clasica/ravel/alcyone.wav: -11.1% -- replaced with
musica/clasica/ravel/alcyone.wav.gz
copia_seguridad/musica/clasica/bolero.wav: 22.2% --
replaced with copia_seguridad/musica/clasica/bolero.wav.gz
copia_seguridad/musica/clasica/espejos.wav: -11.1% --
replaced with copia_seguridad/musica/clasica/espejos.wav.gz
copia_seguridad/musica/clasica/alcyone.wav: 22.2% --
replaced with copia_seguridad/musica/clasica/alcyone.wav.gz
copia_seguridad/musica/clasica/ravel/bolero.wav: 22.2% --
replaced with copia_seguridad/musica/clasica/ravel/bolero.wav.gz
espanol@raspberrypi ~ $
```

➤ También es posible descomprimir directamente los archivos con `gunzip`, que acepta los mismos argumentos que `gzip`. En GNU/Linux, el comando `tar` también se utiliza para generar archivos comprimidos en formato `tar.gz`.

4. Acelerar la escritura de los comandos

La escritura repetitiva de los comandos puede convertirse en algo molesto. Afortunadamente Linux ofrece soluciones para acelerar la escritura.

a. Recordar los comandos anteriores

El primer método es el uso de las fechas arriba y abajo para recordar los comandos introducidos anteriormente.

Uso de las fechas de dirección

Las fechas arriba y abajo dan acceso a la lista de comandos introducidos por un usuario. Una vez que se muestra el comando buscado, puede validarlo de nuevo pulsando la tecla [Intro]. Si desea modificar el comando antes de usarlo, muévase con ayuda de las fechas derecha e izquierda, elimine los caracteres, añada los que necesite y valide de nuevo el comando con la tecla [Intro].

➤ Si cambia de usuario en el mismo puesto, el histórico de comandos queda vinculado al usuario. Por ejemplo, el usuario pi inicia una sesión `root` con `su`, cuando está en la sesión del usuario `root` ya no tiene acceso a su propio histórico. Después, con el comando `exit` sale de la sesión `root` y de nuevo tiene acceso a su histórico.

Uso del comando history

El comando `history` memoriza los últimos comandos de un usuario. Es el comando que se pone en marcha cuando el usuario acciona las teclas arriba y abajo para moverse por los comandos memorizados. Este comando recibe un determinado número de opciones o argumentos. A continuación se muestran los más útiles.

Sintaxis

```
history [![cifra] [![letras] [-c]
```

<code>history</code>	<code>history</code> sin argumento muestra la lista de los últimos comandos que un usuario ha introducido (hasta 500)
----------------------	---

	comandos por defecto). Para recuperar el comando que le interesa: <code>history less</code> muestra el comienzo de la lista. Vaya a la parte superior e inferior con las fechas de dirección, recupere el comando que desea reutilizar y salga con la tecla q .
! 16	Recuerda la línea de comandos nº 16 de la lista mostrada por <code>history</code> .
! ls	Recuerda el último comando que empieza por <code>ls</code> de la lista mostrada por <code>history</code> .
- c	Elimina la lista de comandos almacenados por <code>history</code> .
! !	Recuerda el último comando.
! \$	Recuerda el último argumento. Por ejemplo: <code>mkdir rep ls -l !\$</code>

➤ La lista completa de los comandos de un usuario es visible en su directorio personal `$HOME`. Es el archivo cacheado `.bash_history` el que contiene esta lista.

Ejemplo de uso del comando `history`

```
user@raspberrypi ~ $ history
 1 ls
 2 rm archivo2.txt
 3 ls
 4 cp archivo1.txt archivo2.txt
 5 ls
 6 ls -al
 7 date
 8 history
user@raspberrypi ~ $ !7
date
sábado 20 de julio 2013, 19:43:11 (UTC+0200)
user@raspberrypi ~ $ !ls
ls -al
total 32
drwxr-xr-x 2 user user 4096 julio 20 19:42 .
drwxr-xr-x 4 root root 4096 julio 20 17:51 ..
-rw-r--r-- 1 user user  19 julio 20 19:20 archivo1.txt
-rw-r--r-- 1 user user  19 julio 20 19:42 archivo2.txt
user@raspberrypi ~ $ history -c
user@raspberrypi ~ $ history
 1 history
```

b. Función autocompletar

La función autocompletar es otro de los métodos utilizados para ahorrar tiempo cuando escribe en la línea de comandos. Si pulsa la tecla [Tab], le permite automáticamente escribir el final del comando y de los nombres de archivos y directorios si solo hay una posibilidad de autocompletar.

Si hay varias soluciones posibles, se presenta la lista de las opciones disponibles. Basta con pulsar una o varias letras y de nuevo la tecla [Tab] para terminar de escribir el comando.

Esta función está disponible en el shell (el intérprete de línea de comandos) `bash`, que es normalmente el shell por defecto cuando se ejecuta con Raspbian.

Para comprobar que el shell utilizado en su sistema es `bash`:

Escriba `echo $SHELL` en la línea de comandos y valide. Respete las mayúsculas.

```
root@raspberrypi:/home/user# echo $SHELL
/bin/bash
```

La respuesta del sistema `/bin/bash` confirma que el intérprete de comandos en servicio es `bash`.

Si el comando `echo $SHELL` devuelve otra cosa, por ejemplo:

```
$ echo $SHELL
/bin/sh
```

El shell por defecto se ha modificado. Aquí se usa `sh`.

El comando `cat /etc/shells` enumera los intérpretes de comandos disponibles en su sistema:

```
$ cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/dash
/bin/bash
/bin/rbash
```

Puede cambiar el intérprete de comandos por defecto del usuario `pi` gracias al comando `chsh`. El sistema le pedirá su contraseña, rellene `raspberrypi` como contraseña del usuario `pi`. A continuación, indique la ruta y el nombre del intérprete de comandos que desea utilizar:

```
$ chsh
Contraseña:
Modificación del intérprete de comandos inicial para pi
Introduzca el nuevo valor o "Intro" para conservar el valor del
  intérprete de comandos inicial [/bin/sh]: /bin/bash
```

➤ No use el comando `sudo chsch` porque modifica el shell por defecto del usuario `root`.

Cierre la sesión ([Ctrl] **D**) y vuelva a conectarse como el usuario `pi`. Su shell será `/bin/bash`.

➤ `SHELL` es una variable de sistema que contiene el nombre del intérprete de comandos. El comando `echo` con `$` delante de la variable envía el contenido de esta variable a la pantalla.

Ejemplo de uso de la función autocompletar

En la carpeta `/home/user` hay dos archivos llamados `archivo1.txt` y `archivo2.txt`. El super-usuario desea mostrar el contenido de `archivo2.txt`.

Línea 1: escriba `ca` y pulse en [Tab] (dos veces).

```

1 - root@raspberrypi:/home/user# ca
2 - cal      caller      case      catchsegr
3 - calendar captinfo  cat       catman
4 - root@raspberrypi:/home/user# cat archivo
5 - archiv1.txt archiv2.txt
6 - root@raspberrypi:/home/user# cat archivo2.txt
7 - Este es el contenido del archivo archivo2.txt
8 -

```

Línea 2 y 3: el sistema muestra todos los comandos que empiezan por `ca` (lo que puede ser útil cuando el usuario no recuerda exactamente el comando).

Línea 4: el usuario completa su comando escribiendo una `t`, un [Espacio] y una `a`. A continuación pulsa [Tab]. El sistema completa la palabra `archivo` y se detiene porque hay dos posibilidades: `archivo1.txt` y `archivo2.txt`, que se muestran.

Línea 5: visualización de las dos posibilidades que el sistema ha encontrado.

Línea 6: el usuario completa su comando introduciendo un `2` y después pulsa en [Tab]. El sistema completa el nombre de archivo añadiendo `.txt`. La línea de comandos ahora está completa. El usuario valida con la tecla [Intro].

Líneas 7 y 8: visualización del contenido del `archivo2.txt`.

➤ Incluso si la descripción del funcionamiento de la función autocompletar puede parecer larga, la combinación de la función, que recuerda los comandos anteriores junto con el uso de la función autocompletar hace que se gane mucho tiempo con la costumbre.

5. Administrar el sistema

Aprender a navegar en la arborescencia y a gestionar archivos es la primera etapa en la gestión de un sistema Linux. La administración del sistema es el objetivo de este aprendizaje. La mayor parte de los comandos de administración solo son accesibles por el super-usuario `root`.

a. Administrar los usuarios y los grupos

Un usuario es un individuo identificado por un nombre y autenticado mediante una contraseña. Incluso en un ordenador que solo usa una única persona es interesante crear usuarios con permisos diferentes. Esto permite separar y proteger las diversas actividades que se pueden hacer en la misma máquina.

- Un usuario `webmaster` se conecta directamente al directorio del sitio web para hacer las actualizaciones de este directorio.
- Un usuario de `copia de seguridad` podrá ejecutar las copias de seguridad.

Si varias personas comparten la misma máquina, esto permite separar y proteger sus datos.

Solo se solicitará el usuario `root` para las operaciones de mantenimiento del sistema.

Un grupo es un conjunto de usuarios. Durante la creación de cada usuario, Debian crea por defecto un grupo con el mismo nombre que este usuario y sitúa al usuario en este grupo.

El super-usuario o un usuario convertido en super-usuario es el que gestiona los usuarios y los grupos.

Crear un grupo

La creación de un grupo se hace con el comando `addgroup`. El grupo creado está vacío.

Sintaxis

```
addgroup nombre_grupo
```

Ejemplo de uso del comando addgroup

```

root@raspberrypi:/etc# addgroup musicos
Adición del grupo "musicos" (GID 1004)...
Terminado.

```

Se ha creado el grupo `musicos`. El sistema le ha asignado el GID (*Group ID* = identificador de grupo) 1004.

Solo se ha creado el grupo, no contiene ningún miembro.

Añadir un usuario

La adición de un usuario se hace con el comando `adduser`. Es posible introducir directamente la información por línea de comandos, o dejar al sistema actuar por defecto.

Sintaxis

```
adduser [opciones] nombre_de_usuario
```

<code>nombre_de_usuario</code>	Nombre del usuario a crear.
<code>--home directorio_personal</code>	Es posible asignar un directorio personal, cuyo nombre es diferente del que corresponde al usuario. Por defecto, este directorio se crea en <code>/home</code> .
<code>--ingroup grupo_primaria</code>	Indica el grupo principal del usuario. El grupo debe existir antes de la creación de la cuenta del usuario.

Ejemplo de uso del comando adduser

El usuario se crea con el comando `adduser`, sin indicar opción alguna. El sistema solicita la contraseña del usuario (y después su confirmación). La contraseña no se muestra por motivos de seguridad.

```

root@raspberrypi:/etc# adduser ravel
Adición del usuario «ravel» ...
Adición del nuevo grupo «ravel» (1005) ...
Adición del nuevo usuario «ravel» (1003) con el grupo
«ravel» ...
Creación del directorio personal «home/ravel»...
Copia de los archivos desde «/etc/skel»...
Introduzca la nueva contraseña UNIX
Repita la nueva contraseña UNIX
passwd: la contraseña se ha actualizado con éxito
Modificación de la información relativa al usuario ravel

```

```

Introduzca el nuevo valor o pulse «Intro» para conservar el valor
mostrado
  nombre completo []: Maurice RAVEL
  N° de oficina []:
  Teléfono profesional []:
  Teléfono personal []:
  Otra información []:
¿Esta información es correcta? [S/N]S

```

A continuación, se solicita información administrativa opcional antes de la validación. Sin precisar las opciones, la carpeta personal del usuario se crea en `/home` y forma parte del grupo que tiene el mismo nombre que él (el comando `groups` anterior muestra los grupos de los que forma parte un usuario).

Para comprobar que las operaciones se han desarrollado correctamente, tome la identidad de `ravel` y compruebe la existencia de su directorio personal y su pertenencia al grupo `ravel`.

```

root@raspberrypi:/etc# su ravel
rael@raspberrypi /home $ cd ~
rael@raspberrypi ~ $ pwd
/home/ravel
rael@raspberrypi ~ $ groups ravel

```

➤ Los comandos `addgroup` y `adduser` son específicos de Debian y de las distribuciones que derivan de él. Existen los equivalentes en las demás versiones de Linux.

Cambiar la contraseña de un usuario

El administrador se enfrenta a un problema recurrente: un usuario ha perdido/olvidado su contraseña. En Debian, las contraseñas se almacenan en el archivo `/etc/shadow`, pero cifradas y ningún sistema actual es capaz de descifrarlas.

Sin embargo, un ataque con un diccionario que contenga las contraseñas más habituales puede tener éxito sin necesidad de descifrar la contraseña.

El comando `passwd` permite modificar la contraseña de un usuario. Solo el super-usuario `root` o el propietario de una contraseña la pueden la cambiar.

Sintaxis

```
passwd nombre_de_usuario
```

-d	Solo root puede usar esta opción, que desactiva la contraseña. El usuario se conecta sin contraseña.
-l	Bloqueo (<i>lock</i>) de la cuenta del usuario, poniendo un <code>!</code> delante de la contraseña cifrada. La contraseña ya no se reconoce más.
-u	Desbloqueo (<i>unlock</i>) de la cuenta de un usuario. Elimina la <code>!</code> que añadió la opción <code>-l</code> .

Ejemplo de uso del comando `passwd`

El usuario `mozart` intenta modificar la contraseña del usuario `wolfgangamadeus`. No tiene permiso para hacerlo.

```

mozart@raspberrypi ~ $ passwd wolfgangamadeus
passwd: no puede mostrar o modificar la información
de contraseña de wolfgangamadeus.

```

El super-usuario modifica la contraseña del usuario `wolfgangamadeus`. Debe introducir dos veces la contraseña. Si las dos son diferentes, la contraseña no se modifica.

```

root@raspberrypi:~# passwd wolfgangamadeus
Introduzca la nueva contraseña UNIX:
Repita la nueva contraseña UNIX:
passwd: la contraseña se ha actualizado con éxito

```

Modificar un usuario

Los datos de un usuario se modifican con el comando `usermod`.

Sintaxis

```
usermod [opciones] login_del_usuario
```

-a	Añade el usuario a un grupo adicional. Solo se usa con la opción <code>-G</code> .
-d directorio_personal	Modifica el directorio personal del usuario. Si se indica la opción <code>-m</code> , el nuevo directorio se crea si no existe y los archivos contenidos en el directorio actual del usuario se mueven a este directorio.
-g grupo	Modifica el grupo principal del usuario. El grupo se debe crear antes de usar el comando.
-G grupo1,grupo2,...	Añade el usuario a todos los grupos secundarios listados después de <code>-G</code> . Atención, si el usuario pertenece a un grupo que no se lista, ya no volverá a formar parte de él. Ver la opción <code>-a</code> , que permite añadir al usuario a grupos sin modificar los grupos a los que pertenece.
-l nuevo_login	El nombre del usuario (su login), se convierte en <code>nuevo_login</code> . Sin más cambios, por ejemplo el directorio personal conserva el nombre correspondiente al antiguo login. Será necesario renombrarlo manualmente o con la opción <code>-d</code> .
-m	Mueve el contenido del antiguo directorio personal del usuario a su nuevo directorio personal.
-s nuevo_shell	Modifica el shell del usuario durante su conexión. Para asociar el shell por defecto, deje el campo <code>nuevo_shell</code> vacío. Todos los shells disponibles se listan en <code>/etc/shells</code> . El shell se debe indicar con la ruta absoluta.

Ejemplo de uso del comando `usermod`

El usuario `user` pertenece únicamente al grupo `user`:

```
user@raspberrypi /etc $groups
```

```
user
```

El administrador añade el usuario *user* al grupo *musicos*, además de mantenerlo en el grupo *user*:

```
root@raspberrypi:/etc# usermod -G musicos user
```

El usuario *user* pertenece ahora a los dos grupos:

```
user@raspberrypi /etc $ groups
user musicos
```

Eliminar un usuario

La eliminación de un usuario se hace con el comando `userdel`.

Sintaxis

```
userdel [opciones] login_del_usuario
```

-f	Destruye la cuenta del usuario, incluso si está conectado. También destruye el directorio principal de este usuario, incluso si se usa por otro usuario.
-r	Se eliminan los archivos del directorio del usuario en el directorio personal. Por el contrario, los archivos que pertenecen a este usuario y se ubican fuera se deben buscar y destruir manualmente.

Ejemplo de eliminación de un usuario

El administrador intenta eliminar el usuario *user*, que está conectado en el sistema. El sistema indica que el usuario está conectado. Cuando el super-usuario toma la identidad de *user*, se puede conectar.

```
root@raspberrypi:/etc# userdel user
userdel: user user is currently used by process 2241
root@raspberrypi:/etc# su user
user@raspberrypi /etc $
```

El administrador fuerza con `-f` la eliminación del usuario *user*. El sistema le indica que el usuario está conectado, pero lo elimina. Cuando el super-usuario intenta tomar la identidad de *user*, el sistema lo rechaza.

```
root@raspberrypi:/etc# userdel -f user
userdel: user user is currently used by process 2241
root@raspberrypi:/etc# su user
No passwd entry for user 'user'
root@raspberrypi:/etc#
```

b. Administrar los repositorios

Linux utiliza un sistema de actualización basado en *repositories* (repositorios en español). Un repositorio es un espejo (accesible con el protocolo HTTP), una copia del repositorio de origen de la distribución. Durante la instalación, es posible seleccionar un espejo en ciertas distribuciones. En otras distribuciones, el espejo se asigna automáticamente en función de la situación geográfica de la estación de trabajo.

En la distribución Debian y sus derivados, el archivo `/etc/sources.list` contiene los sitios espejo en los que el sistema debe buscar la lista de archivos disponibles, su versión, etc. Los archivos y sus dependencias se guardan como paquetes que contienen todo lo necesario para la instalación.

APT (*Advanced Packaging Tool* = administrador avanzado de paquetes) es la herramienta utilizada por Debian para actualizar e instalar los paquetes. Los paquetes Debian están en formato `.deb`. Este formato también se ha adoptado por las distribuciones derivadas de Debian.


En Raspbian, el archivo `sources.list` contiene, por defecto:

- La rama *main*, que es la rama principal de Raspbian, completamente libre.
- La rama *contrib*, en la que los paquetes son libres pero utilizan dependencias no libres.
- La rama *non-free*, cuyos paquetes no son libres sino simplemente redistribuibles. Debian considera que la rama no libre no forma parte de Debian. Solo se proporciona por comodidad para los usuarios.
- La rama *rpi*, destinada a recibir paquetes específicos de Raspbian.

El tipo `deb` contiene los programas binarios ejecutables por el microprocesador. El tipo `deb-src` corresponde al código fuente de los paquetes. Los programas se proporcionan en forma de texto y se llaman `sources`. Por ejemplo, se podrían utilizar para modificar y recompilar algunos paquetes. Generalmente el tipo `deb-src` no es necesario para un uso básico de la Raspberry Pi.

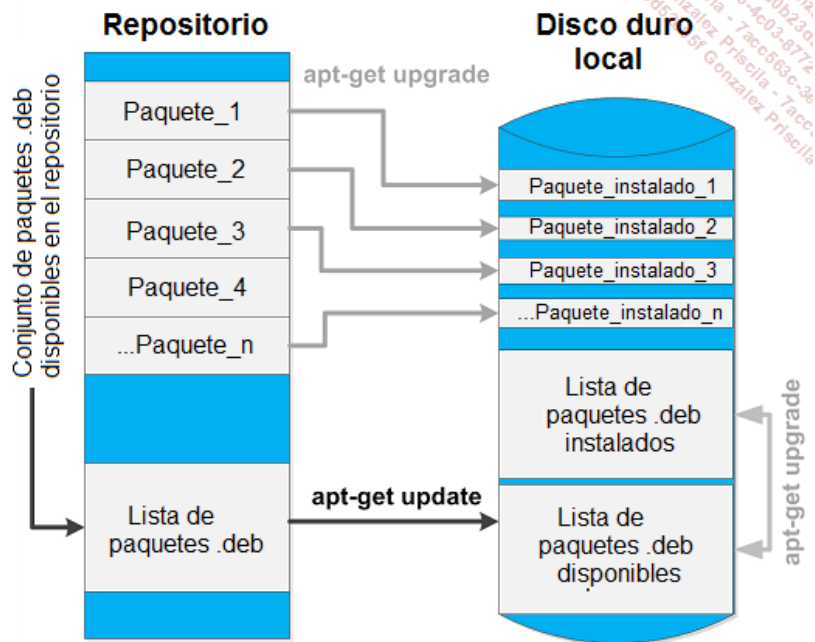
Contenido del archivo `/etc/sources.list` de la distribución Raspbian wheezy

```
deb http://mirrordirector.raspbian.org/raspbian wheezy main
contrib non-free rpi
```

 `http://...` es el enlace al espejo y `wheezy` es el nombre de la versión de Debian de la que deriva Raspbian.

c. Tener el sistema actualizado

APT se compone de varios programas. Uno de ellos, `apt-get`, se usa para la actualización del sistema y la instalación de paquetes nuevos.



La actualización empieza obligatoriamente con el comando `apt-get update`, que va a sincronizar la lista de paquetes disponibles en el disco duro local con la lista del repositorio.

Una vez sincronizados, se pueden ejecutar los comandos `apt-get upgrade` y `apt-get dist-upgrade`.

En algunos casos, es útil actualizar el firmware de la Raspberry Pi. Este programa es el equivalente a la BIOS de un PC. Gracias a él el SoC se dirige al hardware de la tarjeta. En este caso el comando `rpi-update` se ejecutará antes de `apt-get dist-upgrade`.

- Si tiene datos importantes en la tarjeta micro SD, realice sistemáticamente una copia de seguridad de la tarjeta antes de lanzar operaciones de actualización.

Sintaxis

<code>apt-get update</code>	Sincroniza los archivos enumerando los paquetes disponibles en el repositorio y en la estación de trabajo.
<code>apt-get upgrade</code>	Instala las versiones más recientes de los paquetes instalados en el disco duro local. No añade paquetes que no se hubieran instalado. No elimina paquetes instalados en la máquina.
<code>apt-get dist-upgrade</code>	Realiza la actualización de los paquetes, generando los eventuales cambios de dependencia de los paquetes nuevos. Favorece la actualización de los paquetes más importantes frente a los paquetes menores.

Ejemplo de uso del comando `apt-get`

```
root@raspberrypi:~# apt-get update
0% [Working]
Hit http://mirrordirector.raspbian.org wheezy InRelease
0% [Waiting for headers]
Hit http://archive.raspberrypi.org wheezy InRelease
0% [Working]
0% [InRelease gpgv 14.9 kB]
7% [Working]
7% [InRelease gpgv 7,737 B] [Waiting for headers]
Hit http://mirrordirector.raspbian.org wheezy/main armhf Packages
13% [InRelease gpgv 7,737 B]
...
Reading package lists... 83%
Reading package lists... Done...
```

Actualiza la lista de los paquetes disponibles. Una vez actualizada la lista de paquetes con `apt-get update`, es posible ejecutar la actualización de los paquetes con `apt-get upgrade`.

```
root@raspberrypi:~# apt-get upgrade
Reading package lists... Done
Building dependency tree... 87%
...
Reading state information... Done

The following packages have been kept back:
  alsabase gnome-themes-standard lxde lxsession omxplayer
  openssh-client
  openssh-server scratch wpasupplicant
The following packages will be upgraded:
  als-utils apt apt-utils aptitude aptitude-common base-files
  bash binutils
  bsdtails bzip2 ca-certificates console-setup console-setup-
  linux consolekit
  coreutils cpio cpp cpp-4.6 cryptsetup-bin curl dbus dbus-x11
  ...
312 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.
Need to get 263 MB of archives.
After this operation, 25.8 MB disk space will be freed.
Do you want to continue [Y/n]? y
0% [Working]
Get:1 http://archive.raspberrypi.org/debian/ wheezy/main
libpixmap-1-0 armhf 0.26.0-4-raspi [1,200 kB]
0% [Waiting for headers] [1 libpixmap-1-0 156 kB/1,200 kB 13%]
...
```

En el caso anterior, la actualización va a necesitar la descarga de 312 paquetes, es decir, 263 MB. Los paquetes se descomprimen a continuación y, por último, se hace la actualización. Esta operación puede durar cerca de 10 minutos en la Raspberry Pi. La duración depende

de las diferencias existentes entre los paquetes instalados y los paquetes disponibles.

- Se aconseja actualizar regularmente la distribución para instalar las últimas versiones de los paquetes, que corrigen fallos o bugs.

d. Instalar/eliminar un programa

La instalación de un programa en Raspbian también pasa por el uso del comando `apt-get`. La ejecución del comando `apt-get update` antes de la instalación de un programa garantiza que la versión del programa que se va a instalar será la más reciente.

- El programa a instalar debe estar disponible en un repositorio presente en `sources.list`. Por defecto será necesario añadir, con `nano` o con el comando `add-apt-repository`, el repositorio susceptible de ofrecer el paquete correspondiente o instalar el programa siguiendo las indicaciones del desarrollador.

Sintaxis

```
apt-get install nombre_del_programa
apt-get remove nombre_del_programa
```

Ejemplo de instalación y eliminación de programa

El usuario desea ejecutar el programa `tree`, que es una herramienta que muestra por pantalla la arborescencia del sistema de archivos.

```
root@raspberrypi:~# tree /home/user
-bash: /usr/bin/tree: Ningún archivo o directorio de este tipo
```

El usuario instala el programa `tree`, disponible en el repositorio Raspbian.

```
root@raspberrypi:~# apt-get install tree
Lectura de las listas de paquetes... Terminado
Construcción del árbol de dependencias
Lectura de la información de estado ... Terminado
Se instalarán los siguientes paquetes nuevos:
 tree
0 actualizados, 1 instalados de nuevo, 0 eliminados y 10 no actualizados.
Es necesario dejar 0 o/43,4 KB en los archivos.
Después de esta operación, se utilizarán 108 KB de espacio en disco
adicionales.
Selección del paquete tree anteriormente no seleccionado.
(Lectura de la base de datos... 54271 archivos y directorios
ya instalados)
Despliegue de tree (desde ../tree_1.6.0-1_armhf.deb) ...
Tratamiento de las acciones en segundo plano («triggers») para «man-db»...
Configuración de tree (1.6.0-1) ...
```

`tree` está ahora disponible para mostrar la arborescencia del directorio personal del usuario `user`.

```
root@raspberrypi:~# tree /home/user
/home/user
├── documentos
│   ├── archivo1.txt
│   └── archivo2.txt
├── musica
│   └── bolero_de_ravel.txt
└── video

3 directorios, 3 archivos
```

El usuario desinstala ahora el programa `tree` y confirma la eliminación del programa.

```
root@raspberrypi:~# apt-get remove tree
Lectura de las listas de paquetes... Terminado
Construcción del árbol de dependencias
Lectura de la información de estado... Terminado
0 actualizados, 0 instalados de nuevo, 1 eliminados y 10 no
actualizados.
Después de esta operación, se liberarán 108 KB de espacio en disco.
¿Desea continuar [S/N]? S
(Lectura de la base de datos... 54279 archivos y directorios
ya instalados.)
Eliminación de tree ...
Tratamiento de las acciones en segundo plano («triggers») para «mandb»...
```

El programa se ha eliminado: el comando `tree` ya no funciona más.

```
root@raspberrypi:~# tree
-bash: /usr/bin/tree: Ningún archivo o carpeta de esto tipo
```

e. Administrar los permisos

El comando `ls -l` muestra la información de los archivos y directorios, en particular los permisos asociados a cada archivo, así como el propietario del archivo y el grupo propietario.

```
root@raspberrypi:/home/user# ls -l
total 12
-rwxrw-rw- 1 user user 120 Jul 24 13:14 archivo1.txt
-rwxrw-rw- 1 user user 180 Jul 24 13:14 archivo2.txt
drw-r--r-- 2 user user 4096 Jul 24 13:15 letras
root@raspberrypi:/home/user#
```

El comienzo de cada una de las líneas da información acerca del archivo o directorio cuyo nombre figura en el extremo derecho de la línea. El primer bloque de 10 caracteres afecta a los permisos asociados al archivo o directorio listado.

Tipo	Propietario			Grupo propietario			El resto		
d	r	w	x	r	w	x	r	w	x
	4	2	1	4	2	1	4	2	1

La primera letra indica el tipo del archivo:

d para un directorio (*directory* en inglés).

- para un archivo ordinario.

c para un periférico de tipo carácter (puerto serie, módem...).

b para un periférico de tipo bloque (disco duro).

Los nueve caracteres siguientes se reparten en tres bloques:

- Los permisos del propietario del archivo.
- Los permisos del grupo propietario del archivo.
- Los permisos del resto.

Cada bloque tiene tres letras que, cuando existen, indican que el permiso correspondiente está disponible. Cuando el permiso no existe, la letra se sustituye por un guión -.

r permiso de lectura.

w permiso de escritura.

x permiso de ejecución.

- el permiso no existe.

Los permisos también se pueden indicar utilizando las letras o cifras inscritas bajo cada una de las casillas.

A continuación siguen dos nombres: **user user**, que indican para el primero el nombre del propietario del archivo, para el segundo el nombre del grupo propietario del archivo. Aquí el usuario *user* pertenece al grupo *user*, de ahí la repetición del nombre. Esta información proviene del usuario que ha creado el archivo: su login y su grupo principal.

Análisis de los permisos de un archivo

```
-rwxrw-rw- 1 user user 120 Jul 24 13:14 archivo1.txt
```

Tipo	Propietario			Grupo propietario			El resto		
-	r	w	x	r	w	-	r	w	-
	4	2	1	4	2	1	4	2	1
	7			6			6		

El archivo pertenece al usuario *user* y al grupo *user*.

Los permisos corresponden a un archivo, porque la primera letra es un -.

El propietario tiene los permisos *r*, *w* y *x*: puede leer, escribir (modificar, eliminar) y ejecutar este archivo (lo que tiene poco interés en este caso, porque se trata de un simple archivo de texto). Los permisos *rwx* corresponden a 4+2+1 y se pueden expresar con la cifra 7.

Los miembros del grupo *user*, además del propietario *user*, tienen permisos para leer y escribir este archivo (*rw*). Los permisos *rw* corresponden a 4+2 y se pueden expresar con la cifra 6.

Los usuarios que no son ni *user* ni miembros del grupo *user* (el resto) también tienen permisos para leer y escribir sobre este archivo (*rw*). Los permisos *rw* corresponden a 4+2 y se pueden expresar con la cifra 6.

Los permisos de este archivo se pueden escribir *rwxrw-rw-* o **766**, lo que es equivalente.

- Los permisos en Linux se expresan en octales, sistema de numeración en base 8. Cada grupo de 3 signos representa 1 bit y el valor digital es la traducción en octal del número binario correspondiente. *rwxrw-rw-* es equivalente a 111110110 en binario y 766 en octal.

Permisos sobre un directorio

Los permisos sobre un directorio tienen un significado diferente que los permisos de un archivo.

- r** mostrar el contenido del directorio con **ls**.
- w** crear o eliminar archivos en el directorio con **mv**, **cp**, **rm**.
- x** acceder al directorio con el comando **cd**.
- el permiso no existe.

Si un directorio tiene el permiso *w*, se debe dar el permiso *x* para autorizar el acceso al directorio, y también *r* para poder acceder a los archivos contenidos en el directorio. Cuando existe el permiso *w* en un directorio, un usuario puede eliminar todos los archivos que contiene, incluso aunque no tenga permisos explícitos sobre estos últimos. El Sticky Bit permite limitar este permiso, limitando únicamente al propietario de un archivo el permiso para eliminarlo.

Permisos extendidos

SUID (*Set owner User ID up on execution* = ejecutar con la identidad del propietario) permite ejecutar un programa con los permisos de su propietario, en lugar de los permisos del que lanza la ejecución. No tiene efecto en los directorios. La aplicación de SUID sobre un archivo del que root es propietario podría permitir a un atacante ejecutar un programa con los permisos de root, lo que constituye un riesgo para la seguridad.

SGID (*Set Group ID up on execution* = ejecutar con la identidad del grupo) aplicado a un archivo, tiene el mismo efecto que SUID pero a nivel del grupo, con el mismo riesgo para la seguridad. Aplicado a un directorio, SGID hace heredar su grupo propietario a todos los archivos y directorios creados en este directorio. En una compartición, esto permite a todos los miembros de un grupo modificar archivos comunes.

El Sticky Bit definido en un directorio impide a un usuario no propietario de un archivo que pueda eliminarlo.

Modificación del propietario y del grupo propietario

El propietario y el grupo propietario de un archivo o directorio se fija durante la creación, con el login y el grupo principal del que los ha creado.

El cambio de propietario se hace con el comando `chown`, el del grupo propietario con `chgrp`. Solo el super-usuario puede cambiar el propietario y el grupo de un archivo o directorio del que no es propietario.

Cuando se modifica el propietario de un archivo, el antiguo propietario ya no tiene permisos sobre este archivo. Si pertenece al mismo grupo que el nuevo propietario, tendrá los permisos del grupo. Si no pertenece al mismo grupo que el nuevo propietario, tendrá los permisos del resto de usuarios.

Sintaxis

```
chown nuevo_propietario nombre_del_archivo
chgrp nuevo_propietario nombre_del_archivo
```

Ejemplo de cambio del usuario propietario de un archivo

El usuario *user* intenta hacer a *mozart* propietario de su archivo. El sistema indica que esta operación no se permite.

```
user@raspberrypi ~ $ chown mozart archivo1.txt
chown: changing ownership of 'archivo1.txt': Operation not
permitted
```

Solo el super-usuario puede cambiar el propietario de un archivo:

```
root@raspberrypi:/home/user# chown mozart archivo1.txt
root@raspberrypi:/home/user# ls -l
total 12
-rw-r--r-- 1 mozart user 120 Jul 24 13:14 archivo1.txt
-rw-r--r-- 1 user user 180 Jul 24 13:14 archivo2.txt
drw-r--r-- 2 user user 4096 Jul 24 13:15 letras
```

Ejemplo de cambio del grupo propietario de un archivo

El super-usuario cambia el grupo propietario de *archivo1.txt*.

```
root@raspberrypi:/home/user# chgrp mozart archivo1.txt
root@raspberrypi:/home/user# ls -l
total 12
-rw-r--r-- 1 mozart mozart 120 Jul 24 13:14 archivo1.txt
-rw-r--r-- 1 user user 180 Jul 24 13:14 archivo2.txt
drw-r--r-- 2 user user 4096 Jul 24 13:15 letras
```

Modificación de los permisos

La modificación de los permisos de los archivos y directorios se hace con el comando `chmod`. El propietario de un archivo y el super-usuario pueden ejecutar el comando `chmod`.

`chmod` funciona de manera literal (modo simbólico), agregando o eliminando permisos o de manera digital, aplicando directamente el conjunto de los permisos.

Sintaxis

```
chmod [u g o a] [+ - =] [r w x] nombre_del_archivo
chmod NNN nombre_del_archivo
```

u	Aplicar las modificaciones al usuario propietario (<i>user</i>).
g	Aplicar las modificaciones al grupo propietario (<i>group</i>).
o	Aplicar las modificaciones a los demás (<i>others</i>).
a	Aplicar las modificaciones a los tres bloques (<i>all</i>).
+	Añadir un permiso.
-	Retirar un permiso.
=	Eliminar un permiso.
r	Permiso de lectura.
w	Permiso de escritura.
x	Permiso de ejecución.

Ejemplo de uso de `chmod` en modo simbólico

Un archivo llamado *archivo1.txt* tiene los permisos *rw-rw-rw-*.

```
root@raspberrypi:/home/user# ls -l
total 4
-rwxrwxrwx 1 user user 180 Jul 24 13:14 archivo1.txt
```

Para todos los usuarios (*a*), el super-usuario borra los permisos (=) y los sustituye por lectura (*r*).

```
root@raspberrypi:/home/user# chmod a=r archivo1.txt
root@raspberrypi:/home/user# ls -l
total 4
```

```
-r--r--r-- 1 user user 180 Jul 24 13:14 archivo1.txt
```

Para el grupo solamente (g), el super-usuario añade (+) el permiso de ejecución (x).

```
root@raspberrypi:/home/user# chmod g+x archivo1.txt
root@raspberrypi:/home/user# ls -l
total 4
-r--r-xr-- 1 user user 180 Jul 24 13:14 archivo1.txt
```

Para el propietario solamente (u), el super-usuario añade (+) los permisos de escritura y ejecución (wx).

```
root@raspberrypi:/home/user# chmod u+wx archivo1.txt
root@raspberrypi:/home/user# ls -l
total 4
-rwxr-xr-- 1 user user 180 Jul 24 13:14 archivo1.txt
```

Para el propietario y el grupo (ug), el super-usuario retira (-) el permiso de ejecución (x).

```
root@raspberrypi:/home/user# chmod ug-x archivo1.txt
root@raspberrypi:/home/user# ls -l
total 4
-rw-r--r-- 1 user user 180 Jul 24 13:14 archivo1.txt
root@raspberrypi:/home/user#
```

Ejemplo de uso de chmod en modo digital

Con un poco de costumbre, este método es muy rápido. Permite configurar, de una sola vez todos los permisos de un archivo o directorio. Combinado con la función de recuerdo de los comandos anteriores, permite ahorrar un tiempo considerable.

El archivo *archivo1.txt* tiene los permisos *rw-r--r--*.

```
root@raspberrypi:/home/user# ls -l
total 4
-rw-r--r-- 1 user user 180 Jul 24 13:14 archivo1.txt
```

El super-usuario da todos los permisos a todo el mundo.


```
root@raspberrypi:/home/user# chmod 777 archivo1.txt
root@raspberrypi:/home/user# ls -l
total 4
-rwxrwxrwx 1 user user 180 Jul 24 13:14 archivo1.txt
```

El super-usuario da todos los permisos, pero solamente al propietario; al grupo y al resto no le da ningún permiso.

```
root@raspberrypi:/home/user# chmod 700 archivo1.txt
root@raspberrypi:/home/user# ls -l
total 4
-rwx----- 1 user user 180 Jul 24 13:14 archivo1.txt
```

El super-usuario deja todos los permisos al propietario y el permiso de lectura y de escritura al grupo y al resto.

```
root@raspberrypi:/home/user# chmod 766 archivo1.txt
root@raspberrypi:/home/user# ls -l
total 4
-rwxrw-rw- 1 user user 180 Jul 24 13:14 archivo1.txt
```

 Durante la creación de un archivo, se aplica una máscara a los permisos por defecto de Raspbian. El comando *umask* devuelve el valor 0022 que corresponde a *---w--w-*. Esta máscara se sustrae de los permisos fundamentales de la distribución. Los archivos creados no son modificables ni por el grupo propietario ni por el resto.

f. Conocer la ocupación de la tarjeta micro SD

Para saber la ocupación de la tarjeta micro SD y de otros periféricos de almacenamiento masivos conectados a la Raspberry Pi, Linux dispone del comando *df* que indica el tamaño de cada sistema de archivos, el espacio ocupado y el espacio que aún queda disponible. Por defecto *df* muestra los resultados en número de bloques. Las opciones permiten obtener una visualización más legible.

Sintaxis

```
df [-hkm]
```

-h	Añade a cada valor mostrado, una letra de unidad que facilita la lectura. Es la forma más legible para el usuario.
-k	Añade los valores en número de bloques de 1 KB.
-m	Muestra los valores en número de bloques de 1 MB

Ejemplo de uso del comando df

```
pi@raspberrypi:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        15G  4.0G  11G  29% /
devtmpfs         459M    0  459M   0% /dev
tmpfs            463M    0  463M   0% /dev/shm
tmpfs            463M  6.6M  457M   2% /run
tmpfs            5.0M  4.0K  5.0M   1% /run/lock
tmpfs            463M    0  463M   0% /sys/fs/cgroup
/dev/mmcblk0p1   63M   21M   43M  33% /boot
tmpfs            93M    0   93M   0% /run/user/1000
```

A partir de estos datos es posible determinar el momento en que un sistema de archivos o un periférico de almacenamiento necesita una intervención porque está demasiado lleno. Por ejemplo, más atrás vemos que / tiene 4 GB ocupados de los 15 GB disponibles.

g. Escribir un script shell

Un script shell es un archivo de texto que contiene un conjunto de comandos, habitualmente introducidos por línea de comandos. Se agrupan en forma de archivo, cuya ejecución se puede realizar como un comando Linux habitual.

Este párrafo no pretende ofrecer en pocas líneas una formación en la escritura de scripts shell. El objetivo es más modesto y se trata de mostrar el mecanismo desde la escritura de un script hasta su ejecución.

¿Para qué sirve un script shell?

Un script shell sirve para automatizar las tareas que un administrador debe hacer de manera repetitiva. Esto puede ser, por ejemplo, una copia de seguridad regular o una acción iniciada por la presencia de un archivo particular.

Un script de copia de seguridad

Para mostrar de manera sencilla la utilidad de un script shell, a continuación se presenta un script de copia de seguridad del directorio personal de un usuario.

Especificaciones

Un usuario utiliza regularmente su Raspberry Pi. Los documentos que realiza se guardan en su directorio personal. Cuando termina de trabajar, quiere hacer una copia de seguridad de sus directorios y de los archivos que contienen en una llave USB. El nombre del archivo de copia de seguridad debe contener la fecha actual para facilitar la clasificación y la búsqueda.

- En primer lugar, se debe conectar la llave USB. La conexión de la llave USB al sistema de archivos se detalla en el capítulo Utilizar una memoria de almacenamiento externa.

Contenido del script copiaSeguridad.sh

```
#!/bin/bash
# Archivo de copia de seguridad del directorio personal del usuario pi

# El nombre del archivo de copia de seguridad es la fecha actual
# en formato AAAAMMDD
# La llave USB está conectada en el directorio /mnt/llave_usb

# Registro de la fecha en la variable $DATE
# Indicando el formato solicitado: AAAAMMDD
DATE=$(date +"%Y%m%d")

# Directorio personal del usuario pi
DIR_PERSONAL="/home/pi"

# Mensaje que indica que la copia de seguridad empieza
echo "Al inicio de la copia de seguridad"
echo "-----"

# Creación del archivo con compresión de los archivos
# El comando tar crea un archivo que agrupa todos los archivos del
# directorio.
# La opción j indica que el archivo se debe comprimir
# La opción v indica el modo conversación
# La opción f indica que el nombre del archivo sigue a las opciones
# La opción c indica que es necesario crear un nuevo archivo
tar -cjvf /mnt/llave_usb/$DATE.tar.bz2 $DIR_PERSONAL

# Anuncia el fin de la copia de seguridad
echo "Copia de seguridad terminada"
echo "-----"
```

Algunas explicaciones

Los # al comienzo de una línea indican los comentarios. La línea que sigue al # se ignora por el intérprete de comandos.

DATE=... crea una variable llamada DATE y guarda en esta variable lo que está a la derecha del signo =, aquí la fecha en formato AAAAMMDD, con ayuda del comando date y del formato adaptado.

DIR_PERSONAL=... crea la variable DIR_PERSONAL y en ella guarda /home/pi.

echo "... " escribe simplemente en la pantalla el texto entre comillas.

tar crea un archivo en función de los argumentos que siguen.

/mnt/llave_usb/\$DATE.tar.gz2 es la ruta absoluta del archivo. La variable DATE se sustituye por la fecha actual que contiene desde el inicio del programa. El \$ delante de la variable permite llamar a su valor.

Aplicación del script

Conéctese como super-usuario root. Cree un archivo copiaSeguridad.sh con nano. Escriba el texto del script. Eventualmente, modifique la ruta del directorio donde se guardará el archivo. Haga el script ejecutable con el comando chmod +x copiaSeguridad.sh. Ejecute a continuación el script (es necesario obligatoriamente introducir ./ delante del nombre del archivo para poder ejecutar el script desde el directorio actual; también se puede lanzar con la ruta absoluta).

```
pi@raspberrypi ~ $ sudo ./copiaSeguridad.sh
Al inicio de la copia de seguridad
-----
/20130725.tar.bz2
tar: Eliminación de «/» al comienzo de los nombres de los miembros
/home/pi/
/home/pi/.profile
...
...      #Lista de todos los archivos añadidos al archivo
...
/home/pi/python_games/inkspillologo.png
/home/pi/python_games/catgirl.png
/home/pi/.bash_history
Copia de seguridad terminada
-----
pi@raspberrypi ~ $
```

El archivo se ha creado y contiene todos los archivos del usuario pi. Por tanto, el script cumple su función, ya que un solo comando realiza toda una serie de operaciones de manera automática.

- Este script es minimalista, no hace ninguna comprobación antes de realizar la copia de seguridad, para comprobar que el directorio de destino está disponible, por ejemplo.

h. Planificar las tareas

Si es interesante disponer de un script que automatice ciertas operaciones, también sería práctico un script que se lance automáticamente a ciertas horas. Linux dispone de esta función. Es `crontab` y permite la ejecución de un script o de un comando a una hora determinada o siguiendo un ciclo definido por el usuario. `crontab` es el diminutivo de `chronos` (tabla de planificación). Esta tabla especifica las tareas a ejecutar, así como la hora a la que la ejecución se debe iniciar. También se puede indicar una periodicidad para cada una de las tareas.

Para crear una tabla o editarla, el comando es `crontab -e`.

Este comando abre el editor de texto por defecto (*nano* en Raspbian) en la *crontab* personal del usuario. Durante la primera apertura, este archivo solo contiene las líneas de explicación que empiezan con un `#`.

```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# Minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow  command
```

Al final del archivo, bajo los comentarios, el usuario puede programar las tareas y el momento de su ejecución.

Sintaxis de las líneas en la crontab de usuario

```
m h dom mon dow comando
```

m	Minutos de 0 a 59.
h	Hora de 0 a 23.
dom	<i>day of month</i> = día del mes de 1 a 31.
mon	<i>month</i> = mes de 1 a 12 o la abreviatura del mes (en inglés, es decir: jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov y dec).
dow	<i>day of week</i> = día de la semana con 0 y 7 para domingo, 1 para lunes, etc.
comando	Al menos, el script debe tener permisos de ejecución para el usuario que lanza la tarea.

Para cada unidad de tiempo, hay varias notaciones:

- * la tarea se ejecuta cada unidad de tiempo.
- 3-6 la tarea se ejecuta a las unidades de tiempo 3, 4, 5 y 6.
- */2 la tarea se ejecuta cada dos unidades de tiempo 0, 2, 4, 6, 8...
- 2,7 la tarea se ejecuta a las unidades de tiempo 2 y 7.

Ejemplos de programación de la tarea copiaSeguridad.sh

La copia de seguridad se lanza cada media hora:

```
30 * * * * /home/pi/copiaSeguridad.sh
```

La copia de seguridad se lanza todos los días a las 18:00h:

```
00 18 * * * /home/pi/copiaSeguridad.sh
```

La copia de seguridad se lanza cada 10 minutos:

```
10,20,30,40,50,0 * * * * /home/pi/copiaSeguridad.sh
*/10 * * * * /home/pi/copiaSeguridad.sh
```

La copia de seguridad se lanza el jueves a 12:30h:

```
30 12 * * 4 /home/pi/copiaSeguridad.sh
```

La comprobación de las tareas programadas se hace con el comando `crontab -l`, que muestra la lista de tareas presentes en la *crontab* del usuario.

También es posible programar una tarea para que se ejecute cuando arranca la máquina:

Sintaxis

```
@reboot comando
```

La directiva `@reboot` sustituye los cinco campos de fecha y hora.

i. Administrar los procesos

Cada programa que se lanza en Linux se convierte en un proceso mientras dura su ejecución por el microprocesador. Un proceso se ejecuta con los permisos del usuario que lo lanza. Es una de las razones que explican la casi total ausencia de virus en Linux. Si un usuario normal lanza un programa malintencionado, este no podrá realizar operaciones no autorizadas para este usuario. No podrá modificar los archivos de sistema, por ejemplo.

Al inicio de Linux, se arranca un primer proceso: *init*. Tiene el PID 1 (PID = *Process ID* = identificador de proceso). Este proceso es el padre de todos los demás procesos. Arranca los procesos del núcleo y los procesos de sistema, que reciben cada uno un PID diferente. Cada proceso tiene un padre, el proceso que lo ha creado, y él mismo puede ser padre de otros procesos.

Hay disponibles varios comandos para enumerar y analizar los procesos que se están ejecutando.

pstree

El comando `pstree` permite enumerar en forma de arborescencia los procesos activos en un sistema Linux. La opción `-p` muestra el número de los PID entre paréntesis.

```

root@raspberrypi:~# pstree -p
init(1)─┬─console-kit-dae(2437)─┬─{console-kit-dae}(2438)
      │                       └─{console-kit-dae}(2439)
      └─...
          └─cron(2282)
              └─dbus-daemon(2317)
                  └─dhclient(2065)
                      └─getty(2430)
                          └─getty(2431)
                              └─getty(2432)
                                  └─getty(2433)
                                      └─getty(2434)
                                          └─getty(2435)
                                              └─ifplugd(1885)
                                                  └─ifplugd(1912)
                                                      └─login(2429)─┬─bash(2514)
                                                          └─ntpd(2361)
                                                              └─polkitd(2504)─┬─{polkitd}(2506)
                                                                  └─rsyslogd(2235)─┬─{rsyslogd}(2236)
                                                                      │   └─{rsyslogd}(2238)
                                                                      │   └─{rsyslogd}(2239)
                                                                      └─sshd(2395)─┬─sshd(2549)─┬─bash(2556)─┬─pstree(2562)
                                                                          └─thd(2420)
                                                                              └─udev(145)─┬─udev(235)
                                                                                  └─udev(236)

```

top

Para hacerse una idea de los procesos que consumen más recursos, el comando `top` muestra en tiempo real los procesos y su consumo de recursos de sistema. En las primeras líneas se muestra información del tiempo de funcionamiento del sistema, las tareas activas, la ocupación de la CPU y de la memoria.

```

top - 18:37:13 up 7 min, 1 user, load average: 0.07, 0.14, 0.09
Tasks: 171 total, 1 running, 170 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.9 us, 1.7 sy, 0.0 ni, 97.3 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
KiB Mem: 882772 total, 299516 used, 583256 free, 17696 buffers
KiB Swap: 102396 total, 0 used, 102396 free, 145512 cached Mem

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 3489 pi        20   0  85584 20752  8616 S   2.3   2.4   0:01.37 Xtightvnc
 6624 pi        20   0  49972 19632 16112 S   2.3   2.2   0:00.53 lxterminal
 6714 pi        20   0   6812  2524  2132 R   1.0   0.3   0:00.09 top
 1682 pi        20   0  97040 25852 21052 S   0.3   2.9   0:01.64 lxpanel
 3643 pi        20   0  96956 25996 21292 S   0.3   2.9   0:01.29 lxpanel
    1 root         0   0  23856  3948  2736 S   0.0   0.4   0:03.31 systemd
    2 root         0   0    0    0    0 S   0.0   0.0   0:00.00 kthreadd
    3 root         0   0    0    0    0 S   0.0   0.0   0:00.04 ksoftirqd/0
    4 root         0   0    0    0    0 S   0.0   0.0   0:00.00 kworker/0:0
    5 root         0  -20    0    0    0 S   0.0   0.0   0:00.00 kworker/0:+
    6 root         0   0    0    0    0 S   0.0   0.0   0:00.19 kworker/u8+
    7 root         0   0    0    0    0 S   0.0   0.0   0:00.26 rcu_sched
    8 root         0   0    0    0    0 S   0.0   0.0   0:00.00 rcu_bh
    9 root        rt   0    0    0    0    0 S   0.0   0.0   0:00.04 migration/0
   10 root        rt   0    0    0    0    0 S   0.0   0.0   0:00.03 migration/1
   11 root         0   0    0    0    0 S   0.0   0.0   0:00.02 ksoftirqd/1
   13 root         0  -20    0    0    0 S   0.0   0.0   0:00.00 kworker/1:+

```

Este comando dispone de una gran cantidad de opciones (ver `man top`), pero generalmente las más utilizadas son las letras (respetar las mayúsculas/minúsculas):

- P: clasificar los procesos por porcentaje de ocupación del procesador.
- M: clasificar los procesos por porcentaje de ocupación de memoria.
- k: "matar" un proceso.
- q: salir de `top`.

Resultan útiles para localizar un proceso que "bloquea" la Raspberry Pi, monopolizando el tiempo de procesador, por ejemplo. La letra `k` escrita por el teclado abre una petición en la sexta línea de `top`:

```
pid to signal/kill
```

Basta con introducir el PID del proceso a eliminar y pulsar en la tecla [Intro]. La línea se convierte en:

```
Send pid 2602 signal [15/sigterm].
```

Por defecto, `top` ofrece enviar `SIGTERM` al proceso. Esta señal, que tiene el número 15, solicita al proceso que haga una copia de seguridad de sus datos y se detenga. Algunas veces es suficiente para provocar la parada de un proceso, pero el proceso también puede interceptar o ignorar a `SIGTERM`.

Para enviar `SIGTERM` simplemente pulse la tecla [Intro].

Si `SIGTERM` no detiene el proceso, será necesario forzar la parada del mismo enviándole `SIGKILL`, la señal número 9. Basta con introducir 9 y validar para eliminar de un plumazo el proceso.

ps

El comando `ps` muestra los procesos que se están ejecutando, pero en forma de una lista más estética.

Sintaxis

```
ps [opciones]
```

<code>ps</code>	Ejecutado sin opciones, el comando <code>ps</code> muestra los procesos lanzados por el usuario conectado en la consola que se están ejecutando. En general son poco numerosos.
<code>ps -ef</code>	Enumera todos los procesos lanzados por el conjunto de usuarios de todas las consolas del sistema.

	A menudo hay que ejecutar <code>ps -ef less</code> para explotar correctamente esta información.
<code>ps -aux</code>	a enumera todos los procesos. u muestra el usuario de los procesos. x muestra la lista de los procesos sin terminal.
<code>ps -u usuario</code>	Enumera los procesos lanzados por un usuario.

Ejemplo de uso de ps

Procesos del usuario conectado a la consola:

```
root@raspberrypi:~# ps
PID TTY      TIME CMD
2556 pts/0    00:00:01 bash
2659 pts/0    00:00:00 ps
```

Procesos del usuario *pi*:

```
root@raspberrypi:~# ps -u pi
PID TTY      TIME CMD
2672 tty1     00:00:00 bash
2679 tty1     00:00:00 nano
```

Todos los procesos de todos los usuarios, incluido el sistema:

```
root@raspberrypi:~# ps -ef
UID      PID  PPID  C  STIME TTY      TIME  CMD
root      1    0    0  18:44 ?        00:00:00  init [2]
root      2    0    0  18:44 ?        00:00:00  [kthreadd]
root      3    2    0  18:44 ?        00:00:00  [ksoftirqd/0]
root      4    2    0  18:44 ?        00:00:00  [kworker/0:0]
root      5    2    0  18:44 ?        00:00:00  [kworker/u:0]
root      6    2    0  18:44 ?        00:00:02  [rcu_kthread]
root     28    2    0  18:44 ?        00:00:00  [jbd2/sda2-8]
root     29    2    0  18:44 ?        00:00:00  [ext4-dio-unwrit]
root    145    1    0  18:44 ?        00:00:00  udevd -daemon
root    235   145    0  18:44 ?        00:00:00  udevd -daemon
root    236   145    0  18:44 ?        00:00:00  udevd -daemon
|
|          LISTA REDUCIDA POR SER DEMASIADO LARGA...
|
/usr/lib/policykit-1/polkitd -n
root    2514 2429  0  18:58 tty1     00:00:01  -bash
root    2549 2395  0  19:00 ?        00:00:03  sshd: root@pts/0
root    2556 2549  0  19:01 pts/0    00:00:01  -bash
root    2665 2514  0  19:58 tty1     00:00:00  su pi
pi      2672 2665  0  19:58 tty1     00:00:00  bash
root    2678  2    0  19:58 ?        00:00:00  [flush-8:0]
pi      2679 2672  0  19:58 tty1     00:00:00  nano archivol.txt
root    2686 2556  0  20:00 pts/0    00:00:00  ps -ef
```

Podemos localizar en esta lista la siguiente información (primera línea de la lista):

UID	PID	PPID	C	STIME	TTY	TIME	CMD
-----	-----	------	---	-------	-----	------	-----

UID = usuario que ha lanzado este proceso.

- PID = identificador del proceso.
- PPID = identificador del padre de este proceso.
- C = prioridad del proceso (valor grande = prioridad grande).
- STIME = *start time* = hora de comienzo de ejecución del proceso.
- TTY = terminal en el que el proceso se ha lanzado.
- TIME = tiempo de ocupación de la CPU por este proceso.
- CMD = comando que ha lanzado este proceso.

Eliminación de proceso

El comando `kill` permite eliminar un proceso.

Sintaxis

```
kill -numero_signal PID_del_proceso
```

Ejemplo de eliminación de proceso

El usuario *pi* ha abierto el editor *nano* en un terminal.

```
root@raspberrypi:~# ps -upi
PID TTY      TIME CMD
2672 tty1     00:00:00 bash
2750 tty1     00:00:00 nano
```

El super-usuario desea cerrar la ventana de *nano*.

```
root@raspberrypi:~# kill -9 2750
```

Se cierra la ventana *nano* del terminal utilizado por el usuario *pi* y se muestra el mensaje **Proceso terminado**.

- `kill PID` provoca un parada normal del proceso (-15 por defecto). `kill -9 PID` fuerza la parada del proceso, con el riesgo de provocar pérdidas de información.

j. Configurar la fecha del sistema

La Raspberry Pi no tiene reloj RTC mantenido por batería. Cuando se conecta a Internet, recupera la hora de un servidor de tiempo. Para los

modelos A y A+, algunas veces es necesario ajustar la fecha manualmente.

El comando `date` utilizado sin argumentos permite mostrar la hora del sistema.

Muestra la hora del sistema:

```
pi@raspberrypi ~ $ date
miércoles 4 de enero de 2017, 21:01:12 (UTC+0200)
```

El comando `date` utilizado con argumentos permite configurar la hora del sistema. Solo el super-usuario puede modificar la hora del sistema.

Actualización de la hora del sistema:

Sintaxis

```
date MMDDhhmm[YYAA.ss]
```

MM	Mes
DD	Día del mes
hh	Hora
mm	Minuto
YY	Dos primeras cifras del año (opcional)
AA	Dos últimas cifras del año (opcional)
.ss	Segundo (opcional)

Ejemplo:

```
pi@raspberrypi ~ $ sudo date 081008242016.00
miércoles 10 de agosto de 2016, 08:24:00 (UTC+0200)
```

Guardar su configuración

Después de la aplicación de las opciones, la actualización y personalización del sistema, Raspbian está listo para su uso en la Raspberry Pi.

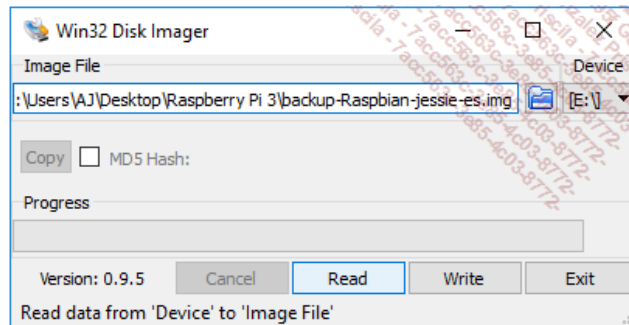
La tarjeta micro SD contiene una versión de Raspbian en buen estado. Es prudente hacer una copia de seguridad, que evitará volver a pasar por todas las etapas anteriores. La Raspberry Pi es una herramienta formidable para experimentar, lo que algunas veces hace imposible reiniciar con un sistema "modificado".

Exactamente como para las imágenes puestas a disposición en el sitio de la Fundación, el sistema presente en la tarjeta SD se va guardar en su estado actual, en forma de archivo *.img*.

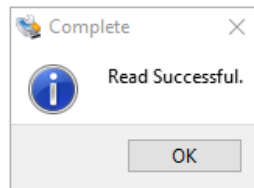
1. Hacer una copia de seguridad de la tarjeta micro SD en Windows

Apague correctamente la Raspberry Pi. Saque la tarjeta SD del conector e introdúzcala en el lector de tarjetas de su PC.

Inicie la herramienta *Win32DiskImager* que ha servido para crear la tarjeta micro SD. Compruebe cuidadosamente que la tarjeta indicada en la etiqueta **Device** de *Win32DiskImager* se corresponde con su tarjeta micro SD. Pulse en la carpeta a la derecha de la zona **Imagen File**, seleccione el directorio en el que desea guardar la imagen y asígnele un nombre. Esta vez, en lugar de escribir la imagen desde el disco a la tarjeta, pulse el botón **Read**. Esto va a provocar la lectura de la tarjeta micro SD, la transferencia de los bytes que contiene al PC y la creación de un archivo *.img* que contiene una copia del sistema.



Cuando la copia de seguridad completa ha terminado, *Win32DiskImager* muestra:



La imagen ahora se guarda en su disco duro, en la ubicación que haya seleccionado. Puede sacar la tarjeta micro SD del lector y usarla. En caso de problema, vuelva a generarla con la imagen.

2. Hacer una copia de seguridad de la tarjeta micro SD en Debian 7

Apague correctamente la Raspberry Pi. Saque la tarjeta micro SD del conector e introdúzcala en el lector de tarjetas de su PC.

En Linux, el comando *dd* que ha servido para la creación de la tarjeta micro SD también se va a utilizar para realizar una imagen de esta.

Identifique su tarjeta micro SD con el comando: `fdisk -l`.

```
root@debian7:~# fdisk -l
.../...

Disco /dev/sdb: 15.7 GB, 15720251392 bytes
64 encabezados, 32 sectores/pistas, 14992 cilindros, total 30703616 sectores
Unidades = sectores de 1 * 512 = 512 bytes
Tamaño del sector (lógico/físico): 512 bytes/512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes/512 bytes
Identificador de disco: 0x000c7b31

Periférico Arranque Inicio Fin Bloques Id Sistema
/dev/sdb1 8192 122879 57344 c W95 FAT32 (LBA)
/dev/sdb2 122880 3788799 1832960 83 Linux
```

Controle el tamaño de las particiones y el tamaño total de la tarjeta para evitar cualquier error.

Su tarjeta micro SD puede aparecer, por ejemplo, en */dev/sda* o */dev/sdb*. Escriba el siguiente comando con una tarjeta detectada en */dev/sdb* y después valide para iniciar la copia de la tarjeta en el archivo que se indica como argumento después de *of=* (*Output File* = archivo de destino).

```
sudo dd if=/dev/sdb of=/home/user/20141023_raspbian_wheezy-es.img bs=1M
```

La copia de la tarjeta tendrá el tamaño de la tarjeta. Por tanto, la imagen de una tarjeta micro SD de 16 GB ocupará 16 GB de espacio en disco. Puede comprimir el archivo para reducir el espacio ocupado con el comando:

```
gzip /home/user/raspbian-jessie-es-20160810.img
```

3. Copia de seguridad de la tarjeta SD en modo gráfico

La versión gráfica de Raspbian Jessie ofrece la posibilidad de realizar una copia de la tarjeta SD de un sistema mientras se está usando ("en caliente"). Vaya al capítulo Utilizar el modo gráfico - Los menús de Raspbian Jessie - Accesorios - SD Card Copier.

Conclusión

La totalidad de los comandos que se han presentado en este capítulo constituye un kit básico para iniciarse en el uso de Linux.

No se han mencionado muchas de las opciones, ni todos los comandos existentes. Solo la práctica permite a un usuario/administrador formarse una caja de herramientas a medida y personal. La consulta de las páginas `man` de estos comandos es un complemento indispensable en la práctica regular de la línea de comandos.

Introducción

Algunas veces, una Raspberry Pi funciona sin pantalla ni teclado conectado, o en remoto y no fácilmente accesible. Por tanto es interesante poder conectarse para realizar tareas de administración. Es posible hacerlo en modo texto en Windows, iniciando una sesión en modo texto con el software PuTTY en SSH. En Linux el comando `ssh` abre un acceso remoto en modo texto. Para ello, el servidor SSH debe estar activado en el menú **Preferencias - Configuración de Raspberry Pi** o en modo texto en `raspi-config`.

La comunicación entre dos ordenadores se puede hacer en abierto (telnet, rlogin, rcp, ftp...). Los códigos ASCII se transfieren tal cual entre las máquinas. Entonces es posible seguir estos intercambios e incluso interferir eventualmente con acciones alintencionadas. SSH (*Secure Shell* = Shell seguro) es un protocolo de comunicación seguro que impone un intercambio de claves de cifrado al inicio de la conexión. Como resultado de la comunicación entre las dos máquinas, se autentifican y se cifran. Resulta imposible espiar el contenido de los intercambios o conectarse fraudulentamente a la Raspberry Pi.

`ssh` también hace referencia al programa informático encargado de implementar el protocolo SSH, por ejemplo en Linux.

Es posible conectar varias sesiones de visualización en la misma Raspberry Pi lo que, por ejemplo, puede ser útil cuando se está aprendiendo. El formador puede observar en tiempo real las acciones de un estudiante conectado en modo remoto a una Raspberry Pi.

En la versión de Raspbian PIXEL, que salió a finales de septiembre de 2016, la Fundación ha integrado el servidor VNC, que se activa marcando una opción en la ventana de configuración de la Raspberry Pi.

- En este capítulo, observe que la dirección IP de la Raspberry Pi algunas veces es diferente. Esto es debido al uso de direcciones diferentes por el DHCP, en función del momento en el que se llevaron a cabo las pruebas.

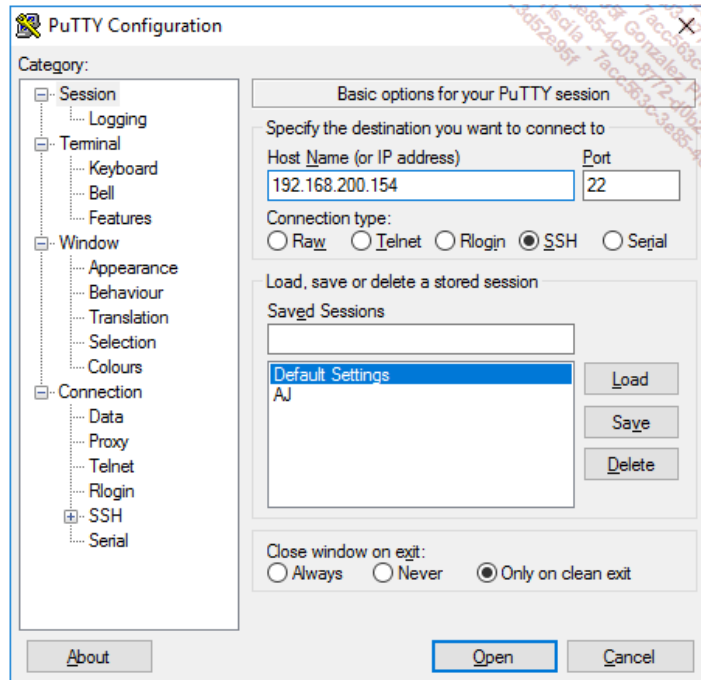
En modo texto

1. En SSH con putty en Windows

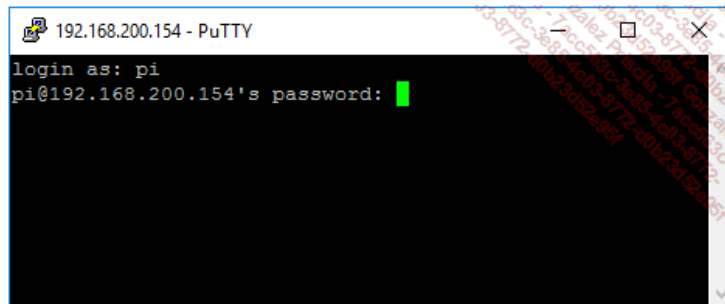
Desde una máquina en Windows, es más sencillo descargar el programa putty disponible como un ejecutable para su descarga gratuita (www.chiark.greened.org.uk/~sgtatham/putty/download.html).

Copie putty.exe en un directorio o en el escritorio.

Muestre la dirección IP de la Raspberry Pi con el control `ifconfig`, por ejemplo (consulte el capítulo Usar la línea de comandos). Ejecute el programa putty.exe en modo administrador.



Verifique que la opción **SSH** está marcada. Rellene la dirección IP de la Raspberry Pi en la zona **Host Name (or IP address)** y pulse el botón **Open**. Se abre una ventana de texto:

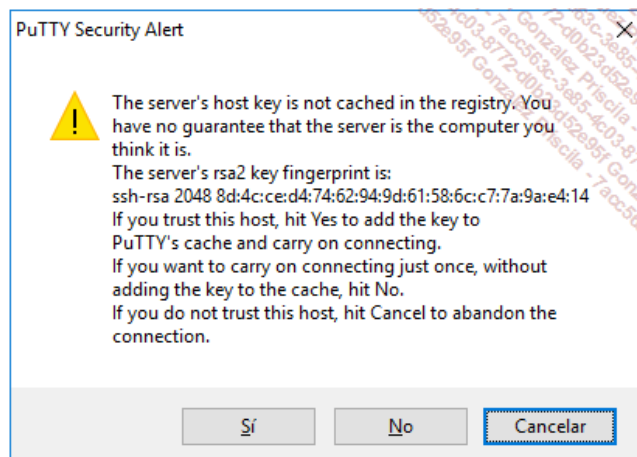


El título de la ventana recuerda a la dirección IP de la máquina en la que el terminal está conectado, así como el nombre del programa utilizado PuTTY.

login as invita al usuario a identificarse. En Raspbian, el usuario por defecto es pi. A continuación, se pide la contraseña para autenticar la solicitud de conexión. La contraseña por defecto de Raspbian es `raspberrypi`.

Durante la primera solicitud de conexión, su máquina recibe la clave pública de la Raspberry Pi. Para evitar toda posibilidad de que una persona malintencionada se cuele en la conversación entre PuTTY y la Raspberry Pi (ataque *Man in the middle* = hombre en medio), una ventana le pide confirmar la clave de la Raspberry Pi, presentando un hash de la clave.

Un hash es el resultado proporcionado por una función, que calcula rápidamente una huella que sirve para identificar un dato proporcionado como entrada. Aquí el dato tratado es la clave pública del servidor. Es posible comprobar de manera segura la integridad de la clave respetando el mismo cálculo en local sobre la clave recibida. Si los dos hash son idénticos, las claves son idénticas. La variación más pequeña (un único bit modificado) produce un resultado totalmente diferente para el hash. Es imposible obtener el dato original a partir del hash.



Acepte esta clave pulsando el botón **Sí**. Se guarda en la base de registro y no se volverá a pedir, salvo si vuelve a instalar Raspbian. En el mundo profesional, la clave se envía por mail o por correo clásico, para garantizar la autenticidad de la conexión.

Para terminar la sesión SSH, introduzca `exit` en la línea de comandos y valide.

➤ Para que los caracteres acentuados se muestren correctamente, es preferible seleccionar en la zona **Window - Translation - Remote Character Set** el juego de caracteres UTF-8. PuTTY dispone de muchas opciones para personalizar la visualización, registrar las sesiones, etc... La información que se ha presentado antes, representa el mínimo necesario para una conexión remota en modo texto.

2. En SSH en Debian

Para conectarse en SSH desde una máquina cuyo sistema operativo sea Debian 7 u otra versión de Linux, el control a utilizar es `ssh`.

Sintaxis

```
ssh nombre_usuario@dirección_IP_raspberry_pi
```

`nombre_usuario` es un nombre de usuario existente en la Raspberry Pi.

`dirección_IP_raspberry_pi` es la dirección IP que se muestra en la Raspberry Pi.

```
root@debian:~# ssh pi@192.168.1.103
The authenticity of host '192.168.1.103 (192.168.1.103)' can't be
established.
ECDSA key fingerprint is
90:28:ec:d8:63:c0:cf:71:aa:14:e3:e9:ce:9f:a3:9c.
Are you sure you want to continue connecting (yes/no)? Yes
Warning: Permanently added '192.168.1.103' (ECDSA) to the list of
known hosts.
pi@192.168.1.103's password:
Linux raspberrypi 3.12.28+ #709 PREEMPT Mon Sep 8 15:28:00 BST 2014
armv6l

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Oct 21 06:28:51 2014 from 192.168.1.102
pi@raspberrypi ~ $
```

Como sucede en Windows, durante la primera conexión el sistema avisa al usuario de que el equipo remoto no está seguro y solicita la confirmación del hash de la clave pública.

Responda **yes** para autorizar la conexión. Informe la contraseña del usuario pi en la Raspberry Pi y valide. La última línea indica que el usuario pi está conectado al ordenador raspberrypi.

El hash de la clave pública se conserva en el directorio personal del usuario que ha solicitado la conexión SSH, en la carpeta `.ssh/known_hosts`.

Para terminar la conexión SSH, escriba `exit`.

```
pi@raspberrypi ~ $ exit
desconexión
Connection to 192.168.1.103 closed.
```

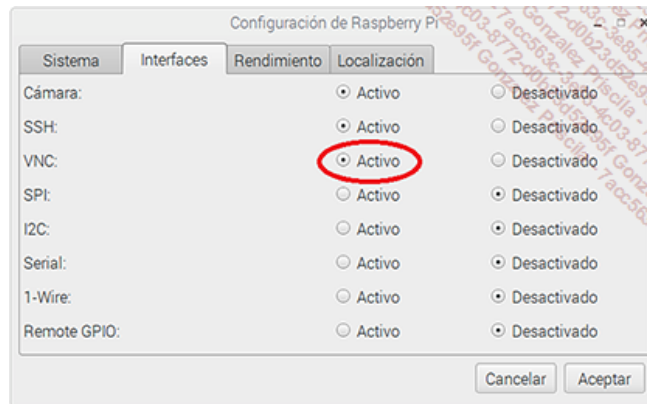
En modo gráfico

1. Principio

Controlar de manera remota la Raspberry Pi también se puede hacer en modo gráfico. En este caso, es necesario activar un servidor en la Raspberry Pi. Su rol será el de transmitir la imagen de la pantalla a la máquina remota. También debe recuperar las pulsaciones del teclado y los movimientos del ratón, transmitidos a través de la red por la máquina remota, y aplicarlos a la sesión actual.

2. Con VNC Server

Para utilizar VNC Server en la Raspberry Pi, en primer lugar es necesario modificar la configuración de la Raspberry Pi (consulte el capítulo Utilizar el modo gráfico).

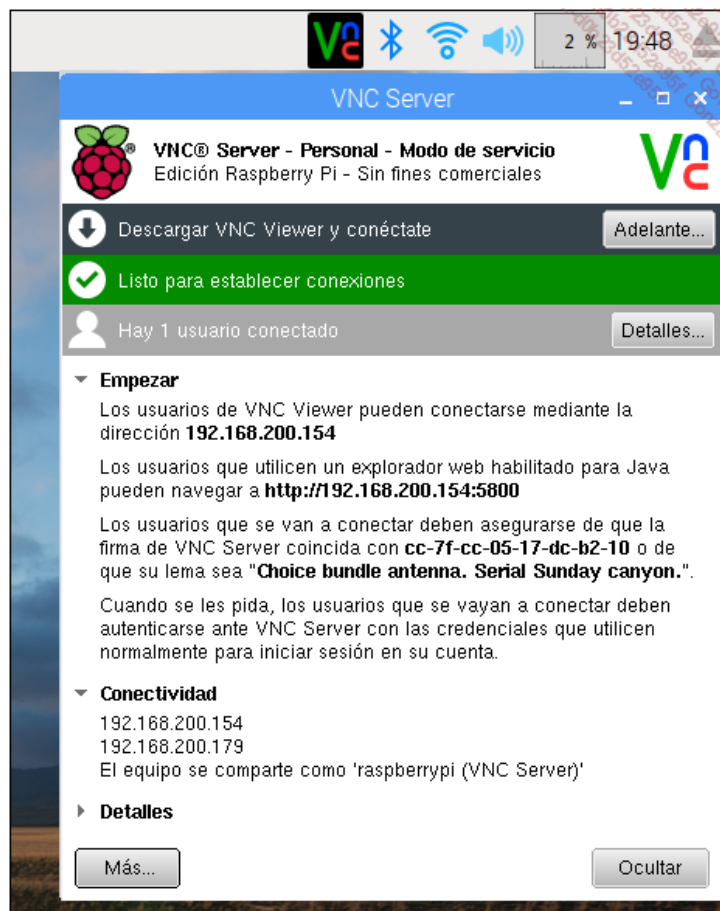


Tras el siguiente inicio, el servidor VNC estará activado.



Esto se indica por la presencia de un icono blanco Vnc a la izquierda del icono Bluetooth.

Un clic izquierdo en este icono abre una ventana de información acerca de VNC Server.



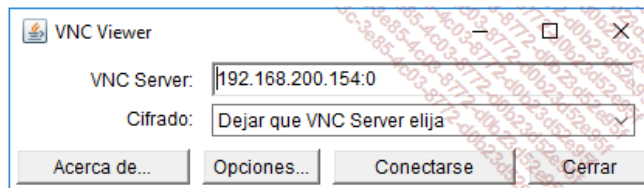
Si se conecta desde un PC equipado con un navegador en el que hay instalado un plugin Java, se puede conectar informando en la barra de dirección de su navegador la dirección de su Raspberry Pi, seguida de :5800.



Aparece una animación Java y debe confirmar si desea ejecutar la aplicación Java VNCviewer.

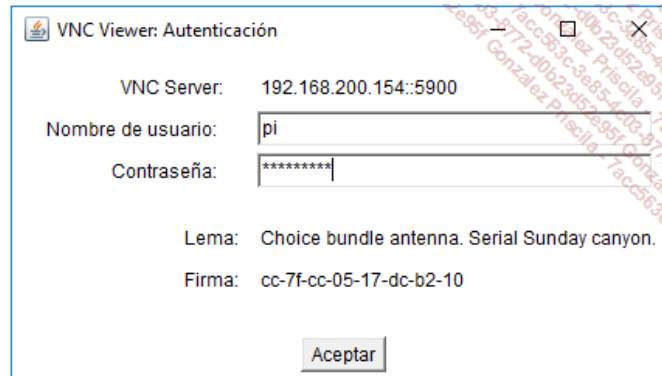
- Java es una tecnología obsoleta que se debería abandonar en un futuro cercano por el gran público. Es una fuente de vulnerabilidad para los navegadores. Sin embargo, hay muchas aplicaciones de negocio desarrolladas en Java y estas aplicaciones deberán persistir algunos años.

Pulse el botón **Ejecutar** para autorizar el inicio de la aplicación.



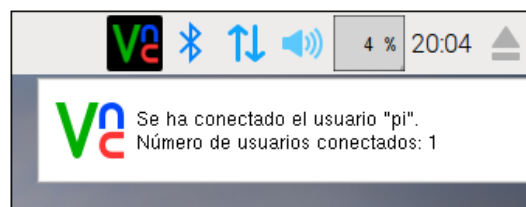
Se abre una nueva ventana con la información relativa a VNC Viewer.

Pulse el botón **Conectarse**.

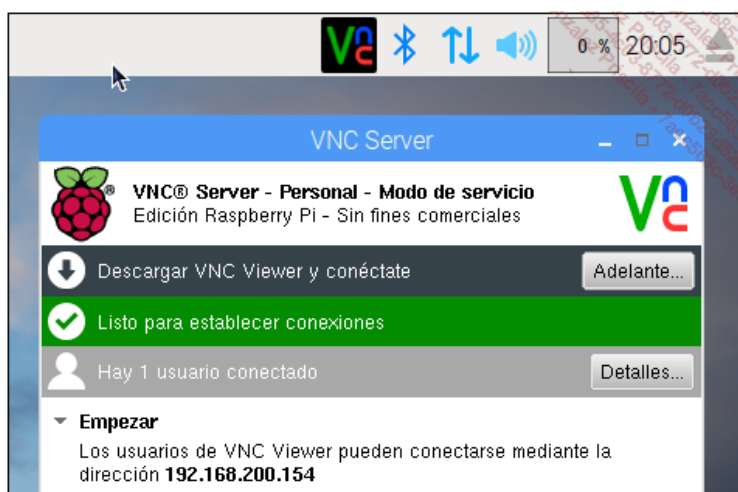


Rellene el login y la contraseña que usa para conectarse a la Raspberry Pi (por defecto **pi** y **raspberrypi**), y valide pulsando el botón **Aceptar**.

Se abre una nueva ventana y la pantalla de la Raspberry Pi se muestra en esta ventana. El icono Vnc se vuelve negro, lo que indica que hay una conexión en curso.



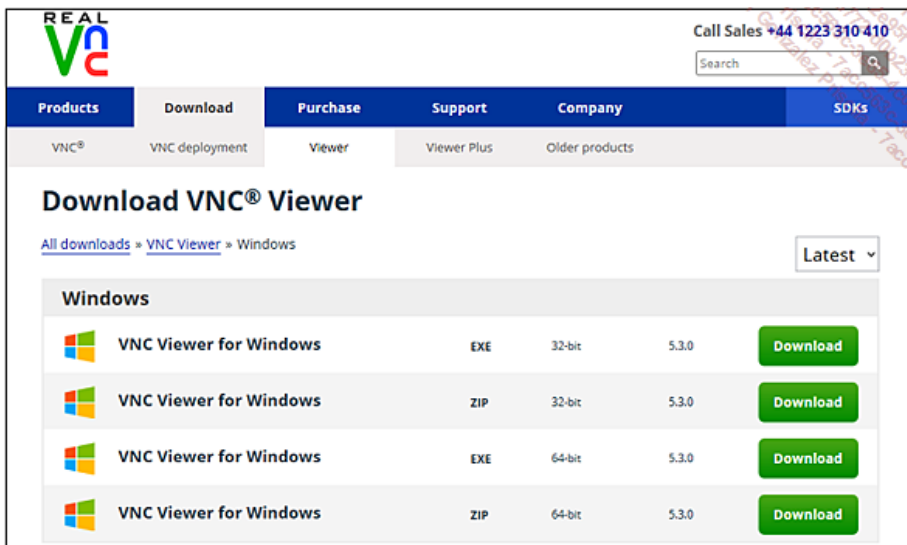
Si pulsa en este icono para mostrar la ventana de información, verá el número de conexiones y podrá obtener más información pulsando en el botón **Detalles**.



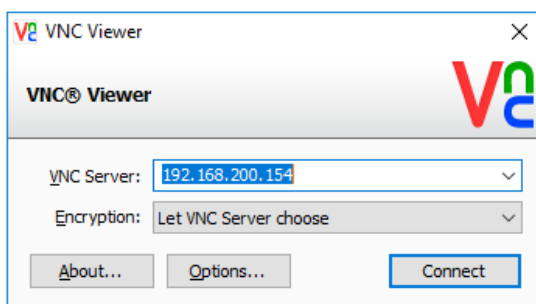
3. Conectarse con VNC Viewer en Windows

Para poder conectarse al servidor VNC instalado en la Raspberry Pi, es necesario instalar un software capaz de comunicarse con VNC Server.

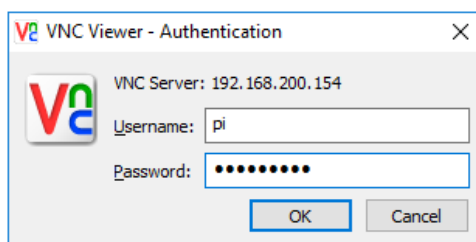
La página <https://www.realvnc.com/download/viewer/windows/latest/> da acceso a la descarga de la versión más reciente de VNC Viewer.



Puede descargar VNC Viewer en versión directamente ejecutable (EXE) o en versión comprimida (ZIP), que habrá que extraer antes de su uso. Pulse dos veces en **VNC-Viewer-5.3.0-Windows-64bit.exe** (el número de versión es susceptible de evolucionar).

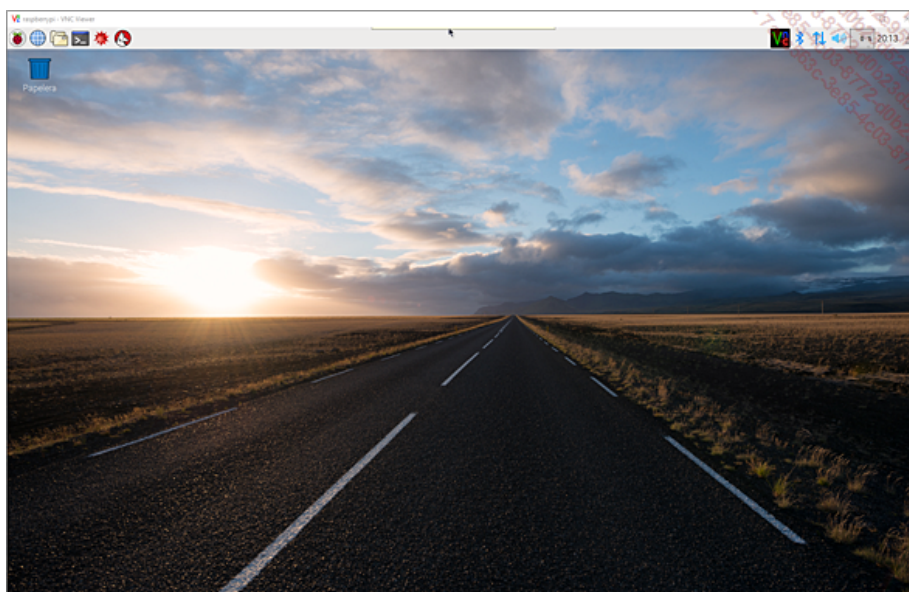


En la ventana **VNC Server**, rellene la dirección de la Raspberry Pi en la que está instalado VNC Server, así como el N° de la pantalla (*display*). Después pulse el botón **Connect** para lanzar **VNC Viewer**.



Se debe autenticar introduciendo el login y la contraseña que usa durante la conexión a la Raspberry Pi (por defecto *pi* y *raspberrypi*). Observe que el número de pantalla se ha cambiado por el puerto utilizado para comunicar con el servidor. Pulse el botón **OK**.

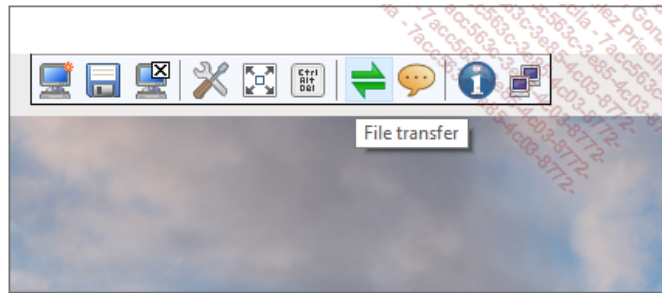
Se abre la ventana del escritorio Raspbian. Está conectado a la Raspberry Pi y la puede utilizar en modo gráfico, de manera remota.



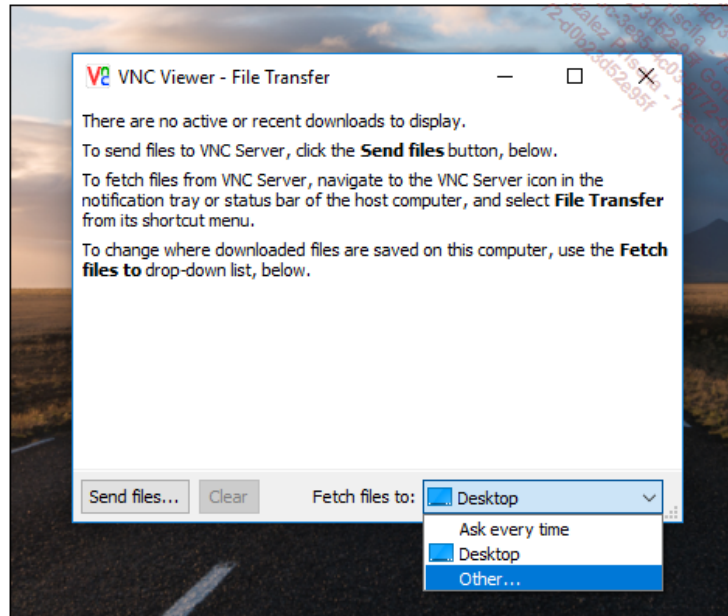
4. Transferir archivos con VNC Viewer

Cuando se establece la unión entre VNC Server, que funciona en la Raspberry Pi, y VNC Viewer en el PC, se hace posible transferir archivos. La transferencia se hace de la Raspberry Pi al PC (del servidor al viewer) o del PC a la Raspberry Pi (del viewer al servidor).

a. Transferencia de archivos de la Raspberry Pi al PC

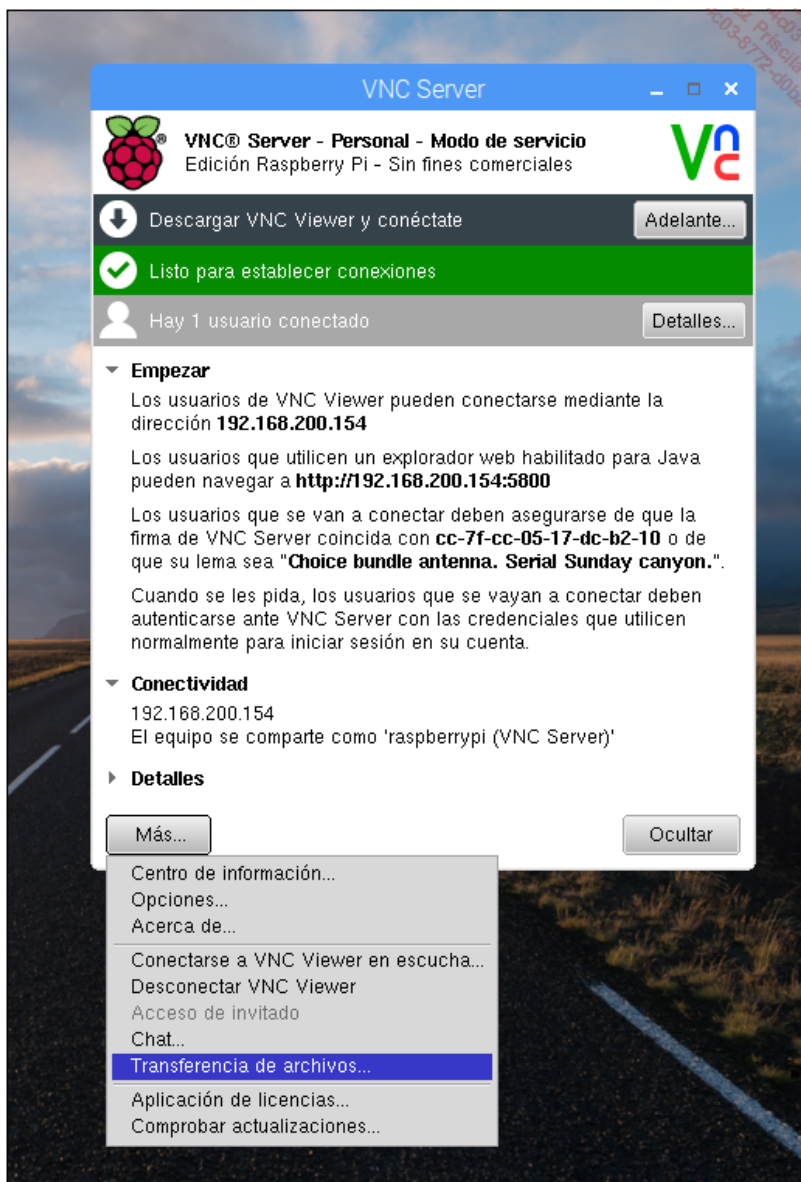


En el PC mueva el ratón por la zona del menú oculto (en la parte superior, en medio de la pantalla). Se abre el menú. Pulse en el icono **File transfer**.



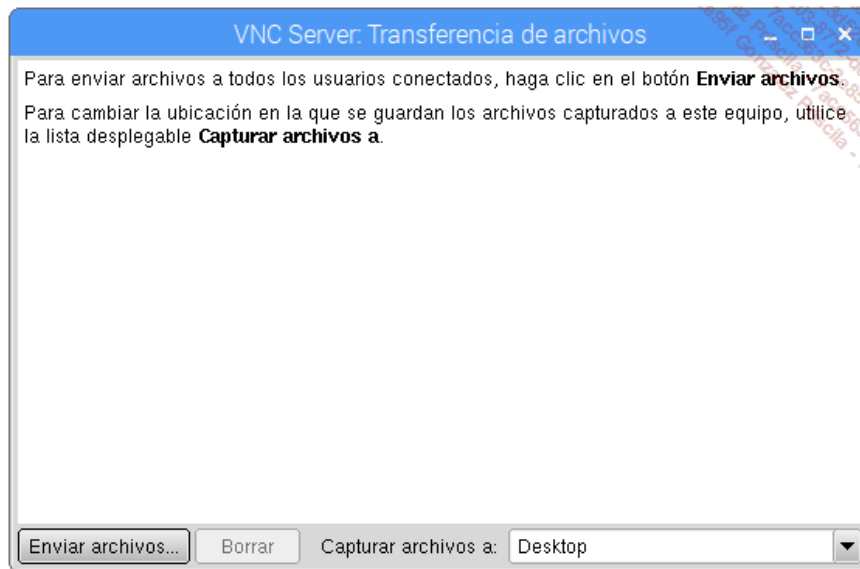
Se abre la ventana de transferencia de archivos en el PC.

En la zona **Fetch files to**: seleccione el lugar donde desea recibir los archivos que se van a transferir.

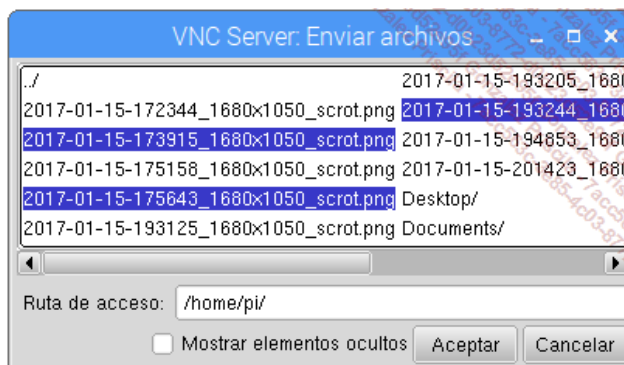


En la Raspberry Pi, abra la ventana de VNC Server pulsando en el icono de la barra de tareas. Se abre la ventana anterior. Pulse el botón **Más** en la parte inferior izquierda de la ventana y después en **Transferencia de archivos**.

Se abre la ventana de transferencia de archivos:

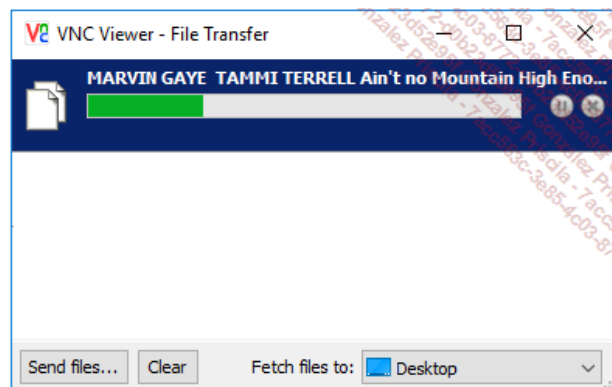


Pulse el botón **Enviar archivos** en la parte inferior izquierda de la ventana. Se abre una ventana que le permite seleccionar los archivos a transferir (use la tecla [Ctrl] para selecciones aisladas, [Shift] para selecciones contiguas).



Cuando haya seleccionado los archivos a transferir, pulse en **Aceptar**.

En el PC una ventana VNC Viewer muestra una barra de progreso, que permite seguir la transferencia en curso. El destino es la carpeta que ha seleccionado en el Viewer.

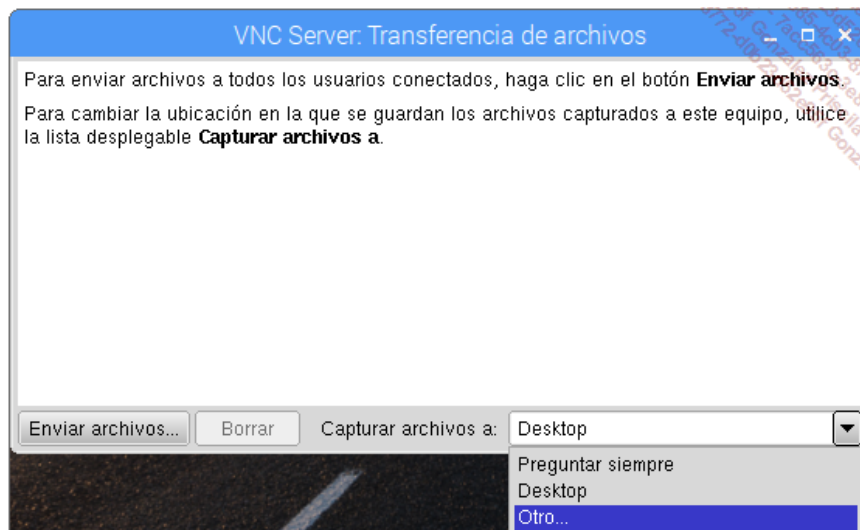


Se le advierte del final de la transferencia de archivos mostrando un mensaje **Descarga terminada**.

b. Transferencia de archivos del PC a la Raspberry Pi

La transferencia en sentido inverso permite enviar archivos a la Raspberry Pi.

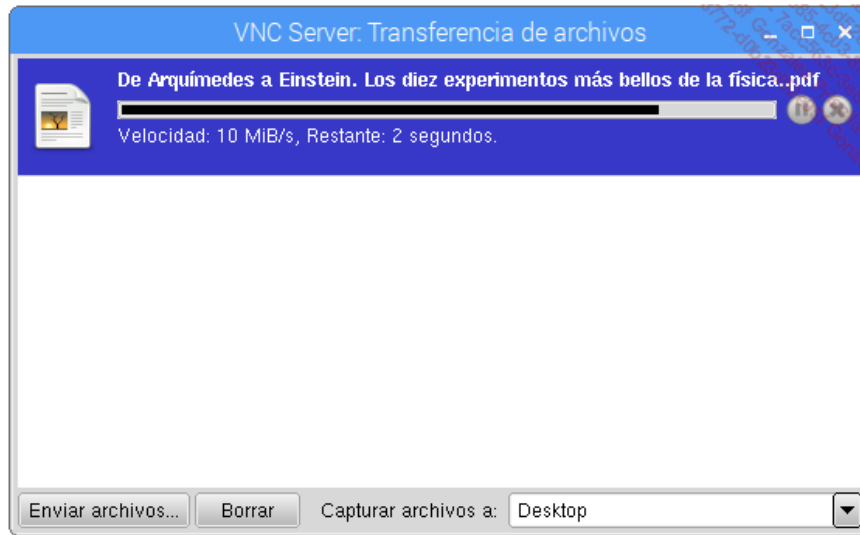
En la Raspberry Pi (VNC Server), abra la ventana pulsando en el icono VNC en la barra de tareas. Pulse el botón **Más** en la parte inferior izquierda de la ventana. Pulse en **Transferencia de archivos**.



Seleccione la carpeta en la que desea recibir los archivos que llegarán del PC, en la zona **Capturar archivos a:**.

En el PC (VNC Viewer) abra el menú oculto en la parte superior de la pantalla y pulse en el icono **Transferencia de archivos**.

Se abre la ventana **Transferencia de archivos**, pulse el botón **Enviar archivos** en la parte inferior izquierda. Seleccione los archivos a enviar en la ventana que se abre, y después pulse el botón **Abrir**. Los archivos se envían a la Raspberry Pi.



En la pantalla de la Raspberry Pi, una ventana muestra una barra de progreso y después anuncia el fin de la descarga.

➤ Si hay un archivo con el mismo nombre en la carpeta destino, los caracteres (1) van a completar el nombre del nuevo archivo.

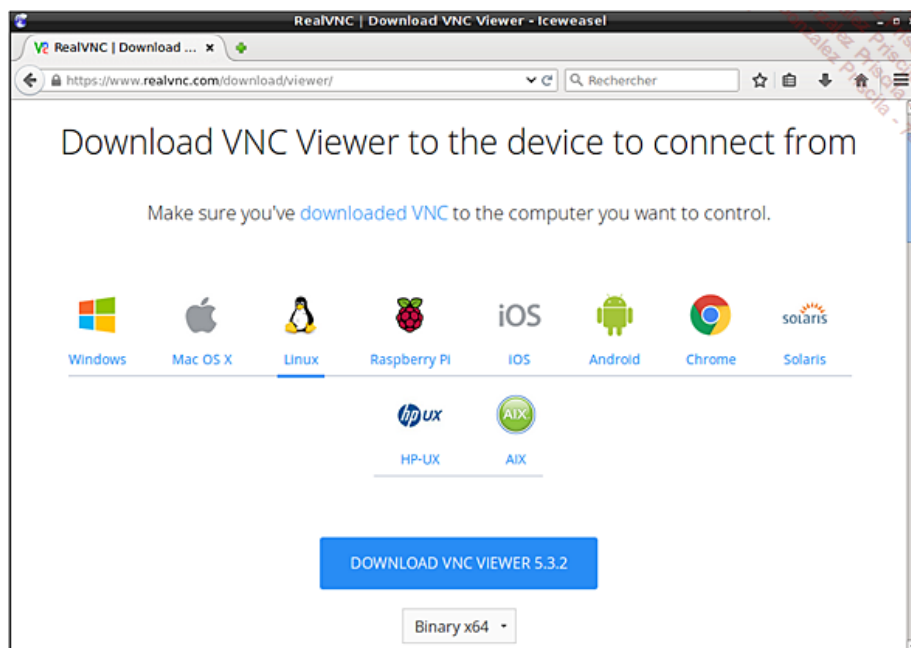
Conclusión

Con la integración de VNC en Raspbian, la Fundación ofrece a los usuarios un medio sencillo de transferir archivos entre la Raspberry Pi y otras máquinas.

5. Conectarse con VNC Viewer en Linux

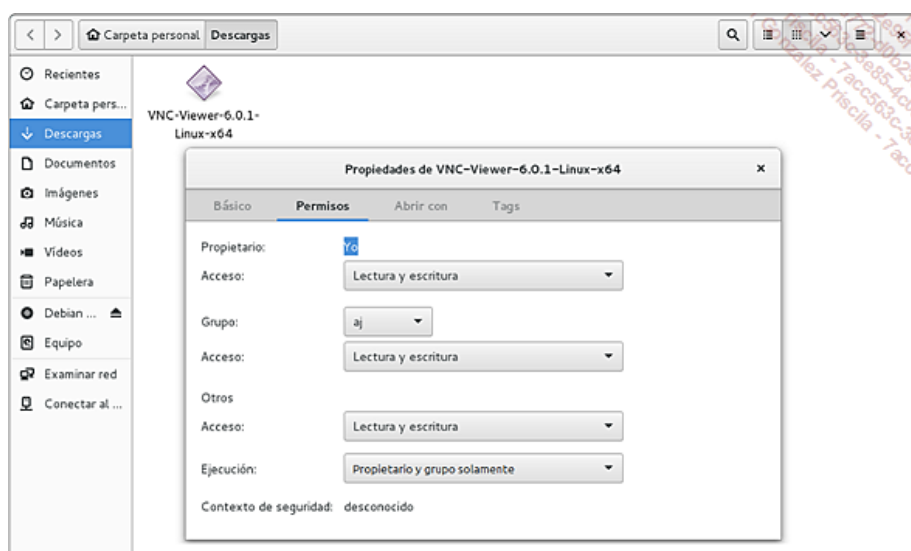
Desde una máquina en Linux, la conexión al servidor VNC instalado en la Raspberry Pi, también necesita la instalación de un software capaz de comunicar con VNC Server. Para Debian, VNC pone a disposición un VNC Viewer que puede cumplir esta función.

En primer lugar, es necesario descargar el cliente VNC Viewer en su máquina Debian.



Vaya a la página de descarga de VNC Viewer y descargue la versión adaptada a su máquina. Aquí, para Debian, es la versión Linux ejecutable (Binary x64) la que se ha seleccionado.

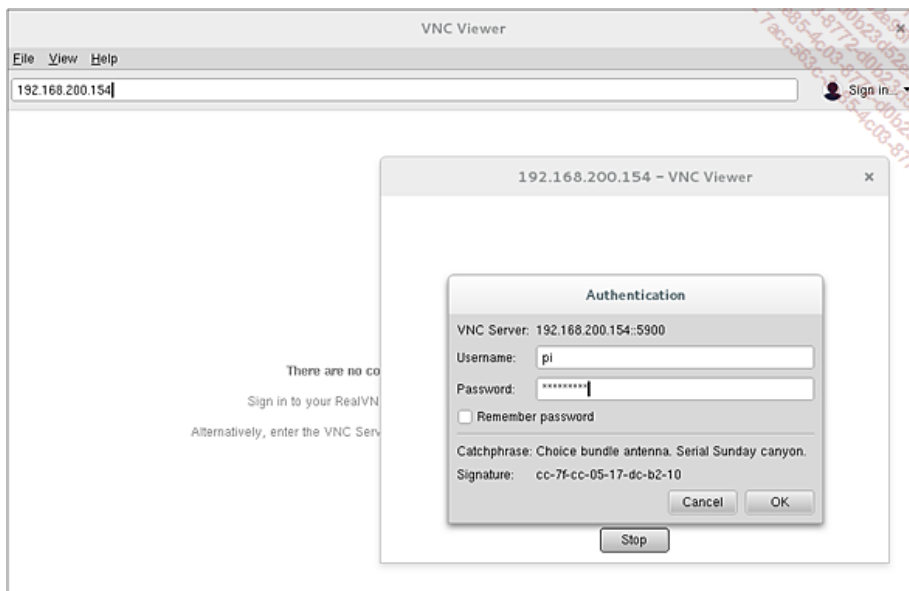
Después de la descarga, encontrará el archivo en la carpeta correspondiente, en función del sistema. En Debian se guarda en la carpeta **Descargas**.



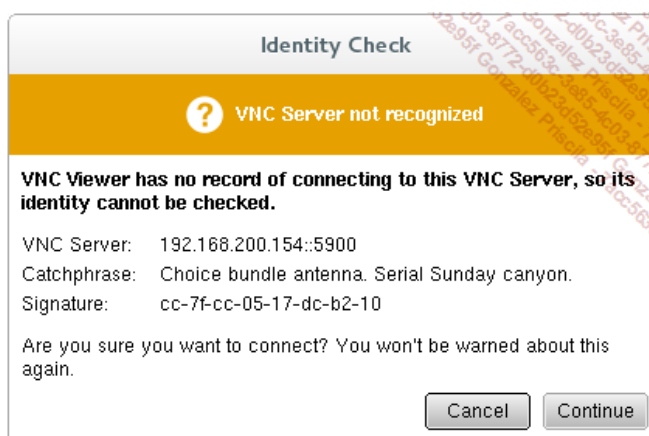
Haga clic con el botón derecho en el archivo VNC-Viewer-6.0.1-Linux-x64 (el nombre puede variar en función de su sistema o del número de versión) y hágalo ejecutable por su propietario y su grupo (lista desplegable **Ejecución**).

Lance el programa haciendo doble clic sobre él. Confirme que desea ejecutarlo.

Encontrará exactamente las mismas pantallas que en la versión Windows.

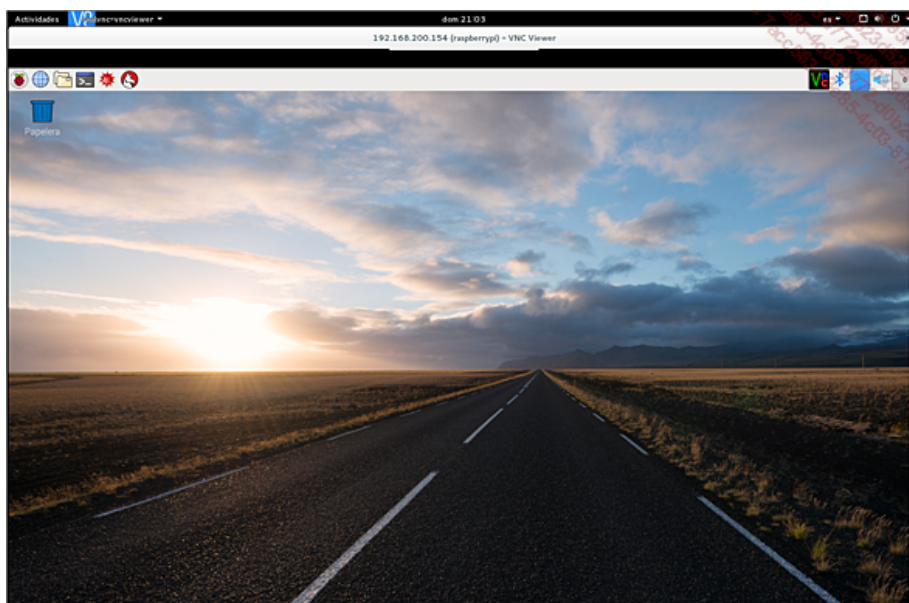


Rellene la dirección y el puerto de la Raspberry Pi, y después el login y la contraseña.



Como cada vez que VNC Viewer se conecta a un servidor VNC desconocido, le solicita que confirme que ha reconocido la firma del servidor y que autoriza la conexión.

Pulse el botón **Continue**.



VNC Viewer ahora está conectado a su Raspberry Pi y su escritorio se muestra en la ventana de VNC Viewer. Puede utilizar la Raspberry Pi como si utilizara directamente el teclado y el ratón que tiene conectados. Todos los movimientos del ratón en alguna de las dos máquinas se reproducen en las dos pantallas. Ocurre lo mismo con las entradas desde el teclado.

Puede salir de la aplicación VNC Viewer pulsando en la cruz situada en la parte superior derecha de la ventana.

6. Instalar TightVncServer en la Raspberry Pi

La instalación de TightVncServer necesita la desinstalación de VNC Server, que está instalado originalmente en la Raspbian PIXEL. Esta usa un servidor VNC alternativo que permite mostrar cómo configurar el arranque automático de una aplicación en systemd.

Desinstale VNC Server:

```
pi@raspberrypi:~ $ sudo apt-get purge realvnc-vnc-server
Lectura de las listas de paquetes... Terminado
Construcción del árbol de dependencias
Lectura de la información de estado... Terminado
```

```
Se eliminarán los siguientes paquetes:
realvnc-vnc-server*
0 actualizaciones, 0 instalaciones, 1 eliminado y 0 no actualizados.
Después de esta operación, se van a liberar 28,3 MB de espacio.
¿Desea continuar? [S/N]
```

Responda "S" para terminar la instalación. Reinicie su Raspberry Pi.

Antes de instalar TightVncServer (o cualquier otro software), realice una actualización de su sistema operativo. Esto garantiza que usará las versiones disponibles más recientes del software.

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

Cuando termine la actualización, instale TightVncServer:

```
sudo apt-get install tightvncserver
```

Cuando se instale TightVncServer, lance su ejecución para comprobar que se ha instalado y funciona correctamente:

```
pi@raspberrypi:~ $ tightvncserver

You will require a password to access your desktops.

Password:xxxxxxx
Warning: password truncated to the length of 8.
Verify:xxxxxxx
Would you like to enter a view-only password (y/n)? n

New 'X' desktop is raspberrypi:1

Creating default startup script /home/pi/.vnc/xstartup
Starting applications specified in /home/pi/.vnc/xstartup
Log file is /home/pi/.vnc/raspberrypi:1.log
```

El software le pide, en primer lugar, una contraseña que autorice el acceso remoto. Observe que esta contraseña se trunca a ocho caracteres. Por tanto, es inútil rellenar más caracteres. Después de volver a introducir la contraseña, TightVncServer solicita si desea introducir una contraseña para el modo visual solamente. En este modo, las acciones del usuario en el teclado y el ratón no se transmiten a TightVncServer. Por tanto, el usuario es un simple espectador, incapaz de actuar en el escritorio mostrado.

La instalación de TightVncServer no garantiza el inicio automático del programa en cada reinicio de la Raspberry Pi. Si desea ejecutar el programa durante el siguiente uso de la Raspberry Pi, habría que solicitarlo manualmente desde la línea de comandos:

```
pi@raspberrypi:~/vnc $ tightvncserver

New 'X' desktop is raspberrypi:1

Starting applications specified in /home/pi/.vnc/xstartup
Log file is /home/pi/.vnc/raspberrypi:1.log
```

7. Arranque automático de TightVncServer

El arranque automático de TightVncServer en systemd necesita algunas operaciones iniciales:

- creación del servicio en systemd
- instalación del servicio
- reinicio de la Raspberry Pi

Creación del servicio en systemd

Cree con nano el archivo `/etc/systemd/system/vncserver@.service`

```
sudo nano /etc/systemd/system/vncserver@.service
```

Inserte en este archivo el siguiente contenido:

```
[Unit]
Description=Servicio de escritorio en modo remoto (VNC)
After=syslog.target network.target

[Service]
Type=forking
User=pi
PAMName=login
PIDFile=/home/pi/.vnc/%H:%i.pid
ExecStartPre=-/usr/bin/vncserver -kill:%i > /dev/null 2>&1
ExecStart=/usr/bin/vncserver -depth 24 -geometry 1280x800:%i
ExecStop=/usr/bin/vncserver -kill:%i

[Install]
WantedBy=multi-user.target
```

➤ Este archivo puede servir de base cuando sea necesario ejecutar una aplicación durante el arranque del sistema. Hay que sustituir la información relativa a `tightvncserver` por aquella del programa a lanzar.

Explicación del contenido del archivo

➤ Estas explicaciones usan nociones avanzadas que no son necesarias para el procedimiento de instalación anterior.

[Unit]

Esta sección [Unit], contiene la información genérica de la unidad.

Description=Servicio de escritorio en modo remoto (VNC)

Esta cadena de caracteres describe la unidad. Su contenido es libre, puede informar lo que quiera que aparezca cuando se muestre el servicio con `systemctl`. Esta línea está destinada a ofrecer información que describe la unidad. Su descripción debe significar algo para el usuario final.

«Servicio de escritorio en modo remoto (VNC)» es un buen ejemplo. «Servidor de control ligero» es peor, porque es muy genérico e incluso «vncserver», que es demasiado específico y carece de sentido para las personas que no conozcan esta aplicación.

After=syslog.target network.target

After indica que el servicio actual VNCserver solo se arrancará después del inicio efectivo de los servicios listados (y separados por un espacio).

[Service]

Los archivos de los servicios deben incluir obligatoriamente una sección « [Service] » con la información acerca del servicio y del proceso que supervisa. Estas opciones, documentadas en `systemd.exec` y `systemd.kill`, son las siguientes:

Type=forking

Si Type se define como **forking**, el proceso configurado con **ExecStart=** (siguiente imagen) llamará a la función `fork ()` durante su arranque. El proceso padre termina cuando el arranque ha terminado y todos los canales de comunicación están establecidos. El proceso hijo continúa funcionando como proceso demonio principal. Es el comportamiento de los demonios UNIX tradicionales. Si se utiliza este argumento, se recomienda utilizar la opción **PIDFile=** para que `systemd` pueda identificar el proceso principal del demonio. `systemd` arrancará las siguientes unidades tan pronto como se detenga el proceso padre.

User=pi

Permite designar al usuario que lanzará el proceso.

PAMName=login

Define bajo qué nombre arranca la sesión. Si se define, el proceso se registrará como sesión PAM (*Pluggable Authentication Modules* = Módulo de autenticación adicional) bajo el nombre de servicio específico. Esto se utiliza en asociación con la opción **User=** (imagen anterior).

PIDFile=/home/pi/.vnc/%H:%i.pid

/home/pi/.vnc/ designa la ruta (el lugar) donde se guardará el archivo de PID.

%H:%i.pid es el nombre del archivo, con **%H** el nombre de la máquina y **%i** el número de la instancia de la unidad.

En la práctica tenemos:

```
ls ~/.vnc
passwd raspberrypi:1.log raspberrypi:1.pid xstartup
```

y en el archivo `raspberrypi:1.pid`:

```
cat ~/.vnc/raspberrypi:1.pid
562
```

Con 562 que es el ID del proceso.

ExecStartPre=/usr/bin/vncserver -kill:%i > /dev/null 2>&1

`ExecStartPre=`: es el control que se ejecuta antes de `ExecStart`.

`usr/bin/vncserver -kill:%i`: este control «mata» cualquier escritorio VNC que previamente estuviera ejecutándose por el servidor. Lo hace matando el proceso cuyo PID se almacena en el archivo « `/home/pi/.vnc/host: instance.pid` » (ver imagen anterior).

`/dev/null 2>&1` redirige cualquier salida de este control a un «agujero negro»: no se muestra nada.

ExecStart=/usr/bin/vncserver -depth 24 -geometry 1280x800:%i

Si el servicio se lanza con `start`, esta línea va a lanzar el servidor VNC con una profundidad de color de 24 bits (**-depth 24**), un tamaño de pantalla de 1280x800 píxeles (**-geometry 1280x800**) en la instancia `:i%`

ExecStop=/usr/bin/vncserver -kill:%i

Si el servicio se detiene con `stop`, esta línea va a «matar» a la instancia actual.

[Install]

WantedBy=multi-user.target

La sección «[Install]» tiene información para la instalación de la unidad. Esta sección no se interpreta por `systemd` durante la ejecución. Solo se utiliza exclusivamente por los controles `enable` y `disable` de la herramienta `systemctl`, durante la instalación de una unidad.

WantedBy=multi-user.target

Los `target` de `systemd` se parecen a los `runlevel` que habíamos visto anteriormente. Tienen un nombre en lugar de un número. Aquí `multi-user` corresponde al `runlevel 3`, es decir el modo texto (no gráfico) multi-usuario... Se crea un enlace simbólico en el directorio `.wants/` de cada una de las unidades listadas cuando esta unidad se instala con `systemctl enable`. El resultado es que la unidad actual se lanzará cuando la unidad cuyo nombre se indica después de `WantedBy` se lance.

Instalación del servicio

Para instalar el servicio, el control `systemctl daemon-reload` recarga `systemd` y permite comprobar si existen nuevas unidades o si algunas unidades se han modificado. Después, el control `systemctl enable` informa a `systemd` que la unidad se debe lanzar durante el arranque de la Raspberry Pi.

```
sudo systemctl daemon-reload
sudo systemctl enable vncserver@1.service
Created symlink from /etc/systemd/system/multi-user.target.wants/
vncserver@1.service to /etc/systemd/system/vncserver@.service.
```

La respuesta de `systemctl`: `Created symlink...` indica que la operación se ha desarrollado correctamente.

➤ La cifra 1 que figura en el control después el carácter @ indica el número de la sesión de `vncserver` que se ejecutará.

Reinicio de la Raspberry Pi

El reinicio de la Raspberry Pi va a provocar el inicio automático de `tightvncserver`.

```
sudo reboot
```

Se puede comprobar que `tightvncserver` funciona correctamente:

```
systemctl | grep vnc
vncserver@1.service
loaded active running Servicio de escritorio en modo remoto (VNC)
system-vncserver.slice
loaded active active system-vncserver.slice
```

Parada del servicio

Si no desea utilizar más `tightvncserver`, lo puede desactivar.

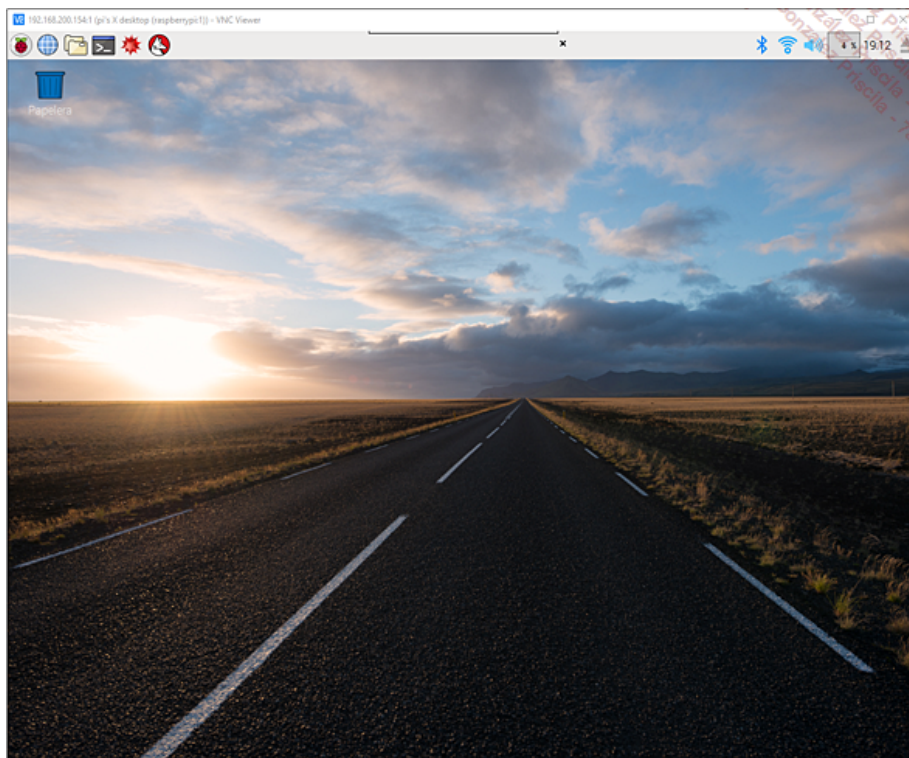
```
sudo systemctl disable vncserver@1.service
```

Si no desea que `tightvncserver` se lance más durante el arranque de la Raspberry Pi, elimine el arranque del servicio.

```
sudo systemctl remove vncserver@1.service
```

Conexión a tightvncserver

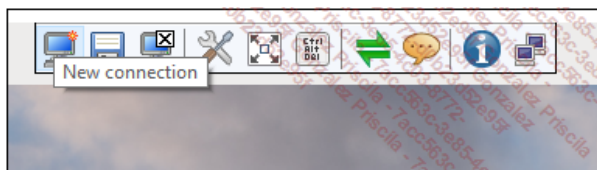
Utilice VNC Viewer como se indica anteriormente, en la implementación de VNC Viewer en Windows o Debian.



Observando esta ventana, puede observar dos diferencias respecto a la ventana original de la Raspberry Pi:



En la parte superior y en el centro de la pantalla figura un rectángulo (1) que, cuando se pasa el ratón por encima, da acceso al menú de VNC Viewer (nueva conexión, registrar la conexión, cerrar la conexión, modo pantalla completa,...):



La otra diferencia es la presencia de un cursor en forma de aspa (2), que es el cursor por defecto de X11. Para modificar el tipo de cursor hay que modificar el archivo `/home/pi/.vnc/xstartup` añadiendo `-cursor_name left_ptr` al final de la línea `xsetroot`:

```
xsetroot -solid grey -cursor_name left_ptr
```

Tras de un reinicio, cuando se vaya a conectar con VNC Viewer, el cursor tendrá la apariencia habitual (una flecha orientada a la izquierda).

Cuando la ventana le dé acceso al escritorio, actúe en la Raspberry Pi como si utilizara el teclado y el ratón que tiene conectados. La conexión en modo remoto funciona igual de bien tanto a pocos metros de distancia como a cientos de kilómetros.

- Con `tightvncserver`, está conectado a otra sesión diferente de la que está abierta en la Raspberry Pi. Los movimientos del ratón y las pulsaciones del teclado no se reproducen en las dos pantallas.

8. Conclusión

Algunas veces es necesario poder utilizar una Raspberry Pi de manera remota. Incluso es posible que este sea el único medio de ejecutar los comandos sobre una Raspberry Pi inaccesible.

El uso de soluciones de conexión de manera remota, facilita la gestión de la Raspberry Pi, independientemente de las circunstancias.

Conceptos básicos de red

1. Introducción

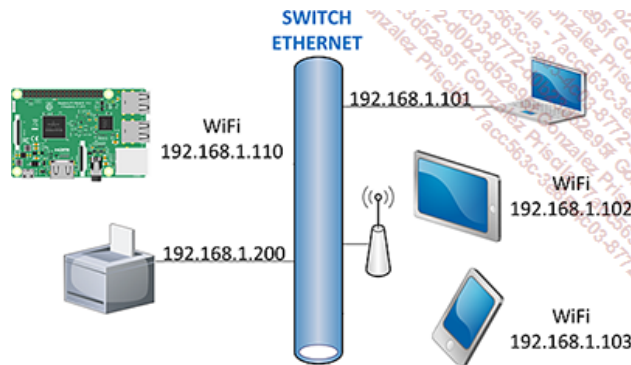
Muchas aplicaciones implican la implementación de la Raspberry Pi en la red. Ya sea sobre una red local (detrás de una unidad de acceso, por ejemplo) o para permitir un acceso desde Internet a una Raspberry Pi situada en una red local, esta conexión plantea normalmente preguntas. En particular, se debe al hecho de que conectar alguna máquina detrás de una unidad de conexión se puede hacer en la actualidad sin ninguna habilidad especial. Por el contrario, en caso de funcionamiento incorrecto, la búsqueda del error solo se puede hacer apoyándose en el entendimiento de los mecanismos implicados.

La presentación de los siguientes conceptos básicos de red no pretende ser ni exhaustiva ni completamente exacta. Contiene la mínima información para que un usuario que descubra este dominio pueda configurar su máquina y entender el funcionamiento de la red.

- Las soluciones que se presentan en este capítulo únicamente tienen un carácter pedagógico y no abordan los problemas de seguridad de las máquinas conectadas a Internet. Sea extremadamente prudente si una Raspberry Pi que realiza funciones importantes es accesible a través de Internet.

2. Como una carta en el correo

Una red informática permite a diferentes máquinas intercambiar datos. Estos datos se presentan como paquetes que contienen unos y ceros, que se envían por los cables de la red o por radio, en el caso del Wi-Fi.



Como en el caso del correo que recibe cada día, los paquetes tienen una dirección de remitente y una dirección de destino. Cuando un paquete se envía a la red, los dispositivos encargados de distribuirlos pueden saber también hacia qué otro dispositivo es necesario enviarlo. Es exactamente lo que pasa con una carta que deposita en un buzón. El enrutamiento de su correo se hace en función de la dirección que se indica en el sobre.

El enrutamiento de los datos se puede hacer con hardware (cable de red) o por radio (Wi-Fi).

a. Asignación de direcciones

En la red que se representa más arriba figuran algunos dispositivos conectados solo a un switch (o al conmutador). La asignación de direcciones se hace en este caso por el usuario. Se dice que el direccionamiento es estático, porque las direcciones se asignan manualmente una vez para todos.

Dispositivo	Dirección
Raspberry Pi	192.168.1.110
Ordenador móvil	192.168.1.101
Tableta	192.168.1.102
Smartphone	192.168.1.103
Impresora	192.168.1.200

b. Dirección física

Antes de que una dirección IP se asigne a un ordenador, este debe poder comunicarse en la red, en particular si debe hacer una consulta para obtener su dirección IP automáticamente. En este caso, el inicio de la "conversación" se hace usando la dirección física de la tarjeta de red. Posteriormente, el diálogo puede continuar usando la dirección IP cuando el ordenador ha recuperado una.

La dirección física también se llama dirección MAC (*Media Access Control*). Es única (salvo si la modifica), porque cada interfaz de red está numerada por su fabricante.

Encontrar la dirección física

```
pi@raspberrypi:~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:84:90:03
          dir inet6: fe80::613:94b0:2db4:d3ae/64 Scope:enlace
...
wlan0     Link encap:Ethernet  HWaddr b8:27:eb:d1:c5:56
          inet dir:192.168.1.110 Bcast:192.168.1.255
          Máscara:255.255.255.0
          dir inet6: fe80::c0da:6a5d:7ed6:6aa3/64 Scope:enlace
```

Un `ifconfig` ejecutado sobre una Raspberry Pi 3 en la que solo se ha configurado la interfaz Wi-Fi muestra que cada interfaz de red tiene una dirección física.

El puerto Ethernet `eth0` tiene la dirección física (**HWaddr** = *Hardware address*) :27:eb:84:90:03. Para la interfaz `wlan0` (Wi-Fi) la dirección es b8:27:eb:d1:c5:56.

La dirección MAC está compuesta por 48 bits, es decir 6 bytes. Los tres bytes de la izquierda identifican al constructor (aquí b8:27:eb = la Fundación Raspberry) y los tres bytes de la derecha representan el identificador único de la interfaz.

- Por convención, en la dirección MAC o dirección física, los bytes se escriben en hexadecimal.

c. ¿Qué es una dirección IP?

IP (*Internet Protocol*) es el protocolo adoptado para gestionar el intercambio de información en la red Ethernet. Las direcciones IP son direcciones utilizadas por el protocolo IP para encaminar los paquetes a su destino.

Una dirección IP se compone de 4 bytes (es decir 32 bits), traducidos en decimal como una sucesión de cuatro números comprendidos cada uno entre 0 y 255, separados por un punto.

192 . 168 . 1 . 110

La dirección anterior es un ejemplo de lo que puede observar en su Raspberry Pi. En modo gráfico, deje un momento el ratón sobre el icono de red para que aparezca la configuración de las interfaces:



La captura de pantalla anterior se ha realizado en una Raspberry Pi conectada al Wi-Fi (wlan0). El puerto Ethernet (toma RJ45) no está conectado.

En modo texto, el comando `ifconfig` proporciona esta información:

```
pi@raspberrypi:~ $ ifconfig
wlan0    Link encap:Ethernet  HWaddr b8:27:eb:d1:c5:56
         inet dir:192.168.1.110  Bcast:192.168.1.255
         Máscara:255.255.255.0
...
```

La dirección de la Raspberry Pi en la interfaz wlan0 (el Wi-Fi) es **192.168.1.110**.

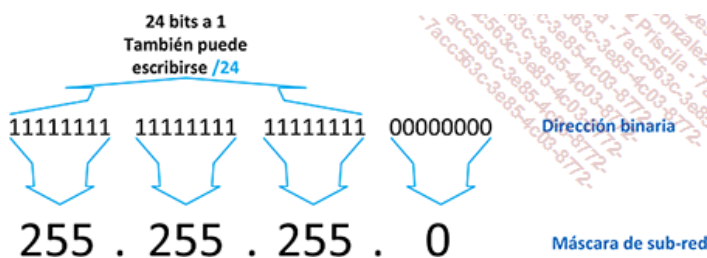
➤ En una misma red, cada dirección IP debe ser única. Si dos ordenadores tienen la misma dirección IP, esto provoca un "conflicto de direcciones".

d. Utilidad de la máscara de sub-red

La dirección IP por sí misma no es suficiente para situar de manera precisa un ordenador en una red. El dato adicional indispensable se llama máscara de sub-red, en inglés *netmask* o *subnet-mask*.

Esta máscara de sub-red se puede expresar de dos maneras:

- Por el número de bits a 1 que tiene, aquí 24 bits, lo que se indica en modo gráfico por /24:



- Por la conversión a decimal de estos 24 bits a 1, seguidos de 8 bits a cero: 255.255.255.0

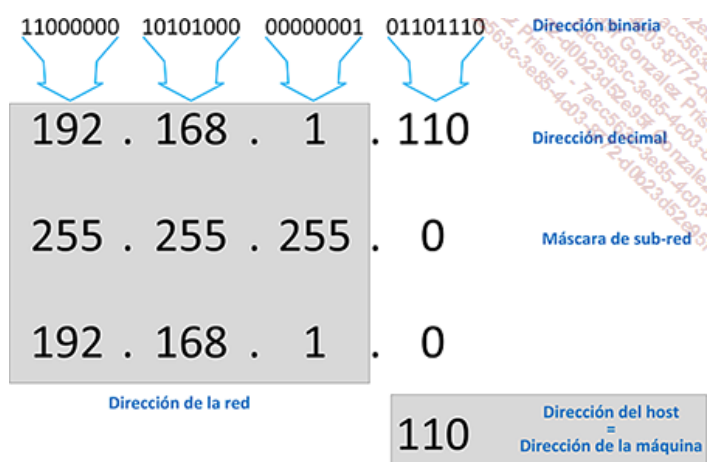
Ambas formas son equivalentes y designan el mismo valor.

La máscara de sub-red permite descomponer la dirección IP en dos partes:

- La dirección de red en la que se encuentra el ordenador.
- La dirección del ordenador en sí misma, también llamada dirección del host o *host address*.

Cálculo de la dirección de red

A partir de la dirección del host y de la máscara de sub-red es posible determinar en qué red se sitúa el ordenador. Además, esta operación se realiza sistemáticamente cuando un ordenador accede a la red.



Si se superponen como se muestra más arriba la dirección IP y la máscara de sub-red, el cálculo de la dirección de red consiste en conservar solo las cifras de la dirección IP en las que figura el número 255. Cuando la máscara vale 0, esta parte de la dirección de red también vale 0. Esta operación proporciona la dirección de la red, aquí 192.168.1.0.

➤ La operación realizada entre la dirección IP y la máscara de sub-red es un Y lógico, realizado en binario. La operación es sencilla cuando la máscara solo está formada por números 255. Sin embargo, se necesitan más cálculos (conversiones decimales > binario y binario > decimal cuando la máscara de sub-red es más complicada, como por ejemplo 255.255.240.0...).

e. Dirección del host

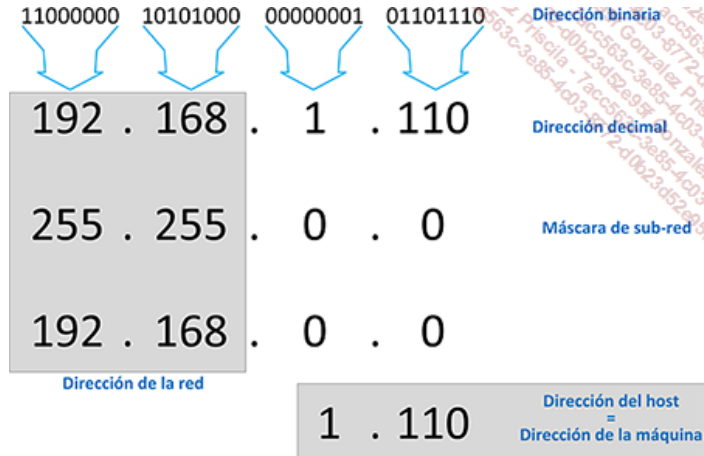
En esta red, el host tiene una dirección que también se puede deducir de su dirección IP y de la máscara de sub-red: en el ejemplo anterior, la dirección del host es el número que se encuentra debajo de los 0 de la máscara de sub-red, es decir 110.

En la red 192.168.1.0, los ordenadores pueden ocupar todas las direcciones comprendidas entre 0 y 255 (salvo algunas excepciones, como se explica más adelante). Esto significa que los ordenadores de esta red podrían tener direcciones comprendidas entre 192.168.1.0 y 192.168.1.255.

Por tanto, habrá posibilidad de conectar 256 máquinas a esta red.

Máscara de sub-red diferente

La máscara de sub-red podría tener un valor diferente. Por ejemplo /16, es decir 255.255.0.0. En este caso el método de cálculo sigue siendo exactamente el mismo:



Esta vez, la red es 192.168.0.0. La gran diferencia viene de la dirección del host, que esta vez está escrita con 2 bytes, es decir 16 bits. Las direcciones de los ordenadores de esta red pueden ir desde 0.0 a 255.255 (con las mismas excepciones que antes). Esta vez son 256 x 256 = 65.536 máquinas las que se podrían conectar a la red.

➤ Hay redes con un máscara de /8 es decir, 255.0.0.0. En este caso, son más de 16 millones de máquinas las que puede acoger la red.

f. Las direcciones particulares

Red

La dirección de la red calculada anteriormente se reserva para la red. Convencionalmente, ninguna máquina de la red tendrá esta dirección. En los ejemplos anteriores, las direcciones 192.168.1.0 y 192.168.0.0 no se podrán asignar a un ordenador, porque son direcciones de red.

Broadcast

Cuando un ordenador realiza una petición a la red sin saber a qué dirección, se envía a una dirección particular, llamada dirección de difusión o *Broadcast Address*. A continuación, los datos se envían a todas las máquinas presentes en la red. Por convención, esta dirección es la última dirección disponible en la red. En los ejemplos anteriores, esta dirección valdría 192.168.1.255 o 192.168.255.255. Ninguna máquina de la red puede tener esta dirección.

➤ El uso de la dirección de red y de la dirección de difusión reduce la franja de direcciones disponibles de dos unidades. Una máscara de sub-red con /24 podría componer 254 direcciones y un máscara con /16 dejaría 65.534 direcciones disponibles para los hosts.

Direcciones privadas

Dentro de una red local, las direcciones utilizadas obligatoriamente son direcciones privadas. Estas direcciones no se pueden enrutar a Internet y se reservan solo al uso en las LAN (*Local Area Network* = red local).

Franja de direcciones	Número de dirección	Red y máscara
10.0.0.0 a 10.255.255.255	16.777.216	10.0.0.0/8
172.16.0.0 a 172.31.255.255	1.048.576	172.16.0.0/12
192.168.0.0 a 192.168.255.255	65.536	192.168.0.0/16

Si tiene que asignar direcciones en una red, es necesario obligatoriamente respetar estas franjas de direcciones. El resto de direcciones son susceptibles de ser utilizadas en Internet y es necesario evitar asignarlas a máquinas en una red LAN.

Hay algunas excepciones que se detallan a continuación.

g. Las excepciones

Las redes reservadas

Hay dos redes reservadas para usos particulares. Por tanto, no hace falta usarlas en su plan de direccionamiento:

- La red 127.0.0.0 está reservada para las pruebas de bucle local. Cuando todas las capas TCP/IP funcionan correctamente en una máquina, la dirección IP 127.0.0.1 es la dirección que permite probar el correcto funcionamiento del conjunto de programas puesto en marcha para TCP/IP (con un ping, por ejemplo). Es el bucle local de su PC, también llamado localhost.

```
pi@raspberrypi:~ $ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data:
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.095 ms
```

Esta prueba muestra que la parte de software de la red, se ha instalado correctamente y funciona en la Raspberry Pi.

- La red 0.0.0.0 también está reservada: se utiliza para definir una ruta por defecto en el router.
- La red 100.64.0.0/10 está reservada desde 2012 para su uso en las NAT (*Network Address Translation* = traducción de dirección red) de grandes dimensiones, a escala de una metrópoli, por ejemplo. Este bloque de dirección no se debe utilizar ni en redes privadas, ni en Internet. Esta red tiene más de cuatro millones de direcciones.

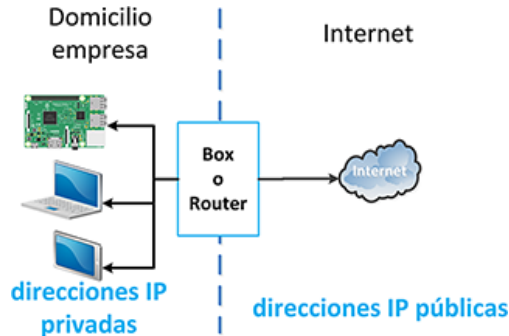
Dirección APIPA

El direccionamiento APIPA (*Automatic Private Internet Protocol Addressing* = direccionamiento IP automática) permite a un sistema operativo asignar automáticamente una dirección IP a un ordenador cuando esta no se ha asignado manualmente y no se puede unir a un DHCP (véase la siguiente imagen). Las direcciones APIPA se asignan en la franja 169.254.0.0/16. Las direcciones van de 169.254.0.0 hasta 169.254.255.255. Esta franja se reserva exclusivamente a este uso.

- Si el comando `ifconfig` indica que su Raspberry Pi tiene una dirección que empieza por 169.254... es que ha habido un problema de conexión en su unidad de conexión o no le ha asignado dirección manualmente, si trabaja con direccionamiento estático.

h. Dirección pública

Las direcciones públicas son aquellas que se utilizan en Internet. Admiten todas las direcciones no mencionadas en las excepciones anteriores.

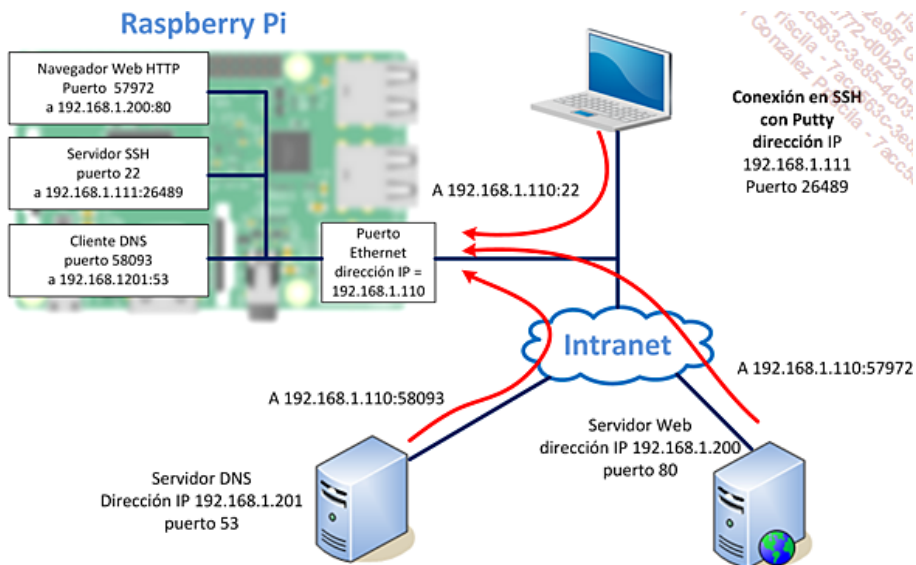


- Internet es el sistema que une las redes locales entre ellas = Inter Network.

3. Llegada a buen puerto

a. Puertos TCP/IP

En una Raspberry Pi o en cualquier otro ordenador hay un gran número de servicios que funcionan al mismo tiempo y deben poder acceder al exterior. Solo hay una única dirección IP en la tarjeta de red o la interfaz Wi-Fi. A cada cliente presente se le asigna un número de puerto, codificado con 2 bytes, es decir 16 bits. Por tanto, el número de puerto puede estar comprendido entre 0 y 65.535.



- El término Intranet utilizado en el esquema anterior, indica una red interna de una empresa. Un servidor web o un DNS que operan en Internet no podrían enviar mensajes a la dirección privada de la Raspberry Pi.

Por defecto, se asignan algunos números de puerto a los servicios:

- 20 y 21: FTP
- 22: SSH
- 23: TELNET
- 53: DNS
- 80: HTTP
- 443: HTTPS

b. Utilización de los puertos

En el esquema anterior, que representa una red local de empresa, funcionan un cliente DNS y un navegador web. Estos dos clientes han recibido números de puerto: 57.972 para el navegador Web y 58.093 para el cliente DNS.

El servidor SSH ha recibido el número de puerto por defecto correspondiente a su función, es decir, el puerto 22. Cuando el ordenador portátil se conecta en SSH con PuTTY, abre una conexión en el socket (véase el siguiente párrafo) 192.168.1.110:22. El servidor SSH de la Raspberry Pi le responderá en el socket 192.168.1.111:26489.

Cuando el navegador Web hace una petición al servidor web, abre una conexión TCP en el socket 192.168.1.200:80. El servidor web responde al socket 192.168.1.110:57972.

El cliente DNS abre una conexión en el socket 192.168.1.201:53. El servidor DNS responde al socket 192.168.1.110:58093.

Este mecanismo permite, con una sola dirección, gestionar múltiples servicios asignado a cada uno un número diferente.

c. Socket TCP/IP

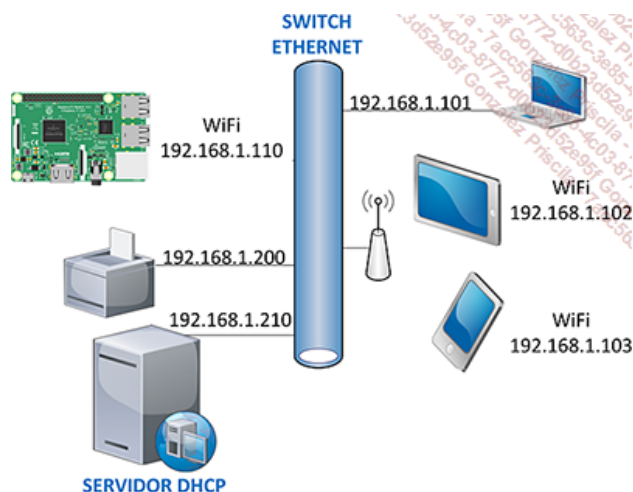
Cuando la Raspberry Pi intercambia datos con el mundo exterior, lo hace a través de su toma Ethernet (o de su interfaz Wi-Fi). Los datos que vuelven hacia él tienen todos como destino la dirección única de la tarjeta red, aquí 192.168.1.110. Pero cada mensaje tiene además de la dirección del destinatario un número de puerto. Este número se indica después de la dirección IP, separada por dos puntos 192.168.1.110:60351. Esta dirección completa tiene la dirección IP de la máquina más el número de puerto del servicio implicado, llamado *socket*. Es el punto de conexión que va a permitir a un cliente conectarse a un servidor, o a la inversa.

4. Roles de la unidad de conexión

Para los particulares, la solución más extendida para conectarse a Internet es usar una unidad de conexión o box. Hay muchos proveedores de acceso a Internet que ofrecen su solución, cada uno rivalizando en términos de funcionalidades, para conseguir el máximo número de clientes. Las funciones principales de una box utilizada en red con la Raspberry Pi se describen a continuación.

a. Servidor DHCP

En la actualidad, el número de dispositivos conectados está en constante aumento. Normalmente tenemos un PC de escritorio, un ordenador portátil, algunas tabletas, varios smartphones e incluso una o varias Raspberry Pi. Cuando no queremos gestionar manualmente las direcciones IP de cada dispositivo conectado en nuestra casa, delegamos esta función a un servidor DHCP (*Dynamic Host Configuration Protocol*= Protocolo de Configuración Dinámica de Host).



El rol de servidor DHCP generalmente queda garantizado por la box instalada en casa. Sin embargo, cualquier otro servidor DHCP podría garantizar la distribución de las direcciones... Una Raspberry Pi, por ejemplo.

Petición DHCP

Cuando enchufa una Raspberry Pi (o cualquier otro ordenador), las interfaces no se inicializan y, por tanto, no tienen dirección IP.

- La Raspberry Pi envía una petición *DHCP DISCOVER* en broadcast, cuyo destino son todas las máquinas de la red. Esta petición tiene su dirección MAC para que pueda recibir una respuesta.
- Un servidor DHCP existente en la red, cuando recibe este tipo de petición, envía una respuesta *DHCP OFFER* cuyo destino es la dirección física del solicitante. Esta respuesta tiene la dirección IP del servidor, la dirección IP y la máscara de sub-red que le ofrece al cliente.
- El cliente recibe la respuesta (o las respuestas si hay varios servidores DHCP en la red). En primer lugar, retienen la respuesta que reciben y después la envían en broadcast *DHCP REQUEST*. Este mensaje contiene la dirección IP del servidor DHCP del destinatario y la dirección IP que este ha propuesto al cliente. Solicita al servidor correspondiente que le sea asignada la dirección IP, así como el envío de determinados parámetros llamados opciones. Este mismo mensaje informa a los servidores DHCP cuya respuesta no se ha retenido de que pueden disponer de nuevo de la dirección IP que habían propuesto.
- El servidor DHCP correspondiente prepara un mensaje de acuse de recibo (*DHCP ACK* = validación de la petición) para transmitir al cliente su dirección IP, su máscara de sub-red, el tiempo de validez para esta dirección y eventualmente, opciones como la dirección IP de la pasarela por defecto, el servidor DNS... (ver más adelante).
- El servidor DHCP registra la dirección IP que ha distribuido con la dirección MAC del cliente implicado.
- El cliente conserva la misma dirección para una duración igual a la mitad del periodo de validez.
A partir de este momento, solicita periódicamente al servidor que le asigne, la renovación de su dirección, que conservará. Como el cliente tiene una dirección IP válida, estos intercambios se realizan a través de las direcciones IP del cliente y del servidor. Si la renovación de la dirección es imposible por una razón u otra, cuando se llega al 87,5 % del tiempo de la validez temporal, el cliente vuelve a hacer una petición de dirección en broadcast.
- Si el cliente está inactivo más tiempo que el tiempo de validez, vuelve a solicitar una nueva dirección durante el siguiente arranque y eventualmente puede obtener una dirección diferente.

Con el uso de un servidor DHCP, la gestión de la asignación de las direcciones en una red local está totalmente automatizada. Es posible definir de manera más fina el funcionamiento del DHCP, como la franja de direcciones distribuidas, las reservas o exclusiones de direcciones, etc... Si tiene que configurar un servidor DHCP, hay una documentación importante en línea disponible que le ayudará.

- El DHCP resuelve el problema de los conflictos de direcciones, porque gestiona la distribución de direcciones IP únicas. Sin embargo, si en la misma red hay una máquina configurada manualmente que tiene la misma dirección IP que alguna de las máquinas gestionadas por el DHCP, esto provocará un conflicto de direcciones.

b. Nombre de dominio

Normalmente, debe consultar muchos sitios web para su trabajo o a título personal. Cada servidor alberga un sitio web que tiene su propia dirección IP. Imagine que tiene que recordar de memoria una cantidad de direcciones:

- 172.217.19.131 para acceder a Google,
- 74.6.50.24 para acceder a Yahoo,
- 176.34.135.167 para acceder a Duckduckgo,
- 145.242.11.48 para acceder al sitio web de la agencia tributaria,
- etc.

Si las máquinas se ajustan perfectamente a esto, la memorización de esta sucesión de cifras rápidamente se hace inmanejable para un usuario

humano. Para evitar este uso de las direcciones IP se ha creado el nombre de dominio.

Es mucho más fácil memorizar un nombre que una cifra. Por tanto, se ha decidido crear dominios para facilitar el uso de Internet. A continuación veremos algunos ejemplos:

.es es el dominio que contienen los ordenadores registrados en dominios .es, que dependen del Ministerio de energía, turismo y agenda digital.

.com es el dominio que contienen los ordenadores que alojan actividades comerciales

.org es el dominio que contienen los ordenadores que alojan a organizaciones sin ánimo de lucro.

Dentro de estos dominios, cada sitio web recibe un nombre único, seleccionado por el creador del sitio web. En función del dominio correspondiente, las condiciones de asignación de nombres pueden ir desde la liberad total (el primero que llega es quien que se lleva el nombre) hasta reglas muy estrictas (identificación, documentación, etc...). A continuación se muestran algunos ejemplos: raspberrypi.org, elpais.com, debian.org. Cada una de las máquinas que alojan estos sitios web tiene una dirección IP, por lo que es necesario un sistema que permita traducir automáticamente la dirección literal informada por el usuario en una dirección IP, utilizable por los sistemas de gestión de Internet. Este es el rol del DNS.

c. DNS

Cuando introduce una dirección en su navegador web, este no sabe a qué dirección debe ir a buscar la página que desea mostrar. Entonces se dirige a DNS (*Domain Name Server* = servidor de nombre de dominio) la dirección que tiene asignada.

En una Raspberry Pi en Raspbian Jessie, el DNS utilizado se registra en el archivo `/etc/resolv.conf`.

```
pi@raspberrypi:~ $ cat /etc/resolv.conf
# Generated by resolvconf
nameserver 192.168.1.254
```

Generalmente la dirección DNS proporcionada por un box o un router es la propia dirección del box o del router que juega el rol de transmisor. Un transmisor DNS envía las peticiones que no puede responder al DNS del ISP (proveedor de acceso a Internet).

El cliente (su navegador web) envía su petición al DNS en forma "¿Me puedes decir cuál es la dirección de www.google.es?" (por ejemplo).

- Si el transmisor DNS local ya ha tratado esta petición, guarda la respuesta en caché (en memoria) y responde instantáneamente enviando la dirección IP correspondiente a www.google.es.
- Si el transmisor DNS local no conoce la respuesta, pregunta al DNS del ISP.
 - Si este conoce la dirección IP de www.google.es, la envía al transmisor DNS de su box.
 - Si el DNS del ISP no conoce la dirección IP solicitada, pregunta al servidor DNS "especialista" del .es, llamado servidor raíz, que le ofrece las direcciones de servidores DNS que le puedan indicar la dirección IP de Google.
 - El DNS del ISP interroga al primer servidor DNS de la lista para obtener la dirección IP de www.google.es. Si la obtiene, la guarda en caché (para evitar solicitarla de nuevo posteriormente) y, por último, transmite la respuesta al transmisor DNS de su box, que la envía a su navegador.
 - Si no obtiene la respuesta, pregunta al segundo servidor DNS de la lista, etc.

Vemos que la obtención de la dirección IP correspondiente a la URL (*Uniform Resource Locator* = localizador uniforme de recursos) www.google.es no es instantánea y puede necesitar varias peticiones antes de que el nombre sea "resuelto" en su dirección IP.

La dirección de DNS que figura en un ordenador algunas veces es directamente la dirección del DNS del ISP. Cuando se registran dos DNS en la configuración, el segundo DNS se llama secundario. Se consulta si el primer DNS no ha podido ofrecer la dirección IP solicitada (congestión, error, etc...).

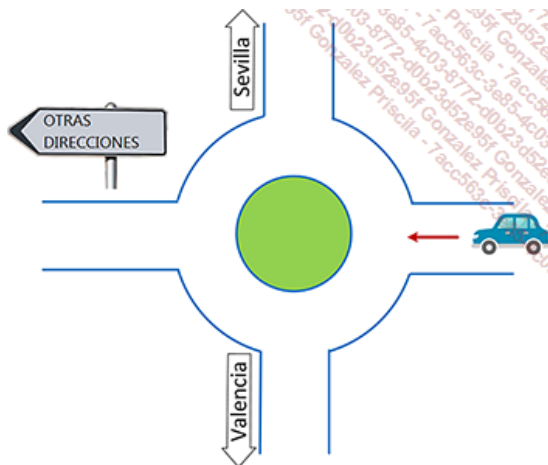
El DNS se puede comparar con un directorio que contiene las direcciones IP de todos los servidores del mundo asociados a su nombre. Esta gigantesca base de datos proporciona a cada solicitud que recibe la dirección IP correspondiente a la URL introducida por un usuario.

Cuando se compra un nombre de dominio, se paga al que vende este servicio para registrar la inscripción del nombre de dominio en la base de datos mundial de nombres de dominio. Al registrarse, se establece una entrada DNS que une al nombre de dominio que ha comprado con la dirección IP del servidor que alberga su sitio web.

d. Pasarela

Otro argumento importante que hay que conservar durante la configuración de una Raspberry Pi en la red es la dirección de la pasarela (Gateway).

Para entender qué representa la pasarela en la red local, imagine que circula en coche. Va dirección a Madrid. Llega a una rotonda:



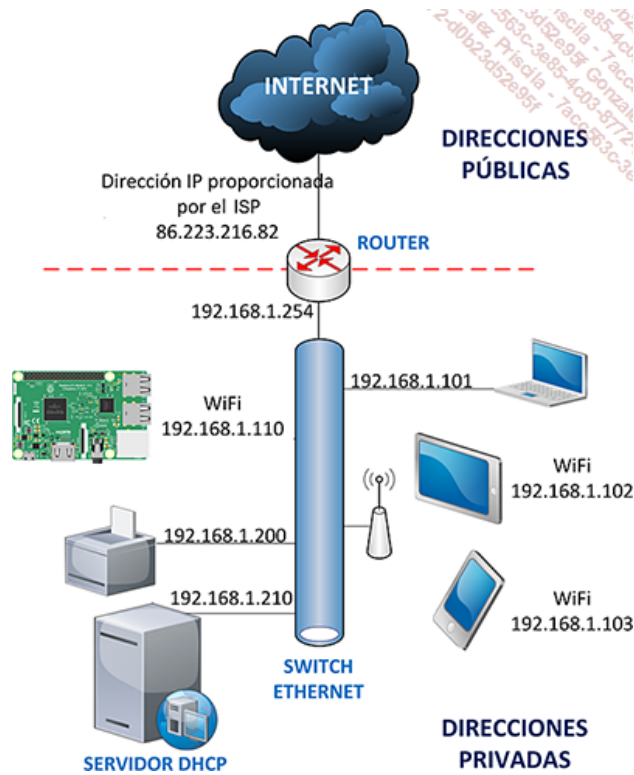
Los paneles indicadores muestran la ruta hacia Sevilla y Valencia. Ninguna de estas direcciones le conviene. Por tanto, toma la dirección que indica "OTRAS DIRECCIONES".

En una red, una máquina que envía datos a una máquina situada en la misma red local sabe que lo puede hacer sin problemas (ver más atrás cómo se calcula la dirección de la red).

Si el cálculo de la dirección de red muestra que la máquina de destino no está en la misma red, la única solución es seguir por la ruta "OTRAS DIRECCIONES", es decir la pasarela.

La pasarela es la dirección a la que se envían los datos que no están destinados a máquinas que pertenezcan a la misma red local que el emisor. Esta dirección corresponde a una interfaz de router situada en la red local. Por tanto, la dirección de esta interfaz está obligatoriamente en la misma red local que las máquinas que la usan. Sin que esto sea una obligación, la dirección IP de la pasarela normalmente es la última dirección disponible en la red (justo antes de la dirección de broadcast). Aquí será 192.168.1.254.

- Un router es un hardware de red encargado de garantizar el enrutamiento entre dos interfaces situadas cada una en una red diferente. En un box, está integrado en la caja.



El otro lado del router está conectado a Internet. La dirección pública del lado de Internet (82.223.216.82) se asigna por el ISP. Para algunos ISP esta dirección pública puede estar fija, para otros ISP cambia regularmente.

Ejemplo de consulta del sitio web

Los elementos de la red empiezan a desplegarse. Revisemos el funcionamiento de este conjunto.

- Enchufe la Raspberry Pi. Esta se pone en contacto con el servidor DHCP para obtener una dirección. El DHCP le asigna la dirección IP 192.168.1.110.
- La Raspberry Pi arranca, y usted inicia el navegador web. En la barra de dirección, escribe google.es.
- El navegador pregunta al relé DNS para obtener la dirección IP de google.es.
- El relé DNS envía la dirección IP 216.58.211.99, que es la dirección de uno de los servidores de google.es.
- El ordenador mira si la dirección 216.58.211.99 está en su red. Para ello usa su propia dirección IP 192.168.1.110 y su máscara de sub-red 255.255.255.0.
- La red del servidor de Google 216.58.211.0 es diferente de la red de la Raspberry Pi 192.168.1.0. La Raspberry Pi no puede enviar la petición a un ordenador situado fuera de su red local. La única solución que tiene es transmitir su petición a la dirección de la pasarela que le ha sido indicada.
- La Raspberry Pi envía su petición a la pasarela (por tanto, al router). El rol del router va a ser gestionar el envío de la petición a Internet hasta el servidor Google a través de muchos otros routers.
- El servidor Google recibe la petición generada por la Raspberry Pi y responde enviando la página de bienvenida del servicio. Esta página vuelve hasta el router del box.

Durante este tiempo...

- Otro usuario ha hecho una petición desde la tableta conectada a la red local, para acceder al motor de búsqueda DuckDuckGo.
- Se produce la misma secuencia anterior hasta devolver la página de bienvenida de DuckDuckGo hasta el router del box.

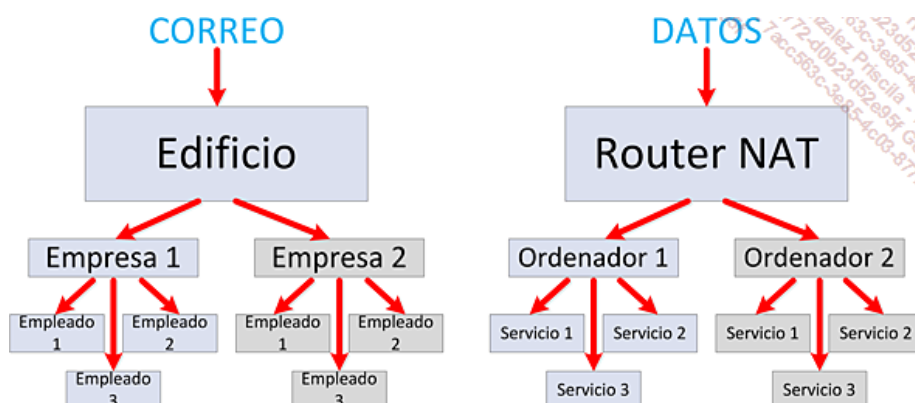
El router acaba de recibir dos páginas web enviadas como destino su dirección pública, porque es él quién ha enviado la petición a través de Internet. Para poder gestionar estas peticiones múltiples, las cuales reciben respuestas, el router pone en marcha la técnica NAT.

e. Router NAT

La implementación de la función NAT en un box o router permite a las múltiples máquinas de una red local disponer cada una de una dirección IP privada diferente para comunicarse con los servidores situados en Internet a partir de una dirección única pública.

Principio de router NAT

La situación del router se parece a la de un conserje de un edificio al que el cartero ha entregado todo el correo destinado a las empresas del edificio.



El conserje deja en el buzón de cada empresa su correo. La persona encargada de la distribución del correo de cada empresa recoge este correo del buzón de mensajes de la empresa. Cuando llega a la empresa, lo distribuye a las personas correspondientes.

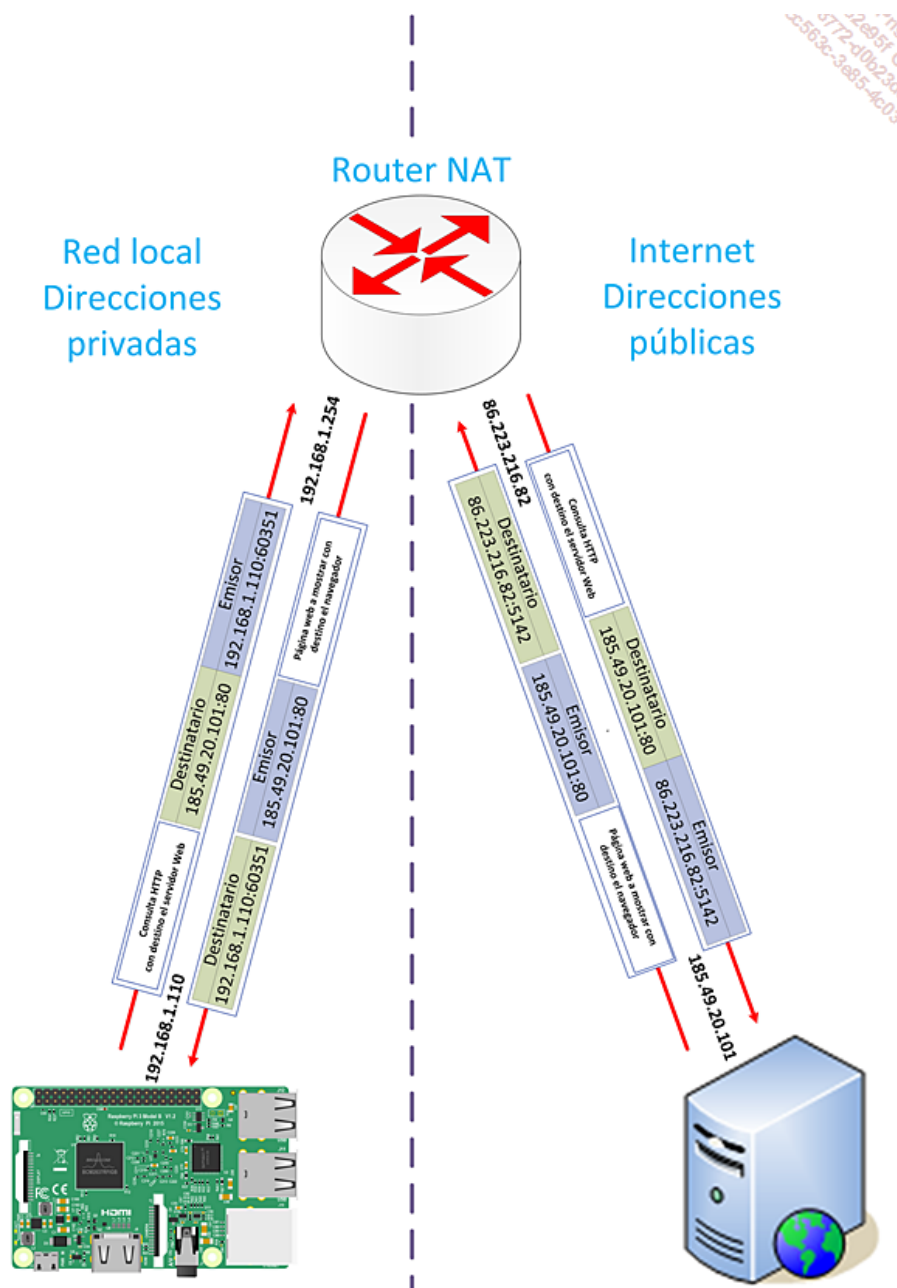
Un router NAT funciona de la misma manera: cuando recibe los datos que llegan de Internet cuyo destino es los ordenadores de la red local, envía los datos a cada ordenador destinatario en función de la dirección IP del destinatario, después estos datos se dirigen a los servicios en función de los numerosos puertos.

Es el principio (simplificado) de la función NAT del router.

Funcionamiento del router NAT

Cuando un ordenador de la red local desea contactar con un servidor en Internet, envía su petición a la pasarela de la red local con su dirección IP y el número de puerto que usa. El router envía la petición a Internet. Sustituye previamente la dirección del ordenador por su propia dirección y el número de puerto por un número aleatorio que memoriza en una tabla.

El servidor que ha recibido la petición responde enviando los datos en dirección al router en el número de puerto que le ha indicado:



En el esquema anterior (disponible para su descarga en formato PDF para una mejor legibilidad) la Raspberry Pi envía una petición HTTP a un servidor web situado en Internet a través de un box que garantiza la función de router NAT. Para gestionar los datos que se envían del servidor, el router NAT mantiene actualizada una tabla que le permite saber a dónde enviarlo en la red local:

	Puerto externo	Hacia	Puerto interno	Protocolo	Dirección IP interna
HTTP	5142		60351	TCP	192.168.1.110

En el ejemplo anterior, la tabla indica que todo el tráfico IP recibido por el router en el puerto externo 5142 se debe enviar al puerto 60.351 de la Raspberry Pi, en la dirección IP 192.168.1.110.

Conclusión

El router NAT integrado en los box ofrece la posibilidad de compartir una única dirección pública entre muchos ordenadores de una red local en la que se usa la traducción de dirección. Por otro lado, diferentes redes locales pueden utilizar las mismas franjas de direcciones sin interferir las unas con las otras, porque estas direcciones se ocultan por el router NAT y nunca aparecen en Internet.

El NAT ha sido una de las formas utilizadas para resolver momentáneamente la escasez de direcciones IPv4. El arranque difícil de la nueva norma IPv6 sin duda va a perpetuar esta situación.

5. Acceder a una Raspberry Pi desde Internet

Cuando se usa una Raspberry Pi, en particular para las aplicaciones domóticas, es interesante poder conectarse desde cualquier sitio, para comprobar una temperatura o para controlar el funcionamiento de un dispositivo en modo remoto.

Dos dificultades frenan este uso:

- Es difícil recordar la dirección pública del box y, con algunos proveedores, esta dirección cambia regularmente.
- El box no deja pasar las peticiones que vienen del exterior con destino a las máquinas de la red local que gestiona.

Cada uno de estos obstáculos se puede salvar usando los medios que se presentan en los siguientes párrafos.

a. DNS dinámico

El DNS dinámico permite asociar un nombre de dominio a la dirección IP pública de su box, incluso si esta dirección cambia regularmente.

Las soluciones disponibles

Hay varias soluciones en línea para cumplir esta función. Históricamente es DynDNS (www.dyn.com), que ha creado un servicio gratuito. El servicio ahora es de pago (33 € sin IVA/año para un particular).

Actualmente (julio 2016) NoIP ofrece una solución de DNS dinámica gratuita (www.noip.com). La opción del nombre de dominio está limitada y es necesario confirmar la reserva cada 30 días respondiendo a un correo electrónico.

Funcionamiento del DNS dinámico

Algunas box y routers incorporan un software cliente para DynDNS y/o NoIP. En un router TP-Link TD-W8980 existe una sección Dyn DNS que permite configurar una cuenta NoIP para acceder a la red local desde Internet.



Vemos que el nombre de dominio raspxxxxx.no-ip.biz es un sub-dominio de no-ip. Esto significa que es el DNS de NoIP el que actualizará su dirección IP, en relación con el nombre de dominio que haya seleccionado. Cuando usted (u otro usuario) intenta conectarse usando el nombre de dominio registrado por NoIP, la petición DNS se transmitirá al servidor DNS de NoIP, que devolverá la dirección IP de su box al solicitante.

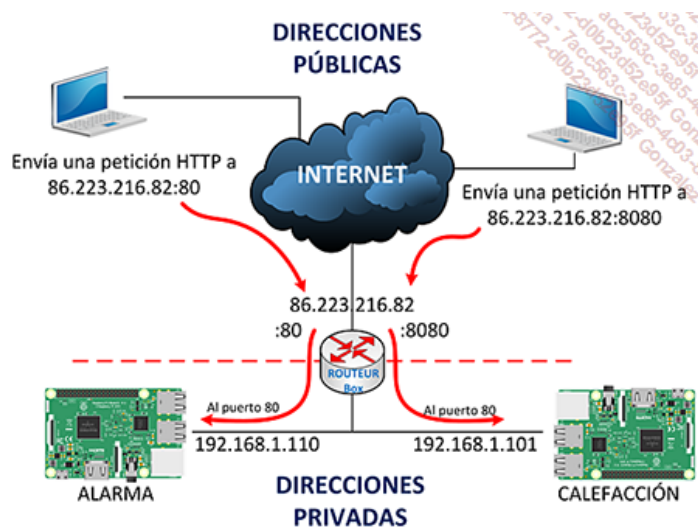
No todo es tan sencillo

Si es un navegador web el que hace la petición, la envía al puerto 80 de su box, o bien el box no tiene un servidor web... Igualmente, si intenta conectarse con PuTTY, su petición se enviará a la dirección pública del box, esta vez en el puerto 22. En ambos casos, el box no podrá responder, porque no dispone de los servidores solicitados. No transmitirá más las peticiones a los servidores que existen en la red local. No hay ningún medio de saber a qué dirección de red debe reenviar estas peticiones.

b. Redirección de puerto

También llamado *Port Forwarding* o redirección de puerto, el mapeo del puerto permite responder a esta problemática. Esta técnica va a leer un puerto de la dirección pública (Internet) con la dirección IP y el puerto de un ordenador conectado a la red local (dirección privada).

Tomemos como ejemplo una casa equipada de dos Raspberry Pi encargadas una de la vigilancia (alarma) y otra de la gestión de la calefacción.



La Raspberry Pi encargada de la alarma (a la izquierda en el esquema anterior) está accesible desde la página Web con la dirección 192.168.1.110:80. También dispone de un acceso SSH que permite conectarse en modo texto de manera remota.

La Raspberry Pi encargada de la gestión de la calefacción está accesible en la dirección 192.168.1.101:80.

Uno de los usuarios de los dos ordenadores portátiles conectados a Internet desea consultar el estado de la alarma. Se conecta con un navegador Web a la dirección pública del box, bien directamente o pasando por NoIP. Su petición llega al puerto externo 80. La box envía esta petición al puerto 80 de la Raspberry Pi 192.168.1.110. El servidor web de la Raspberry envía de vuelta al navegador la página web que contiene los elementos solicitados.

Otro usuario quiere ajustar la calefacción. Se conecta también a la dirección del box. Pero esta vez la dirección utilizada está seguida de :8080. La petición que llega a este puerto se envía por el box al puerto 80 de la Raspberry Pi 192.168.1.101. El servidor web de esta Raspberry Pi envía al navegador la página de gestión de la calefacción.

Puede suceder que sea necesaria una conexión SSH. Es la Raspberry Pi 192.168.1.110 la que ofrece este servicio en el puerto 22. Para no facilitar la tarea a eventuales intrusos, el puerto externo se obtiene al azar: 6954. Se mapeará en el puerto 22 de la Raspberry Pi.

- Algunas box o routers, usan el puerto externo 80 para la gestión remota. En este caso habría que utilizar otro puerto y agregar el puerto en la dirección en el navegador.

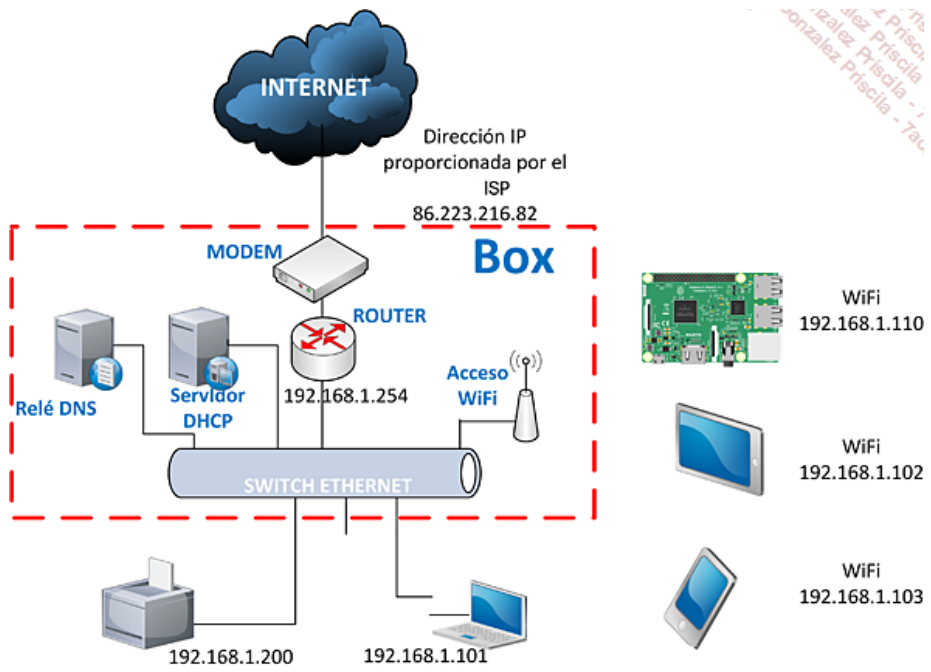
	Puerto externo	Hacia	Puerto interno	Protocolo	Dirección IP interna
HTTP	80		80	TCP	192.168.1.110
SSH	6954		22	TCP	192.168.1.110
HTTP	8080		80	TCP	192.168.1.101

La tabla anterior indica la configuración que hay que realizar en el box o router para implementar la redirección del puerto.

En la práctica, el router gestiona una única tabla que agrupa la información estática de redirección de puertos y la información dinámica generada por el enrutamiento NAT (vista más atrás). Esta tabla le sirve para determinar los respectivos destinatarios de los paquetes recibidos.

6. Box Internet

Llega al final de esta inicialización de red y ahora es posible mostrar el esquema de un box de Internet. Todavía no se han presentado todas las funciones, pero las principales relacionadas con la red figuran en este resumen.



En las empresas, las funciones que se presentan aquí quedan garantizadas por servidores, switches, enrutadores y accesos Wi-Fi separados.

Gestionar la conexión de red alámbrica de la Raspberry Pi

1. Mostrar la información de la conexión

El comando `ifconfig` (*interfaz configuración*) muestra la información de la conexión de red.

```
pi@raspberrypi:~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:37:bc:ee
          inet addr:192.168.1.230  Bcast:192.168.1.255
            Mask:255.255.255.0
          inet6 addr: fe80::76c2:5219:50c4:ac63/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:296694 errors:0 dropped:163 overruns:0
            frame:0
          TX packets:187970 errors:0 dropped:0 overruns:0
            carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:416225559 (396.9 MiB)  TX bytes:18700015
            (17.8 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:244 errors:0 dropped:0 overruns:0 frame:0
          TX packets:244 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:19296 (18.8 KiB)  TX bytes:19296 (18.8 KiB)
```

`eth0` corresponde a la tarjeta Ethernet de la Raspberry Pi. La dirección IP es 192.168.1.230 y la máscara de sub-red 255.255.255.0.

`lo` es el bucle local, que permite probar la pila TCP/IP (el correcto funcionamiento de la capa TCP/IP). Su dirección es 127.0.0.1.

2. Modificar los argumentos de la red

➤ Los argumentos que siguen se aplican a Raspbian Jessie.

Una de las novedades que han aparecido con Raspbian Jessie es la gestión del Wi-Fi. Esta gestión ahora está en manos de `dhcpcd`. De hecho, el archivo `/etc/network/interfaces` ya no juega el rol que jugaba en las anteriores versiones de Raspbian. La Fundación recomienda no añadir modificaciones al archivo `interfaces` en Jessie.

```
auto lo
iface lo inet loopback

auto eth0
allow-hotplug eth0
iface eth0 inet manual

auto wlan0
allow-hotplug wlan0
iface wlan0 inet manual
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf

auto wlan1
allow-hotplug wlan1
iface wlan1 inet manual
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

Si ha modificado el archivo `interfaces`, es absolutamente necesario restaurarlo al estado inicial, como antes.

a. Configuración automática por DHCP

La ventaja de `dhcpcd` cuando se usa con una box o un router configurado en DHCP (caso mayoritario en una instalación doméstica o escolar) es que funciona en modo automático sin tener que intervenir en la configuración.

b. Configurar una dirección estática

El uso de una dirección estática normalmente no es la mejor solución. Si quiere/necesita una dirección IP fija, defina una reserva estática en su router (si es posible) o use una implementación libre de Zeroconf como `avahi`.

Si define `ip_address`, `dhcpcd` no intentará obtener un contrato DHCP y utilizará el valor que le haya indicado para la dirección con una duración de contrato indefinida.

A continuación se muestra un ejemplo en el se configura una dirección estática, la pasarela y un DNS.

Modifique el archivo `/etc/dhcpcd.conf` como sigue (extracto de la página `man` de `dhcpcd.conf`):

```
sudo nano /etc/dhcpcd.conf
```

Añada las siguientes líneas al final del archivo:

```
interface eth0

static ip_address=192.168.0.10/24
static routers=192.168.0.1
static domain_name_servers=192.168.0.1
```

Asegúrese de que la dirección que asigna es única en la red, que la máscara de red (`/24 = 255.255.255.0`) y la dirección de la pasarela (`static routers`) son correctas, así como la del DNS (`static domain_name_servers`). Por defecto, su Raspberry Pi no podrá conectarse a las máquinas de la red, acceder a Internet o resolver los nombres de dominio.

Después de intervenir en el archivo `/etc/dhcpcd.conf`, solo se ha modificado este archivo de texto. Hay que forzar que se tengan en cuenta las modificaciones por parte del servicio que gestiona la red en Raspbian.

c. Reinicio de la red para tener en cuenta las modificaciones

Para tener en cuenta las modificaciones realizadas en el archivo `dhcpd.conf`, reinicie el demonio `systemctl` para que recargue la configuración de los servicios. La modificación del archivo `/etc/dhcpd.conf` se tendrá en cuenta después de este reinicio.

```
sudo systemctl daemon-reload
```

Configurar una conexión Wi-Fi

La Raspberry Pi 3 integra una interfaz Wi-Fi. En la Raspberry Pi Zero habría que conectar una llave Wi-Fi al puerto USB con un adaptador.

1. Mostrar la información de la conexión

➤ Raspbian reconoce la mayor parte de las llaves Wi-Fi USB automáticamente. En caso de duda, compruebe la compatibilidad de su llave en http://elinux.org/RPi_VerifiedPeripherals#USB_Wi-Fi_Adapters.

El comando `ifconfig` muestra la información relativa a la conexión de red. En la Raspberry Pi Zero no hay puerto Ethernet alámbrico (`eth0`). Solo aparece el Wi-Fi.

```
pi@raspberrypi:~ $ ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr:::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wlan0     Link encap:Ethernet  HWaddr 00:90:4c:0e:81:23
          inet addr:192.168.1.110  Bcast:192.168.1.255
            Mask:255.255.255.0
          inet6 addr: fe80::8290:be4d:43ff:3026/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45177 errors:0 dropped:29433 overruns:0
            frame:0
          TX packets:10548 errors:0 dropped:0 overruns:0
            carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6288763 (5.9 MiB)  TX bytes:1148318 (1.0 MiB)
```

En el ejemplo anterior, la llave Wi-Fi que equipa la Raspberry Pi Zero está conectada a la red y ha recuperado la dirección **192.168.1.110** proporcionada por el DHCP. A continuación se muestran las etapas que permiten tener en cuenta la llave Wi-Fi y su uso en la Raspberry Pi Zero.

a. Comprobación de la detección de la llave USB

El comando `lsusb` muestra los periféricos reconocidos en el bus USB:

```
pi@raspberrypi:~ $ lsusb
Bus 001 Device 004: ID 046d:c52b Logitech, Inc. Unifying Receiver
Bus 001 Device 003: ID 0a5c:bd1e Broadcom Corp.
Bus 001 Device 002: ID 05e3:0610 Genesys Logic, Inc. 4-port hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

La segunda línea confirma que la llave Broadcom se ha identificado correctamente en el bus USB.

b. Modificación del archivo `/etc/wpa_supplicant/wpa_supplicant.conf`

Abra el archivo `/etc/wpa_supplicant/wpa_supplicant.conf`

```
sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

Contiene:

```
country=GB
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
```

Modifique la primera línea para indicar que se utiliza el Wi-Fi en España. Los canales autorizados, así como la potencia, se ajustarán automáticamente.

Al final del archivo, añada la definición de su red.

```
country=ES
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
    ssid="el_SSID_de_su_red"
    psk="su clave "
    key_mgmt=WPA-PSK
}
```

La definición de su red se sitúa entre llaves. `ssid` designa el identificador de su red Wi-Fi (`ssid = Service Set Identifier`) y `psk` (*pre-shared key* = clave compartida) la clave que permite acceder a la red Wi-Fi.

Esta configuración supone que su punto de acceso Wi-Fi usa WPA (*Wi-Fi Protected Access* = acceso Wi-Fi protegido). El WEP (*Wired Equipment Privacy* = Red alámbrica privada) es demasiado vulnerable, de modo que no se debe utilizar.

c. Asignar una dirección fija a una llave USB Wi-Fi

La forma de proceder es la misma que para la asignación de una dirección IP fija a un puerto alámbrico Ethernet. Será necesario tener las mismas precauciones para la asignación de direcciones.

Después de abrir el archivo `/etc/dhcpd.conf`

```
sudo nano /etc/dhcpd.conf
```

añada estas líneas al final del archivo

```
interface wlan0
static ip_address=192.168.1.43/24
static routers=192.168.1.254
static domain_name_servers=192.168.1.254
```



Esta configuración se ha probado con varias llaves Wi-Fi de marcas y orígenes diferentes, así como con varios routers y box de Internet.

Introducción

La tarjeta micro SD (*Secure Digital*) de la Raspberry Pi ofrece a los usuarios una gran flexibilidad de uso. Basta con cambiar la tarjeta micro SD para iniciar rápidamente en otro sistema.

El arranque de la Raspberry Pi requiere, necesariamente, una tarjeta micro SD. Sin embargo, nada le impide, una vez arrancado el sistema, hacer que el sistema funcione en una memoria de almacenamiento externa, llave USB o disco duro.

Este capítulo presenta los elementos que debe tener en cuenta durante la elección de una solución, así como un ejemplo de puesta en marcha con una llave USB de 4 GB dividida en dos particiones de 2 GB. Esta descripción se aplica a los soportes externos de cualquier tamaño.

Después de la preparación de la llave USB para su uso con la Raspberry Pi, el montaje manual o automático de las particiones presentes en el soporte externo permitirá ampliar las posibilidades de la Raspberry Pi.

El uso de la línea de comandos para realizar todas las operaciones facilitará adquirir el control de este entorno, que desde un punto de vista de su uso es sencillo y habitualmente mucho más potente que las herramientas en modo gráfico. Normalmente esto no es más que una forma de ocultar la flexibilidad y riqueza del shell.

El uso de una memoria de almacenamiento masivo externa en modo gráfico, se trata en el capítulo Utilizar el modo gráfico.

¿Por qué una memoria externa?

Cuando el uso de la Raspberry Pi implica frecuentes escrituras en la tarjeta micro SD o necesita una capacidad de almacenamiento elevada, es porque la tarjeta micro SD ha alcanzado el límite de su capacidad. En ese caso, el usuario opta por otros medios de almacenamiento.

1. Características de la tarjeta microSD



La tarjeta micro SD, creada en el 2000 por Panasonic, SanDisk y Toshiba, se ha convertido en un soporte necesario para el almacenamiento de los datos en muchos dispositivos con un sistema embebido (aparatos de toma de imágenes, reproductores de música, smartphones, tabletas, etc.).

Se divide en miniSD y microSD, que mantienen las características de la tarjeta micro SD original, pero con un tamaño adaptado a la miniaturización de los dispositivos actuales.

A mediados del año 2016, la capacidad más elevada de las tarjetas micro SD disponibles habitualmente en el mercado era de 128 GB, aunque las tarjetas de 256 GB todavía están disponibles, así como las tarjetas de 512 GB.

Tecnología de las tarjetas micro SD

Las tarjetas SD están construidas a partir de EEPROM (*Electrically Erasable Programmable Read-Only Memory* = memoria de solo lectura programable y borrable). La escritura de los datos en estas memorias de tipo NAND (*Not-AND* = memoria construida a base de puertas lógicas NO-SÍ) necesita el uso de una tensión más elevada que la utilizada en lectura. Si bien la lectura de las tarjetas SD puede hacerse indefinidamente sin deterioro, las escrituras repetidas provocan a la larga una degradación de las células de memoria.

Los fabricantes implementaron procesos que administran el lugar en que se producen las escrituras en las memorias. Un componente de la tarjeta guarda el número de escrituras por zona de memoria y optimiza el uso de estas zonas.

Tiempo de vida de las tarjetas micro SD

El tiempo de vida de las tarjetas micro SD actuales se estima en alrededor de 100.000 escrituras. Esto puede parecer suficiente y sería el caso, por ejemplo, para un aparato de fotos personal o un lector MP3. Por el contrario, para un uso más intensivo por la Raspberry Pi (actualización frecuente de datos, swap, etc.) se corre el riesgo de que la tarjeta micro SD alcance rápidamente los límites de su tiempo de vida.

2. Comparativa de los dispositivos de almacenamiento

La Raspberry Pi puede usar varios tipos de dispositivos de almacenamiento, para el almacenamiento de los programas y los datos.

Las diferentes opciones posibles

1. Tarjeta micro SD de clase 4 a 10
2. Llave USB
3. Disco duro externo USB
4. Disco duro externo USB SSD (*Solid-State Drive* = disco duro únicamente compuesto de memorias electrónicas, sin partes móviles)
5. Almacenamiento en una ubicación de red compartida (para la Raspberry Pi Zero con un adaptador)

Cada uno de estos tipos de memoria tiene tiempos de acceso diferentes. Para las memorias estáticas (tarjeta micro SD, llave USB o disco USB), el tiempo determinante es el de acceso en modo escritura.

Velocidad de transferencia

La velocidad de transferencia de las tarjetas SD se indica como una cifra con "C", que indica la clase de la tarjeta y corresponde a una velocidad media de escritura en esta tarjeta. La tarjeta que ilustra la sección anterior es una tarjeta de clase 10. Por tanto, tiene una velocidad media de escritura de 10 MB/s.

Las llaves USB se estabilizan habitualmente a 10 MB/s en modo escritura, la velocidad disminuye durante la escritura de muchos pequeños archivos.

Un disco duro externo USB 2.0 alcanza una tasa de máxima de 30 MB/s.

El disco SSD es capaz de transferencias más elevadas, porque está equipado con una interfaz USB 3.0. La Raspberry Pi solo usará las posibilidades de USB 2.0, es decir, 60 MB/s en el mejor de los casos.

Coste del GB almacenado

Las tarjetas micro SD de clases 4 a 10 y de tamaño de 16 GB solo cuestan algunos euros. El GB de memoria supone alrededor de 0,50 €.

Una tarjeta micro SD de 128 GB cuesta cerca de 50 €, lo que sitúa el precio del GB a 0,39 €, etc.

Las llaves USB están construidas con una tecnología parecida a la de las tarjetas SD, por lo que tienen las mismas limitaciones y suponen aproximadamente 0,50 € por GB. La diferencia de precio puede ser importante entre dos modelos de la misma capacidad, algunas veces el doble. La velocidad de los componentes utilizados en la llave USB podría justificar estas diferencias de precio, a no ser que se trate de una simple cuestión de marketing...

Un disco duro USB externo de 2,5 pulgadas y 500 GB cuesta en torno a 50 €, lo que sitúa el GB a 0,10 €. Es el modelo clásico de disco duro, formado por una o varias bandejas rotativas, recorridas por un cabezal lector. Es sensible a los golpes y el acceso a los datos con un sistema mecánico impone ciertas restricciones (estabilización, vibraciones, etc.) que ralentizan la transferencia de datos.

El disco duro SSD está formado por componentes electrónicos estáticos, sin ninguna restricción mecánica. Sin embargo, la tecnología utilizada es parecida a la de las tarjetas SD y tienen los mismos límites en relación a la escritura. El disco SSD puede soportar centenares de miles de escrituras. Su tiempo de vida se amplía con el uso de discos de gran capacidad y por la gestión del reparto de la escritura en las celdas de memoria menos utilizadas. El precio del GB varía de 0,50 a más de un euro.

El almacenamiento en un disco en red solo está diseñado para una máquina que disponga de estas posibilidades. Poner en marcha un servidor con el único objetivo de ofrecer acceso compartido a la red a la Raspberry Pi implica un gasto desproporcionado. Sin embargo, un PC o una box pueden desempeñar esta función.

3. Elección de una memoria de almacenamiento

La elección de una memoria de almacenamiento externo está relacionada con la aplicación prevista y el coste de la operación. También puede resultar de la disponibilidad de un hardware para la ocasión.

Tabla comparativa de las diferentes soluciones

La siguiente tabla indica la velocidad de escritura y el precio del GB para los diferentes soportes. Se trata de un precio medio conseguido después de analizar las ofertas de varios proveedores, a mediados del año 2016.

- Algunos periféricos de almacenamiento tienen una interfaz USB 3.0. En la Raspberry Pi solo se pueden usar con USB 2.0, que es la norma de la interfaz USB de la Raspberry Pi.

Comparativa de los dispositivos de almacenamiento		
Tipo de memoria	Velocidad de escritura MB/s	Precio del GB en €
Llave USB	10	0.50
Tarjeta micro SD clase 4	4	0.40
Tarjeta micro SD clase 10	10	0.50
Disco duro externo	30	0.10
Disco duro SSD	60	0.50

Es hora de elegir

Con toda esta información, ¿cómo orientar su elección?

El uso de su Raspberry Pi será uno de los criterios determinantes:

- Si la Raspberry Pi solo es una herramienta de iniciación y desea mantener el precio lo más bajo posible, con una facilidad de uso máxima y sin preocuparse de la fiabilidad, la tarjeta micro SD se adapta a sus necesidades. Pasará fácilmente de un sistema a otro, simplemente cambiando la tarjeta micro SD. De esta manera puede experimentar rápidamente diferentes sistemas operativos. En caso de problema, la reconstrucción de una tarjeta micro SD es muy rápida.
- Si va a usar la Raspberry Pi para aplicaciones más críticas, para las que el tiempo de acceso a la memoria de almacenamiento es importante, decántese por una llave USB o un disco duro.
- Si necesita capacidades de almacenamiento importantes (vídeos para un centro multimedia, por ejemplo), el disco duro o el almacenamiento en red son las opciones preferidas. El disco SSD está en una gama de precio elevado, que no guarda relación con el bajo coste de la Raspberry Pi.

- Una solución (si dispone de un disco duro SATA o IDE de 2,5 pulgadas o 3,5 pulgadas) puede consistir en comprar un adaptador USB-IDE o USB-SATA disponible en cualquier gran superficie, con un precio aproximado de 20 €.

Discos y particiones en Linux

En el sistema Linux, los discos no aparecen directamente en el sistema de archivos (la arborescencia) como en Windows. El acceso a los discos y particiones se hace a través de los archivos especiales, que sirven de interfaz entre el sistema y el periférico.

1. Identificación de los discos en Linux

Un disco externo USB, conectado a una máquina que funciona con Windows, se detecta y aparece como una partición en el administrador de discos. Estas particiones se llaman *F:*, *G:* u otra, según la primera letra disponible después de las reservadas para los lectores locales.

En Linux, un disco aparece en la carpeta */dev* como un archivo especial que permite al sistema acceder al disco.

Los discos IDE (*Integrated Drive Electronics* = disco duro de integración electrónica) conectados a la placa base de los PC por una capa de 40 cables tienen el nombre de *hda*, *hdb...* en la carpeta */dev*. Desde hace algunos años, se han sustituido por los discos SATA (*Serial Advanced Technology Attachment* = unión en serie en tecnología de puntos).

Los discos SATA o SCSI (*Small Computer System Interface* = interfaz para pequeños sistemas informáticos) se llaman *sda*, *sdb...* en la carpeta */dev*.

2. Identificación de las particiones en Linux

En Linux, las particiones reciben un nombre que se forma con el nombre del disco duro en el que se encuentran (nombre del disco completo seguido de una cifra). Por ejemplo, un disco llamado *sda* y con dos particiones, se mostrará como *sda1* y *sda2*. Las particiones están accesibles en el directorio */dev*, que agrupa todos los archivos especiales del sistema Linux.

Archivos especiales

Para tener una visión general de los archivos especiales presentes en el directorio */dev* de la Raspberry Pi, escriba el comando:

```
pi@raspberrypi ~ $ ls /dev
```

```
pi@raspberrypi:~ $ ls /dev
autofs      loop5      ram3       tty16     tty43     urandom
block       loop6      ram4       tty17     tty44     vc-cma
bsg         loop7      ram5       tty18     tty45     vchiq
btrfs-control loop-control ram6       tty19     tty46     vcio
bus         mapper     ram7       tty2      tty47     vc-mem
cachefiles  mem        ram8       tty20     tty48     vcs
char        memory_bandwidth ram9       tty21     tty49     vcs1
console     mmcblk0    random     tty22     tty5      vcs2
cpu_dma_latency mmcblk0p1  raw       tty23     tty50     vcs3
cuse        mmcblk0p2  rfskill   tty24     tty51     vcs4
disk        mmcblk0p5  sda       tty25     tty52     vcs5
fb0         mmcblk0p6  sda1      tty26     tty53     vcs6
fd          mmcblk0p7  serial1   tty27     tty54     vcsa
full        mqueue     sg0       tty28     tty55     vcsa1
fuse        net        snd       tty29     tty56     vcsa2
gpiomem    network_latency spidev0.0 tty30     tty58     vcsa3
hidraw0    network_throughput spidev0.1 tty31     tty59     vcsa5
hidraw1    null       stderr    tty32     tty6      vcsa6
hidraw2    ppp        stdin     tty33     tty60     vcsm
hidraw3    ptmx       stdout    tty34     tty61     vchi
hwrng     pts        tty       tty35     tty62     watchdog
initctl    ram0       tty0      tty36     tty63     watchdog0
input      ram1       tty1      tty37     tty7      xconsole
kmsg       ram10      tty10     tty38     tty8      zero
log        ram11      tty11     tty39     tty9
loop0      ram12      tty12     tty4      ttyAMA0
loop1      ram13      tty13     tty40     ttyprintk
loop2      ram14      tty14     tty41     uhid
loop3      ram15      tty15     tty42     uinput
loop4      ram2
```

Esto son cerca de 160 archivos especiales que existen en */dev*. Es aquí donde aparecen los discos y particiones que nos interesan.

Nomenclatura de los archivos especiales

La tarjeta micro SD se identifica con el nombre *mmcblk0* (*Memory Card Block*) y sus particiones *mmcblk0p1* (partición 1) a *mmcblk0p6* (partición 6).

La llave USB aparece como *sda*. Está en la categoría de los discos SCSI/SATA (*sd* = *SCSI Disk*) y como es el primer disco tiene la letra *a*. Un segundo disco conectado a un puerto USB de la Raspberry Pi toma el nombre *sdb*. Las particiones de estos discos se indican con cifras:

- **sda1** = primera partición del primer disco
- **sda2** = segunda partición del primer disco
- **sdb1** = primera partición del segundo disco

Linux es capaz de administrar varias particiones en los dispositivos extraíbles (llave USB, tarjeta micro SD). En estos mismos soportes, Windows solo reconoce una única partición, siempre y cuando esté en un formato que reconozca (de tipo FAT para File Allocation Table = tabla de asignación de archivos). Las demás particiones se ignoran.

3. Label y UUID

Las particiones, una vez formateadas, se pueden identificar con su nombre de archivo especial, pero también se les puede asignar una etiqueta o usar el UUID (*Universal Unique Identifier* = identificador único universal) del sistema de archivos generado durante el formateo.

Identificación de una partición con etiqueta

La etiqueta es una facilidad que permite al usuario llamar a una partición según sus costumbres. No hay ninguna seguridad en la nomenclatura y se pueden asignar etiquetas idénticas a particiones diferentes.

Identificación de una partición por UUID

Linux asigna el UUID de manera pseudo-aleatoria a cada sistema de archivos. Está formado por 128 bits, y la probabilidad de que dos UUID sean idénticos es extremadamente baja.

Según las versiones de los generadores de UUID, se utilizan ciertas características de la máquina (dirección MAC, número de disco duro, etc.), así como la hora de creación en décimas de microsegundo, para crear el número. A esto se puede añadir funciones de hash criptográfico. El número astronómico de posibilidades garantiza que cada UUID generado es único.

Ejemplo de UUID: 548da5o2-ebde-45c0-9ab2-de5e2431ee0b

Preparación de la llave USB

Para realizar las operaciones, hágase con una llave USB. El modelo seleccionado aquí es una clave de 4 GB, pero se puede adaptar en función de la disponibilidad.

Para empezar, enchufe su llave USB y localice el nombre del archivo especial correspondiente:

```
pi@raspberrypi ~ $ ls /dev/sd*
/dev/sda
```

Si su llave USB es la única memoria de almacenamiento conectada a la Raspberry Pi, debería existir un archivo **sda** en la carpeta */dev*.

La llave utilizada para estas pruebas solo se ha conectado a una máquina Windows desde que se compró y no tiene partición.

Creación de las particiones

Atención: a partir de este momento las operaciones que se realicen en la llave USB van a eliminar los datos presentes en ella. Haga una copia de seguridad de los datos existentes en la llave en otro soporte.

La llave USB utilizada para este ejemplo es una clave 4 GB, que se dividirá en dos particiones de 2 GB.

Inicie el comando `fdisk` en el periférico *sda* para crear las particiones en la llave USB.

```
pi@raspberrypi ~ $ sudo fdisk /dev/sda
Device contains neither a valid DOS partition table, nor Sun, SGI
or OSF disklabel
Building a new DOS disklabel with disk identifier 0x06b2422c.
Changes will remain in memory only, until you decide to write
them.
After that, of course, the previous contain won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be
corrected by w(rite)

Command (m for help):
```

Escriba `n` para crear una nueva partición. Cree una partición primaria por defecto (pulse la tecla [Intro]), asígnele el número 1 por defecto e indique el tamaño deseado para esta partición `+2G`. Compruebe por último que la partición se ha guardado correctamente con el comando `p`.

```
Command (m for help): n
Partition type:
  p   primary (0 primary, 0 extended, 4 free)
  e   extended
Select (default p):
Partition number (1-4, default 1):
Using default value 1
First sector (2048-7831551, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-7831551, default
7 831 551): +2G

Command (m for help): p

Disk /dev/sda: 4009 MB, 4 009 754 624 bytes
124 heads, 62 sectors/track, 1018 cylinders, total 7831552 sectors
Unites = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x06b2422c

Device Boot      Start         End      Blocks   Id System
/dev/sda1        2048     4 196 351      2 097 152   83 Linux
```

Cree una segunda partición, que ocupará la totalidad del espacio disponible en la llave. Compruebe la tabla de partición.


```
Device Boot      Start         End      Blocks   Id System
/dev/sda1        2048     4 196 351      2 097 152   83 Linux
/dev/sda2        196 352     7 831 551      1 817 600   83 Linux
```

De momento, esta tabla de partición solo existe en la memoria RAM de la Raspberry Pi. Hay que escribirla en la llave para que las modificaciones tengan efecto. El comando `w` (*write* = escribir) lleva a cabo esta operación:

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 22:
Argument invalide.
The kernel still uses the old table. The new table will be used
at the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.
```

 La tecla `q` permite salir de `fdisk` sin guardar las modificaciones.

Después de escribir la tabla de partición, el sistema indica que el núcleo Linux utiliza todavía la antigua tabla de partición. Aquello que se escriba solo tendrá efecto durante el próximo reinicio.

Por tanto, hay que reiniciar el sistema para tener en cuenta las modificaciones:

```
pi@raspberrypi ~ $ sudo reboot
```

Después de reiniciar el sistema, la llave USB aparece siempre bajo el nombre *sda*, y las dos particiones creadas *sda1* y *sda2* también se muestran.

```
pi@raspberrypi ~ $ ls /dev/sd*
/dev/sda  /dev/sda1  /dev/sda2
```

Formateo de las particiones

Las particiones existen, pero no contienen nada. En particular, falta el sistema de archivos, necesario para que el sistema pueda almacenar y después encontrar los archivos.

La siguiente etapa es el formateo de la partición. Esta operación va a preparar la partición para que el sistema pueda almacenar sus datos. Es un poco como si hubiera pedido un armario (la partición) para guardar los elementos de bricolaje: habrá que poner las estanterías y cajones (el sistema de archivos) para poder almacenar sus tornillos, tuercas y arandelas (los datos), etc. Sin esto, el armario no se podrá utilizar.

Existen diferentes sistemas de archivos que se pueden usar con GNU/Linux:

- *ext2* o *ext2fs* (*second extended file system* = segundo sistema de archivos extendido), es el sistema de archivos histórico de GNU/Linux.
- *ext3* es una evolución de *ext2*. El seguimiento del log que apareció con *ext3* permite recuperar más fácilmente los datos en caso de parada repentina del sistema operativo (fallo del sistema o corte de la alimentación eléctrica). El tamaño máximo de un archivo es de 2 TB y el tamaño máximo del sistema de archivos es de 16 TB.
- *ext4* se adapta mejor a los equipamientos modernos. Gestiona archivos de 16 TB como máximo y un sistema de archivos de 1024 PB. Es más fiable y con mejor rendimiento para los archivos voluminosos.

➤ 1024 PB = 1024 Peta Byte, es decir, 1024×10^{15} bytes, es decir 10^{18} bytes.

El comando `mkfs` (*make file system* = crear un sistema de archivos) realiza esta operación. Necesita dos argumentos:

- El tipo de sistema de archivos a crear (si este argumento se omite, la partición se formateará en *ext2*).
- El nombre de la partición a formatear.

Formatee la primera partición de la llave USB con el sistema de archivos *ext4*.

```
pi@raspberrypi ~ $ sudo mkfs -t ext4 /dev/sda1
mke2fs 1.42.5 (29-Julio-2012)
Etiqueta del sistema de archivos=
Tipo de sistema operativo: Linux
Tamaño del bloque=4096 (log=2)
Tamaño del fragmento =4096 (log=2)
"Stride" = 0 bloques, "Stripe width" = 0 bloques
131 072 i-nodos, 524 288 bloques
26 214 bloques (5.00%) reservados para el super usuario
Primer bloque de datos=0
Número máximo de bloques del sistema de archivos=536870912
16 grupos de bloques
32 768 bloques por grupo, 32 768 fragmentos por grupo
8192 i-nodos por grupo
Super bloques de seguridad almacenados en los bloques:
32 768, 98 304, 163 840, 229 376, 294 912

Asignación de las tablas de grupo: 0/16 1/16 2/16 3/16 4/16 5/16
6/16 7/16 8/16 9/1610/1611/1612/1613/1614/1615/16 terminada
Escritura de las tablas de i-nodos: 0/16 1/16 2/16 3/16 4/16 5/16
6/16 7/16 8/16 9/1610/1611/1612/1613/1614/1615/16 terminada
Creación del log (16 384 blocs): terminada
Escritura de los super bloques y de la información de compatibilidad
del sistema de archivos: 0/16 1/16 2/16 3/16 4/16 5/16 6/16 7/16 8/16
9/1610/1611/1612/1613/1614/1615/16 terminada
```

Formatee de la misma manera la segunda partición de la llave USB.

Asignación de una etiqueta a las particiones

Las particiones *sda1* y *sda2* están formateadas, pero no tienen nombre. En la captura de pantalla anterior, la zona que muestra la etiqueta está vacía:

```
Etiqueta del sistema de archivos=
```

Es posible, pero no obligatorio, dar a las particiones *sda1* y *sda2* una etiqueta que facilite su identificación. El tamaño máximo de la etiqueta es de 16 caracteres.

El programa *e2label*, instalado por defecto con Raspbian, se encarga de esta operación.

Asigne un nombre a la primera partición:

```
pi@raspberrypi ~ $ sudo e2label /dev/sda1 data_01
```

Llame de la misma manera a la segunda partición: *data_02*.

Mostrar etiqueta y UUID

El comando `blkid` muestra, para cada partición, la información disponible:

```
pi@raspberrypi ~ $ sudo blkid
/dev/mcblkp1: LABEL="RECOVERY" UUID="C6C9-01A3" TYPE="vfat"
/dev/mcblkp5: SEC_TYPE="msdos" LABEL="boot" UUID="5D2D-B09A"
TYPE="vfat"
/dev/mcblkp6: UUID="41cd5baa-7a62-4706-b8e8-02c43ccee8d9"
TYPE="ext4"
/dev/sda1: LABEL="data_01" UUID="90424b84-50cf-4d80-a45e-
65e85a6e1a80" TYPE="ext4"
/dev/sda2: LABEL="data_02" UUID="b1efcef0-1fb4-4a19-8a2f-
e0b736ac85cd" TYPE="ext4"
```

La partición de la tarjeta micro SD */dev/mcblkp01*, de tipo *vfat*, tiene la etiqueta *RECOVERY*. Esta etiqueta se ha asignado durante la creación de la partición por NOOBS, herramienta que facilita el inicio de la Raspberry Pi y la creación de la tarjeta micro SD. Su UUID es corto: *C6C9-01A3*, es decir, 4 bytes (32 bits). La probabilidad de que haya otra partición con el mismo UUID es mayor para las particiones *vfat* que para las particiones Linux, cuya longitud del UUID es de 16 bytes (128 bits).

La siguiente partición */dev/mcblkp05*, de tipo *msdos*, tiene la etiqueta *boot* y también tiene un UUID corto.

/dev/mcblkp06 es una partición *ext4*. No tiene etiqueta, pero tiene un UUID.

/dev/sda1 y */dev/sda2*, creadas anteriormente en la llave USB, tienen cada una un UUID y una etiqueta.

➤ vfat corresponde al sistema FAT32 utilizado por las últimas versiones de Windows 95 y posteriores. msdos es el tipo utilizado por MS-DOS y las primeras versiones de Windows 95 (FAT16).

Definir un punto de conexión

En Windows, cada partición se identifica con una letra distinta C:, D:, etc. En Linux las particiones simplemente se "cuelgan" en la arborescencia general del sistema, con un "Punto de conexión".

1. La arborescencia en Linux

En Linux, el sistema de archivos se organiza en forma de árbol (consulte el capítulo Usar la línea de comandos).

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

Hay dos directorios que nos interesan particularmente:

- **/media** destinado a la conexión de los medios extraíbles.
- **/mnt** habitualmente utilizado para la conexión temporal de sistemas de archivos.

Para acceder a las particiones de la llave USB, es necesario crear un directorio a partir del cual el sistema operativo podrá acceder al sistema de archivos presente en el soporte.

Este directorio se puede situar en cualquier parte del árbol, pero una buena práctica consiste en usar los directorios previstos para este efecto.

Para probar las etapas de conexión/desconexión de un sistema de archivos, */mnt* es el directorio adecuado.

2. Crear un punto de conexión

La creación de un punto de conexión se resume en la creación de un directorio */mnt/partition1*, destinado a acoger el sistema de archivos situado en la llave USB.

Después de la conexión de la partición situada en la llave USB, los archivos escritos en este directorio */mnt/partition1* se escribirán en la partición correspondiente de la llave USB.

Para poder montar las dos particiones de la llave USB, será necesario crear dos directorios con los comandos:

```
pi@raspberrypi /mnt $ sudo mkdir partition1
pi@raspberrypi /mnt $ sudo mkdir partition2
```

Utilizar una memoria externa

Tan pronto como se han creado los directorios que van a servir de punto de conexión, es posible colgar sistemas de archivos externos a la arborescencia de la Raspberry Pi.

1. Conexión manual de la llave USB

Después de crear los puntos de conexión, es necesario montar las particiones en estos directorios para que aparezcan en el sistema de archivos de la Raspberry Pi. El comando `mount` realiza esta operación.

La partición presente en la llave USB, llamada `/dev/sda1` por el sistema, se conecta en el punto de conexión `/mnt/partition1` con el comando:

```
pi@raspberrypi /mnt $ sudo mount -t ext4 /dev/sda1
/mnt/partition1
```

Sintaxis

El comando `mount` recibe tres argumentos:

- **-t ext4** indica el tipo de partición a montar. Los tipos de partición reconocidos son muchos (`ext`, `ext2`, `ext3`, `ext4`, `hfs`, `msdos`, `vfat`, `nfs`, `iso9660`, `smbfs`, etc.). De esta manera, es posible montar una gran variedad de soportes de memoria de almacenamiento en el sistema de archivos de la Raspberry Pi.
- **/dev/sda1** es la partición que es necesario montar en el punto de conexión.
- **/mnt/partition1** es el directorio en el que aparecerá `/dev/sda1`.

El comando `mount` también acepta opciones con argumentos `-o`, entre los que figuran:

- **ro** autoriza únicamente la lectura en el sistema de archivos montado.
- **rw** autoriza la lectura y la escritura en el sistema de archivos montado.
- **user** autoriza a los usuarios normales (los que no son `root`) a montar un sistema de archivos.

Monte la segunda partición de la llave USB en el directorio `/mnt/partition2` y dele permisos de solo lectura. Compruebe con el comando `mount` que la opción `ro` se ha aplicado correctamente en `/mnt/partition2`. El comando `mount` utilizado solo muestra la lista de los archivos montados.

```
/dev/sda1 on /mnt/partition1 type ext4 (rw,relatime,data=ordered)
/dev/sda2 on /mnt/partition2 type ext4 (ro,relatime,data=ordered)
```

La conexión manual de las particiones da una gran flexibilidad para administrar memorias de almacenamiento externas. Sin embargo, el sistema Linux utiliza un sistema que ahorra las transferencias USB. Cuando escribe en la llave, el sistema guarda los archivos en la memoria RAM de la Raspberry y solo escribe en la llave USB si el tamaño de los datos a escribir es suficiente, o para garantizar una desconexión correcta.

Esto significa que entre el momento en que escribe "lógicamente" en la llave y el momento en que se produce la escritura "física" en el soporte puede transcurrir un periodo no despreciable, y totalmente invisible para el usuario.

La extracción de la llave USB sin precaución se puede hacer siempre y cuando la sincronización entre la memoria RAM y la llave USB no se haya realizado, o tras una escritura en la llave USB. Puede ser que haya archivos presentes en la llave USB que estén bloqueados o que no se hayan cerrado correctamente y tengan incoherencias.

Desafortunadamente, el defecto solo aparecerá cuando se vuelva a conectar la llave USB. Con un poco de suerte, solo uno o varios archivos no serán legibles. En el peor de los casos será imposible acceder a la llave USB.

- En Linux es importante retirar una llave USB o un disco duro externo asegurándose de que los datos se han escrito correctamente en el soporte extraíble. La sección siguiente describe esta operación.

2. Desmontar una partición

La desconexión de una partición en Linux se hace con el comando `umount`. Este comando va a desconectar correctamente el sistema de archivos presente en la partición de la llave USB del sistema Linux.

En primer lugar, `umount` llama al comando `sync` cuya función es transferir los datos de la memoria RAM del sistema a la partición de la llave USB con el objetivo de sincronizar los archivos y terminar las escrituras actuales.

En segundo lugar, `umount` va a desconectar del sistema Linux de la partición presente en la llave USB. Como esto se hace con motivo de la sincronización, ya no hay riesgo de corromper los datos. `umount` devuelve un error si un archivo queda abierto y hay alguna aplicación que lo utiliza. En este caso, es el usuario el responsable de cerrar la aplicación.

Desconexión de la partición `/dev/sda1`:

```
pi@raspberrypi /mnt $ sudo umount /dev/sda1
```

Comprobación con el comando `mount` de la desconexión de `/dev/sda1`:

```
pi@raspberrypi /mnt $ mount
...
/dev/sda2 on /mnt/partition2 type ext4 (ro,relatime,data=ordered)
```

La partición `/dev/sda1` ya no aparece más en la lista de los sistemas de archivos montados.

- Desmontar la partición `/dev/sda1` no elimina el directorio `/mnt/partition1`, que continúa estando presente (pero vacío) en `/mnt`.

Atención: la desconexión de la partición `sda1` no afecta a la partición `/dev/sda2`, que continúa montada. Por tanto, hay que desmontar también la partición `sda2` para poder retirar la llave USB con total seguridad. Esta precaución vale para todas las particiones que se hayan montado en el sistema.

3. Conexión automática al inicio

En Linux es posible montar automáticamente una partición durante el inicio del sistema. En este caso la llave (o el disco duro USB) se debe conectar a la Raspberry Pi antes de iniciar el sistema. El archivo de configuración que permite automatizar la conexión de una partición se llama `fstab` (*file system table* = tabla de los sistemas de archivos). Está situado, como la mayoría de los archivos de configuración, en el

directorio `/etc` de la arborescencia del sistema operativo Linux.

Para examinar su contenido, utilice el comando `cat`:

```
pi@raspberrypi /etc $ cat fstab
proc /proc proc defaults 0 0
/dev/mmcblk0p5 /boot vfat defaults 0 2
/dev/mmcblk0p6 / ext4 defaults,noatime 0 1
# a swapfile is not a swap partition, so no using swapon|off from
here on, use dphys-swapfile swap[on|off] for that
pi@raspberrypi /etc $
```

En conexión automática al inicio figuran en el orden:

- **/proc**, un sistema de archivos especial que da, en tiempo real, información acerca del sistema y los procesos que se están ejecutando (consulte el capítulo Usar la línea de comandos). Está montado en el punto de conexión `/proc`. El tipo de partición también es especial:`proc`.
- **/dev/mcbk0p5**, la partición de inicio del sistema en la Raspberry Pi. Está situada en la tarjeta micro SD y montada en `/boot`. El tipo de esta partición es `vfat`.
- **/dev/mcbk0p6**, la partición que contiene el sistema de archivos raíz del sistema Linux en la Raspberry Pi. Está situada en la tarjeta micro SD y montada en `/`. El tipo de esta partición es `ext4`.

Para que las particiones de la llave USB se carguen al inicio del sistema es necesario añadir dos líneas al final de `fstab`, respetando la sintaxis necesaria (use `nano`):

```
/dev/sda1 /mnt/partition1 ext4 defaults 0 2
```

- Antes de modificar `fstab`, haga una copia de seguridad. Un error de sintaxis en este archivo puede impedir que el sistema pueda arrancar. Volviendo al `fstab` original en lugar del modificado es posible retomar el control de la Raspberry Pi.

En orden, esto es lo que encontraremos en esta línea:

- **/dev/sda1**: el nombre de la partición a montar.
- **/mnt/partition1**: el punto de conexión de la partición.
- **ext4**: el tipo de partición a montar.
- **defaults**: los parámetros por defecto (`rw,suid,dev,exec,auto,nouser,async`).
- **0**: indica a `dump` no hacer backup de este sistema de archivos.
- **2**: orden de comprobación por `fsck` (`file system check` = comprobación del sistema de archivos).

La modificación se tendrá en cuenta durante el próximo inicio de la máquina, o forzando la carga de `fstab` con el comando `mount -a`, que monta todos los sistemas de archivos presentes en `fstab` si no están montados:

```
pi@raspberrypi /mnt/partition1 $ sudo mount -a
```

La ausencia de mensajes de error indica que todo ha ido bien. Compruebe que la partición se ha montado correctamente:

```
pi@raspberrypi /mnt/partition1 $ mount
...
/dev/sda1 on /mnt/partition1 type ext4 (rw,relatime,data=ordered)
```

Agregue la segunda partición presente en la llave USB a `fstab` y compruebe que el sistema la ha conectado correctamente.

Atención: la conexión automática de particiones externas no implica la automatización de la desconexión. Respete el procedimiento de desconexión si tiene que desconectar un soporte extraíble de la Raspberry Pi.

4. Uso de los UUID

En ocasiones, cuando están conectados varios discos o llaves USB a una Raspberry Pi, puede suceder que la detección de los periféricos de almacenamiento no se realice sistemáticamente en el mismo orden.

Por ejemplo, la partición que desea montar está en la llave USB y se identifica como `/dev/sda1` durante un primer inicio. Durante el siguiente inicio, se detecta otro periférico en primer lugar y se convierte en `sda`. Por tanto, su llave USB se verá como `sdb` y la partición que le interesa se convertirá en `/dev/sdb1`.

Este riesgo puede tener pocas o ninguna consecuencia, ya que se trata de una partición que contiene datos. Pero no será igual cuando se trate de una partición que se usa durante el arranque. En este caso, el inicio se interrumpirá por no poder acceder a los archivos del sistema.

La solución a este problema reside en el uso de los UUID, que permiten identificar de manera determinada y única las particiones de los dispositivos de almacenamiento.

En este caso, la conexión de la primera partición de la llave USB en `fstab` se convierte en:

```
UUID=90424b84-50cf-4d80-a45e-65e85a6e1a80 /mnt/partition1
ext4 defaults 0 2
```

El comando `mount` permite comprobar que la conexión de la partición se ha realizado correctamente:

```
pi@raspberrypi /etc $ mount
...
/dev/sda1 on /mnt/partition1 type ext4 (rw,relatime,data=ordered)
```

Solo se modifica el inicio de la línea en `fstab`: `/dev/sd1`, cambiándola por `UUID=90424b84-50cf-4d80-a45e-65e85a6e1a80`. El resto de la línea permanece inalterada.

- El UUID se puede obtener con el comando `blkid`.

Al inicio del sistema, el comando `mount` va a cargar las particiones presentes en `fstab`. Cuando llega a la línea correspondiente a la primera partición de la llave USB, esta se identifica por su UUID. Poco importa si se ha detectado en `/dev/sda1` o `/dev/sdb1`: su identificador único garantiza que esta será la partición que se montará.

Monta automáticamente la segunda partición de su llave USB, utilizando su UUID. Compruebe con `mount` que la operación se ha realizado correctamente.

Modificación de los permisos

Durante la preparación de la llave USB se aplican los permisos del administrador. Para que un usuario normal pueda usar el periférico, es necesario modificar los permisos de acceso.

Aplicar el permiso 777 al conjunto del disco puede parecer sencillo, pero resulta peligroso porque abre el disco a cualquier acceso sin ningún control.

1. Partición FAT o NTFS

Para una partición FAT o NTFS es mejor crear un grupo que tenga acceso al disco y darle los permisos al conjunto del disco. Basta con añadir a continuación los usuarios a este grupo para asignarles dichos permisos. Estos usuarios pueden ser usuarios reales o usuarios establecidos por las aplicaciones.

Cree el grupo *diskuser*, autorizado a usar el disco:

```
pi@raspberrypi ~ $ sudo groupadd diskuser
```

A continuación hay que añadir el usuario *pi*, y el resto que desee, al grupo *diskuser*:

```
pi@raspberrypi ~ $ sudo usermod -a -G diskuser pi
```

Añada ahora la siguiente línea al archivo */etc/fstab*, para montar automáticamente la partición FAT:

```
/dev/sda2 /mnt auto auto,async,gid=diskuser,umask=007 0 2
```

🔸 El argumento *umask* elimina los permisos indicados. Aquí, el valor 7 indica que los usuarios que no son ni propietarios ni miembros del grupo propietario no tendrán ningún permiso. Este valor es opcional.

Una vez se ha montado la partición, todas las carpetas y archivos de la partición pertenecen al grupo *diskuser*. El comando *mount* indica que *elgid* es el 1005, que es el *id* del grupo *diskuser*.

```
pi@raspberrypi /mnt $ mount
... /...
/dev/sda2 on /mnt type vfat
(rw,relatime,gid=1005,mask=0007,dmask=0007,allow_utime=0020,code
page=cp437,iocharset=ascii,shortname=mixed,errors=remount-ro)
```

Cuando el usuario *pi* o cualquier otro usuario que pertenezca al grupo *diskuser* crea un archivo, este pertenece al grupo *diskuser*.

```
pi@raspberrypi /mnt $ touch Pepe
pi@raspberrypi /mnt $ ls -al
total 20
drwxrwx--- 2 root diskuser 16 384 noviembre  5 19:28 .
drwxr-xr-x 22 root root      4096 noviembre  5 06:24 ..
-rwxrwx--- 1 root diskuser    0 noviembre  5 19:28 Pepe
```

2. Partición Linux

Para las particiones Linux *ext2*, *ext3*, *ext4*, etc. los permisos que se aplican no dependen del usuario que monta la partición. Se basan únicamente en los metadatos almacenados en el sistema de archivos.

En este caso, es necesario usar el comando *bindfs* para ofrecer un acceso a un sistema de archivos, con permisos diferentes. Por defecto *bindfs* no se instala. En primer lugar hay que instalarlo con el comando:

```
pi@raspberrypi ~ $ sudo apt-get install bindfs
```

El sistema de archivos se debe haber montado previamente. Aquí, el sistema de archivos *ext4* presente en el soporte externo se ha montado en */mnt*. Se ha creado un directorio *hdd* en el directorio del usuario *pi*, destinado a acoger al sistema de archivos.

bindfs permite indicar el propietario de todos los archivos (argumento *-u*), así como el grupo propietario (argumento *-g*). Después de la ejecución de *bindfs*, el usuario *pi* se convierte en propietario de los archivos y directorios accesibles desde */home/pi/hdd*.

```
pi@raspberrypi ~ $ sudo bindfs -u pi -g pi /mnt /home/pi/hdd
```

Si comparamos el contenido de */mnt*:

```
pi@raspberrypi /mnt $ ls -al
total 100
drwxr-xr-x 22 root root 4096 agosto  12 10:09 .
drwxr-xr-x 22 root root 4096 noviembre  5 06:24 ..
drwxr-xr-x  2 root root 4096 julio  26 14:19 bin
...
```

con el de */home/pi/hdd*:

```
pi@raspberrypi ~/hdd $ ls -al
total 100
drwxr-xr-x 22 pi pi 4096 agosto  12 10:09 .
drwxr-xr-x 38 pi pi 4096 noviembre  6 06:01 ..
drwxr-xr-x  2 pi pi 4096 julio  26 14:19 bin
```

veremos que es posible acceder al sistema de archivos cuyo propietario y grupo propietario son ahora *pi*.

bindfs utiliza FUSE (*Filesystem in Userspace* = sistema de archivos en el espacio del usuario). El uso de FUSE es susceptible de aumentar la ocupación de memoria y provocar una ralentización de la máquina, provocada por el consumo de recursos.

Conclusión

El uso de una tarjeta micro SD limita la cantidad de memoria disponible. El uso de una memoria de almacenamiento externo da acceso a capacidades de almacenamiento más extendidas, particularmente considerables si la Raspberry Pi se usa como centro multimedia.

Introducción

El capítulo anterior (Utilizar una memoria de almacenamiento externa) muestra cómo usar una memoria de almacenamiento externo para almacenar los datos. Esta memoria puede ser una llave USB o un disco duro USB. El disco duro externo, conectado en modo USB, presenta la ventaja de tener una velocidad de transferencia elevada y un bajo coste por cada GB de memoria.

Cuando arranca la Raspberry Pi B+ es necesaria la tarjeta micro SD. Es en la primera partición de esta tarjeta, formateada en FAT, donde obligatoriamente el SoC va a cargar los primeros archivos de su secuencia de inicio. El capítulo Preparar la tarjeta micro SD explica en detalle las diferentes etapas de este procedimiento de inicio.


Con la Raspberry Pi 3 y el SoC BCM2837 es posible no arrancar desde la tarjeta micro SD, sino directamente desde la llave USB o el disco duro conectado a la Raspberry Pi 3. Esta funcionalidad está disponible como experimento (agosto 2016). Se describe en este capítulo a título informativo.

Obligatoriamente será necesario prever una alimentación externa para el disco duro si se encuentra problemas durante la conexión del disco duro a la Raspberry Pi. Algunos discos duros (cada vez menos numerosos) se entregan con una fuente de alimentación. Otros discos duros disponen de dos tomas USB, una reservada únicamente a la alimentación y la otra para la alimentación y la transferencia de los datos en el bus USB.

Hay que enchufar el cable de alimentación y datos a una de las tomas de la Raspberry Pi, y conectar la toma de alimentación a otra fuente de alimentación, por ejemplo un hub USB alimentado.

Para los discos duros más actuales, que solo tienen una toma USB y consumen poco, en ocasiones puede ser suficiente la conexión a la Raspberry Pi. En caso de pérdida de velocidad durante los accesos a disco será necesario conectar la toma USB a un hub alimentado.

Vigile también la aparición de una franja coloreada en la parte superior derecha de la pantalla o el parpadeo del diodo LED rojo PWR. Esto indica que la tensión de la alimentación de la tarjeta Raspberry Pi es inferior a 4,6 voltios, lo que puede provocar fallos o problemas de escritura en la tarjeta micro SD.

 No todos los hubs USB funcionan con la Raspberry Pi. Compruebe la lista de hubs en http://elinux.org/RPi_VerifiedPeripherals#Working_USB_Hubs.

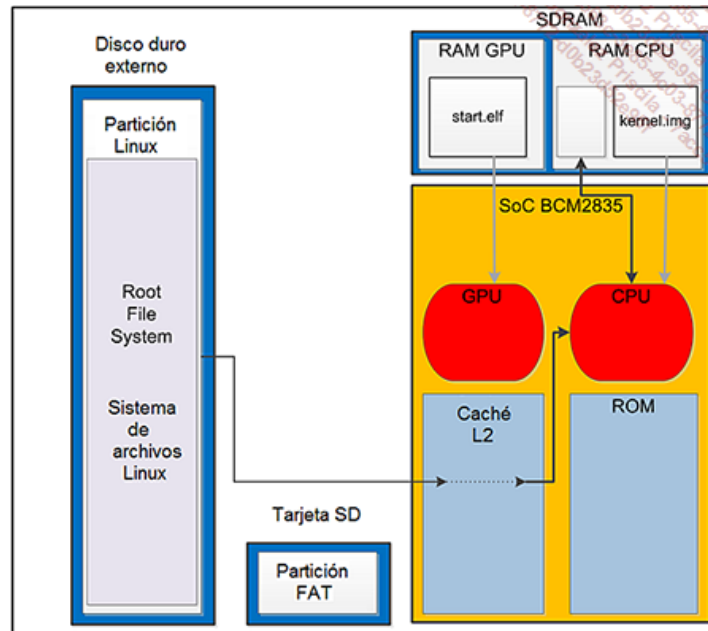
Secuencia de boot de la Raspberry Pi

En la partición FAT de la tarjeta micro SD se encuentran los archivos:

- *bootcode.ini*: la segunda etapa del bootloader.
- *start.elf*: el firmware de la GPU proporcionado en binario.
- *kernel.img*: el núcleo del OS, es decir, de Linux con Raspbian.
- *cmdline.txt*: el archivo que contiene los argumentos que se pasan al núcleo durante el boot.
- *config.txt*: archivo opcional, que lee la GPU antes de inicializar la CPU ARM. Contiene información del reparto de la RAM, el overclocking, el modo de vídeo a usar, etc.

En primer lugar, la GPU carga en memoria el archivo *bootcode.ini*, que luego ejecuta. *bootcode.ini* carga *start.elf*, el firmware de la GPU, en la SDRAM. *start.elf* realiza la compartición de la memoria entre la CPU y la GPU y después carga el núcleo Linux *kernel.img* en la memoria reservada a la CPU.

El siguiente esquema muestra el funcionamiento de la Raspberry Pi cuando el sistema de archivos raíz no está situado en la tarjeta micro SD, sino en un disco duro externo.



kernel.img busca el sistema de archivos raíz (RFS = *Root file System*) en función del argumento *root* (*root=/dev/mmcblk0p2*) que se pasa en *cmdline.txt*. Por defecto es */dev/mmcblk0p2*, que contiene la RFS. Por tanto, será necesario modificar este argumento para que el núcleo acceda al disco duro.

Después del inicio, el sistema no accede más a la tarjeta micro SD.

Preparación del disco

El disco duro seleccionado para el inicio de la Raspberry Pi es un disco duro USB 3.0, de la marca Toshiba y de 1 TB de capacidad. Funcionará en modo USB 2.0 en la Raspberry Pi.

Cuando se compra el disco, solo tiene una partición exFAT de 1 TB.

La preparación del disco va a consistir en eliminar esta partición y crear una o varias particiones para el uso del disco con la Raspberry Pi.

Si el disco se debe utilizar tanto en un PC con Windows como en la Raspberry Pi, por ejemplo para leer en la Raspberry Pi vídeos del PC, es necesario prever dos particiones. Una partición Linux para alojar el sistema de archivos raíz y una partición FAT para almacenar los vídeos.

1. Eliminar la partición origen

El disco duro se ha utilizado inicialmente en una máquina que funciona con Windows. Por tanto, solo tiene una partición del mismo tamaño que el disco.



a. Identificación de la partición a eliminar

El administrador de los discos de Windows identifica el disco como Disco 1, con una partición principal de 931 GB con formato exFAT.

Conecte el disco duro en el hub USB conectado a la Raspberry Pi.

El disco arranca y el sistema lo reconoce, lo que se puede comprobar con el comando `dmesg` (*Display Message* = mostrar mensajes), que muestra en pantalla los mensajes enviados por el núcleo Linux.

Use el comando `dmesg` para visualizar los mensajes del núcleo.

```
[ 1336.537049] usb 1-1.3.2: Product: Portable USB 3.0 Drive
[ 1336.537063] usb 1-1.3.2: Manufacturer: Toshiba
[ 1336.537075] usb 1-1.3.2: SerialNumber: 21302B120345
[ 1336.543037] scsi0 : usb-storage 1-1.3.2:1.0
[ 1340.639583] scsi 0:0:0:0: Direct-Access    TOSHIBA
MQ01ABD100      AX00 PQ: 0 ANSI: 2
[ 1340.645009] sd 0:0:0:0: [sda] 1953525164 512-byte logical
blocks: (1.00 TB/931 GiB)
[ 1340.646124] sd 0:0:0:0: [sda] Write Protect is off
[ 1340.646161] sd 0:0:0:0: [sda] Mode Sense: 00 06 00 00
[ 1340.647250] sd 0:0:0:0: [sda] No Caching mode page present
[ 1340.647284] sd 0:0:0:0: [sda] Assuming drive cache: write
through
[ 1340.656251] sd 0:0:0:0: [sda] No Caching mode page present
[ 1340.656322] sd 0:0:0:0: [sda] Assuming drive cache: write
through
[ 1340.685400] sda: sda1
[ 1340.696530] sd 0:0:0:0: [sda] No Caching mode page present
[ 1340.696600] sd 0:0:0:0: [sda] Assuming drive cache: write
through
[ 1340.696624] sd 0:0:0:0: [sda] Attached SCSI disk
```

Como se acaba de conectar el disco, los mensajes que le afectan se encuentran al final de la cola de mensajes.

Si la conexión del disco duro provoca un reinicio de la Raspberry Pi, la lectura de los mensajes del núcleo ya no se puede realizar.

La herramienta `fdisk` se usa en todos los sistemas operativos para modificar las particiones de los discos duros. En Linux, es accesible por línea de comandos.

Sintaxis

`fdisk`

d	Destruir una partición.
l	Enumerar los tipos de partición conocidos.
n	Añadir una partición.
p	Mostrar la tabla de particiones.
q	Salir sin aplicar las modificaciones.
t	Modificar el identificador de sistema de una partición.
v	Comprobar la tabla de partición.
w	Escribir la tabla de partición.

➤ Esta tabla indica los principales comandos que se pueden usar en modo interactivo después de ejecutar `fdisk`. Para obtener la lista de las particiones existentes, use el comando `fdisk -l` como super-usuario.

Use el comando `fdisk -l` para mostrar los discos conectados a la Raspberry Pi.

```
pi@raspberrypi ~ $ sudo fdisk -l

Disk /dev/mmcblk0: 4008 MB, 4008706048 bytes
4 heads, 16 sectors/track, 122336 cylinders, total 7829504 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00047c7a

Device Boot      Start         End      Blocks   Id  System
/dev/mmcblk0p1    8192      122879       57344    c   W95 FAT32 (LBA)
/dev/mmcblk0p2   22880     7829503     3853312   83   Linux
```

```

Disk /dev/sda: 1000.2 GB, 1000204883968 bytes
255 heads, 63 sectors/track, 121601 cylinders, total 1953525164
sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x40bc56d3

Device Boot      Start          End      Blocks   Id  System
/dev/sda1        2048      1953521663   976759808    7  HPFS/NTFS/exFAT
pi@raspberrypi ~ $

```

En primer lugar, *fdisk* muestra la tarjeta micro SD y sus dos particiones *mmcblk0p1* y *mmcblk0p2* y después el disco */dev/sda* de 1.000 GB, con su partición única */dev/sda1*. Este es el disco duro que se va a particionar de nuevo para recibir el sistema de archivos raíz de la Raspberry Pi y la partición destinada a los datos.

Como sucede con el uso de una llave USB, como vimos en el capítulo anterior, la identificación del disco es una etapa capital que evita la pérdida de datos en caso de error.

b. Eliminación de la partición

Atención: a partir de este punto, las operaciones que realice sobre el disco duro van a eliminar los datos presentes en el disco. Haga una copia de seguridad de los datos que desea conservar en otro soporte.

Ejecute el comando: `sudo fdisk /dev/sda`

```
pi@raspberrypi ~ $ sudo fdisk /dev/sda
```

```
Command (m for help):
```

Escriba *p*, seguido de la tecla [Intro], para mostrar la tabla de particiones y comprobar que puede trabajar correctamente en el disco duro externo.

```
Command (m for help): p
```

```

Disk /dev/sda: 1000.2 GB, 1000204883968 bytes
255 heads, 63 sectors/track, 121601 cylinders, total 1953525164
sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x40bc56d3

```

```

Device Boot      Start          End      Blocks   Id  System
/dev/sda1        2048      1953521663   976759808    7  HPFS/NTFS/exFAT

```

```
Command (m for help):
```

Escriba *d* para eliminar una partición. Como solo existe una partición, esta se selecciona por defecto.

Escriba *p* para confirmar que la partición se ha eliminado.

```
Command (m for help): d
```

```
Selected partition 1
```

```
Command (m for help): p
```

```

Disk /dev/sda: 1000.2 GB, 1000204883968 bytes
255 heads, 63 sectors/track, 121601 cylinders, total 1953525164
sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x40bc56d3

```

```

Device Boot      Start          End      Blocks   Id  System

```

```
Command (m for help):
```

La tabla de particiones está vacía, pero estas modificaciones solo tendrán efecto cuando se escriba en disco la tabla de particiones modificada.

Escriba *w*; *fdisk* escribe la tabla de particiones en disco y termina su ejecución.

2. Creación de dos particiones

La siguiente tabla muestra la información que se usará para preparar el disco duro.

Partición	Sistema	Tipo	Formateo	Nombre	Tamaño
1	Linux	Primaria	ext4	BOOT	300 GB
2	Dos	Primaria	FAT32	DATA	700 GB

Creación de la primera partición

Escriba el comando `sudo fdisk /dev/sda` para modificar la tabla de particiones del disco duro.

Escriba *n* para crear una nueva partición, *p* para crear una partición primaria (también es el valor por defecto). Escriba *1* para indicar a *fdisk* que es la partición número 1, guarde el valor por defecto para el sector de comienzo de la partición (2048) e indique el tamaño de la partición: `+300G`, es decir, 300 GB.

```
Command (m for help): n
```

```
Partition type:
```

```
  p  primary (0 primary, 0 extended, 4 free)
```

```
  e  extended
```

```
Select (default p): p
```

```
Partition number (1-4, default 1): 1
```

```
First sector (2048-1953525163, default 2048):
```

```
Using default value 2048
```

```
Last sector, +sectors or +size{K,M,G} (2048-1953525163, default
```

```
1953525163): +300G
```

Creación de la segunda partición

Escriba `n` para crear una nueva partición, `p` para crear una partición primaria (también es el valor por defecto). Escriba `2` para indicar *fdisk* que es la partición número 2, guarde el valor por defecto para el sector de inicio de la partición. Por defecto, *fdisk* ofrece crear una partición que ocupe todo el espacio todavía disponible. Valide esta opción.

```
Command (m for help): n
Partition type:
  p   primary (1 primary, 0 extended, 3 free)
  e   extended
Select (default p): p
Partition number (1-4, default 2): 2
First sector (629147648-1953525163, default 629147648):
Using default value 629147648
Last sector, +sectors or +size{K,M,G} (629147648-1953525163,
default 1953525163):
Using default value 1953525163
```

Muestre de nuevo la tabla de particiones para comprobar la configuración.

```
Command (m for help): p

Disk /dev/sda: 1000.2 GB, 1000204883968 bytes
255 heads, 63 sectors/track, 121601 cylinders, total 1953525164
sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x40bc56d3

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            2048     629147647    314572800   83  Linux
/dev/sda2    629147648    1953525163    662188758   83  Linux
```

Las dos particiones se visualizan correctamente. Solo se modifica el tipo de partición indicado por el identificador `Id` 83. El tipo 83 corresponde a una partición Linux. Es necesario modificar el tipo de la partición 2 en FAT32 con un identificador `Id` a `c`.

Modifique el `Id` de sistema de la partición 2 para que cambie a `c`. Para ello, escriba `t` y después el valor `c`.

```
Command (m for help): t
Partition number (1-4): 2
Hex code (type L to list codes): c
Changed system type of partition 2 to c (W95 FAT32 (LBA))
```

Muestre de nuevo la tabla de particiones para comprobar la configuración.

```
Command (m for help): p

Disk /dev/sda: 1000.2 GB, 1000204883968 bytes
255 heads, 63 sectors/track, 121601 cylinders, total 1953525164 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x40bc56d3

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            2048     629147647    314572800   83  Linux
/dev/sda2    629147648    1953525163    662188758   c   W95 FAT32 (LBA)
```

➤ La partición `/dev/sda2` es de tipo W95 FAT32 a causa de la modificación del `Id` de sistema. El formateo que se realiza en la siguiente sección va a hacer efectiva la configuración del sistema de archivos FAT32.

La tabla de particiones está lista para la escritura en el disco. Después de esta acción, las modificaciones serán irreversibles.

Escriba `w` para finalizar las modificaciones.

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: If you have created or modified any DOS 6.x partitions,
please see the fdisk manual page for additional information.
Syncing disks.
```

El disco duro está listo para recibir los sistemas de archivos seleccionados.

3. Formateo de las particiones

La partición `sda1` se va a formatear en `ext4`, formato presente por defecto en el núcleo Linux. Por el contrario, la partición `sda2` va a recibir un sistema de archivos FAT. Hay que añadir las herramientas de gestión `msdos` al sistema antes de poder usarlas.

Instale las herramientas de gestión de sistemas de archivos `msdos` con el comando `sudo apt-get install dosfstools`.

```
pi@raspberrypi ~ $ sudo apt-get install dosfstools
```

a. Partición Linux

Inicie el formateo de la partición `sda1` en `ext4` con el comando `sudo mkfs.ext4 /dev/sda1 -L BOOT`.

➤ La opción `-L` permite definir la etiqueta de la partición.


```
pi@raspberrypi ~ $ sudo mkfs.ext4 /dev/sda1 -L BOOT
mke2fs 1.42.5 (29-Jul-2012)
Etiqueta de sistema de archivos=BOOT
Tipo de sistema operativo: Linux
```

```
Tamaño de bloque=4096 (log=2)
Tamaño de fragmento=4096 (log=2)

...
Asignación de las tablas de grupo: terminado
Escritura de las tablas de i-nodos: terminado
Creación del log (32768 bloques): terminado
Escritura de los superbloques y de la información de compatibilidad
del sistema de archivos: terminado
```

b. Partición FAT

Inicie el formateo de la partición *sda2* en FAT32 con el comando `sudo mkfs.vfat /dev/sda2 -n DATA`.

 La opción `-n` permite definir el nombre de la unidad de la partición.

```
pi@raspberrypi ~ $ sudo mkfs.vfat /dev/sda2 -n DATA
mkfs.vfat 3.0.13 (30 Jun 2012)
```

Copia del sistema de archivos

El disco está listo para recibir el sistema de archivos raíz. El uso del comando `cp`, si bien está previsto para copiar el contenido de la tarjeta micro SD al disco duro, es sensible a los permisos que existan en los archivos y directorios. Más radical, el comando `dd` realiza una copia sector por sector, sin tener en cuenta el contenido de los sectores que se están copiando.

Atención: el comando `dd` es peligroso si se utiliza incorrectamente. Si los orígenes y destinos `if` y `of` son erróneos, hay riesgo de eliminación irreversible de datos.

Inicie la copia de la partición `mmcblk0p2` de la tarjeta micro SD a la partición `sda1` del disco duro con el comando `dd`: `sudo dd if=/dev/mmcblk0p2 of=/dev/sda1 bs=32M`.

```
pi@raspberrypi ~ $ sudo dd if=/dev/mmcblk0p2 of=/dev/sda1 bs=32M
117+1 líneas leídas
117+1 líneas escritas
3945791488 bytes (3,9 GB) copiados, 253,2 s, 15,6 MB/s
```

Al cabo de algunos minutos, el contenido del sistema de archivos raíz se copia al disco duro.

Ahora es posible montar la partición `sda1` en el sistema.

Cree un directorio `/mnt/diskusb`.

Monte la partición 1 del disco duro en este directorio.

```
pi@raspberrypi /mnt $ sudo mkdir diskusb
pi@raspberrypi /mnt $ sudo mount /dev/sda1 /mnt/diskdur
pi@raspberrypi /mnt $ cd diskdur
pi@raspberrypi /mnt/diskdur $ ls
bin dev home lost+found mnt proc run selinux sys usr
boot etc lib media opt root sbin srv tmp var
```

El comando `ls` muestra que el sistema de archivos raíz presente en la tarjeta micro SD se ha copiado correctamente en la partición `sda1`.

Muestre el uso del espacio en disco con el comando `df-h`.

```
pi@raspberrypi /mnt/diskdur $ df-h
Sys. fich. 1K-blocks Util. Disponible Uti% Montada en
rootfs 3791568 1565704 2053488 44% /
/dev/root 3791568 1565704 2053488 44% /
devtmpfs 183620 0 183620 0% /dev
tmpfs 38376 264 38112 1% /run
tmpfs 5120 0 5120 0% /run/lock
tmpfs 76740 0 76740 0% /run/shm
/dev/mmcblk0p1 57288 34760 22528 61% /boot
/dev/sda1 3791568 1565696 2053496 44% /mnt/diskusb
```

➤ Para obtener una visualización más legible, use el comando `df` con la opción `-h`. El resultado obtenido se muestra en M (megabytes) y G (gigabytes).

El sistema de archivos `/dev/sda1` es efectivamente la copia de `/dev/root`. Tiene el mismo tamaño, es decir, 1,5 GB (1.565.696 bytes) y 2 GB (2.053.496 bytes, el 44 % del tamaño de la partición) están disponibles. Sin embargo, el tamaño de la partición creada en el disco duro era de 300 GB. Por tanto, hay que ampliar el sistema de archivos que se acaba de copiar a la totalidad del espacio disponible.

➤ La herramienta **Expand Filesystem** presente en `raspi-config` hace esta operación, pero únicamente en la partición 2 de la tarjeta micro SD. Por lo tanto, no se puede usar en el caso del disco externo.

El comando `resize2fs` va a permitir redimensionar el sistema de archivos `/dev/sda1`.

Inicie el redimensionamiento del sistema de archivos `/dev/sda1` con el comando `sudo resize2fs /dev/sda1`.

```
pi@raspberrypi /mnt/diskusb $ sudo resize2fs /dev/sda1
resize2fs 1.42.5 (29-Jul-2012)
El sistema de archivos de /dev/sda1 se ha montado en /mnt/diskusb;
el cambio de tamaño se debe realizar en línea
old_desc_blocks = 1, new_desc_blocks = 19
El sistema de archivos /dev/sda1 ahora tiene una tamaño de
78643200 bloques.
```

El comando `df` muestra la efectividad de esta acción:

```
pi@raspberrypi /mnt/diskusb $ df
Sys. fich. 1K-blocks Util. Disponible Uti% Montada en
rootfs 3791568 1565704 2053488 44% /
/dev/root 3791568 1565704 2053488 44% /
devtmpfs 183620 0 183620 0% /dev
tmpfs 38376 264 38112 1% /run
tmpfs 5120 0 5120 0% /run/lock
tmpfs 76740 0 76740 0% /run/shm
/dev/mmcblk0p1 57288 34760 22528 61% /boot
/dev/sda1 309637120 1568096 295471332 1% /mnt/diskusb
```

El espacio disponible ahora es de 295 GB y el sistema de archivos solo ocupa el 1 % de este espacio.

Modificación de los archivos de inicio

El sistema de archivos raíz está listo. Para que se pueda informar al sistema de los cambios a realizar durante el inicio también se deben modificar los archivos de inicio.

1. cmdline.txt

El archivo *cmdline.txt* contiene los argumentos que se pasan al núcleo durante su inicio. En particular contiene la ruta de acceso al sistema de archivos raíz, que por defecto es */dev/mmcbk0p2*.

En primer lugar, haga una copia de seguridad de *cmdline.txt*. Siempre puede volver atrás en caso de problema.

```
sudo cp cmdline.txt cmdline.org
```

Abra el archivo */boot/cmdline.txt* con *nano* y sustituya *root=/dev/mmcbk0p2* por *root=/dev/sda1* en la línea de comandos.

Compruebe cuidadosamente la modificación y guarde el archivo modificado.

2. fstab

Durante el inicio, el archivo *fstab* indica al sistema las particiones que debe montar, y en qué punto de conexión debe hacerlo.

El *fstab* que nos interesa aquí es el que se solicitará durante el inicio en el disco duro. El *fstab* situado en la tarjeta micro SD no tendrá más utilidad en este momento. Es el *fstab* situado en el disco duro el que es necesario modificar.

En primer lugar, haga una copia de seguridad de *fstab*.

En */mnt/diskdur/etc/fstab* comente (ponga una *#* al comienzo de línea) la línea que contiene */dev/mmcbk0p2*, y añada una línea empezando por */dev/sda1* con los mismos argumentos.

```
#/dev/mmcbk0p2 / ext4 defaults,noatime 0 1
/dev/sda1 / ext4 defaults,noatime 0 1
```

➤ También es posible usar el UUID para la conexión. Es útil si, por motivo del uso de varios periféricos de almacenamiento USB, la partición */dev/sda1* cambia de nombre y se convierte en */dev/sda2*. El uso del UUID garantiza que se montará la partición correcta.

Comprobación del funcionamiento

Reinicie la Raspberry Pi. Si todas las modificaciones se han realizado correctamente, el inicio de la Raspberry se hará en la tarjeta micro SD para las primeras etapas y después el sistema de archivos raíz situado en el disco duro se montará en la raíz.

Compruebe con el comando `df` que el sistema de archivos situado en el disco duro está en servicio.

```
login as: pi
pi@10.25.100.15's password:
Linux raspberrypi 3.6.11+ #474 PREEMPT Thu Jun 13 17:14:42 BST
2013 armv6l

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Aug 13 13:36:13 2013
pi@raspberrypi ~ $ df
Sys. fich.      1K-blocks    Util. Disponible  Uti% Montado en
rootfs         309637120   1568180  295471248      1% /
/dev/root      309637120   1568180  295471248      1% /
devtmpfs       183620      0        183620         0% /dev
tmpfs          38376       260      38116          1% /run
tmpfs          5120        0        5120           0% /run/lock
tmpfs          76740       0        76740          0% /run/shm
/dev/mmcblk0p1 57288       34768    22520         61% /boot
pi@raspberrypi ~ $
```

La Raspberry Pi dispone ahora de 300 GB de espacio para su sistema de archivos.

Ir más allá

El disco duro dispone de una segunda partición formateada en FAT.

Monte manualmente la segunda partición en una carpeta */data*.

Modifique */etc/fstab* para que la partición *sda2* se conecte automáticamente al inicio del sistema.

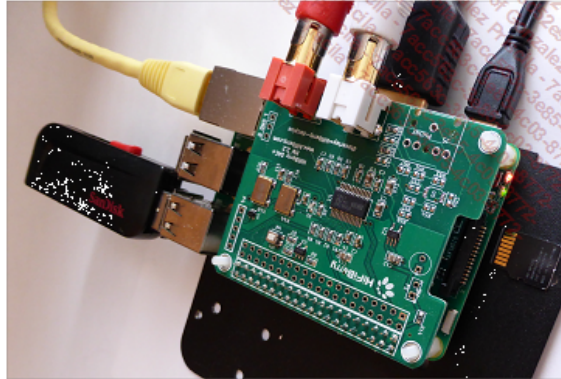
Consulte la solución de estos dos ejercicios en el anexo.

Arrancar en una llave USB sin tarjeta micro SD

Como se ha indicado en la introducción de este capítulo, durante la escritura de estas líneas esta posibilidad ofrecida por el SoC BCM2837 todavía es experimental. Funciona con Raspbian y únicamente con algunas llaves USB y discos duros (clásicos o SSD).

1. Principio

Cuando se enchufa, solo está activada la GPU. El procesador ARM está en modo de espera y la SDRAM está desactivada. El SoC tiene una ROM (*Read Only Memory* - Memoria de acceso de solo lectura) en la que el fabricante ha programado la primera etapa del bootloader. Durante su reactivación, la GPU se configura para ejecutar el programa que está en esta ROM. Esto es lo que hace y el único objetivo del firmware contenido en la ROM es acceder a la tarjeta micro SD, que debe estar formateada en FAT (al menos la primera partición), salvo si se ha modificado un registro de la OTP (*One Time Programmable memory*), que dice al BCM2837 que, cuando arranque, compruebe si hay una tarjeta SD operativa. Si no es el caso, puede intentar arrancar en el USB el SPI o Ethernet...



En la foto anterior, la Raspberry Pi equipada de una tarjeta HiFiBerry funciona normalmente. Arranca directamente en la llave USB, sin intervención de la tarjeta micro SD.

2. Actualización del sistema operativo

Empiece por actualizar Raspbian.

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

Después, el firmware (*start.elf* y *bootcode.bin*) en */boot*, se debe sustituir por la versión de la rama NEXT (experimental): `sudo BRANCH=next rpi-update`:

```
pi@raspberrypi:~ $ sudo BRANCH=next rpi-update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
*** Performing self-update
*** Relaunching after update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
*** Downloading specific firmware revision (this will take a few minutes)
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 168 0 168 0 0 294 0 --:--:-- --:--:-- --:--:-- 294
100 51.1M 100 51.1M 0 0 413k 0 0:02:06 0:02:06 --:--:-- 468k
*** Updating firmware
*** Updating kernel modules
*** depmod 4.4.15-v7+
*** depmod 4.4.15+
*** Updating VideoCore libraries
*** Using HardFP libraries
*** Updating SDK
*** Running ldconfig
*** Storing current firmware revision
*** Deleting downloaded files
*** Syncing changes to disk
*** If no errors appeared, your firmware was successfully updated to
47d64831ccbc99080f98c1bc8948717c3416cb0f
*** A reboot is needed to activate the new firmware
```

3. Autorización del boot USB

La autorización del boot en un puerto USB se hace modificando un registro de la memoria OTP interna al BCM2837. Esta modificación es la que indica al SoC que puede buscar una partición de arranque en otra parte diferente de la tarjeta micro SD. Esta modificación se hace añadiendo `program_usb_boot_mode=1` al final del archivo *config.txt*. Esta adición se puede hacer manualmente con `nano` o rellenando la siguiente línea de comando:

```
echo program_usb_boot_mode=1 | sudo tee -a /boot/config.txt
```

Ahora se puede reiniciar la Raspberry Pi y comprobar que el registro 17 de la OTP se ha modificado correctamente:

```
sudo reboot
```

Cuando la Raspberry Pi ha terminado de arrancar, inicie una sesión y compruebe que el registro 17 tiene el valor correcto:

```
pi@raspberrypi:~ $ vcgencmd otp_dump | grep 17:
17:3020000a
```

El valor hexadecimal por defecto del registro 17 es 1020000a. La línea añadida a *config.txt* ha cambiado el valor del registro. Compruebe que el valor que se muestra es correcto: 3020000a.

👉 `vcgencmd` es una herramienta que permite obtener información de sistema de la Raspberry Pi.

Si lo desea, puede eliminar la línea `program_usb_boot_mode` al final del archivo *config.txt* (no deje la línea en blanco al final del archivo). De esta manera, si utiliza su tarjeta micro SD en otra Raspberry Pi cuyo SoC no se ha modificado, no programa el modo de boot en USB. Puede

utilizar el comando `sudo nano/boot/config.txt`, por ejemplo.

4. Elección de la llave USB

Las pruebas, que se han hecho en agosto de 2016, se han realizado con

- Sandisk Cruzer Fit 16GB
- Sandisk Cruzer Blade 16GB
- Samsung 32 GB USB 3.0 drive
- MeCo 16 GB USB 3.0
- Sandisk Cruzer Slice16 GB



Los ensayos descritos aquí, se han realizado en un modelo de llave USB SanDisk Cruzer Slice de 16 GB (imagen anterior). Si su llave no figura en la lista anterior o en la de la Fundación (<https://www.raspberrypi.org/blog/pi-3-booting-part-i-usb-mass-storage-boot/>) es posible que no permita a la Raspberry Pi arrancar.

➔ Atención, la llave USB se eliminará y todo su contenido se perderá definitivamente.

5. Preparación de la llave USB

Empiece insertando la llave USB en uno de los puertos USB de la Raspberry Pi 3 en funcionamiento. Es el sistema operativo actual funcionando en la Raspberry Pi 3 el que se va a transferir a la llave USB. La fuente (tarjeta micro SD) estará en `/dev/mmcblk0` y el destino (llave USB) debería estar en `/dev/sda`, suponiendo que no tiene otros periféricos USB conectados. Verifique si es el caso para evitar borrar otro periférico USB.

La primera etapa de preparación de la llave consiste en crear dos particiones:

- una partición FAT32 de 100 de MB
- una partición Linux ext4 que ocupará el resto del espacio disponible.

La herramienta `parted` es la que se usará para crear las particiones en la llave USB conectada en `/dev/sda`.

```
sudo parted /dev/sda
```

El software le deja crear las dos particiones con los comandos:

```
mkpart primary fat32 0% 100M
mkpart primary ext4 100M 100%
```

La opción `print` muestra el resultado del particionamiento para que pueda comprobarlo.

A continuación encontrará el avance de la operación:

```
pi@raspberrypi:~ $ sudo parted /dev/sda
GNU Parted 3.2
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) mktable msdos
Warning: The existing disk label on /dev/sda will be destroyed and
all data on
this disk will be lost. Do you want to continue?
Yes/No? Y
(parted) mkpart primary fat32 0% 100M
(parted) mkpart primary ext4 100M 100%
(parted) print
Model: SanDisk Cruzer Slice (scsi)
Disk /dev/sda: 16,0GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number Start End Size Type File system Flags
 1 1049kB 99,6MB 98,6MB primary fat32 lba
 2 99,6MB 16,0GB 15,9GB primary ext4 lba

(parted) quit
Information: You may need to update /etc/fstab.
```

Las particiones se han creado y es necesario formatearlas para poner en marcha los sistemas de archivos necesarios para el funcionamiento de la Raspbian. La primera partición se llama BOOT y está formateada en FAT, la segunda se llama ROOT y está formateada en ext4. En función del tamaño del soporte utilizado (llave USB, disco duro, etc...), esta operación puede ser más o menos larga. Déjela hasta que termine.

```
pi@raspberrypi:~ $ sudo mkfs.vfat -n BOOT -F 32 /dev/sda1
mkfs.fat 3.0.27 (2014-11-12)
pi@raspberrypi:~ $ sudo mkfs.ext4 /dev/sda2
mke2fs 1.42.12 (29-Aug-2014)
Creando un sistema de archivos con 3884032 4k bloques
y 971040 i-nodos.
UUID de sistema de archivos=545cd535-8405-42eb-aafe-e92300belf10
Superbloques de seguridad almacenados en los bloques:
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632,
2654208

Asignación de las tablas de grupo: terminado
Escritura de las tablas de i-nodos: terminado
Creación del archivo de traza (32768 bloques): terminado
Escritura de los superbloques y de la información de compatibilidad
del sistema de archivos: terminado
```

6. Transferencia de Raspbian a la llave USB

La primera etapa para copiar el sistema operativo Raspbian de la tarjeta micro SD a la llave USB es montar la llave USB en el sistema de archivos actual. Para ello es necesario crear un directorio `/mnt/target` y otro `/mnt/target/boot` en la tarjeta micro SD. Estos dos directorios están destinados al montaje de las particiones presentes en la llave USB.

```
pi@raspberrypi:~ $ sudo mkdir /mnt/target
pi@raspberrypi:~ $ sudo mount /dev/sda2 /mnt/target/
pi@raspberrypi:~ $ sudo mkdir /mnt/target/boot
pi@raspberrypi:~ $ sudo mount /dev/sda1 /mnt/target/boot/
```

La transferencia del sistema operativo se va a confiar a `rsync` (herramienta de sincronización en modo remoto). Hay que instalar esta herramienta en Raspbian:

```
pi@raspberrypi:~ $ sudo apt-get install rsync
```

➤ Es posible que `rsync` ya esté instalado en su versión de Raspbian. En este caso, recibirá el mensaje: `rsync ya es la versión disponible mas reciente`. No tenga en cuenta este mensaje y continúe la preparación de la llave USB.

Ahora que los sistemas de archivos de la llave USB están montados en el sistema de archivos de la Raspberry Pi 3, se puede copiar ahí el `boot` y el `root file system (rfs)`... Tardará media hora, incluso más, según el tamaño del sistema.

```
pi@raspberrypi:~ $ sudo rsync -ax --progress /boot /mnt/target
```

Como resultado de la sincronización habrá disponible una copia de la tarjeta micro SD en la llave USB.

7. Regeneración de las claves SSH

La seguridad de SSH se basa en el intercambio de claves (archivos) entre los diferentes protagonistas (consulte el capítulo Conectarse a la Raspberry Pi en remoto). Estas claves permiten identificar de manera segura al remitente de un mensaje. El nuevo sistema que se está creando no puede tener las mismas claves que el que funciona en la Raspberry Pi actualmente. Tiene que crear sus propias claves:

```
pi@raspberrypi:~ $ cd /mnt/target
pi@raspberrypi:/mnt/target $ sudo mount --bind /dev dev
pi@raspberrypi:/mnt/target $ sudo mount --bind /sys sys
pi@raspberrypi:/mnt/target $ sudo mount --bind /proc proc
pi@raspberrypi:/mnt/target $ sudo chroot /mnt/target
root@raspberrypi:/# rm /etc/ssh/ssh_host*
root@raspberrypi:/# dpkg-reconfigure openssh-server
Creating SSH2 RSA key; this may take some time ...
2048 7a:f7:33:4c:01:71:a3:fc:c1:2c:b4:9c:3c:3f:4b:19
/etc/ssh/ssh_host_rsa_key.pub (RSA)
Creating SSH2 DSA key; this may take some time ...
1024 28:c6:4b:d1:5e:d9:cb:8c:ca:ad:70:fe:87:08:a1:21
/etc/ssh/ssh_host_dsa_key.pub (DSA)
Creating SSH2 ECDSA key; this may take some time ...
256 f7:f1:9e:3f:6b:13:41:d2:d9:ee:a1:00:e2:ef:ed:db
/etc/ssh/ssh_host_ecdsa_key.pub (ECDSA)
Creating SSH2 ED25519 key; this may take some time ...
256 fc:98:8c:c3:52:45:95:c1:67:78:41:0c:00:c6:23:f0
/etc/ssh/ssh_host_ed25519_key.pub (ED25519)
[ ok ] Restarting OpenBSD Secure Shell server: sshd.
root@raspberrypi:/# exit
```

La Raspbian recién creada dispone ahora de sus propias claves SSH.

8. Modificación del archivo `cmdline.txt`

El sistema presente en la llave USB era una copia del que existe en la tarjeta micro SD. Es necesario modificar los parámetros de arranque para que Linux no busque el sistema de archivos `root` en la partición `mmcblk0p2` de la tarjeta micro SD, sino en la partición `sda2` de la llave USB. El comando `sed` permite realizar esta operación sustituyendo directamente en el archivo la cadena de caracteres `root=/dev/mmcblk0p2` por `root=/dev/sda2`.

```
pi@raspberrypi:/mnt/target $ sudo sed -i
"s,root=/dev/mmcblk0p2,root=/dev/sda2,"
/mnt/target/boot/cmdline.txt
```

➤ El comando entre `sudo...` y `... txt` se debe introducir en una sola línea.

9. Modificación del archivo `fstab`

El montaje automático de los diferentes sistemas de archivos se configura en el archivo `/etc/fstab`. Originalmente, el montaje de los sistemas de archivos se hacía desde la tarjeta micro SD (`mmcblk0p`). Hay que sustituir el origen de los sistemas de archivos por la llave USB (`sda`). De nuevo, es el comando `sed` el que se va a encargar de la operación:

```
pi@raspberrypi:/mnt/target $
sudo sed -i "s,/dev/mmcblk0p,/dev/sda," /mnt/target/etc/fstab
```

➤ El comando entre `sudo...` y `... fstab` se debe introducir en una sola línea.

10. Desmontar el sistema de archivos

La preparación del nuevo sistema ya ha terminado. Los sistemas de archivos de la llave USB se deben desmontar correctamente:

```
pi@raspberrypi:/mnt/target $ cd ~
pi@raspberrypi:~ $ sudo umount /mnt/target/boot
```

```
pi@raspberrypi:~ $ sudo umount /mnt/target
```

11. Encender la Raspberry Pi 3

Para tener en cuenta el nuevo sistema operativo, detenga el sistema operativo:

```
pi@raspberrypi:~ $ sudo poweroff
```

Cuando el sistema se haya detenido (el LED ACT parpadea varias veces y después permanece apagado) puede desenchufar la toma micro USB de alimentación.

12. Reinicio de la Raspberry Pi 3

Retire la tarjeta micro SD de su conector. Vuelva a enchufar la toma de alimentación. La Raspberry Pi 3 debe arrancar en Raspbian desde la llave USB.

Si el sistema no arranca, verifique que ha respetado todas las etapas de implementación y, eventualmente, inténtelo con otra llave USB.

Es posible obtener más información del sistema con los comandos `fdisk` y `df`:

```
pi@raspberrypi:~ $ fdisk -l
/dev/sda1 2048 194559 192512 94M c W95 FAT32 (LBA)
/dev/sda2 194560 31266815 31072256 14,8G 83 Linux
```

El comando `fdisk` muestra que se utilizan correctamente las particiones de la llave USB (`sda`).

```
pi@raspberrypi:~ $ df -h
Sys. de archivos. Tamaño Utilizado Dispo Uti% Montado en
/dev/root 15G 4,0G 9,8G 29% /
devtmpfs 459M 0 459M 0% /dev
tmpfs 463M 0 463M 0% /dev/shm
tmpfs 463M 6,4M 457M 2% /run
tmpfs 5,0M 4,0K 5,0M 1% /run/lock
tmpfs 463M 0 463M 0% /sys/fs/cgroup
/dev/sda1 93M 20M 73M 22% /boot
tmpfs 93M 0 93M 0% /run/user/1000
```

El comando `df` muestra que los sistemas de archivos están conformes a lo que se ha implementado durante la transferencia del sistema operativo:

`/` ocupa 15 GB (`/dev/root`) y `/boot` ocupa prácticamente 100 MB (93 MB exactamente para `/dev/sda1`).

Conclusión

La Raspberry Pi ahora arranca en el disco duro o en llave USB. La tarjeta micro SD ya solo se pide para las primeras etapas del boot en el primero caso y se puede eliminar en el segundo caso. No se somete a más desgaste debido a las frecuentes escrituras realizadas por Linux durante su funcionamiento.

El tiempo de arranque no se ha modificado fundamentalmente. Pero la capacidad disponible ha aumentado enormemente, lo que puede ser útil para algunas aplicaciones.

La tarjeta micro SD utilizada en este capítulo para el arranque sobre el disco duro, ya no se puede utilizar sola. Marque la tarjeta cuidadosamente para que permanezca asociada al disco duro. Bastará con cambiar la tarjeta para probar otras distribuciones, el disco duro seguirá estando disponible para almacenar los datos.

Por el contrario, la tarjeta micro SD utilizada al inicio del capítulo para transferir Raspbian a una llave USB siempre se puede utilizar sola, porque el sistema simplemente se ha clonado y la tarjeta micro SD no se ha modificado.

Introducción

Diseñada con fines educativos, la Raspberry Pi ha demostrado su adaptabilidad. El objetivo inicial era sacar a los jóvenes de su estado de consumidores controlados por la informática, y animarlos a desarrollar aplicaciones.

De hecho, la Raspberry Pi ha alcanzado un público mucho más amplio, porque su flexibilidad le ha abierto el camino a decenas de aplicaciones tan variadas como inesperadas. La encontramos en las escuelas y universidades, en la industria, en organismos locales, etc.

Este ordenador, con el tamaño de una tarjeta de crédito, ha llegado al espacio. Ha tomado fotos que ha enviado a la tierra desde 40 km de altura. Un Airbus A380 no pasa de los 13 km...

De manera más cercana, equipa instalaciones domóticas, alarmas, media centers, estaciones meteorológicas, centrales telefónicas domésticas, etc. Hace funcionar a robots, cuadricópteros, simuladores de consola de videojuegos, etc.

En este capítulo descubrirá algunas aplicaciones sencillas de implementar y que muestran la diversidad de las aplicaciones al alcance de la Raspberry Pi.

Un puesto de escritorio

Para las escuelas dotadas de pocos medios, en países emergentes por ejemplo, el uso de una suite de Microsoft Office en un PC equipado con un sistema operativo Windows no es factible por el coste de las licencias y del hardware.

1. Usar LibreOffice

La suite LibreOffice se instala con Raspbian Jessie. Ahora puede disfrutar de una suite de escritorio que permite trabajar con todos los tipos de documentos habituales. El funcionamiento en la Raspberry Pi 3 es aceptable, teniendo en cuenta el precio de la solución completa. Vaya al capítulo Utilizar el modo gráfico para iniciar la suite LibreOffice.



La captura de pantalla anterior muestra este capítulo en LibreOffice.

2. Imprimir

La impresión con la Raspberry Pi necesita la instalación de un servidor de impresión CUPS (*Comon Unix printing system* = sistema de impresión común a los sistemas Unix y sistemas compatibles).

Empiece comprobando que su sistema esté actualizado e instale CUPS en la Raspberry Pi.

```
pi@raspberrypi ~ $ sudo apt-get update
pi@raspberrypi ~ $ sudo apt-get upgrade
pi@raspberrypi ~ $ sudo apt-get install cups cups-client
```

Añada al grupo *lpadmin* el usuario que debe gestionar la impresión. Aquí será *pi*.

```
pi@raspberrypi ~ $ sudo usermod -a -G lpadmin pi
```

Puede comprobar que *pi* pertenece al grupo *lpadmin*, mostrando todos sus grupos de pertenencia:

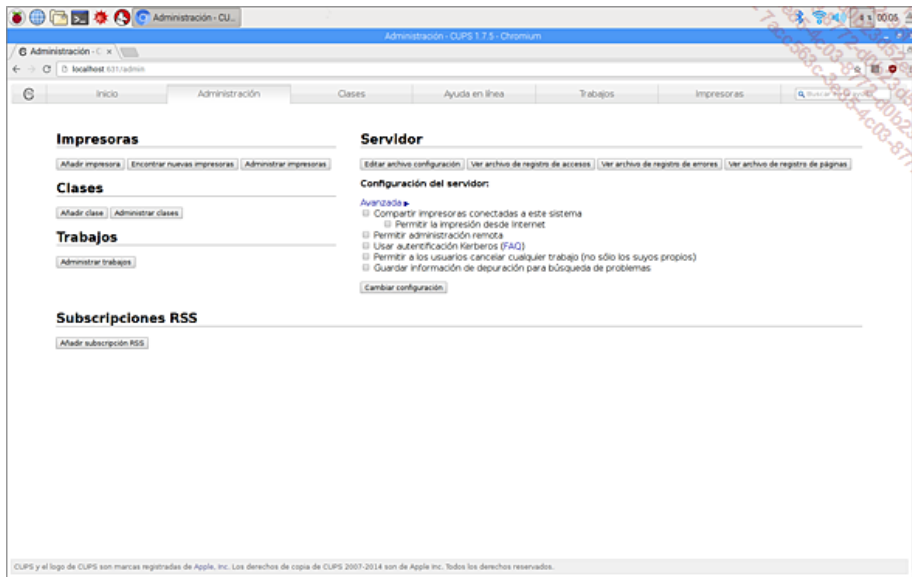
```
pi@raspberrypi:~ $ groups pi
pi : pi adm dialout cdrom sudo audio video plugdev games users
input netdev spi i2c gpio lpadmin
```

El acceso a CUPS se hace con un navegador local en la Raspberry Pi, en la dirección *localhost* a través del puerto 631. Por defecto, el acceso remoto está prohibido. La interfaz de CUPS está en español.

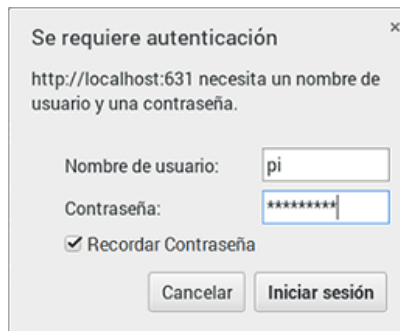


La versión de CUPS en el momento de escribir estas líneas es la 1.7.5. Esta versión podría ser diferente a la que instale CUPS, pero el uso no será fundamentalmente diferente.

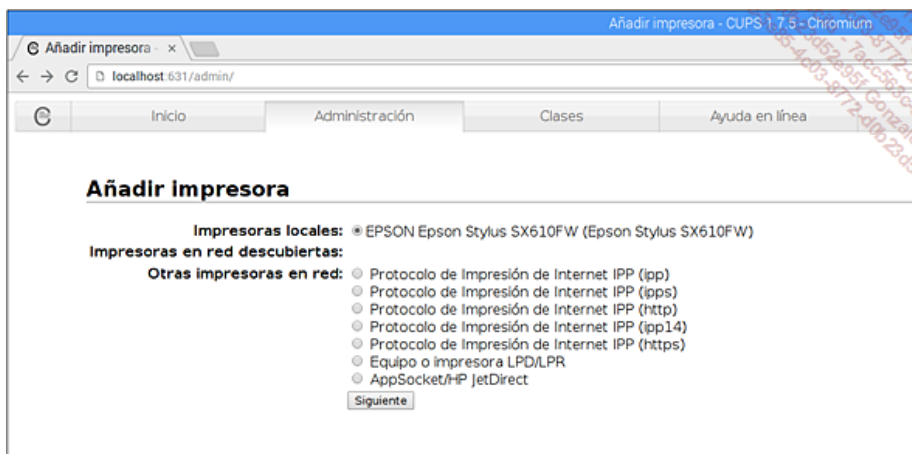
Dentro de la sección **CUPS para administradores**, seleccione la opción **Añadiendo impresoras y clases**.



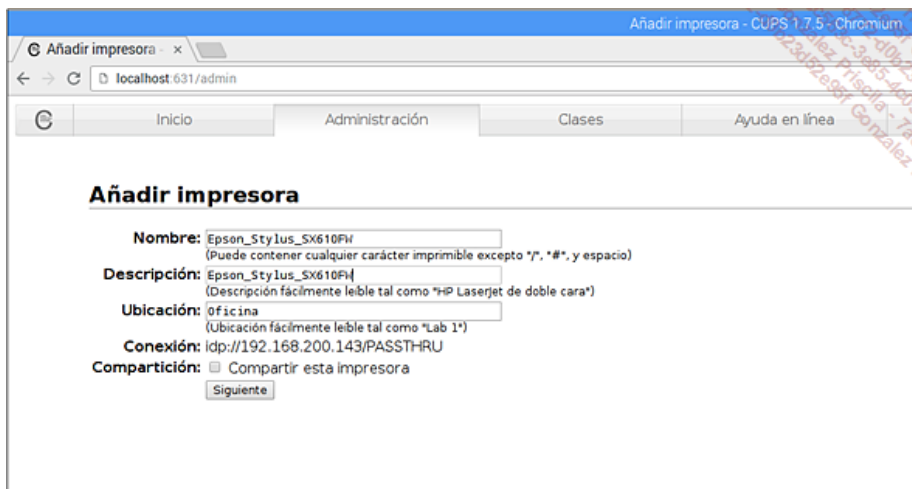
Se abre la ventana que permite agregar impresoras (imagen anterior). En **Impresoras**, pulse el botón **Añadir impresora**.



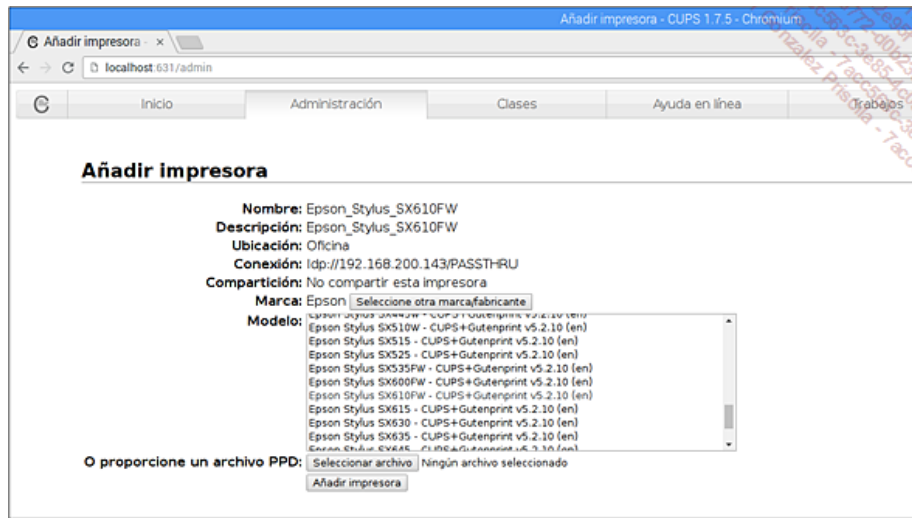
Se solicita autenticación. Introduzca el identificador y la contraseña del usuario que ha añadido anteriormente al grupo *lpadmin*. Si solo va a utilizar CUPS, seleccione **Recordar Contraseña** y a continuación valide pulsando en el botón **Aceptar**.



Seleccione la impresora que desea añadir. Si la impresora está conectada a la Raspberry Pi (en la red o en un puerto USB), se reconocerá y se mostrará. Aquí será una Epson Stylus. Adapte esta selección en función de su impresora. A continuación, será necesario indicar a CUPS cómo enviará la información a la impresora.



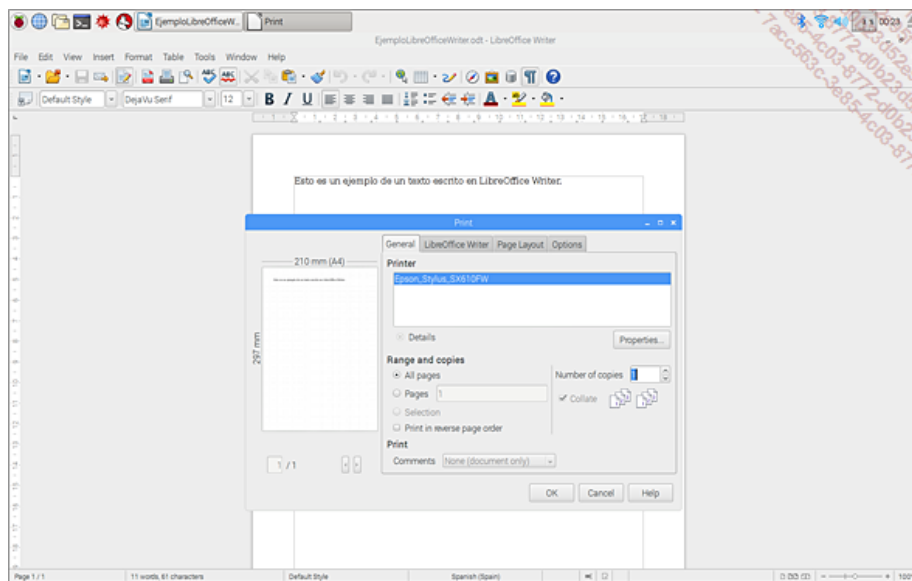
La siguiente pantalla está parcialmente rellena con la información recogida por CUPS. Agregue información acerca de la ubicación de la impresora (aquí: **Oficina**) y pulse en el botón **Siguiente**. Si desea compartir esta impresora con otros usuarios, marque la opción **Compartir esta impresora**.



La pantalla anterior permite realizar una selección en la (larga) lista de modelos disponibles de un fabricante. Seleccione el modelo exacto de su impresora o el modelo más cercano y pulse en el botón **Añadir impresora**.



A continuación podrá definir las opciones por defecto que usará su impresora. En caso de la Epson Stylus puede definir, entre otros, el formato de papel, el modo de impresión de los colores, la calidad de impresión, la resolución, etc... También puede agregar banners entre las diferentes impresiones si hay varios usuarios (menú en la parte superior de la página).



Ahora, la impresora aparece en las aplicaciones. Aquí se va a imprimir un documento abierto en LibreOffice.

Pulse en **File - Print** para abrir el menú de impresión. Si tiene varias impresoras, aparecen en la ventana **Printer**. Seleccione la impresora que desea utilizar y pulse en el botón **Properties** para configurar la impresión.

La impresión de páginas de texto es prácticamente instantánea. Los documentos con imágenes o los PDF tardan más tiempo en salir, en particular si la Raspberry Pi está ocupada haciendo otras tareas simultáneamente.

Un centro multimedia

Con cuatro núcleos funcionando a 1,2 GHz, el SoC de la Raspberry Pi 3 está provisto de una GPU que permite tratar vídeo en 1.080p. Por lo tanto, está adaptado para la realización de un centro multimedia doméstico.

El bajo consumo de la Raspberry Pi permite usar la tarjeta sin ventilación forzada. Normalmente, el ruido de la ventilación es un inconveniente en los aparatos destinados a reproducir vídeo y sonido. Con la Raspberry Pi 3, este inconveniente se elimina totalmente. Sin embargo, para un uso intensivo la utilización de un radiador adaptado permite mantener una temperatura razonable del SoC.

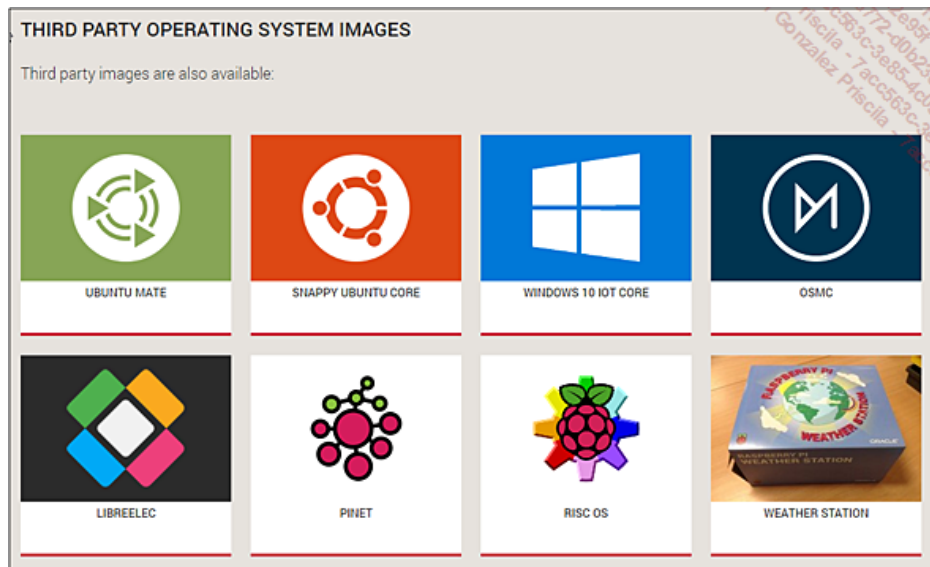
El centro multimedia XBMC se ha adaptado a la Raspberry Pi y varias distribuciones ofrecen un paquete sencillo de instalar y configurar: OpenELEC, Raspbmc o XBian.

- Las capturas de pantalla de LibreElec son muy oscuras en blanco y negro, el fondo de pantalla de esta aplicación era azul oscuro y negro. Sin embargo en una pantalla el texto se lee perfectamente.

1. Instalar LibreElec

LibreElec (*Libre Embedded Linux Entertainment Center* = Centro multimedia libre incorporado en Linux) es el elegido porque es relativamente sencillo de desplegar para un primer centro multimedia. LibreElec es una distribución ligera que integra el centro multimedia KODI.

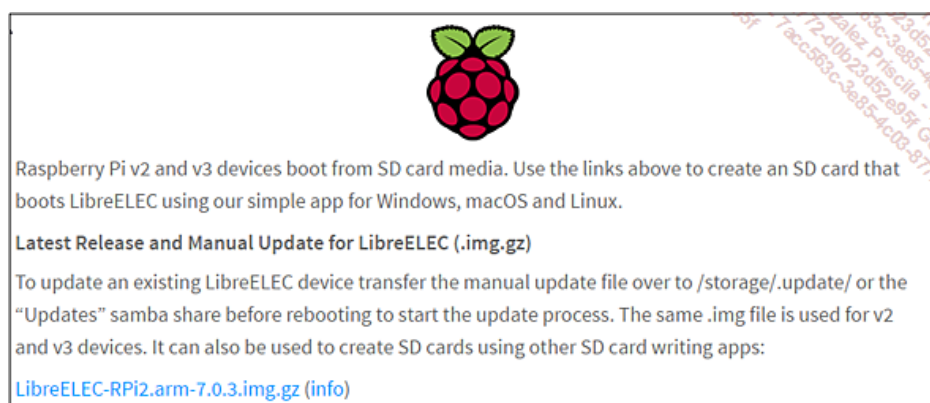
Para descargarlo, vaya a la página **Download** del sitio web raspberrypi.org.



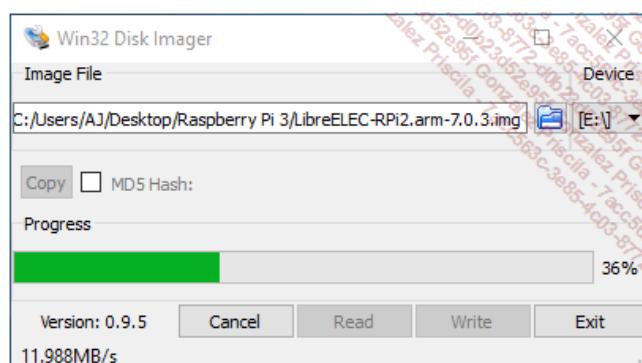
LibreElec está situado en la parte inferior izquierda en la captura de pantalla anterior, pero la página de descarga de la fundación evoluciona y los sistemas propuestos por terceros así como las ubicaciones pueden cambiar cuando instale LibreElec.

Cuando pulse en el logo LibreElec, el enlace le dirige a la página de bienvenida de LibreElec. En el menú situado en la parte superior de la página de bienvenida de LibreElec, pulse en **Download**.

Una vez en la página de descarga, busque en la página la zona siguiente.

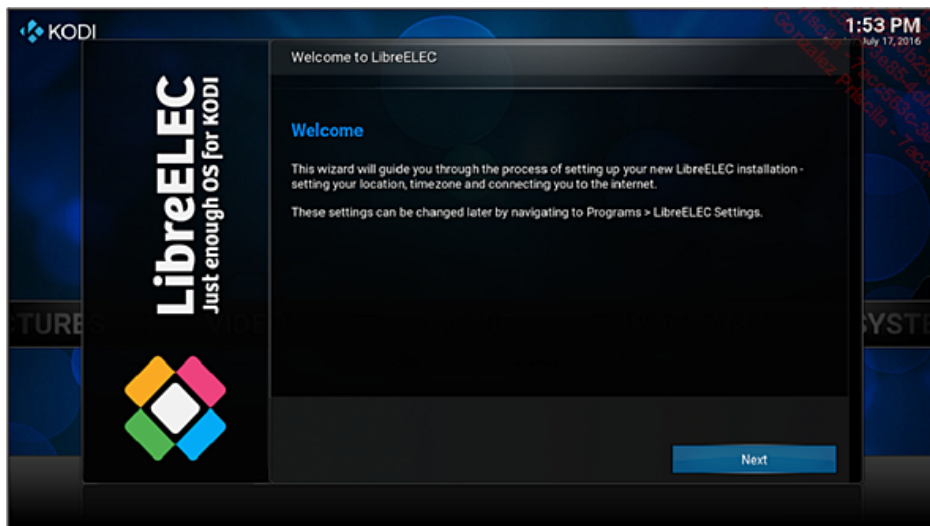


Pulse en **LibreELEC-RPi2.arm-7.0.3.img.gz (info)** para descargar la imagen en formato .gz que desea instalar en Raspberry Pi 3. Solo ocupa 116 MB. Guárdela después y descomprima el archivo para extraer el archivo .img.



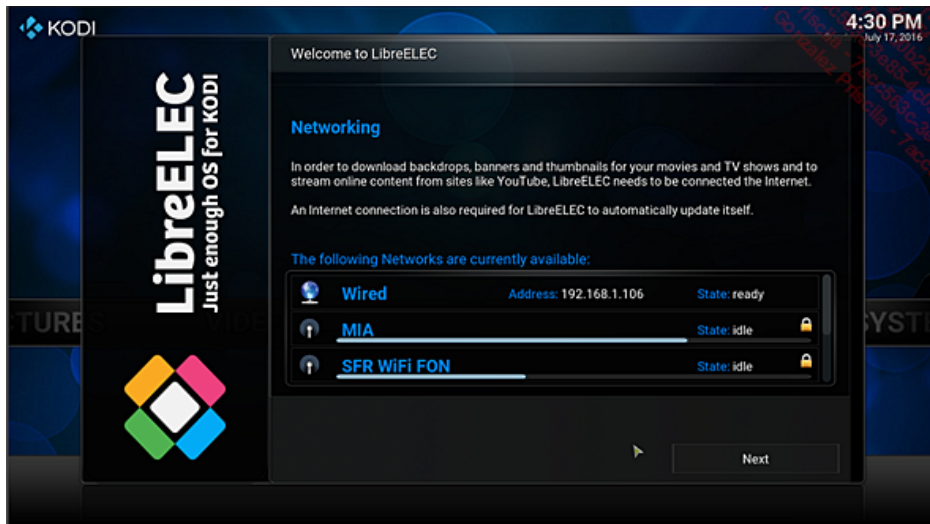
El procedimiento de instalación con Win32DiskImager (imagen anterior) es idéntico al de Raspbian. Vaya al capítulo Preparar la tarjeta micro SD para obtener más detalles.

Cuando la escritura de la imagen en la tarjeta micro SD haya terminado, inserte la tarjeta en el conector de una Raspberry Pi 3 y conecte la toma de alimentación. LibreElec arranca, muestra una pantalla de bienvenida y redimensiona las particiones, y a continuación reinicia.

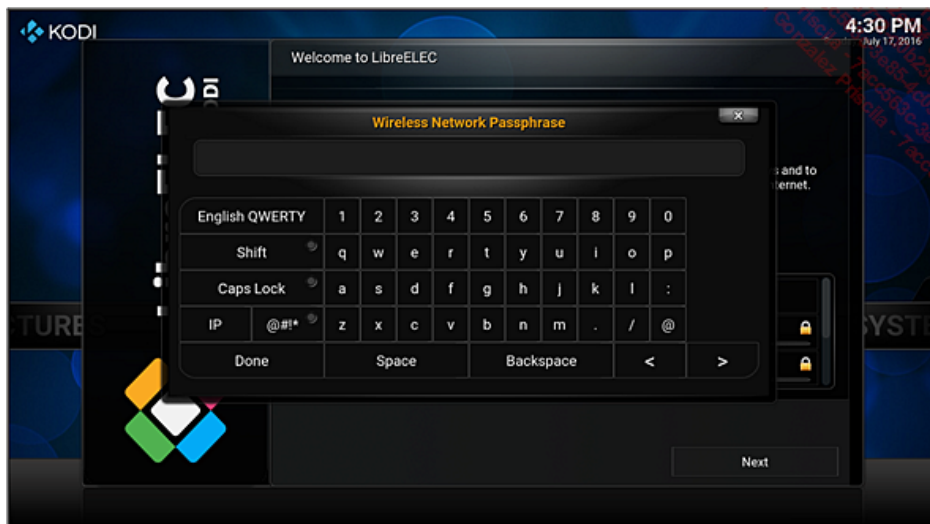


LibreElec muestra una primera pantalla de bienvenida en inglés. Pulse el botón **Next**. La pantalla siguiente le ofrece renombrar el host, que se llama LibreELEC por defecto. Este es el nombre que designará a la Raspberry Pi en la red. Modifíquelo si le parece necesario y después pulse el botón **Next**.

La siguiente etapa afecta a la conexión de red de su Raspberry Pi.

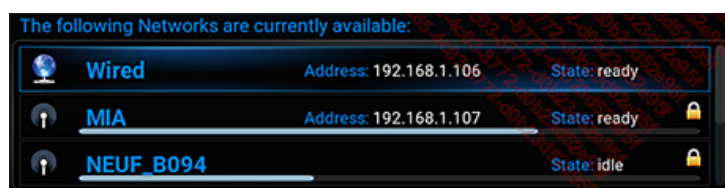


La red alámbrica (Wired) que estaba conectada a esta Raspberry Pi se ha configurado y está lista. Si desea conectar su máquina al Wi-Fi, seleccione su punto de acceso de la lista (aquí **MIA**). Después pulse en **Next** y en **Connect** en la siguiente pantalla.



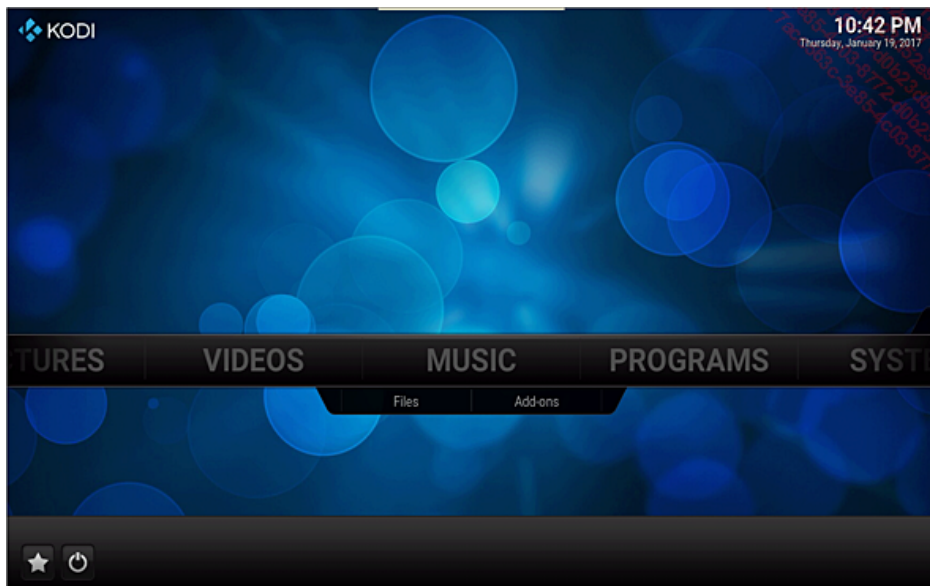
Ahora debe rellenar la **Passphrase** (la contraseña) de su punto de acceso: el teclado está en QWERTY.

Una vez comprobada la Passphrase, la Raspberry Pi se conecta al Wi-Fi en el punto de acceso y se une a la red.



Ahora LibreElec está conectado a la red. La siguiente pantalla le permite seleccionar si se desea conectar en modo remoto con SSH (utilizando PuTTY, por ejemplo: consulte el capítulo Conectarse a la Raspberry Pi en remoto). En este caso, el login es root y la contraseña libreelec.

Por defecto, está marcada la opción **Samba**. Esto permite compartir LibreElec en la red y poder agregar películas, música, etc... en la memoria de la Raspberry Pi desde un ordenador conectado a la red.



Termine de arrancar LibreElec pulsando el botón **Next**. Llegará a la pantalla de bienvenida de Kodi (imagen anterior).

2. Configuración

Después de la configuración básica, ahora es necesario configurar la distribución para adaptarla a sus necesidades. La primera acción a realizar es pasar Kodi a español.

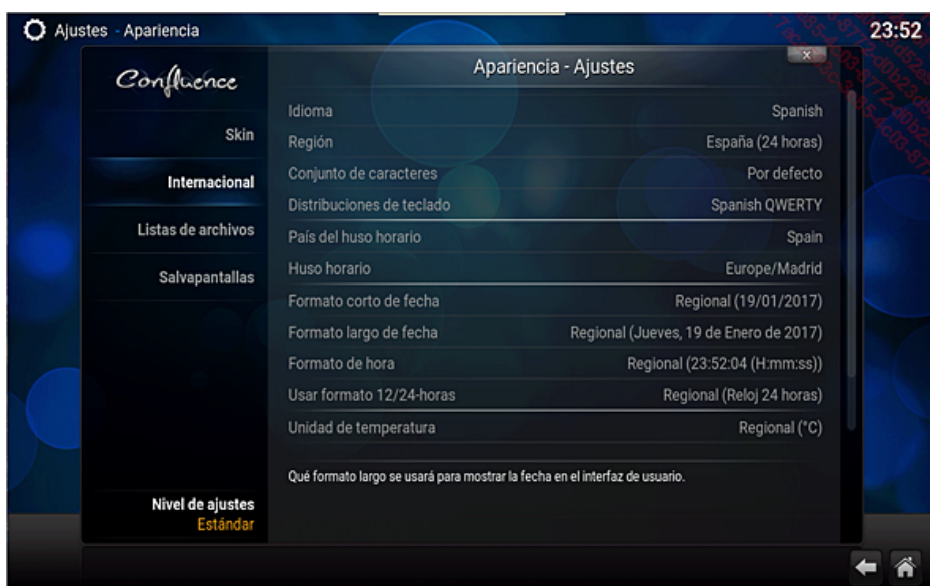
Vaya al menú **System - Settings - Appearance - International** (siguiente imagen).



Modifique los parámetros de esta pantalla para 'españolizar' la distribución:

- Lenguaje (Idioma): Spanish
- Region (Región): España (24 horas)
- Keyboard layouts (Distribuciones de teclado): Spanish QWERTY
- Timezone country (País del huso horario): Spain

Pulse en la zona a modificar y seleccione la opción adecuada de la lista desplegable.



Con esta configuración, los menús de LibreElec estarán en español y la información se mostrará en un formato adaptado a las costumbres locales.

Observe en la parte inferior izquierda de la captura de pantalla anterior la posibilidad de ajustar el **Nivel de ajustes** en función de su dominio

del tema (Básico, Estándar, Avanzado, Experto).

Los dos iconos en la parte inferior derecha de la pantalla le permiten volver a la pantalla anterior o al menú principal.

Para terminar de configurar la distribución, vaya al menú **System - LibreElec**.



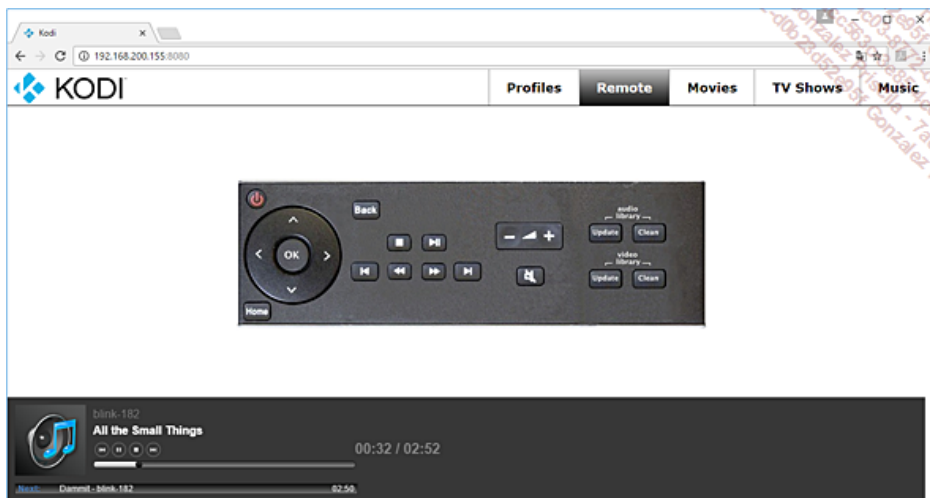
Ajuste la disposición del teclado en **es-latín9**, configure las actualizaciones en modo **auto**, con lo que no tendrá que preocuparse de hacer una búsqueda manual a partir de este momento.

Si desea controlar su centro multimedia desde otro ordenador/tableta, es posible partir de un navegador web. Vaya al menú **Sistema - Ajustes de LibreELEC - Servicios - Servidor Web** y rellene los campos de la siguiente manera.



El acceso remoto por http está permitido desde un navegador, en la dirección de la Raspberry Pi a través del puerto **8080**: **http://192.168.1.107:8080** (cambie la dirección por la de su Raspberry Pi).

Durante la conexión debe guardar un login (por defecto kodi) y una contraseña (por simplicidad, he conservado kodi).



Ahora tiene acceso a un pantalla que tiene un mando a distancia con el que puede controlar su centro multimedia (en la pantalla anterior, lectura de música).

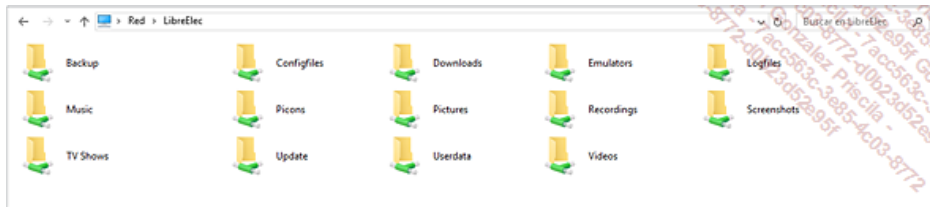
3. Utilizar el centro multimedia

Su centro multimedia ahora está listo para usarse. Para mostrar las posibilidades del software, los siguientes párrafos van a explicar cómo leer música desde la tarjeta micro SD de la Raspberry Pi 3 y después cómo leer una película desde un recurso compartido remoto (PC, Box, NAS...).

a. Escuchar la música en local

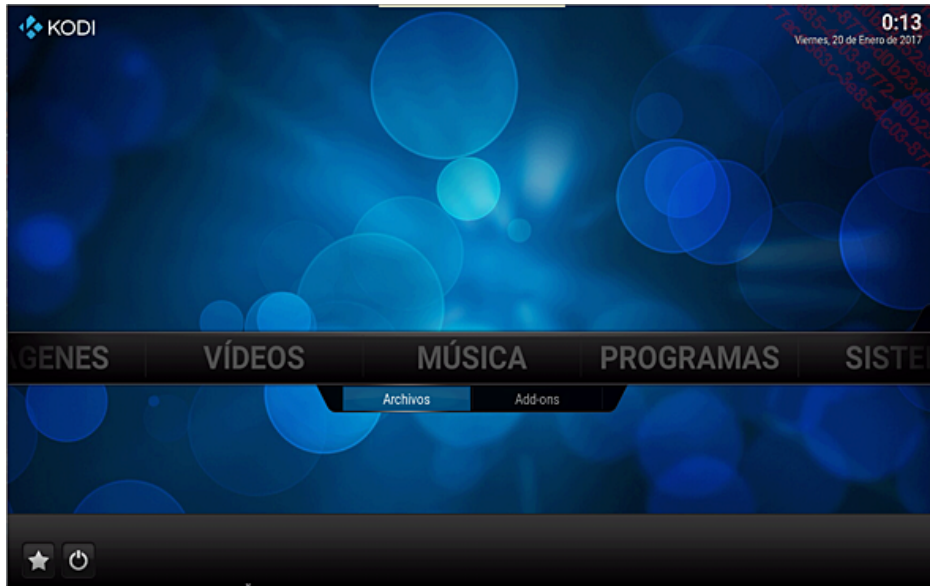
En primer lugar es necesario copiar la música disponible en el disco duro de su PC a la carpeta **Music**, accesible en el recurso compartido LibreElec.

En el Explorador de archivos de su PC, pulse en la sección **Red**. LibreElec debería aparecer entre el resto de recursos compartidos disponibles (impresoras, carpetas, etc...). Si no está, abra el menú de Windows y pulse en **Ejecutar**. En la zona de texto introduzca `\\LibreElec` o `\\direcciónIP_de_la_Raspberry_Pi`. Aparecen las carpetas compartidas por LibreElec.

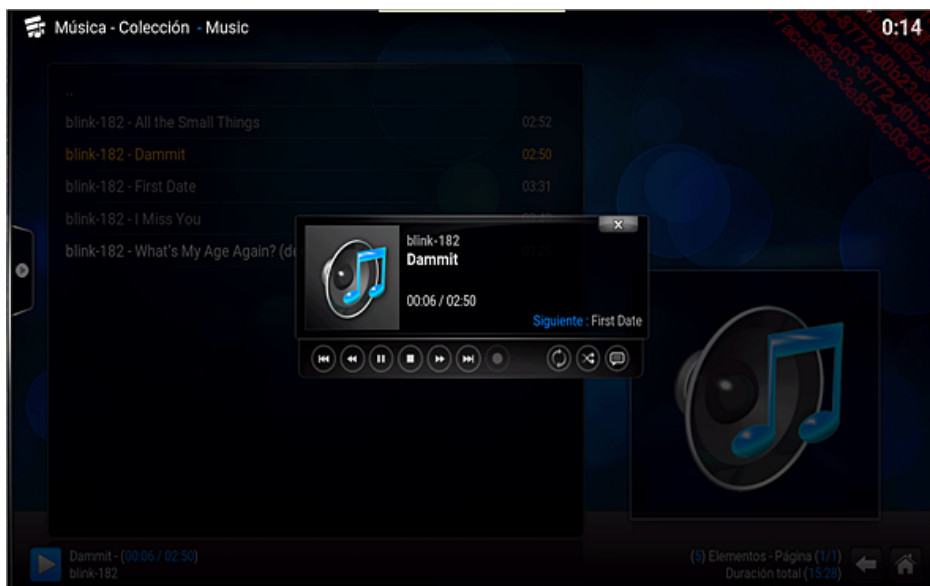


La carpeta Music está a la derecha de esta captura de pantalla. En su PC, seleccione los títulos mp3 que desee escuchar, haga clic derecho en estos archivos y seleccione **Copiar**. Abra la carpeta compartida **Music** y copie allí los títulos seleccionados.

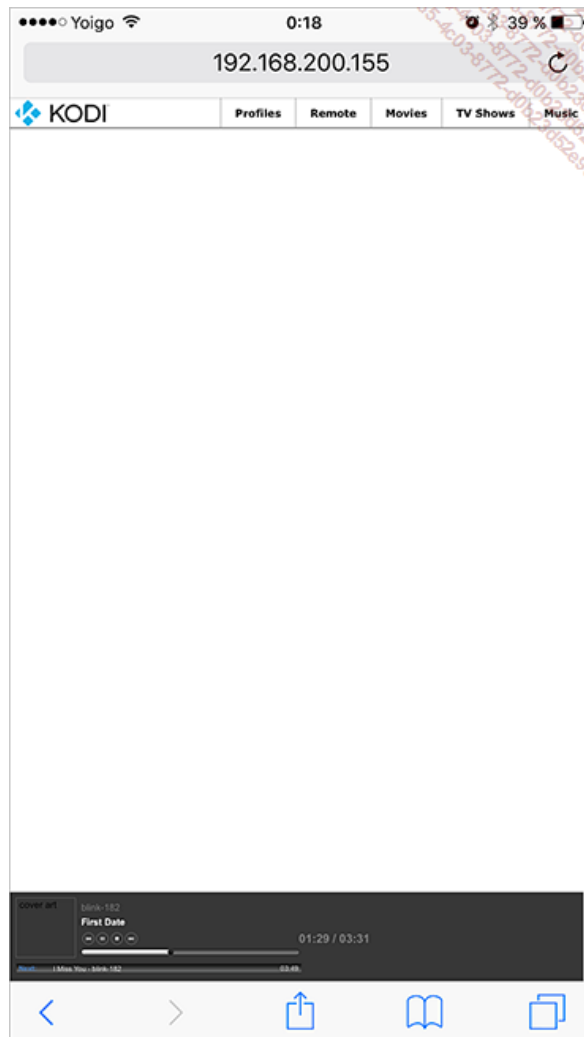
En LibreElec, vuelva al menú principal y seleccione **MÚSICA - Archivos**, como en la siguiente captura de pantalla.



En la pantalla siguiente, pulse en la carpeta **Music**. Esta se abre y muestra la lista de canciones disponibles. Haciendo doble clic en una canción, comenzará a sonar.



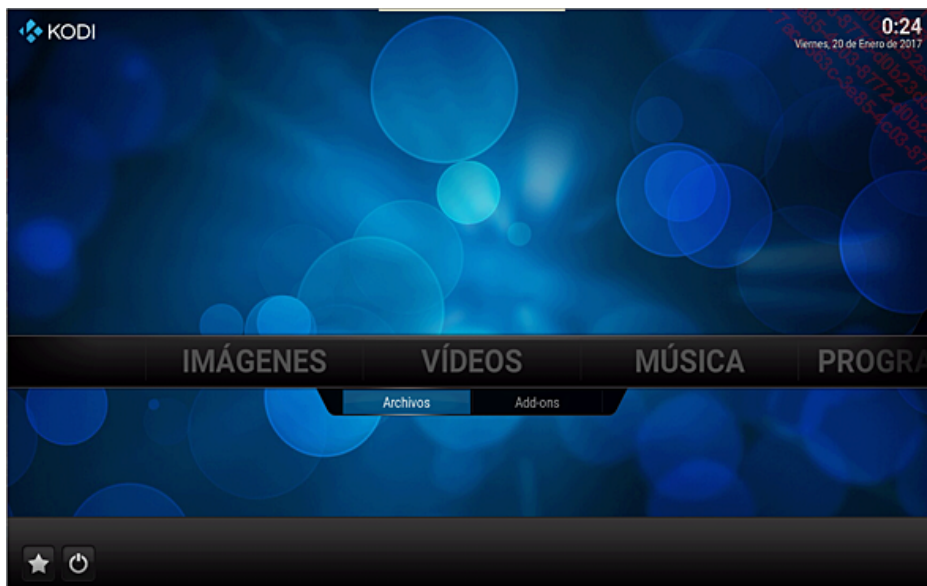
El control de lectura está disponible en la pantalla de LibreElec como antes, pero también en modo remoto en un PC si el sitio web se ha configurado, o incluso en un smartphone como se muestra a continuación.



Esta captura de pantalla, realizada en un Samsung Galaxy S4, muestra que es perfectamente viable gestionar la reproducción de manera remota desde un ordenador, un smartphone o una tableta. Si la Raspberry Pi está conectada a un televisor HDMI compatible con CEC, el control remoto del televisor también se podrá utilizar, las órdenes se enviarán a LibreElec a través del cable HDMI (ver más abajo, en el párrafo Utilizar el mando a distancia).

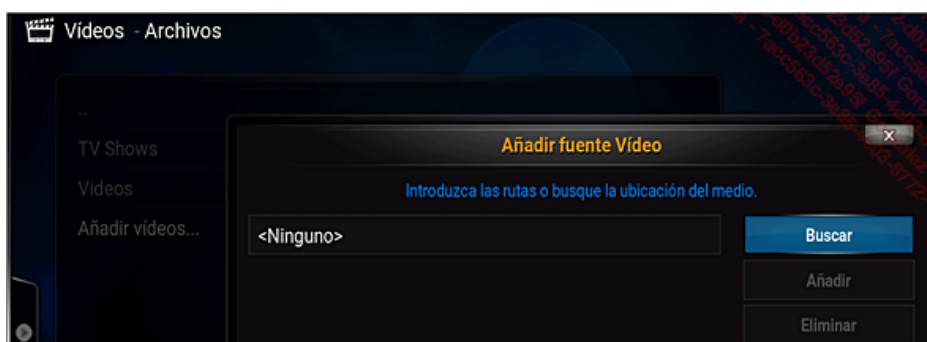
b. Ver una película en modo remoto

El almacenamiento de las películas generalmente no se hace en la tarjeta micro SD de la Raspberry Pi. Se pueden almacenar en un PC familiar, en el disco duro de una box Internet o en un disco duro de copia de seguridad compartido.



Para la siguiente demostración, las películas están guardadas en un PC con Windows 7, en una carpeta compartida con el nombre de recurso compartido *FILMS*. El PC y la Raspberry Pi están conectados en una red por cable. Hay que indicar a LibreElec dónde se encuentran las películas que desea ver, añadiendo una fuente de vídeo.

En el menú principal, pulse en **Vídeo - Archivos** y después en la pantalla siguiente **Añadir Vídeos**. En la ventana que se abre puede indicar la ruta de la fuente de los vídeos, si la conoce. En caso contrario, pulse en el botón **Buscar** (siguiente captura de pantalla).

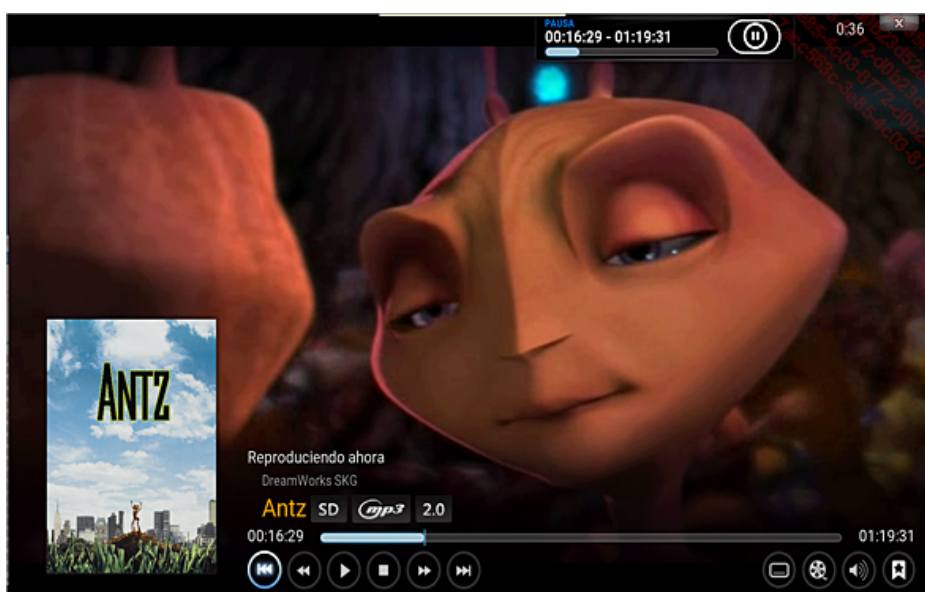


Una lista le permite seleccionar las posibles fuentes. Pulse en **Examinar** y después en **Red Windows (SMB)**. Examine la red hasta encontrar el recurso compartido al que desea conectarse. Selecciónelo y pulse en **OK**.

Debe registrar el nombre de usuario y la contraseña para conectarse en la máquina remota. Si tiene que usar varios recursos compartidos, repita la operación tantas veces como sea necesario y después valide la lista de las fuentes de vídeo pulsando en **OK**.



Una vez agregados el o los recursos compartidos, puede seleccionar la carpeta de la lista en **Videos - Archivos**. Se muestra la lista de las películas disponibles. Pulse en la película que desea ver.



La lectura empieza (imagen anterior). La calidad es excelente, sin ningún salto de imagen, a pesar de un vídeo de 1080p.

El uso de LibreElec es muy intuitivo y la reproducción de imágenes, música o vídeos ahora está al alcance de su mando a distancia.

4. Utilizar el mando a distancia

La Raspberry Pi es compatible con HDMI CEC (*Consumer Electronics Control* = controlado por el usuario). La mayor parte de las marcas que producen aparatos HDMI garantizan la compatibilidad con esta norma. El nombre comercial dado a la norma varía según los fabricantes: Anynet+ (Samsung), Aquos Link (Sharp), Regza Link (Toshiba), Bravia Theater Sync (Sony), Viera Link (Panasonic), etc.

La norma CEC permite a los diferentes aparatos de un home cinema dialogar entre sí. Por tanto, el mando a distancia de su televisor podrá sustituir al teclado y al ratón de su Raspberry Pi. Muchos usuarios conectan su tarjeta únicamente en HDMI, sin teclado ni ratón.

En este caso, el movimiento por la pantalla del centro multimedia se controla con las teclas del mando a distancia que tienen flecha. Las teclas de lectura, pausa, parada, avance y retroceso rápido se utilizan para la lectura de vídeos.

Son muchas las posibilidades de configuración y ajuste de LibreElec, pero la configuración básica permite usarlo de manera sencilla. Recorra los diferentes menús disponibles para descubrir todas las opciones.

Un servidor web

Hoy en día, cualquier persona sube a la web la información que desea compartir. Llega el momento de elegir el almacenamiento. Almacenamiento compartido, servidor dedicado en un datacenter, servidor instalado en un antiguo PC, etc. Las opciones son amplias. La llegada de la Raspberry Pi ofrece una solución para los sitios web que alcanzan un número limitado de visitas. El bajo consumo de la Raspberry Pi permite dejarla conectada permanentemente con un coste mínimo. La Raspberry Pi consume 700 mA bajo 5 voltios, es decir 3,5 W. Durante un año, el consumo es de $3,5 \text{ W} \times 24 \text{ h} \times 365 \text{ d} = 30.660 \text{ Wh}$, es decir, alrededor de 31 kWh. El kWh cuesta una media de 15 céntimos de euro y el coste de almacenamiento de su servidor web sería inferior a 5 euros/año, aproximadamente.

1. Instalar el servidor

Una gran parte de los servidores web instalados (50 %) corren en Apache (datos de Web Server Survey - Marzo de 2016). Este servidor puede funcionar en la Raspberry Pi pero tiene un consumo importante de recursos. Para adaptar el servidor a la Raspberry Pi es preferible una solución ligera.

Los servidores web como Nginx o lighttpd consumen muchos menos recursos, proporcionando un servicio eficaz.

a. Objetivo de la instalación

El objetivo es publicar en Internet un blog alojado en una Raspberry Pi. Se adquirió el nombre de dominio *raspi.blog.es* en el registro de dominios BookMyName. La solución que se aplica en este ejemplo es la instalación en Raspbian de un servidor lighttpd. Esta instalación básica funciona perfectamente con páginas estáticas. En segundo lugar, se añade PHP y MariaDB para permitir instalar WordPress y poner en marcha el blog. Será necesario aceptar una ralentización en la carga de las páginas con esta configuración más pesada.

Por último, el router a la que la Raspberry Pi está conectada se configurará para garantizar la publicación del sitio web en Internet. La dirección IP fija del router se debe añadir en la configuración DNS de BookMyName, para que los internautas que escriban <https://www.raspberrypi.org/blog/> terminen en la Raspberry Pi. Consulte la documentación en línea del registro para encontrar los procedimientos de configuración del DNS.

b. Instalación de Raspbian Lite

El sistema Raspbian Lite se ha elegido para este servidor. La interfaz gráfica no tiene utilidad en este caso y Raspbian se instala en modo texto únicamente en una tarjeta micro SD dedicada para la realización del blog.

La primera operación que hay que hacer es actualizar el sistema.

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

Ejecute *raspi-config* y modifique los argumentos correspondientes (nombre del host, teclado, huso horario, etc...). Inicie de nuevo la Raspberry Pi hasta el final de la configuración.

c. Modificación del nombre del host

Esta modificación no es obligatoria, pero es interesante llamar a cada Raspberry Pi que acceda a la red de manera personalizada. Esto permite identificar rápidamente las máquinas en caso de consulta del entorno de red o de los clientes que tengan un contrato sobre el DHCP del Box/del router.

2. Instalación de lighttpd

La instalación de *lighttpd* se hace en modo consola:

```
pi@raspi.blog ~ $ sudo apt-get install lighttpd
```

Esta instalación es rápida. El servidor web está operativo muy rápidamente. A este nivel, puede conectarse desde un navegador web a la dirección de la Raspberry Pi. Se muestra la página de bienvenida de *lighttpd*. La prueba se volverá a hacer después de la modificación de los permisos de acceso al directorio del sitio web.

a. Modificación de los permisos sobre el sitio web

Lighttpd accede al sitio web con el usuario *www-data*. Para facilitar este acceso y permitir al usuario *pi* realizar modificaciones en el sitio web, hay que proceder a realizar algunas modificaciones:

```
pi@raspi.blog ~ $ sudo chown -R www-data:www-data /var/www/html
pi@raspi.blog ~ $ sudo usermod -a -G www-data pi
pi@raspi.blog ~ $ groups pi
pi : pi adm dialout cdrom sudo audio www-data video plugdev games
users netdev input lpadmin indiecity spiuser spi gpio
pi@raspi.blog ~ $ sudo chmod -R g+ws /var/www/html
```

El comando *chown* hace que el usuario y el grupo *www-data* sean propietarios del directorio */var/www/html*, así como de los sub-directorios si existen (-R).

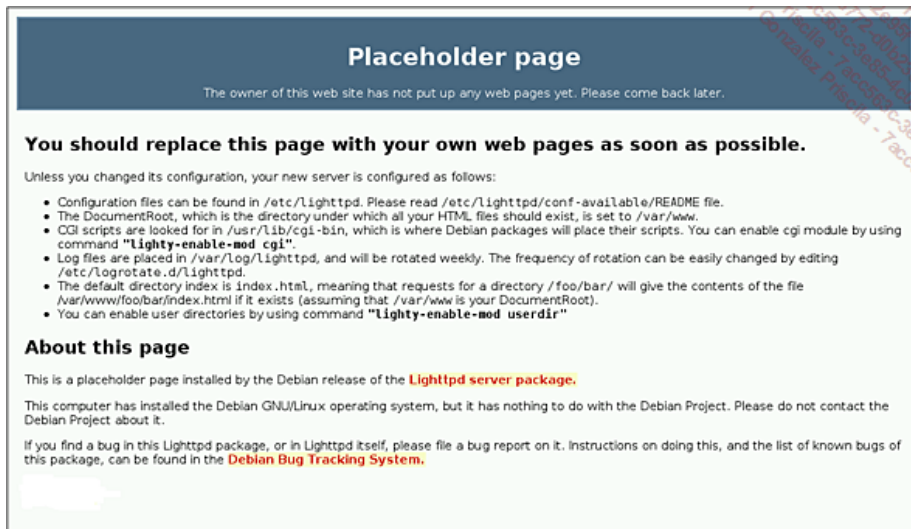
El comando *usermod* añade al usuario *pi* al grupo *www-data*, lo que confirma la visualización de los grupos a los que el usuario *pi* pertenece (comando *groups pi*).

Para terminar, el comando *chmod -R g+ws /var/www/html* establece el bit Setgid. Gracias a este bit Setgid, todos los archivos y directorios creados en */var/www/html* por cualquier usuario pertenecerán obligatoriamente al grupo *www-data*. Esto garantiza el acceso a los archivos y directorios por *lighttpd*, así como por el usuario *pi*.

➔ Para que estos nuevos permisos se puedan tener en cuenta, desconéctese (cierre la sesión) y conéctese de nuevo.

b. Verificación del servidor web

Conéctese con un navegador web desde una máquina en Linux, Windows o Android que esté en la misma red que la Raspberry Pi. En la barra de dirección, escriba la dirección de la Raspberry Pi. El navegador muestra la página por defecto de *lighttpd*.



La página web mostrada está en el directorio `/var/www/html` y se llama `index.lighttpd.html`. Esta es la página de bienvenida por defecto.

Basta con crear una página `index.html` en el directorio `/var/www/html` para que la página de lighttpd se sustituya por esta última.

- Cree la página `index.html` con el editor de texto nano. Introduzca el contenido siguiente:

```
<html>
<body>
<h1>Página de Test</h1>
</body>
</html>
```

Refresque y compruebe con su navegador que se muestra la página `index.html`. Debe obtener el siguiente resultado:



Si estas comprobaciones son correctas, puede poner en marcha un sitio web con páginas estáticas. Es posible usar LibreOffice para crear páginas web. Las imágenes se guardarán en el directorio `/var/www/html` para evitar errores de visualización.

- Evite dar a sus páginas web nombres que contengan mayúsculas, caracteres acentuados, espacios u otros caracteres especiales.

3. Instalación de PHP

Aunque un sitio formado por páginas estáticas es suficiente para empezar en Internet, pronto se convierte en algo pesado de mantener. Establecer un lenguaje de programación ejecutado en el servidor facilita el desarrollo de sitios dinámicos.

El lenguaje PHP presente en las páginas web se ejecuta en el servidor y el resultado del programa PHP se envía al navegador del internauta.

- PHP 7 es la última versión de PHP, más rápida y con menos consumo de recursos que la versión PHP 5. En el momento de escribir estas líneas, esta versión no está disponible en Raspbian.

a. Instalar PHP 5 en la Raspberry Pi

```
pi@raspbilog :/var/www/html $ sudo apt-get install php5-cgi php5-mysql
Lectura de las listas de paquetes... Terminado
Construcción del árbol de dependencias
.../...
Se instalarán los siguientes paquetes nuevos:
 libmysqlclient18 libonig2 libperl4-corelibs-perl libqdbm14 lsof
 mysql-common
 php5-cgi php5-cli php5-common php5-json php5-mysql php5-readline
0 actualizados, 8 instalados de nuevo, 0 eliminados
y 0 no actualizados.
Es necesario 5 803 KB de espacio para los archivos.
Después de esta operación se utilizarán 30,0 MB de espacio adicional
en disco.
¿Desea continuar? [S/N] s
```

b. Indicar a lighttpd que debe usar PHP

```
pi@raspbilog /var/www/html $ sudo lighttpd-enable-mod fastcgi
Enabling fastcgi: ok
Run /etc/init.d/lighttpd force-reload to enable changes
pi@raspbilog /var/www/html $ sudo lighttpd-enable-mod fastcgi-php
Enabling fastcgi-php: ok
Run /etc/init.d/lighttpd force-reload to enable changes
pi@raspbilog /var/www/html $ sudo service lighttpd force-reload
```

La instalación de los módulos `fastcgi` establece una pasarela que permite a lighttpd comunicarse con PHP. CGI significa *Comon Gateway Interface*, es decir, interfaz de pasarela común.

Después de la instalación de PHP, el refresco de la página `index.html` no muestra ninguna modificación en esta página. Para que PHP funcione, la página debe tener como extensión `.php`.

c. Renombrar la página `index.html` por `index.php`

```
pi@raspiblog /var/www $ sudo mv index.html index.php
```

El refresco de la página sigue sin cambiar nada: la página PHP se ejecuta, pero solo contiene el código HTML anterior.

d. Comprobación de la instalación de PHP

Para probar el correcto funcionamiento de PHP, es necesario añadir código PHP en la página. La función `phpinfo()` muestra mucha información de la instalación de PHP. En general, es la primera función que se ejecuta en PHP. Obligatoriamente hay que eliminar el archivo que contiene `phpinfo()` cuando el sitio esté en línea. La cantidad de información proporcionada por esta función facilita los ataques contra su sitio Web.

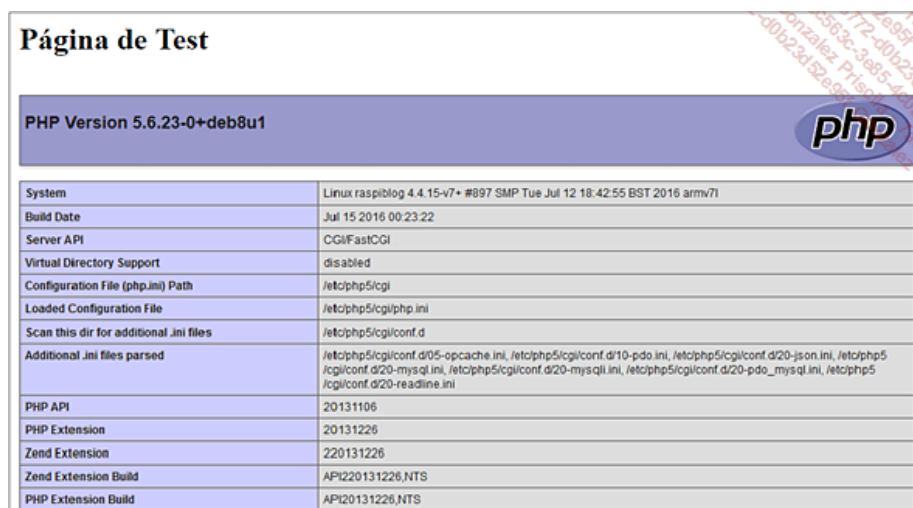
Modifique la página `index.php` con el editor de texto `nano`. Añada la línea que contiene `phpinfo()`:

```
<html>
<body>
<h1>Página de Test </h1>
<br> <?php phpinfo();?>
</body>
</html>
```

`<?php` abre la etiqueta PHP e indica al servidor que la línea contiene código que se debe interpretar con PHP.

- `phpinfo()` es el código PHP a ejecutar.
- `;` marca el final de la línea de código. Este punto y coma es obligatorio.
- `?>` cierra la etiqueta PHP e indica al servidor que el código que viene a continuación es de nuevo HTML.

Refresque y compruebe con su navegador que la página `index.php` muestra ahora el título **Página de Test**, seguido de la información proporcionada por `phpinfo()`.



Página de Test	
PHP Version 5.6.23-0+deb8u1	
System	Linux raspiblog 4.4.15-v7+ #897 SMP Tue Jul 12 18:42:55 BST 2016 armv7l
Build Date	Jul 15 2016 00:23:22
Server API	CGIFastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/cgi
Loaded Configuration File	/etc/php5/cgi/php.ini
Scan this dir for additional .ini files	/etc/php5/cgi/conf.d
Additional .ini files parsed	/etc/php5/cgi/conf.d/05-opcache.ini, /etc/php5/cgi/conf.d/10-pdo.ini, /etc/php5/cgi/conf.d/20-json.ini, /etc/php5/cgi/conf.d/20-mysql.ini, /etc/php5/cgi/conf.d/20-mysqli.ini, /etc/php5/cgi/conf.d/20-pdo_mysql.ini, /etc/php5/cgi/conf.d/20-readline.ini
PHP API	20131106
PHP Extension	20131226
Zend Extension	220131226
Zend Extension Build	API220131226,NTS
PHP Extension Build	API20131226,NTS

Solo con añadir la función `phpinfo()`, se ha generado en el servidor una página web de varios centenares de líneas. Esto es una típica página web dinámica: un código de pocas líneas que genera la visualización automática de centenares de líneas de información.

4. Instalación de MariaDB

Después de instalar PHP, la siguiente etapa es la instalación de una base de datos. La empresa creadora y propietaria de MySQL fue comprada por Sun Microsystems en 2008. En 2009 la compra de Sun Microsystems por Oracle ha generado cierta polémica, ya que Oracle desarrolla su propia base de datos: Oracle Database. Michael Widenius, creador de MySQL creó MariaDB para poder continuar el desarrollo del proyecto en open source. Las pruebas muestran que MariaDB no tiene nada que envidiar a MySQL en la mayor parte de los dominios de uso. En Raspbian, durante mucho tiempo solo ha estado disponible la base de datos MySQL, pero MariaDB ahora está integrada en la distribución. Las dos bases de datos son totalmente compatibles y el cambio de una por otra se hace sin modificaciones particulares. En el futuro, será posible instalar MariaDB en Raspbian.

a. Instalar la base de datos MariaDB MySQL en la Raspberry Pi

```
pi@raspiblog:/var/www/html $ sudo apt-get install mariadb-server
Lectura de las listas de paquetes... Terminado
Construcción del árbol de dependencias
.../...
Se instalarán los siguientes paquetes nuevos:
  mariadb-client-10.0 mariadb-client-core-10.0 mariadb-common
  mariadb-server
  mariadb-server-10.0 mariadb-server-core-10.0
0 actualizados, 11 instalados de nuevo, 0 eliminados y 0 no
actualizados.
Es necesario 10,5 MB de espacio para los archivos.
Después esta operación, se utilizarán 116 MB de espacio en disco
adicionales.
¿Desea continuar? [S/N]
```

Durante la instalación, es necesario introducir una contraseña que se utilizará para el acceso a la base de datos.

- Si la Raspberry Pi está conectada realmente a Internet, seleccione una contraseña robusta formada, al menos, por 8 caracteres y que contenga una mezcla de minúsculas, mayúsculas, cifras y caracteres especiales.

b. Seguridad de MariaDB

El uso de MariaDB, en particular en una Raspberry Pi conectada a Internet, obliga a dar seguridad de manera rigurosa a la base de datos. El comando `mysql_secure_installation` realiza esta operación. El procedimiento (en inglés) funciona en forma de pregunta-respuesta, para guiarnos con una instalación segura.

```
pi@raspiblog:/var/www/html $ mysql_secure_installation
```

```
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED
FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

In order to log into MariaDB to secure it, we'll need the current password for the root user. If you've just installed MariaDB, and you haven't set the root password yet, the password will be blank, so you should just press enter here.

```
Enter current password for root (enter for none):
```

Para poder ayudarle a configurar MariaDB, a continuación se muestra la traducción de las preguntas en español:

Set root password? [Y/n] = ¿Configurar una contraseña para root?

La protección de la base de datos empieza por el uso de una contraseña robusta. Evite elegir contraseñas que pudieran figurar en un diccionario utilizado para forzar el acceso: nombres comunes, apellidos, diminutivos, años de nacimiento, etc...). Use una sucesión de letras (mayúsculas y minúsculas), signos aleatorios.

Remove anonymous users? [Y/n] = ¿Eliminar usuarios anónimos?

Por defecto, MariaDB tiene un usuario anónimo. Esto permite a cualquiera conectarse a la base de datos sin tener una cuenta personal. Esto está destinado a facilitar las pruebas, pero se debe evitar si la Raspberry Pi está conectada a Internet.

Disallow root login remotely? [Y/n] = ¿Prohibir el acceso remoto a root?

Normalmente, el administrador *root* solo debe poder conectarse en la máquina en local. Esto prohíbe a un usuario conectarse en remoto, como *root*, incluso si conoce la contraseña.

Remove test database and access to it? [Y/n] = ¿Eliminar la base de datos de prueba?

Por defecto, la base de datos MariaDB integra una base de datos llamada 'test', a la que puede acceder cualquiera. Está prevista para un entorno de pruebas y no se debe conservar para un uso normal.

➤ En algunas distribuciones, la base de datos 'test' no existe y puede que se le avise mediante un mensaje de error, lo cual no afecta en nada a la configuración de MariaDB.

Reload privilege tables now? [Y/n] = ¿Recargar la tabla de permisos?

Al final de esta operación, la tabla de permisos debe recargarse para tener en cuenta las modificaciones realizadas anteriormente. Cuando la tabla se ha recargado, un mensaje le advierte de que la operación ha terminado y que su base de datos MariaDB es segura.

```
All done! If you've completed all of the above steps,
your MariaDB installation should now be secure.
Thanks for using MariaDB!
```

c. Probar la instalación de MariaDB

La etapa anterior ha mostrado que MariaDB funciona. Pero después de asignar la seguridad, es interesante probar el funcionamiento de MariaDB.

Indique la contraseña que ha definido para el acceso a MariaDB, el prompt `MariaDB [(none)]>` aparece después de algunas líneas de información. Para salir de MariaDB, introduzca el comando `exit`; y después [Intro] para volver a la línea de comando.

➤ MariaDB y MySQL eran compatibles, los comandos son idénticos. Esto permite a las aplicaciones diseñadas para funcionar con MySQL migrar a MariaDB sin modificación.

```
pi@raspbilog:/var/www/html $ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 51
Server version: 10.0.25-MariaDB-0+deb8ul (Raspbian)

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current
input statement.

MariaDB [(none)]> help

General information about MariaDB can be found at
http://mariadb.org

List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?          (\?) Synonym for 'help'.
clear      (\c) Clear the current input statement.
connect    (\r) Reconnect to the server. Optional arguments are db
and host.
.../...
warnings  (\W) Show warnings after every statement.
nowarning (\w) Don't show warnings after every statement.

For server side help, type 'help contents'

MariaDB [(none)]> exit
```

Esta conexión a la base de datos y la ejecución del comando `help` son una muestra de que la base de datos funciona correctamente.

d. Creación de la base de datos para WordPress

El CMS (*Content Management System* = sistema de gestión de contenido) WordPress se basa en el uso de una base de datos. En esta base de datos se han registrado los artículos y sus comentarios, los enlaces a las imágenes, los usuarios, etc... Para permitir la instalación de WordPress, inicialmente es necesario crear una base de datos que le permita trabajar.

La creación de una base de datos necesita ejecutar MariaDB como usuario *root*. Introduzca la contraseña que haya utilizado durante la instalación de MariaDB. La base de datos creada para WordPress se llama `raspbilog`. El comando `CREATE DATABASE raspbilog;` realiza esta operación. `USE raspbilog` indica a MariaDB que debe usar esta base de datos. MariaDB indica el éxito de la operación, mostrando el nombre de la base de datos entre paréntesis en el prompt. `SELECT DATABASE();` permite mostrar la información relativa a la base de datos que se está usando.

- El punto y coma con el que termina cada comando, indica a MariaDB que el comando ha terminado y que puede ejecutarlo. Ante todo, no hay que omitirlo.

```
pi@raspiblog:/var/www/html $ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 52
Server version: 10.0.25-MariaDB-0+deb8u1 (Raspbian)

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current
input statement.

MariaDB [(none)]> CREATE DATABASE raspiblog;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> USE raspiblog;
Database changed
MariaDB [raspiblog]> SELECT DATABASE();
+-----+
| DATABASE() |
+-----+
| raspiblog  |
+-----+
1 row in set (0.00 sec)

MariaDB [raspiblog]>
```

e. Asignación de los permisos en la base de datos

Ahora hay que asignar al usuario de la base de datos (*root*) los permisos que le permitirán acceder a todas las tablas de la base de datos. Los permisos se dan en raspiblog.* para garantizar este acceso.

- El signo * es un carácter llamado comodín. Sustituye a todos los caracteres. De esta manera, el usuario tendrá acceso a todas las tablas de la base de datos.

Sustituya contraseña por la contraseña seleccionada durante la instalación de MariaDB.

```
MariaDB [raspiblog]> GRANT ALL PRIVILEGES ON raspiblog.* TO
root@localhostIDENTIFIED BY "contraseña";
Query OK, 0 rows affected (0.00 sec)
```

Para tener en cuenta las modificaciones de los permisos asignados a root, ejecute el comando `FLUSH PRIVILEGES;`.

```
MariaDB [raspiblog]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)
```

La configuración de la base de datos ha terminado y puede salir de MariaDB con el comando `exit;`.

```
MariaDB [raspiblog]> exit;
Bye
```

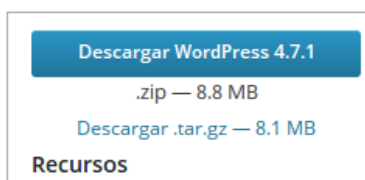
El sitio web está listo para la puesta en marcha de páginas web dinámicas. La primera solución es desarrollar las páginas en PHP-MariaDB, aunque actualmente existen soluciones de CMS. Implementan de manera transparente para el usuario los lenguajes de programación de la parte servidora (PHP) y las bases de datos (MariaDB).

5. Instalación de WordPress

Entre estos CMS están Drupal, Joomla y WordPress. Los dos primeros permiten realizar sitios web de envergadura, mientras que WordPress inicialmente estaba más orientado a la creación de blogs, aunque ahora se utiliza para la creación de sitios web estándares. WordPress representa el 38 % de los CMS utilizados (estadísticas 2016 <http://trends.builtwith.com/cms>). Actualmente es el producto más extendido en este dominio.

a. Descarga del archivo WordPress

Descargue del sitio web es.wordpress.org la última versión española de WordPress. El paquete disponible en el repositorio Raspbian no es la última versión disponible. Las versiones más recientes contienen actualizaciones de seguridad importantes si su Raspberry Pi está conectada a Internet.



En la página de inicio de WordPress en español, sitúe el ratón en el texto **Descargar .tar.gz - 8.1 MB**. Haga clic con el botón derecho del ratón y copie la dirección del enlace (en la captura de pantalla anterior el enlace se muestra junto al texto para facilitar la lectura). Esta dirección va a servir para descargar el archivo de WordPress en la Raspberry Pi. Sitúese en el directorio `/var/www/html`.

- Compruebe el número de la última versión y adapte los comandos que tienen `wordpress-4.7.1-es_ES` a la versión actual en el momento en que instale WordPress.

```
pi@raspiblog:/var/www/html $ wget https://es.wordpress.org/
wordpress-4.7.1-es_ES.tar.gz
--2016-07-20 19:27:46-- https://es.wordpress.org/
wordpress-4.7.1-es_ES.tar.gz
Resolución de es.wordpress.org (es.wordpress.org)...
66.155.40.249, 66.155.40.250
Conexión a es.wordpress.org (es.wordpress.org)
|66.155.40.249|:443... conectado.
```

```
Petición HTTP enviada, esperando respuesta... 200 OK
Tamaño: 8210897 (7,8M) [aplicación/byte-stream]
Hacer una copia de seguridad en: «wordpress-4.7.1-es_ES.tar.gz»

wordpress-4.7.1-es_100%[=====>] 7,83M
418KB/s ds 20s

2016-07-20 09:10:47 (405 KB/s) - « wordpress-4.7.1-es_ES.tar.gz »
registrado [8210897/8210897]
```

b. Extracción de WordPress

El archivo que contiene WordPress ahora está almacenado en la tarjeta micro SD, en el directorio `/var/www/html`. El comando `tar` extrae todos los archivos y crea de nuevo la arborescencia necesaria para el funcionamiento de WordPress.

```
pi@raspiblog:/var/www/html $ tar -xvf wordpress-4.7.1-es_ES.tar.gz
```

Este comando crea una carpeta `wordpress` que contiene los archivos del CMS.

```
pi@raspiblog:/var/www/html $ ls -al
total 8040
drwxrwsr-x 3 www-data www-data 4096 jul. 20 09:15 .
drwxr-xr-x 3 root root 4096 jul. 19 21:55 ..
-rw-rwSr-- 1 www-data www-data 3373 jul. 19 21:55
index.lighttpd.html
-rw-r--r-- 1 pi www-data 84 jul. 19 22:34 index.php
drwxr-sr-x 5 pi www-data 4096 jun 22 10:17 wordpress
-rw-r--r-- 1 pi www-data 8210897 jun 22 10:19 wordpress-4.7.1-es_ES.tar.gz
```

El directorio `/var/www/html` está en el estado en el que le ha dejado la instalación y la prueba de `lighttpd`. Para tener una instalación "propia", es conveniente eliminar los archivos utilizados durante las pruebas:

```
pi@raspiblog:/var/www/html $ rm index.*
```

Para que WordPress se ejecute en la raíz del sitio Web, mueva todos los archivos y directorios contenidos en el directorio `wordpress` a la raíz del sitio Web y después elimine el directorio `wordpress`.

⚠ Atención, esta copia eliminará los archivos que tengan el mismo nombre en la raíz del sitio web. Piense si necesita hacer una copia de seguridad de sus archivos antes de mover el contenido de este directorio.

```
pi@raspiblog:/var/www/html $ mv wordpress/* -t ./
pi@raspiblog:/var/www/html $ rmdir wordpress/
pi@raspiblog:/var/www/html $ rm wordpress-4.7.1-es_ES.tar.gz
```

Después de mover el directorio `wordpress` a la raíz del sitio web, los dos comandos `rmdir` eliminan el directorio `wordpress` vacío de su contenido y el archivo `wordpress-4.7.1-es_ES.tar.gz`.

Puede comprobar que WordPress se ha transferido correctamente a `/var/www/html` con el comando `ls`:

```
pi@raspiblog:/var/www/html $ ls
index.php          wp-comments-post.php  wp-login.php
license.txt        wp-config-sample.php  wp-mail.php
readme.html        wp-content             wp-settings.php
wordpress-4.7.1-es_ES.tar.gz  wp-cron.php           wp-signup.php
wp-activate.php    wp-includes           wp-trackback.php
wp-admin           wp-links-opml.php     xmlrpc.php
wp-blog-header.php wp-load.php
```

El sitio web está preparado. Tan solo queda configurar WordPress para poder usarlo.

c. Configuración de WordPress

La primera operación de configuración consiste en autorizar la escritura en la arborescencia `wp-content` de WordPress para que los uploads no provoquen mensajes de error.

```
pi@raspiblog:/var/www/html $ sudo chmod -R g+w wp-content
```

La configuración de WordPress se hace en el archivo `wp-config.php`. Este archivo no existe inicialmente. En primer lugar, hay que copiar el archivo `wp-config-sample.php`, renombrándolo por `wp-config.php`.

```
pi@raspiblog:/var/www/html $ cp wp-config-sample.php wp-config.php
```

A continuación abra `wp-config.php` con `nano`.

```
pi@raspiblog:/var/www/html $ nano wp-config.php
```

Escriba los campos que se indican a continuación:

```
/** nombre de la base de datos de WordPress. */
define('DB_NAME', 'nombre de su base de datos' );

/** Usuario de la base de datos MySQL. */
define('DB_USER', 'usuario de su base de datos');

/** contraseña de la base de datos MySQL. */
define('DB_PASSWORD', 'contraseña de su base de datos');

/** Dirección del almacenamiento MySQL. */
define('DB_HOST', 'localhost');
```

Escriba respectivamente el nombre de la base de datos, el nombre de su usuario, la contraseña que haya definido durante la instalación de MariaDB y por último el nombre de host del ordenador.

Ahora puede acceder a la página de configuración rápida `install.php`. Esta página permite configurar de WordPress en algunos minutos.

➔ Sustituya la dirección IP por la de su Raspberry Pi.

Hola

¡Bienvenido al famoso proceso de instalación de WordPress en cinco minutos! Simplemente completa la información siguiente y estarás a punto de usar la más enriquecedora y potente plataforma de publicación personal del mundo.

Información necesaria

Por favor, debes facilitarnos los siguientes datos. No te preocupes, siempre podrás cambiar estos ajustes más tarde.

Título del sitio

Nombre de usuario
Los nombres de usuario pueden tener únicamente caracteres alfanuméricos, espacios, guiones bajos, guiones medios, puntos y el símbolo @.

Contraseña
Fuerte
Importante: Necesitas esta contraseña para acceder. Por favor, guárdala en un lugar seguro.

Tu correo electrónico
Comprueba bien tu dirección de correo electrónico antes de continuar.

Visibilidad para los buscadores Disuade a los motores de búsqueda de indexar este sitio
Depende de los motores de búsqueda atender esta petición o no.

Escriba el título del sitio Web y el identificador. Es aconsejable no usar admin como identificador, ya que los ataques a blogs WordPress se hacen generalmente suponiendo que el identificador es *admin*. Use otro identificador, en la configuración detallada de WordPress será posible mostrar *admin* como administrador del blog, lo que reducirá la efectividad de los ataques, porque este no será el verdadero nombre del administrador.

WordPress le propone una contraseña "robusta" que puede utilizar. Si desea utilizar otra contraseña, escriba una para el administrador del blog. No dude en seleccionar una contraseña larga, de más de 10 caracteres, que contenga una mezcla aleatoria de minúsculas, mayúsculas, cifras y caracteres especiales. Esto garantizará una buena resistencia del blog frente a los ataques.

➔ No use la contraseña que figura en las capturas de pantalla de este capítulo.

Título del sitio

Nombre de usuario
Los nombres de usuario pueden tener únicamente caracteres alfanuméricos, espacios, guiones bajos, guiones medios, puntos y el símbolo @.

Contraseña
Fuerte
Importante: Necesitas esta contraseña para acceder. Por favor, guárdala en un lugar seguro.

Tu correo electrónico
Comprueba bien tu dirección de correo electrónico antes de continuar.

Introduzca también una dirección de correo electrónico válida y pulse el botón **Instalar WordPress**. Después de una espera durante la que WordPress inicializa el servidor web obtiene esta pantalla que confirma el éxito de la instalación:

¡Lo lograste!

WordPress ya está instalado. ¡Gracias y disfrútalo!

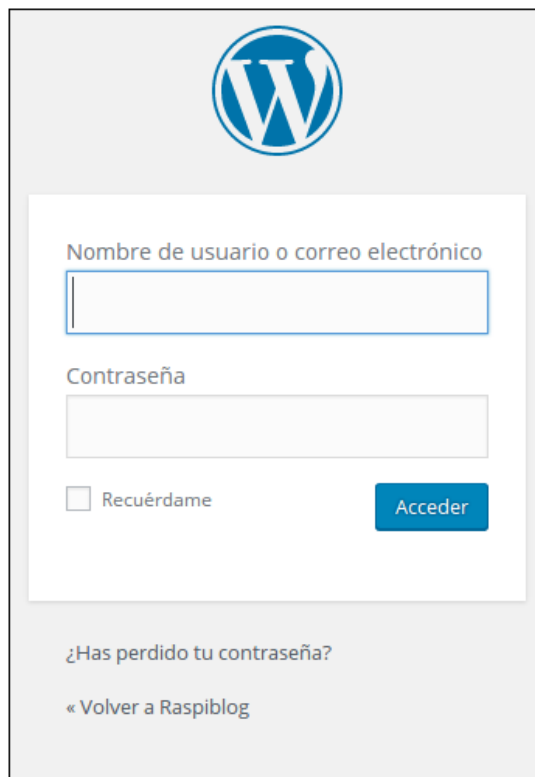
Nombre de usuario Francois

Contraseña Tu contraseña elegida.

Pulse el botón **Acceder** para pasar a la etapa siguiente, es decir, probar el sitio web.

6. Comprobación del sitio web

La aparición de la ventana de conexión cuando pulsa en el botón **Acceder** ya es una primera prueba concluyente.



Escriba el identificador y la contraseña que ha seleccionado para el administrador del sitio web, y después pulse en el botón **Acceder**.

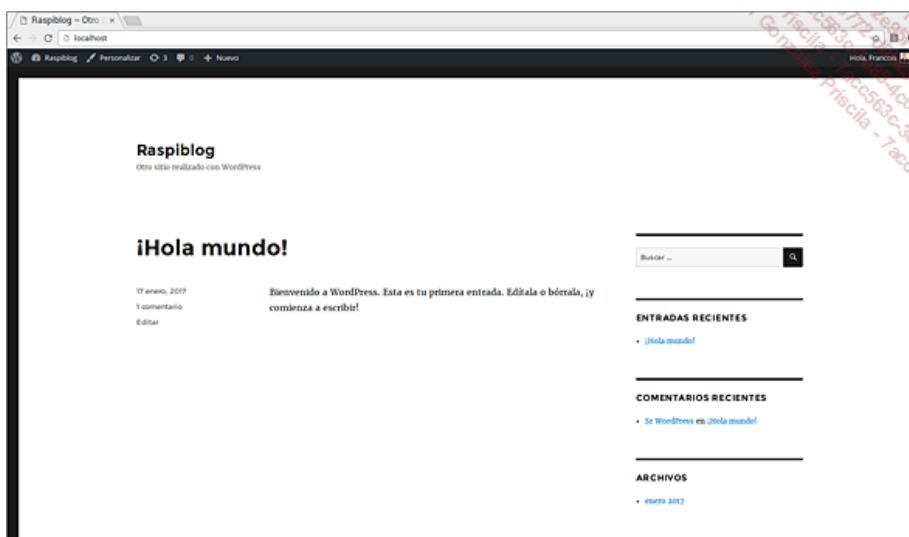
El panel de control de WordPress le da la bienvenida. Está preparado para crear las páginas y los artículos.

Para empezar la personalización de su sitio, pulse el botón **Personaliza tu sitio**. Un conjunto de pestañas situado en la parte izquierda de la pantalla le permite definir un determinado número de parámetros.



El libro de Christophe Aubry: *WordPress 4.5 y 4.6 - Un CMS para crear y administrar blogs y sitios web* publicado por Ediciones ENI en la colección *Objetivo Web* le será de gran ayuda para continuar el desarrollo de su sitio web.

Conéctese a la dirección local de su Raspberry Pi para descubrir la interfaz por defecto de sitio web totalmente nuevo WordPress.



Por defecto, WordPress 4.7.1 utiliza el tema Twenty Seventeen. Puede modificar este tema o cambiarlo por otro. Se presenta un primer artículo y un comentario a efectos demostrativos. Puede eliminarlos y empezar a crear su propio contenido.

7. Poner en línea el sitio web

Después de tener preparado su sitio web en local, llega el momento de ponerlo a disposición de los internautas del mundo entero.

a. Nombre de dominio

Para poner en línea un sitio web, se necesita una dirección pública, es decir, una dirección IP accesible desde Internet (para más información de las direcciones IP, consulte el capítulo *Gestión de la red*). Las direcciones distribuidas en los box o routers, generalmente del tipo 192.168.1.xxx, son direcciones privadas. Los paquetes de datos que salen de su box, así como los que llegan, utilizan la dirección pública de la box. Es su responsabilidad encontrar la información de la máquina correcta en función de una tabla de traducción de direcciones (NAT) que

tiene actualizada permanentemente.

Es posible indicar la dirección IP del box para acceder al sitio web, conservar una gran cantidad de direcciones públicas no es sencillo. La solución la proporcionan los DNS, encargados de convertir los nombres de hosts y de dominios en direcciones IP. De esta manera, los usuarios de Internet memorizan solo los nombres.

Por tanto, hay que asociar el nombre de dominio del blog raspiblog.es con una dirección IP. El nombre de dominio se ha reservado en BookMyName. Este registro ofrece al usuario la posibilidad de definir su propia información de DNS.

Después de reservar el nombre de dominio, se ha definido una zona DNS personalizada en BookMyName para asociar el nombre del servidor web del blog <https://www.raspberrypi.org/blog/> con la dirección pública del router, detrás del cual se encuentra la Raspberry Pi.

Esta solución es posible porque la dirección pública del router es fija. Algunos proveedores de acceso modifican regularmente la dirección pública proporcionada a la box. En este caso una solución como No-IP o DynDNS garantiza la actualización del DNS para cada cambio de la dirección IP pública.

b. Configuración del router

Las peticiones que llegan al router destinadas a la Raspberry Pi se emiten con el protocolo http. Cada protocolo utiliza un puerto diferente, lo que permite separar la información que circula por Internet. El protocolo http utiliza el puerto 80. Cuando una petición llega a la router en el puerto 80, no sabe tratarla. Por tanto, hay que indicarle que reenvíe todo lo que llegue al puerto 80 a la dirección interna de la Raspberry Pi. Esta operación se llama "redirección de puerto". El servidor web lighttpd activo en la Raspberry Pi recibirá la petición http y podrá responderla.

El reparto de las direcciones en el router es la siguiente:

- 192.168.1.10 a 192.168.1.50, direcciones distribuidas por DHCP a las máquinas que se conectan al router.
- 192.168.1.53, dirección fija de la Raspberry Pi.
- 192.168.1.200, dirección fija de la impresora láser.
- 192.168.1.254, dirección del router = dirección de la pasarela para las máquinas en red.

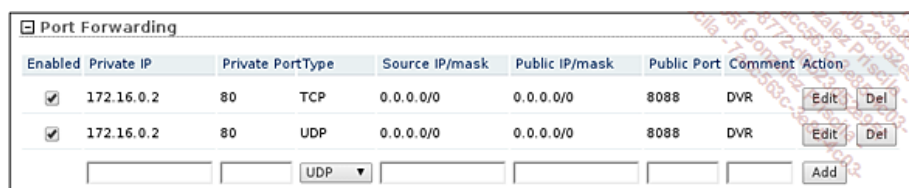
La redirección de puerto del router se ajusta en la sección **Configuración básica - Red local IPV4**.

Escriba el puerto que se va a redirigir: 80 en la zona **Puerto o rango de puertos (externo) WAN**.

El protocolo por defecto es **TCP**, manténgalo.

Indique la dirección de la Raspberry Pi 192.168.1.53 en la zona **Puerto o rango de puertos (INTERNO) LAN**.

Configure el **Puerto o rango de puertos (interno) LAN** igual a 80, es en este puerto de la Raspberry Pi donde se enviarán las peticiones que lleguen al puerto 80 de la box desde Internet.



Pulse en **Add** para validarla y añadir una línea de redirección adicional vacía.

Rellene la ventana **Port forwarding**, en función de su red.

A continuación, pulse el botón **Apply** en la parte inferior de la ventana. Esto valida la modificación. Reinicie su router para que tenga en cuenta la redirección de puerto.

Después del inicio completo de su router, su sitio web es accesible.

Hay procedimientos equivalentes en otras box de Internet. Consulte la ayuda en línea de su proveedor de acceso a Internet.

➤ En función de la versión de su router y proveedor y de las actualizaciones, el procedimiento puede ser diferente.

8. Conectarse al servidor web

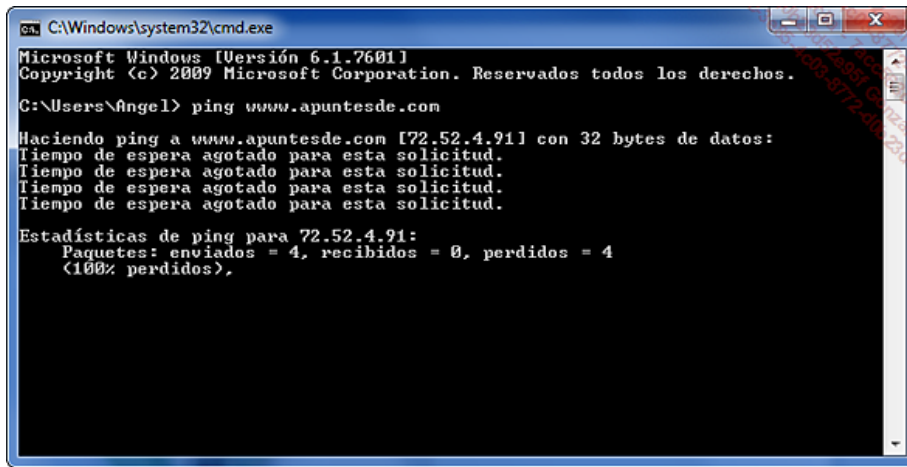
Para probar el acceso al servidor web desde la red local situada en su box, será necesario introducir dirección IP de la Raspberry Pi en la barra de direcciones del navegador. Este acceso interno confirma el correcto funcionamiento del sitio alojado en la Raspberry Pi.

Para los accesos desde máquinas conectadas a Internet (fuera de su red), será necesario introducir en la barra de direcciones del navegador el nombre del sitio que haya seleccionado.

Pasados algunos segundos, su sitio web debe aparecer.



En caso de problema, es necesario implementar un procedimiento de resolución de problemas. Empiece haciendo un ping al nombre de dominio. Incluso si su box o su router está configurado para no responder al ping, puede comprobar que la dirección IP enviada por el DNS corresponde con su dirección pública.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Angel> ping www.apuntesde.com

Haciendo ping a www.apuntesde.com [72.52.4.91] con 32 bytes de datos:
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.

Estadísticas de ping para 72.52.4.91:
    Paquetes: enviados = 4, recibidos = 0, perdidos = 4
              (100% perdidos),
```

Si esta dirección es correcta, compruebe que la redirección en la box está configurada correctamente y se ha tenido en cuenta. Si estas comprobaciones no aportan una solución al problema, un análisis de los paquetes IP con Wireshark puede aportar elementos adicionales. Wireshark es un analizador de protocolo que hace necesario tener buenos conocimientos de redes para explotar las capturas que contiene.

- Para acceder a la sección de control, entre en la barra de dirección del web http://direccion_ip_del_servidor/wp-admin y conéctese.

Una cámara de vídeo vigilancia con Pi Zero

La calidad de las imágenes proporcionadas por los 8 mega píxeles de la cámara de la Raspberry Pi, la disponibilidad de cámaras infrarrojas que captan imágenes nocturnas y el precio moderado de la Raspberry Pi Zero, permiten hacerse con una cámara de vigilancia con un coste bajo.



Será posible alojar el conjunto de componentes en una caja de cámara ficticia (foto anterior), que se vende por menos de diez euros en las tiendas. Provista de articulaciones, el índice de protección IP44 de esta caja protege al equipamiento que contiene de cuerpos sólidos de dimensiones superiores a 1 mm y del impacto del agua que provenga de cualquier dirección.

- La puesta en marcha de una cámara de vigilancia en su casa, en una empresa o en la vía pública está sujeta a obligaciones reglamentarias y legislativas que debe respetar.

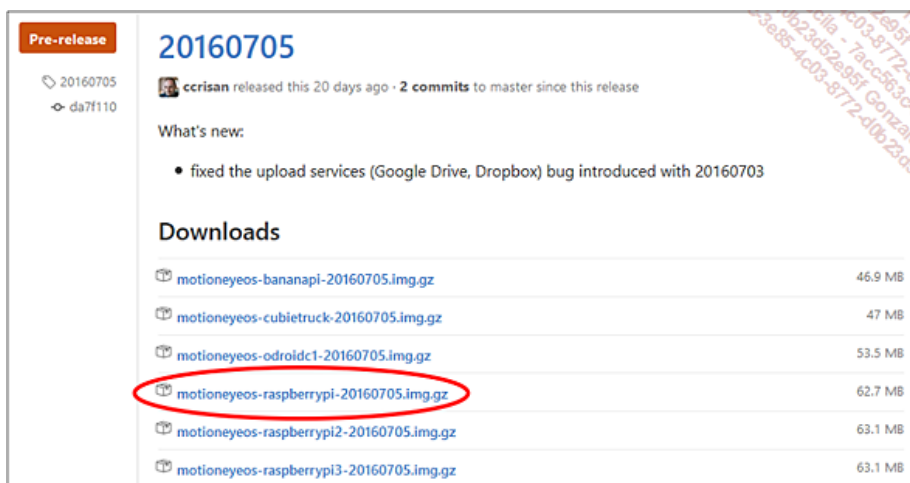
1. La distribución motionEyeOS

Desarrollada por Calin Crisan, esta distribución reducida (menos de 70 MB), dispone de todos los ingredientes necesarios para la puesta en marcha de una cámara de vigilancia. Por el momento, está disponible en inglés. Después de la instalación de motionEyeOS veremos las diferentes opciones para configurar las más importantes en la aplicación que nos interese.

motionEyeOS puede gestionar varias cámaras. La cámara de la Raspberry Pi y también webcams USB. Será necesario comprobar que estas últimas sean compatibles con Raspbian.

2. Instalación de motionEyeOS

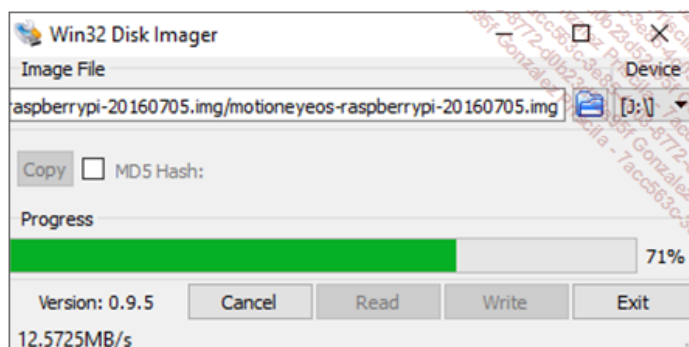
Vaya a la página de la distribución: <https://github.com/ccrisan/motioneyeos/releases>



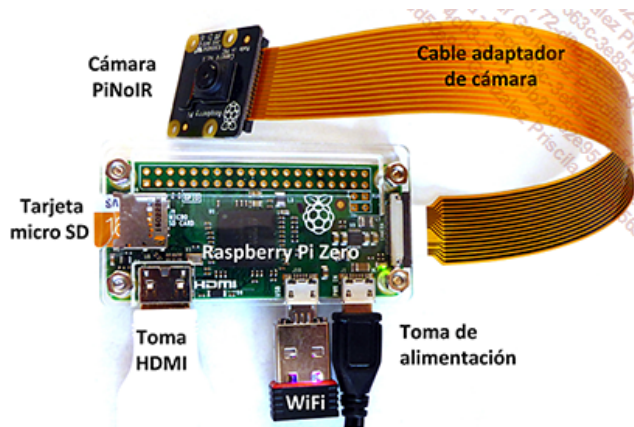
Descargue la última versión de motionEyeOS para la Raspberry Pi. Esta versión funcionará con la Raspberry Pi Zero.

- Si desea utilizar otro modelo de Raspberry Pi, descargue la versión correspondiente a su tarjeta.

Descomprima el archivo obtenido para extraer la imagen (con 7zip, por ejemplo). Vaya al capítulo Preparar la tarjeta micro SD para más información. En Windows, la herramienta Win32DiskImager transferirá la imagen de la distribución a la tarjeta micro SD (siguiente imagen). Visto el tamaño reducido de la distribución, esta operación es muy rápida.



Inserte la tarjeta micro SD en el conector de la Raspberry Pi Zero, conecte los cables HDMI y USB si es necesario.



El sistema anterior que ha servido para realizar esta cámara, está compuesto de:

- una Raspberry Pi Zero
- una caja transparente para la Raspberry Pi Zero
- una tarjeta micro SD
- una cámara infrarroja PiNoIR
- un cable adaptador para la cámara
- un adaptador de toma HDMI mini <-> HDMI
- una llave Wi-Fi USB equipada de un adaptador micro USB <-> USB
- una alimentación 5v - 2,5A

➤ La Raspberry Pi Zero no tiene puerto Ethernet. motionEyeOS transmite las imágenes y se configura a través de la red, de ahí el uso de una llave Wi-Fi y de un adaptador micro USB.

motionEyeOS gestiona automáticamente la dirección IP en una conexión alámbrica. Si usa una llave USB Wi-Fi como antes, debe crear en la partición /boot de la tarjeta micro SD un archivo *wpa_supplicant.conf* que contenga:

```
update_config=1
ctrl_interfaz=/var/run/wpa_supplicant

network={
    scan_ssid=1
    ssid="SSID_de_su_wifi"
    psk="contraseña_de_su_wifi"
}
```

La distribución utilizará esta información en el momento del arranque, para conectarse a través del Wi-Fi al punto de acceso designado (unidad de conexión Internet, router, etc...).

➤ La partición /boot es accesible directamente en Windows, cuando la tarjeta micro SD está en un lector de tarjetas SD.

3. Arrancar motionEyeOS

Conecte el cable de alimentación a la Raspberry Pi. Si ha conectado una pantalla, en primer lugar verá una frambuesa que representa el único núcleo del SoC. Tras algunas inicializaciones, entre ellas la extensión de la partición a toda la tarjeta micro SD, motionEyeOS arranca. Al final de la secuencia de arranque, la dirección IP de la Raspberry Pi se muestra.

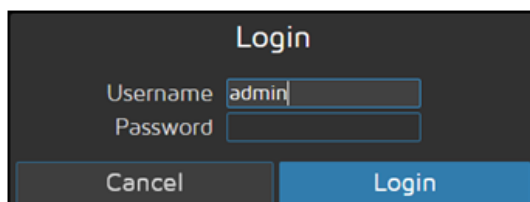
ID	Client Name	MAC Address	IP Address	Valid Time
6	meve-d2c1cfd	00:0F:55:A7:C9:5D	192.168.1.108	19.22.07

En ausencia de pantalla habrá que enumerar, como se muestra aquí arriba, los clientes del servidor DHCP de su red y descubrir la dirección IP de aquel cuyo nombre empiece por *meve-*.

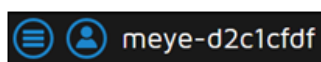
4. Conexión a motionEyeOS

Una vez ha arrancado motionEyeOS, es posible conectarse desde un navegador que funcione en una máquina situada en la misma red. Aparece una ventana de login.

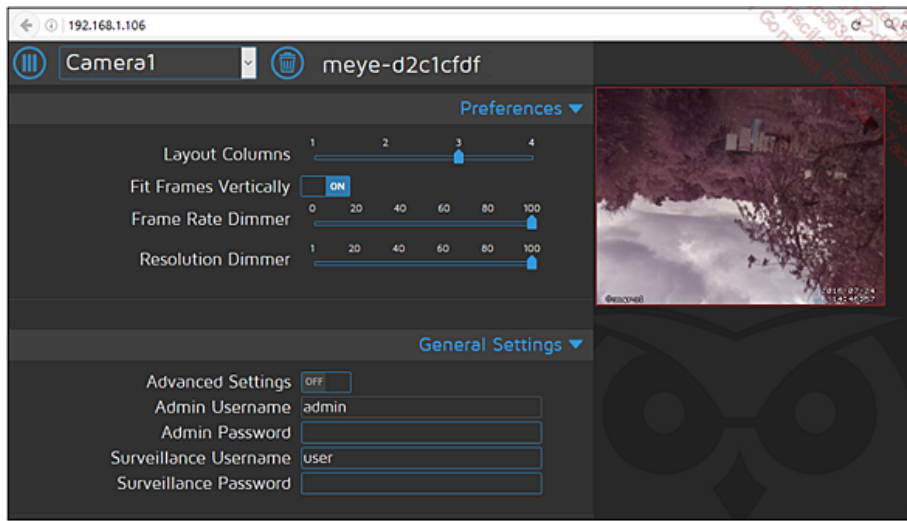
Introduzca el login **admin** sin contraseña y valide.



Durante los siguientes reinicios, la conexión se establece por defecto para un usuario llamado *user*. Este solo tiene acceso a la visualización de la imagen y a la configuración de la pantalla (número de columnas y ajuste de las imágenes).

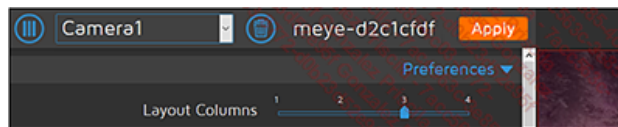


Para conectarse como administrador, pulse en la silueta que representa una persona en la parte superior de la pantalla, a la izquierda del nombre del ordenador. El icono situada más a la izquierda representa tres líneas horizontales, y abre la ventana de configuración.

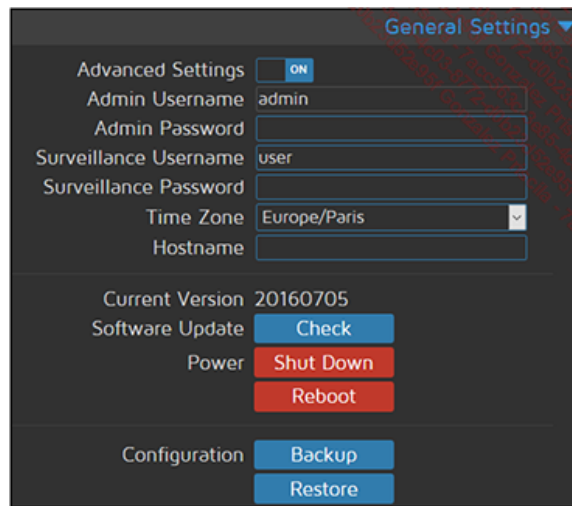


Obtiene la pantalla anterior con una configuración mínima por defecto.

Para acceder a la configuración completa, configure **General Settings - Advanced Settings** a **ON**.



Para validar esta modificación, así como todas las modificaciones futuras, pulse el botón rojo **Apply** (imagen anterior) que aparece en la parte superior de la página web cuando modifique algún argumento.



Ahora dispone de opciones adicionales. Ajuste el huso horario (**Time Zone**) en función de su situación geográfica. Esto relacionará la fecha y la hora de las imágenes y de las películas con la hora local.

Si lo desea, puede modificar los logins y las contraseñas del administrador y del usuario. Este no es necesario para realizar las pruebas pero es indispensable si se conecta la cámara a Internet.

En la parte inferior de la captura de pantalla anterior aparecen los botones que permiten comprobar la presencia de nuevas versiones de la distribución (**Check**), detener el sistema (**Shut Down**) o reiniciarlo (**Reboot**).

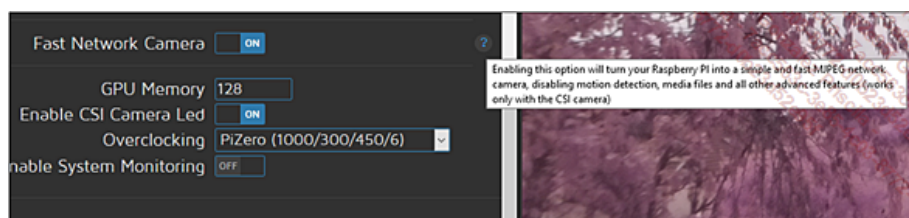
- Cuando haya configurado motionEyeOS según sus necesidades, es posible (y muy aconsejable) guardar la configuración en su disco duro o en una llave USB (**Backup**). También puede recargar una configuración guardada con antelación (**Restore**).

5. Configuración de motionEyeOS

La distribución tiene muchas opciones. Las principales se verán aquí. Para el resto, dispone de una ayuda en forma de icono que aparece junto a cada opción cuando pasa por encima de la zona.

a. Cámara Ethernet

Para ilustrar esta ayuda, despliegue la zona de argumentos **Expert Settings**.



Junto a la opción **Fast Network Camera** (Cámara Ethernet rápida) aparece un signo de interrogación cuando pasa por encima del texto. Pulse en el signo de interrogación para que aparezca un cuadro de texto. En él encontrará información sobre esta opción: solo se puede usar con la cámara conectada al bus CSI. Además, anula la detección de movimiento y el registro de archivos. La Raspberry Pi ahora se ha convertido en una simple cámara que envía un flujo.

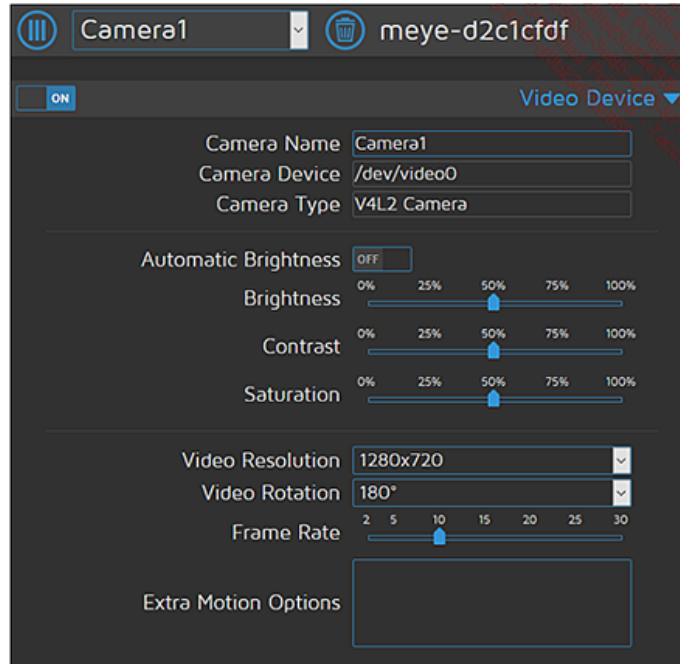
Para probar el funcionamiento como cámara Ethernet, configure la opción **Fast Network Camera** a **ON**. Valide pulsando en **Apply** en la parte superior de la página.

Conéctese a la dirección de la Raspberry Pi desde un navegador para ver las imágenes transmitidas por la cámara.

Para continuar con la configuración de motionEyeOS, configure la opción **Fast Network Camera** a **OFF**.

b. Configuración del vídeo

La función principal de motionEyeOS es tratar el vídeo de la cámara. Por tanto, hay que configurarlo para que responda a sus necesidades.

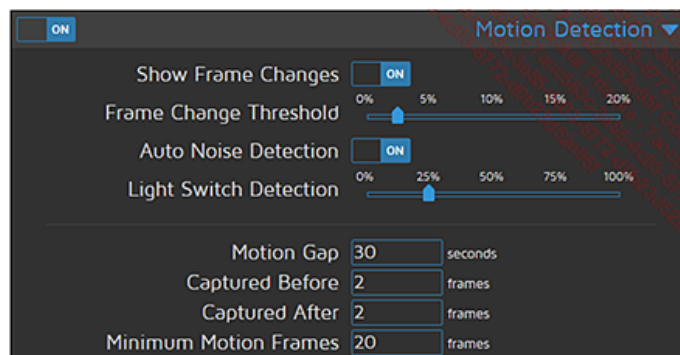


En la sección **Video device** puede seleccionar activar el ajuste automático de luminosidad o ajustar los parámetros de forma manual. La resolución del vídeo es accesible también, así como el número de imágenes por segundo. No hay que olvidar que cuando más aumenta el tamaño de la imagen, más importante es el débito. Lo mismo ocurre con el número de imágenes por segundo.

Con la cámara de la Raspberry Pi, es frecuente que la imagen sufra una rotación respecto a la posición de la cámara. La opción **Video Rotation** permite poner la imagen en el sentido correcto.

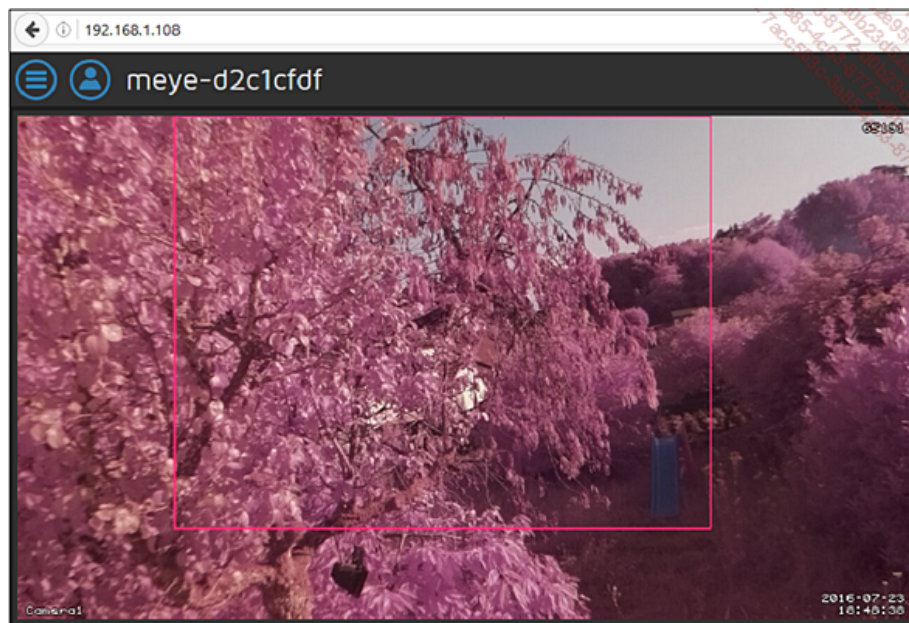
c. Ajuste de la detección de movimiento

motionEyeOS integra el software motion. Su principal funcionalidad es detectar un movimiento, comparando las sucesivas imágenes proporcionadas por la cámara. Memoriza temporalmente las imágenes, lo que permite crear vídeos que empiecen antes de detectar el movimiento, tomando las imágenes almacenadas en memoria. Motion también puede almacenar imágenes fijas durante la detección de eventos y enviar un correo electrónico a uno o varios destinatarios. Se pueden añadir imágenes de la escena a este correo electrónico.



En la sección **Motion Detection** el interruptor situado a la izquierda de la línea del título se debe configurar a **ON** si quiere utilizar la función de detección de movimiento.

Show Frame Changes rodea con un rectángulo rojo la zona de la imagen en la que motion ha detectado un movimiento. En la siguiente captura de pantalla, la detección de movimiento la ha provocado el movimiento de las ramas debido al viento. Se recomienda activar esta opción durante la puesta a punto.



Frame Change Threshold ajusta el umbral a partir del cual el número de píxeles cambia de una trama (imagen), a la otra. Cuando más bajo sea este valor, más sensible es la instalación, pero aumenta mucho las falsas detecciones.

Auto Noise Detection permite el ajuste automático del nivel de ruido de la imagen.

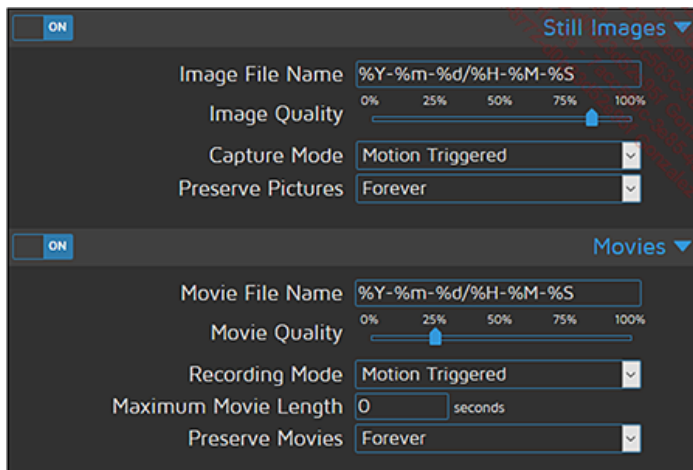
Light Switch Detection indica el porcentaje de la imagen que debe cambiar entre dos tramas, para que se interprete como un cambio importante de luminosidad y no como un movimiento. Ponga el valor a 0 para desactivar esta función.

La parte inferior de esta sección permite ajustar:

- El tiempo durante el que no se hará ninguna detección, tras una primera detección (**Motion Gap**).
- El número de tramas capturadas antes (**Captured Before**) o después (**Captured After**) de una detección.
- El número mínimo de imágenes contenidas en una película (**Minimum Motion Frames**).

d. Elección del nombre de los archivos guardados

Puede decidir la forma que tomarán los nombres de los archivos que se guardan en la tarjeta micro SD. Por defecto, los dos tipos de archivos se guardan siguiendo los mismos criterios.



Se crea una carpeta correspondiente a la fecha del día (año-mes-día) en la que los archivos se nombran en función de la hora (hora-minutos-segundos).

El registro de las imágenes fijas (**Still Images**) se puede desencadenar (**Sensor Mode**) por un movimiento (**Motion Triggered**). También se puede tratar de un timelapse (**Interval Snapshot**), con fotos tomadas a intervalos de tiempo regulares. Para terminar, es posible guardar todas las imágenes (**All Frames**).

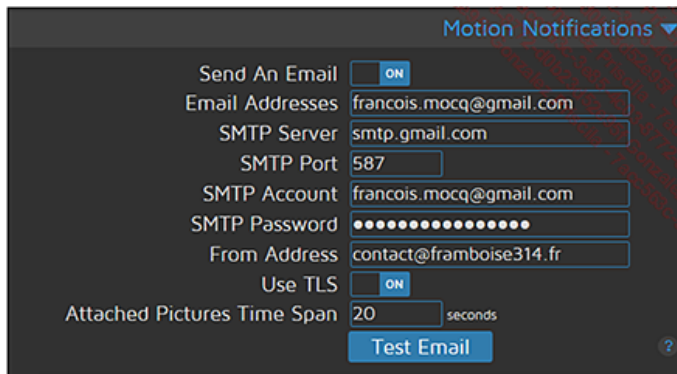
La conservación de las imágenes (**Preserve Pictures**) puede durar indefinidamente (**Forever**) o un tiempo determinado.

Las opciones **Image Quality** y **Movie Quality** ajustan la calidad de las imágenes registradas.

Para las películas, las opciones son prácticamente idénticas a las de las imágenes fijas. Es posible registrar vídeo continuo (**Continuous Recording**). Será necesario dimensionar la memoria masiva para garantizar un registro durante todo el periodo previsto.

e. Configuración de las notificaciones

Detectar un movimiento es una cosa, advertir es otra. motionEyeOS permite configurar el envío de uno o varios correos electrónicos para alertar. Es posible adjuntar un mensaje a las imágenes capturadas por la cámara.



La configuración anterior en la sección **Motion Notifications** funciona con la aplicación de correo electrónico. Será necesario adaptarla en función de la aplicación de correo electrónico que use.

Después de configurar el interruptor **Send An Email** a **ON**, indique la dirección de correo electrónico a la que se enviará el mensaje. Puede agregar otras direcciones separándolas por comas.

Guarde la información relativa a la cuenta de correo electrónico (SMTP) utilizada para el envío de correos. En caso de Gmail, si usa la conexión en dos etapas, habría que obtener una contraseña de aplicación para motionEyeOS, porque su contraseña habitual no funcionará aquí.

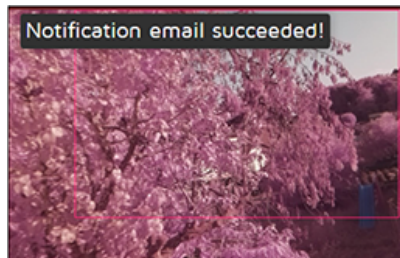
Vaya a la página <https://security.google.com/settings/security/apppasswords>. Después de haber rellenado la contraseña de su cuenta Gmail, acceda a una página que permita generar contraseñas para sus aplicaciones. Pulse en **Seleccionar aplicación** y guarde el nombre de su aplicación pulsando en **Otras (nombre personalizado)**. Valide su petición pulsando el botón **Generar**. Se abre una ventana:



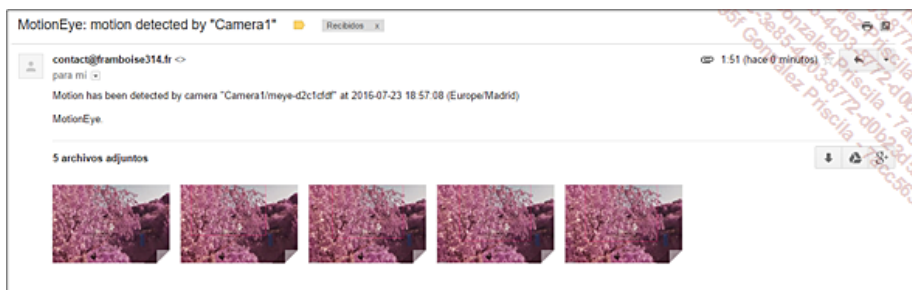
Contiene una contraseña compuesta de 16 caracteres. Esta contraseña es la que usará con motionEyeOS. Obsérvela cuidadosamente porque no podrá encontrarla. Recibirá un correo de confirmación que no contiene la contraseña. La única posibilidad, en caso de olvidarla, es volver a la página de generación de contraseñas, revocar la contraseña olvidada y generar otra.

Puede indicar una dirección de envío en la zona **From Address**. Esta es la dirección con la que se enviarán los mensajes.

El botón **Test Email** fuerza el envío de un mensaje de detección. Si obtiene como respuesta un mensaje escrito en rojo en la parte superior de la página, analice cuidadosamente la causa del rechazo del mensaje para remediarlo. Si el envío del correo electrónico se desarrolla normalmente, debe obtener un reporte afirmativo, como sigue.



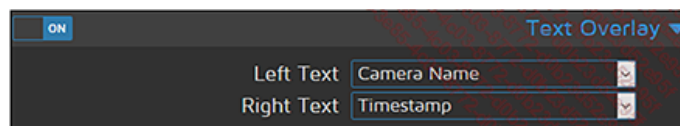
Cuando la cámara detecta un movimiento, envía un correo electrónico que señala el evento. El valor **Attached Pictures Time Span** se utiliza para fijar el intervalo de tiempo que separa las imágenes adjuntas al mail. Puede desactivar esta opción poniendo el valor a 0.



El mensaje anterior señala una detección de movimiento. Cinco imágenes se han añadido como elementos adjuntos. Las zonas en las que los movimientos se han detectado, se enmarcan en rojo.

f. Información superpuesta en las imágenes

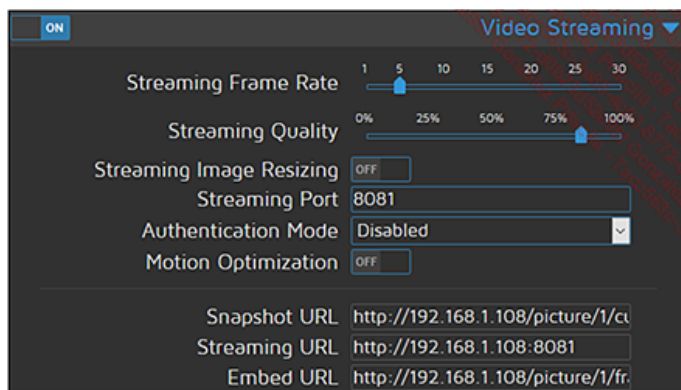
motionEyeOS le permite seleccionar la información que desea ver aparecer como texto superpuesto en las imágenes.



La sección **Text Overlay** anterior da acceso a la configuración de los textos situados a la derecha y a la izquierda de la imagen. Para estos dos textos, puede seleccionar mostrar el nombre de la cámara (**Camera Name**), la fecha y hora (**Time Stamp**), un texto personalizado (**Custom Text**) o ningún texto (**Disabled**).

g. Difusión de las imágenes

Los usuarios de un sistema de vídeo vigilancia habitualmente desean poder echar un vistazo al espacio que se está vigilando. Como es lógico, motionEyeOS ofrece esta función. La cámara Ethernet presentada al inicio de este capítulo es un ejemplo, aunque su uso implica abandonar otras posibilidades del software.



En la sección **Video Streaming**, el número de imágenes por segundo se ajusta con **Streaming Frame Rate** y la calidad con **Streaming Quality**. Estos dos argumentos influyen en el ancho de banda necesario para transmitir el flujo de imágenes. Será necesario permanecer atento y conservar los argumentos que garantizan una transmisión correcta de las imágenes. Cuando **Streaming Image Resizing** está a **ON**, las imágenes se redimensionan antes de su envío al navegador web remoto. Con un procesador poco potente es aconsejable dejar este argumento a **OFF**.

Por defecto, el puerto por el que se transmite el flujo de imágenes es el 8081 para la cámara 1, el 8082 para la cámara 2, etc. Si lo desea, puede modificar estos puertos en la zona de texto **Streaming Port**. Para observar el flujo de imágenes, conéctese con un navegador web a la dirección de la Raspberry Pi e indique el puerto 8081, por ejemplo: 192.168.1.110:8081.

Authentication Mode (tipo de autenticación) está desactivado (**Disabled**) por defecto. Esto significa que no importa quién se pueda conectar a la dirección de la Raspberry Pi para ver las imágenes. Esto no es deseable necesariamente. Seleccione uno de los modos disponibles (**Basic** o **Digest** en función de las aplicaciones que use) para la autenticación. Cuando una persona se conecte, se deberá identificar para visualizar las imágenes. Si no ha modificado los argumentos generales (**General Settings**), se puede conectar como user, sin contraseña, para visualizar las imágenes enviadas.

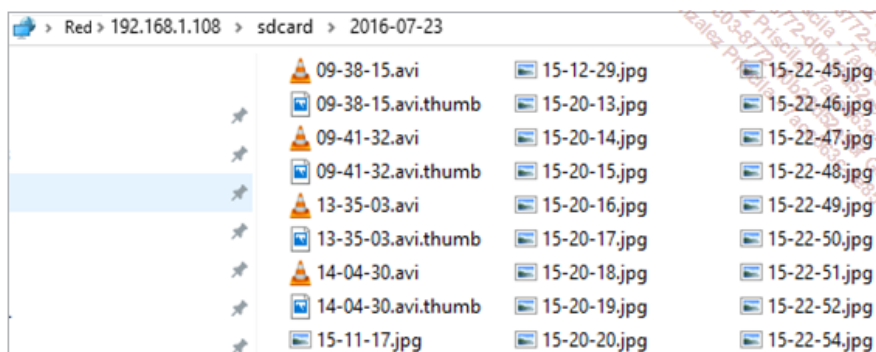
Motion Optimization está a **OFF** por defecto. Si pasa este argumento a **ON** el flujo de vídeo se gestionará mejor y el número de imágenes enviadas por segundo se reducirá cuando no se detecte ningún movimiento.

Los argumentos **Snapshot URL**, **Streaming URL** y **Embed URL** proporcionan, respectivamente, la última imagen fija capturada por el software, el flujo de imágenes y por último un flujo contenido en una página web ligera (menos de 3KB) que es posible incorporar a una página web.

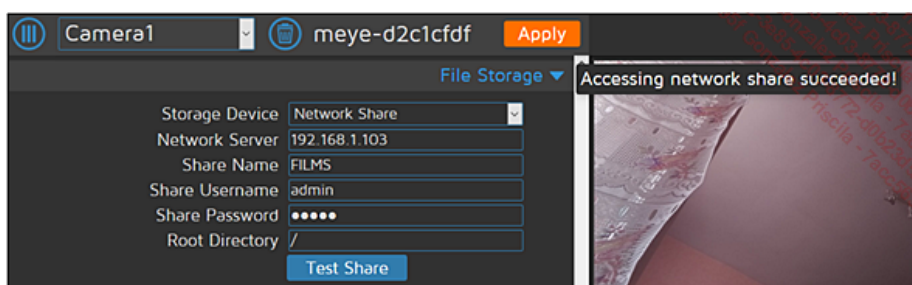
h. Registro de imágenes

Para conservar la prueba de una intrusión, es importante registrar las imágenes en la tarjeta micro SD o en un servidor remoto.

Si activa el servicio Samba, puede visualizar las imágenes y los vídeos registrados conectándose al recurso compartido por defecto, como sigue.



En la sección **File Storage** puede seleccionar registrar los archivos (**Storage Device**) en la tarjeta micro SD, en un directorio de su elección o en un recurso compartido remoto (**Network Share**).



Esta es la opción que se ha tomado aquí. Introduzca el nombre o la dirección IP del servidor (**Network Server**) e indique el nombre del recurso compartido (**Share Name**). En la captura de pantalla anterior, el recurso compartido se llama **Films**. Introduzca el login y la contraseña para acceder al recurso compartido. Verifique la disponibilidad del recurso compartido pulsando el botón **Test Share**. Si obtiene como antes **Accessing network share succeeded!** es que todo ha ido bien y las imágenes se registrarán en el recurso compartido remoto.

6. Conclusión

La distribución motionEyeOs es reciente y todavía presenta defectos e inestabilidades. Pero las funciones ya disponibles conforman una herramienta interesante para el establecimiento de un sistema de vigilancia. El tratamiento de imágenes para detectar movimientos implica una latencia de varios segundos en una Raspberry Pi Zero. El flujo visualizado en pantalla está cuatro o cinco segundos retrasado en relación a la toma de la imagen. Lo importante es que las imágenes registradas durante la detección de un movimiento se guardan.

Incluso aunque esta distribución todavía está en plena evolución, la información contenida en esta presentación le permitirá probarla y familiarizarse con motionEyeOS mientras se estabiliza.

Presentación

Aprender a programar puede ser desagradable. Scratch, con su enfoque visual, colorido y lúdico, permite abordar la programación con los niños. Se presentan los conceptos fundamentales de la programación en forma de fichas que permiten representar visualmente su acción. La puesta a punto de los programas se facilita resaltando las instrucciones que se están ejecutando y la posibilidad de modificar el código durante la ejecución del programa.

1. Una larga historia

Scratch es la herencia de una línea de lenguajes de programación. Desde los años 1960 a los años 1980, el lenguaje Logo invadió progresivamente los ordenadores de la época. La democratización de la informática ayudó a su difusión, particularmente en los entornos educativos. El objetivo principal de Logo era programar el movimiento de un objeto móvil bautizado como "tortuga". Logo también podía tratar listas y ejecutar operaciones. Muchos estudiantes descubrieron la programación a través de las instrucciones de Logo.

Squeak es la evolución de Logo. Es un entorno gráfico en el que el niño crea diseños compuestos por muchos objetos.

Scratch es el descendiente de estos pioneros. Este software libre orientado a los niños a partir de 8 años se ha diseñado por la rama educativa del MIT (*Massachusetts Institute of Technology*). Es un entorno de programación visual. Los objetos implementados en la escena de Scratch se llaman *sprites* (del inglés "duendes"). Se trata de objetos que se desplazan y conforman la decoración de la escena siguiendo las instrucciones contenidas en los scripts. Los duendes pueden ser fijos o animados. Las instrucciones se presentan en forma de fichas de colores que facilitan el seguimiento de las funciones:

- Movimientos = azul
- Apariencia = violeta
- Variables = rojo
- Sonidos = rosa
- Controles = naranja
- etc.

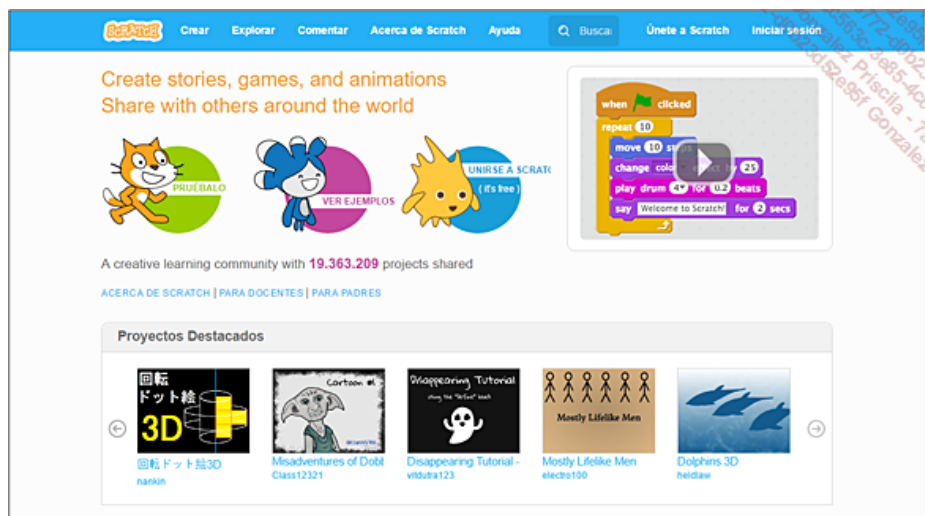
2. Los proyectos

El lema de Scratch es "Imagina Programa Comparte". En el sitio del MIT hay disponibles cerca de 16 millones de proyectos compartidos (agosto de 2016). Cada uno se puede estudiar, modificar, compartir de nuevo, etc. Es una fuente importante de ejemplos, desde los más sencillos a los más elaborados.

Como sucede habitualmente en programación, después de descubrir las bases de un lenguaje, es interesante ver lo que han creado otros desarrolladores. Esto permite apropiarse de las técnicas y forjarse su propio enfoque sobre la programación a partir de los ejemplos.

Las buenas prácticas (*good practices, best practices*) de programación también se adquieren estudiando los proyectos de programadores que han alcanzado buenos resultados utilizando medios elegantes.

La práctica de un lenguaje informático pasa por el descubrimiento de los medios que ofrece y las restricciones a las que están sometidos los desarrolladores.



Los proyectos son accesibles en línea desde el sitio <http://scratch.mit.edu/>. Un motor de búsqueda permite seleccionar los proyectos. El proyecto seleccionado se carga. Entonces es posible ejecutar el programa y mostrar los scripts para estudiarlos... El menú **Archivo** permite descargar el proyecto para poder disponer de él en su ordenador local.

- La versión de Scratch disponible en el sitio es la versión 2. En la Raspberry, la versión que se proporciona con Raspbian es la 1.4. Los proyectos descargados desde el sitio están en .sb2, por lo que son compatibles con la versión 1.4.

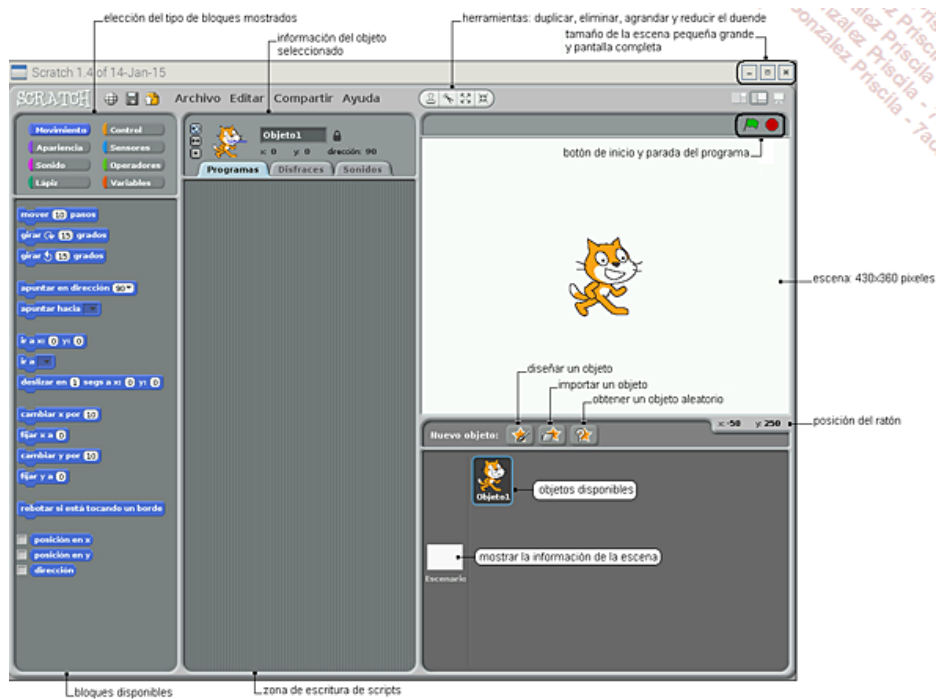
3. Acceso al hardware

Las últimas versiones de Scratch en la Raspberry Pi se benefician de un motor optimizado. La velocidad de ejecución ha aumentado mucho. Además, se ha integrado un servidor GPIO en Scratch y es accesible a través de la opción de menú **Edición**. Encontrará ejemplos de aplicación de este servidor GPIO en el capítulo La GPIO de la Raspberry Pi. También hay un servidor Mesh. Se accede a través de la opción de menú **Compartir**. Este servidor permite hacer de interfaz entre Scratch y las tarjetas HAT. Encontrará ejemplos de aplicación del uso del servidor Mesh en el capítulo Los periféricos, en la sección Las tarjetas de extensión - PiFace Digital 2.

El entorno de Scratch

Para abrir la ventana del entorno de desarrollo Scratch, pulse en **Menú** y después en **Programación**. Para terminar, pulse en **Scratch**.

1. La interfaz gráfica



Un menú situado en la parte superior de la interfaz da acceso a un determinado número de funciones. Hay tres iconos situados en la parte izquierda:

- El globo para definir el idioma utilizado.
- El disquete para guardar el proyecto actual.
- El documento con la flecha para compartir el proyecto en el sitio de Scratch.

El menú textual está situado en la parte derecha de estos iconos:

- Archivo
 - Crear un nuevo proyecto
 - Abrir un proyecto existente
 - Guardar el proyecto actual
 - Importar un proyecto
 - Exportar un duende
 - Hacer anotaciones al proyecto
 - Salir
- Edición
 - Recuperar un objeto eliminado pulsando en **No eliminar**.
 - Ejecutar el programa en modo paso a paso, resaltando el bloque que se está ejecutando
 - Definir cómo funciona el modo paso a paso
 - Comprimir imágenes y sonidos
 - Mostrar los bloques de comando de motores Lego
 - Arrancar el servidor GPIO
- Compartir
 - Compartir del proyecto en línea en el sitio de Scratch
 - Acceso al sitio de Scratch
 - Host Mesh
 - Join Mesh
- Ayuda
 - Acceso a la ayuda en línea en el sitio de Scratch
 - Acceso a las pantallas de ayuda en el sitio de Scratch
 - Muestra la versión y la licencia de Scratch
- A la derecha del menú hay una sección de herramientas que contiene:
 - Un botón para duplicar elementos.
 - Unas tijeras para eliminar elementos.
 - Una herramienta para aumentar el tamaño de un duende.
 - Una herramienta para reducir el tamaño de un duende.

Los bloques se muestran en la parte izquierda de la interfaz. Por encima de la zona en la que aparecen los bloques, un menú gráfico permite seleccionar el tipo de bloque que se debe mostrar en la paleta. Los bloques tienen la gran ventaja para los principiantes de eliminar los errores de sintaxis. El montaje de los bloques está guiado por su forma durante la creación de los scripts. Las ubicaciones previstas en los bloques para recibir valores u operaciones terminan con un punto o por un redondel. No hay ningún riesgo de inversión. Para facilitar el aprendizaje de Scratch, este capítulo contiene una descripción de cada bloque.

En el centro de la pantalla, la parte superior corresponde al objeto seleccionado. Aparece la imagen del objeto y sus propiedades. Una línea azul indica en qué dirección se dirige el duende. Un clic con el ratón y un movimiento de este provocan la rotación del duende alrededor de su

centro. Tres botones a la izquierda del duende permiten seleccionar si este puede:

- pivotar
- solamente retroceder derecha-izquierda y a la inversa
- no pivotar

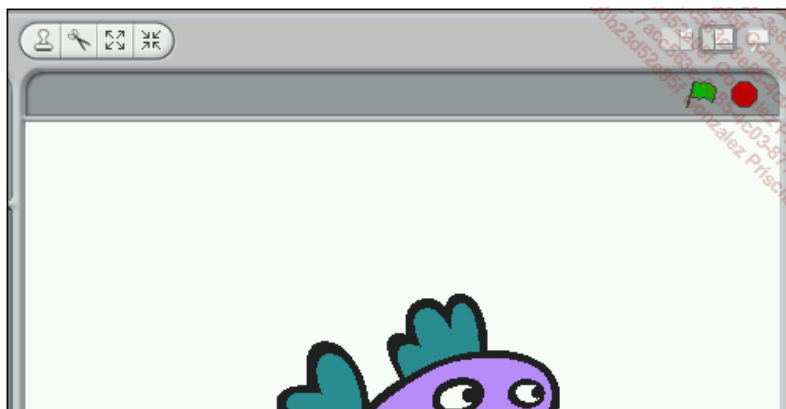
Tres pestañas muestran en la parte inferior de la ventana:

- Los scripts relacionados con el objeto: son las instrucciones que el duende ejecutará. Es aquí donde los scripts se desplazan y unen los bloques que se encuentran en la parte izquierda de la pantalla.
- Los disfraces del objeto: permiten dar diferentes apariencias a un duende (triste, alegre...) o animarlo (caminar, explosión...).
- Los sonidos del duende: se reproducirán en función de las instrucciones.



A la derecha de la pantalla está la escena, una zona de 480 píxeles de largo por 360 píxeles de alto. Por encima y en la parte derecha de la escena hay dos series de botones:

- Ajuste del tamaño de la escena
 - Pequeña escena: reduce el tamaño de la escena para dar prioridad a la zona de los scripts, lo que facilita el trabajo durante el desarrollo del programa.
 - Gran escena: la escena ocupa más espacio. El espacio disponible para el desarrollo es reducido. Si hay bloques que se salen de la zona de visualización, una barra de desplazamiento aparece en la parte inferior de la zona de los scripts.
 - Pantalla completa: es el modo de uso durante la ejecución de un programa ya terminado, para aprovechar totalmente el proyecto.
- Inicio y parada del programa:
 - La bandera verde arranca el programa.
 - El octógono rojo, que recuerda una señal de stop por su forma, detiene la ejecución de todos los scripts.



Debajo la escena un indicador muestra en píxeles las coordenadas X e Y del ratón respecto al centro de la escena.

La parte inferior derecha de la interfaz gráfica de Scratch recoge los objetos disponibles para realizar el programa.

La escena forma parte de estos objetos. Se le pueden adjuntar scripts y sonidos, pero no disponen de indumentaria, sino de planos que son la decoración en la que se desenvuelven los duendes.

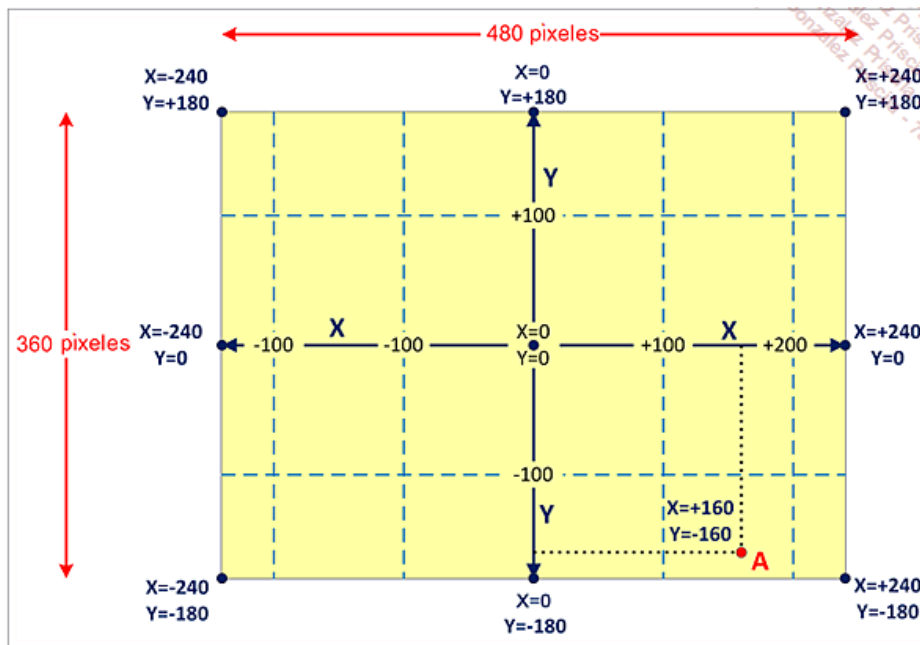
La zona de los duendes reúne los objetos implementados en el programa. Para depositar un objeto en la escena, es necesario pulsar sobre él con el botón izquierdo del ratón y llevarlo a la escena, dejando pulsado el botón del ratón. Cuando el botón se suelta, el objeto se ubica en la escena.

Por encima de la zona de los objetos disponibles hay tres iconos que permiten:

- crear un nuevo objeto con el editor gráfico de Scratch.
- importar un nuevo objeto almacenado en un directorio de la Raspberry Pi.
- obtener un objeto llamado "objeto sorpresa", que Scratch muestra de manera aleatoria.



Posicionar un duende en la escena



La escena en la que se desenvuelven los objetos o duendes es un rectángulo de 480 píxeles por 360 píxeles.

➤ Recuerde que la definición del color CGA en los inicios del PC era de 320 x 200 píxeles para toda la pantalla.

Cada píxel de la escena queda identificado por sus coordenadas orto-normales. El centro de la escena tiene la coordenada $X=0$ e $Y=0$. Es el punto donde se unen los ejes ortogonales, es decir, que forman entre ellos un ángulo de 90 grados. Todos están posicionados partiendo del centro.

- valores positivos hacia arriba y a la derecha.
- valores negativos hacia abajo y a la izquierda.

El tamaño es idéntico en ambos ejes. Cada unidad vale un píxel.

El eje X de ordenadas está graduado de -240 a la izquierda hasta +240 a la derecha.

El eje Y de abscisas está graduado de -180 en la parte inferior hasta +180 en la parte superior.

Es este sistema de representación en el plano el que permite posicionar con precisión los duendes en la escena y moverlos de un punto a otro.

Ejemplo de posicionamiento de un objeto en la escena

El punto A se define en la parte derecha inferior de la escena. Sus coordenadas en los ejes son $X=+160$ e $Y=-160$.

Si un duende se posiciona con $X=+160$ e $Y=-160$, su centro vendrá a ocupar el punto A.

Dirección del movimiento de un duende

Un duende se puede posicionar en la escena de manera absoluta, imponiendo las coordenadas en las que se debe ubicar el centro del objeto.

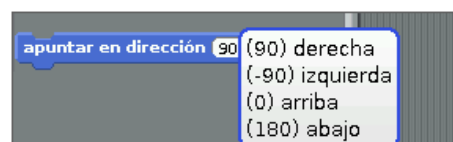
También puede moverse de manera relativa, es decir, respecto al lugar en el que encuentra.



Los bloques de movimiento anterior permiten:

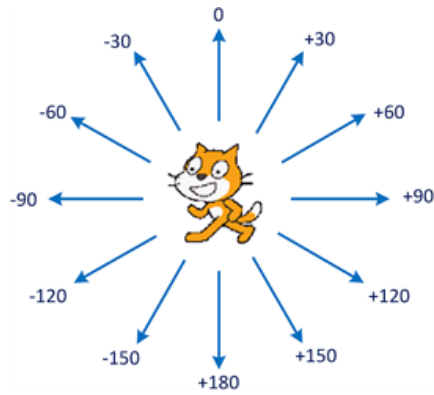
- El movimiento de 10 pasos del duende en la dirección a la que apunta. Pulsando en el valor 10, es posible cambiarlo por un número positivo o negativo.
- La rotación del duende de 15 grados en el sentido de las agujas del reloj. Entonces, apunta a una dirección diferente a la anterior a la ejecución del bloque. El valor se puede modificar haciendo clic en la cifra.
- La rotación del duende 15 grados en el sentido contrario a las agujas del reloj. Este valor se puede modificar.

Un bloque de movimiento permite orientar al duende en la escena, proporcionándole una dirección hacia la que girar.



Por tanto, el duende se puede dirigir en una de estas cuatro direcciones seleccionadas de la lista. Pulsando en la cifra es posible introducir manualmente un valor.

Es conveniente respetar las convenciones de localización de los ángulos previstas en la aplicación para evitar errores de funcionamiento de los scripts.



La dirección 0 grado apunta hacia arriba, +180 grados apunta hacia abajo. +90 grados indica la derecha de la escena y -90 grados indica la izquierda.

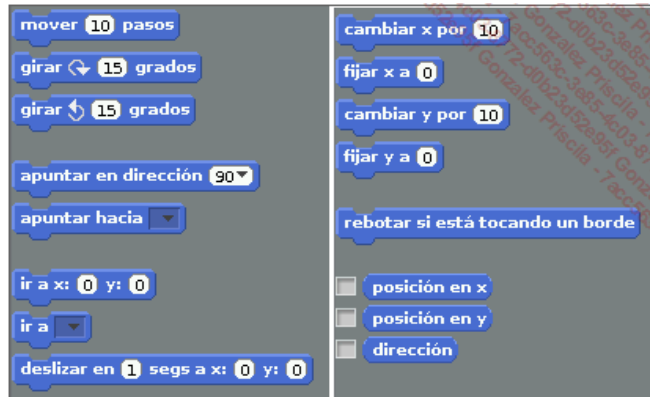
Por supuesto, son posibles todos los valores intermedios.

Para garantizar un funcionamiento correcto de los scripts, obligatoriamente el valor de los ángulos debe estar comprendido entre -179 grados y +180 grados. De hecho, utilizando valores de ángulo como +300 grados el funcionamiento de los scripts se puede ver afectado. El duende se dirigirá a la dirección -60 grados, pero será el valor 300 el que figure en el bloque. Esta diferencia puede provocar errores en la ejecución de algunos programas.

2. Los bloques de movimiento

Los duendes pueden avanzar y realizar una rotación. Los bloques de movimiento permiten darles las instrucciones correspondientes.

- Los bloques de movimiento, como todos los bloques utilizados en Scratch, disponen de una ayuda (en inglés) accesible haciendo clic con el botón derecho del ratón en el bloque y seleccionando después **Ayuda**.



Mover 10 pasos: el duende avanza en la dirección hacia la que está mirando. Para modificar el valor, basta con pulsar en él. Se aceptan valores positivos y negativos.

Girar 15 grados: estos dos bloques provocan la rotación del duende 15 grados en el sentido de las agujas del reloj o contrario a las agujas del reloj. El valor se puede modificar.

Apuntar en dirección 90: hace pivotar el duende para que esté frente a la dirección seleccionada. Por defecto, una lista desplegable ofrece arriba, abajo, derecha e izquierda. Es posible introducir manualmente un valor comprendido entre -179 y +180 grados. El objeto avanzará en esta dirección si se activa a continuación el bloque **Mover 10 pasos**.

Apuntar hacia: una flecha permite abrir una lista desplegable. Cuando solo hay un único duende en la zona de objetos disponibles, solo aparece en la lista el cursor del ratón. Este bloque va a obligar al duende a girar permanentemente hacia el cursor del ratón. Cuando hay varios duendes disponibles, podemos forzar a que un duende siempre se gire hacia otro.

Ir a x: 0 y: 0: obliga al duende a dirigirse a un lugar concreto de la escena. Con x=0 y y=0, el duende se posiciona en el centro de la escena. Este bloque es útil cuando es necesario posicionar un objeto en la escena, al comienzo de una animación por ejemplo. Si este bloque se usa durante una animación, el duende desaparece instantáneamente del lugar donde estaba, para volver a aparecer en las coordenadas fijadas por este bloque. Los dos valores son modificables.

Ir a: este bloque obliga al duende a dirigirse a un lugar seleccionado de la lista desplegable. Si se selecciona otro duende, el duende que ejecutará este bloque se situará en el otro duende. Si se selecciona el cursor de ratón, el duende seguirá fielmente el cursor del ratón.

Deslizar en 1 segs a x: 0 y: 0: el duende va a moverse, con un movimiento deslizante, desde el lugar donde está hasta las coordenadas fijadas por este bloque. Cuanto más largo sea el tiempo, más lento es el movimiento. El tiempo y las coordenadas son modificables.

Cambiar x por 10: se trata de una traducción aproximada. De hecho se aumenta el valor de x en un valor numérico, no se sustituye. Este bloque permite mover el duende horizontalmente sin modificar su altura en la escena.

Fijar x a 0: este bloque fuerza un nuevo valor de x. Este valor se puede modificar.

Cambiar y por 10: como en el caso de **Cambiar x**, la traducción es aproximada, y se va a aumentar en un valor numérico determinado. Este bloque permite mover el duende verticalmente sin modificar su posición horizontal en la escena.

Fijar y a 0: este bloque fuerza un nuevo valor de y. Este valor se puede modificar.

Rebotar si está tocando un borde: si el duende llega al borde de la escena, vuelve en sentido inverso, respetando la elección hecha en la zona de las propiedades del objeto, en medio de los tres botones que gestionan el pivoteo del duende.

Posición en x: muestra la posición x del duende en la escena.

Posición en y: muestra la posición y del duende en la escena.

Dirección: muestra en la escena la dirección a la que apunta el duende.

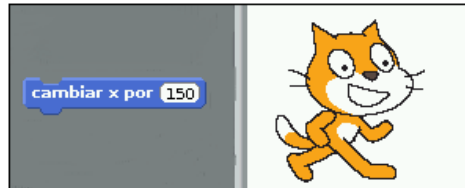
Estos tres últimos bloques se utilizan para la puesta a punto de los programas. Permiten controlar las coordenadas de un duende en función de sus movimientos. Es una ayuda considerable cuando es necesario localizar un funcionamiento incorrecto en un script. Una vez que el script funciona, basta con desmarcar las opciones para que desaparezcan estas indicaciones.

Comprobar un bloque de movimiento

Para probar un bloque de movimiento, seleccione un objeto de la lista de duendes disponibles. Cuando inicia Scratch, solo está disponible el gato, logotipo y mascota de Scratch.

Pulse en el gato en la parte de la escena de la pantalla para seleccionarlo. Sitúelo aproximadamente a izquierda de la escena, centrándolo verticalmente. Intente mostrar x:-150 e y:-25 (aproximadamente) en el indicador de posiciones x e y, situado bajo la escena. Suelte el ratón.

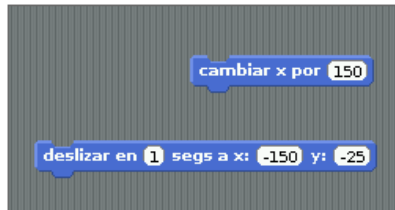
Pulse en el bloque **Cambiar x por 10**, bloque que permite añadir un valor a la posición x actual del gato. Deslice el bloque en la zona de script y suéltelo.



Modifique el valor numérico del bloque, sustitúyalo por 150 y después pulse en la parte coloreada de azul del bloque.

El gato se mueve repentinamente hacia la derecha 150 píxeles.

Pulse en el bloque **Deslizar en 1 segs a x: 0 y: 0** y llévelo a la zona de script sin encajarlo en el primer bloque. Los dos bloques deben permanecer separados.



Modifique los valores x:-150 e y:-25, y después pulse en una parte coloreada de azul del bloque.

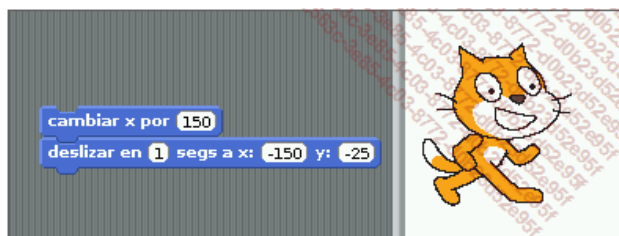
El gato que antes de desplazó vuelve a la posición x:-150, y:-25.

Pulse alternativamente en cada uno de los bloques. El gato repite estos movimientos.

Primer script

Un script es el conjunto de instrucciones separadas en un único bloque. Observe los dos bloques presentes en la zona de script. Presentan muescas que tienen la misma forma. Por tanto, se pueden encajar.

Deslice uno de los dos bloques hacia el otro, de manera que las muescas queden juntas. Cuando los dos bloques estén lo suficientemente cerca, aparece una zona clara entre los dos bloques. Suelte el ratón, los dos bloques se unen por sus muescas.



Pulse en la zona coloreada de azul de uno de los bloques. El conjunto de los dos bloques se resalta durante la ejecución.

Las dos instrucciones se ejecutan una después de la otra: el gato se mueve bruscamente a la derecha y después vuelve a su posición deslizándose.

Acaba de realizar su primer script. A continuación veremos cómo es posible ejecutar un script, además de pulsando encima de él con el ratón.

Eliminar una instrucción

Para eliminar una instrucción, en primer lugar es necesario separar las dos instrucciones que forman el script.

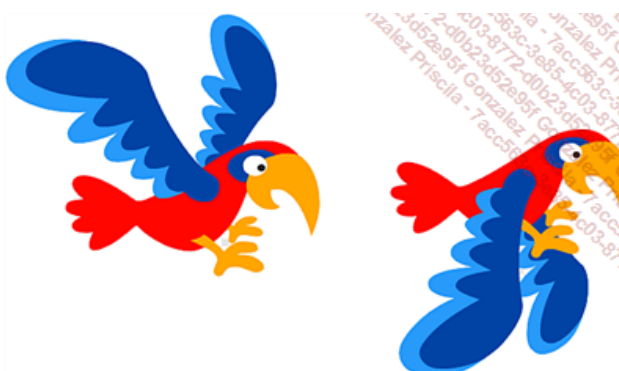
Pulse en el bloque inferior y muévelo hacia abajo.

Es la única manera de separar dos bloques. Si mueve el bloque superior hacia arriba, mueve los dos bloques juntos hacia arriba. Compruebe también este movimiento de bloque para ver la diferencia.

Cuando el bloque inferior se separe, llévelo a la zona donde se almacenan los bloques de movimiento y suelte el ratón. El bloque ha desaparecido de la zona de los scripts. También puede hacer clic con el botón derecho del ratón en el bloque y seleccionar **Eliminar** en el menú contextual.

3. Los bloques de apariencia

Algunos duendes no tienen indumentaria: sirven solo para la apariencia en pantalla. Las únicas modificaciones que aceptan son los movimientos y las rotaciones. Otros duendes pueden tener diferentes aspectos, es decir, pueden presentar formas diferentes. Un duende puede parecer que anda o vuela, cuando los aspectos se suceden en pantalla.



El loro anterior parece batir las alas cuando las dos imágenes correspondientes a sus dos aspectos se suceden durante su movimiento y crean una animación.

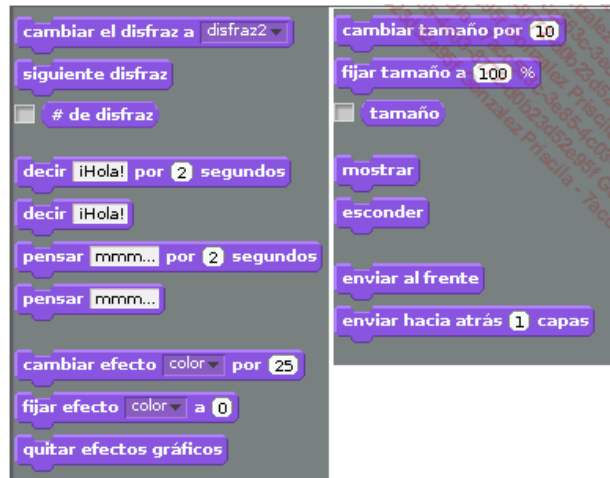
El tiburón que se presenta a continuación tiene tres disfraces que puede adoptar según las necesidades del script:

- **shark-a:** el tiburón triste busca su presa.
- **shark-b:** el tiburón ve una presa.
- **shark-c:** el tiburón atrapa la presa.



Los bloques de apariencia permiten seleccionar la apariencia en pantalla para el tiburón en cada momento, de ahí el nombre de aspecto.

Los bloques de apariencia van a modificar el disfraz del duende, permitiéndole hablar o pensar al estilo de los bocadillos que aparecen en los dibujos animados, modificar su apariencia con diferentes efectos, cambiar su tamaño, ocultarlo o hacer que aparezca, administra la superposición de los duendes cambiando el plano en el que se muestran, etc.



Cambiar el disfraz a [disfraz2]: este bloque permite seleccionar de manera absoluta el disfraz que se va a mostrar en pantalla. La lista desplegable muestra los disfraces disponibles. Si el duende solo dispone de un disfraz, la lista solo tendrá un elemento.

Siguiente disfraz: cuando un duende dispone de varios disfraces, se muestra el siguiente disfraz.

Nº de disfraz: muestra en la escena el número del disfraz actual adoptado por el duende. Este bloque se usa para ajustar los scripts.

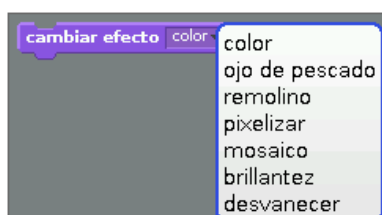
Decir [¡Hola!] por 2 segundos: el duende dice Hola en un bocadillo como el de los dibujos animados, durante un tiempo definido, en este caso dos segundos. La frase pronunciada y el tiempo de visualización son modificables, lo que permite jugar en las escenas con los diálogos.

Pensar [mmm...] por 2 segundos: el duende piensa mmm... en un bocadillo como el de los dibujos animados, durante un tiempo definido, en este caso dos segundos. El pensamiento y el tiempo de visualización son modificables.



Cambiar efecto [color] por 25: modifica los efectos gráficos aplicados al duende. La modificación por defecto es la del color. Una lista permite seleccionar entre varios efectos:

- **Color:** modifica los colores del duende.
- **Ojo de pescado:** aplica un efecto de dilatación circular.
- **Remolino:** crea una espiral a partir de los píxeles del duende.
- **Pixelizar:** solo conserva un determinado número de píxeles del duende.
- **Mosaico:** divide el duende en pequeñas copias de sí mismo. El número de copias depende del valor del efecto.
- **Brillantez:** modifica la luminosidad del duende, aumentando la intensidad de los colores.
- **Desvanecer:** estampa uniformemente los colores del duende.



Pruebe con diferentes valores (10, 20, 40) para visualizar los efectos que se generan. Un valor negativo reduce el efecto aplicado al duende. Observe la diferencia entre la luminosidad que se aplica a la intensidad de los colores y el efecto desvanecer, que estampa progresiva y uniformemente los colores del duende. Este último se utilizará mucho para hacer desaparecer o aparecer el duende lentamente.

Fijar efecto [color] a 0: si el valor permanece a 0, vuelve a 0 el efecto seleccionado en la lista anterior (y solo este). También permite fijar un valor para uno de los efectos gráficos, sustituyendo 0 por este valor.

Quitar efectos gráficos: anula todos los efectos gráficos aplicados al duende. Este vuelve su apariencia a la normalidad.

Cambiar tamaño por 10: modifica el tamaño del duende en incrementos del 10 %. El valor se puede modificar y puede ser negativo para reducir el tamaño del duende en pantalla. Esta modificación permite sugerir el enfoque o la distancia de un duende.

Fijar tamaño a 100 %: recuperar el valor inicial para el tamaño del duende.

Tamaño: muestra en la escena el tamaño actual del duende. Este bloque se usa para ajustar los scripts.

Mostrar: cuando este bloque está en funcionamiento durante la ejecución de un script, el duende se hace visible.

Esconder: cuando este bloque está en funcionamiento durante la ejecución de un script, el duende desaparece. Atención: a no ser que se haya ejecutado un bloque **Mostrar**, el duende permanecerá invisible.

Enviar al frente: cuando varios duendes se unen, permite traer a un duende a primer plano para que no quede oculto por el resto de duendes.

Enviar hacia atrás 1 capas: gestión del orden en el que los duendes se superponen. La modificación del valor numérico permite administrar la superposición de los duendes.

4. Las estructuras de control

Para controlar el desarrollo de un script desde su inicio hasta su finalización, Scratch pone a disposición del programador un conjunto de bloques. Estos bloques, llamados bloques de control, permiten administrar la ejecución de las acciones, su repetición o incluso ejecutar un conjunto de bloques si una condición es verdadera.



Al presionar la bandera: los bloques conectados a este control se ejecutan cuando se pulsa la bandera verde, situada en la parte superior derecha de la escena.

Al presionar tecla [espacio]: los bloques conectados a este control se ejecutan cuando se pulsa la barra espaciadora. Es posible seleccionar otra tecla de la lista desplegable en su lugar. Las flechas de dirección figuran en la lista y permiten el "control" de un duende pulsando estas teclas. Pero todas las teclas alfanuméricas del teclado están disponibles.

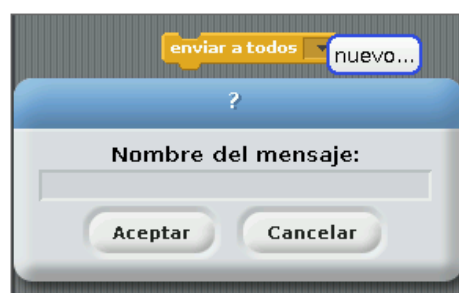
Al presionar Objeto1: se ejecutan los bloques ubicados bajo este control si se hace clic con el botón izquierdo del ratón en el duende Objeto1. Cuando un duende se selecciona, este control muestra automáticamente su nombre.

Esperar 1 segundos: cuando el script llega a este bloque, espera durante un segundo. Es posible modificar el tiempo de espera cambiando el valor de la cifra (en segundos).

Por siempre: los bloques insertados en la agrupación formada por este control se repetirán indefinidamente cuando el bloque se activa. Solo un clic en el icono rojo de stop detendrá el script.

Repetir 10: los bloques insertados en la agrupación formada por este control se repetirán 10 veces cuando el bloque se active. El número de repeticiones de los bloques es configurable modificando el valor numérico.

Enviar a todos: envía un mensaje al resto de duendes, así como a la escena. La flecha de dirección hacia abajo abre una lista si ya existe un mensaje. En caso contrario, permite crear uno nuevo. Haciendo clic en **nuevo** se abre una ventana.



Escriba un nombre de mensaje en la zona de texto de esta ventana y después pulse en **Aceptar**. El mensaje se enviará al resto de duendes y a la escena cuando el bloque se active. Esto permite, por ejemplo, administrar los duendes desde un lugar único. La escena también puede enviar mensajes. Es una manera de agrupar en un único lugar la gestión de los duendes. Por ejemplo, la escena envía un mensaje tras un cambio en su imagen de fondo. Este mensaje posiciona los duendes que intervienen en este decorado y lanza sus acciones.

Enviar a todos y esperar: envía un mensaje al resto de duendes, así como a la escena. Funciona como **enviar a todos**. Cuando se envía el mensaje, se espera a que todas las acciones lanzadas por el mensaje terminen.

Al recibir: los bloques conectados a este control se ejecutan al recibir un mensaje seleccionado de la lista desplegable. Esto permite sincronizar las acciones ejecutadas por los duendes.

Por siempre si <condición>: repite indefinidamente los bloques insertados en la agrupación si la condición presente después del **si** es verdadera. La ubicación en la que se recibe la condición termina en la parte derecha e izquierda por una figura puntiaguda. Esta ubicación solo se puede ocupar por un bloque de la misma forma. Esto evita las sintaxis erróneas u otro tipo de errores.



Si <condición>: ejecuta los bloques insertados en la agrupación si la condición es verdadera.

Si <condición> ... si no ...: si la condición es verdadera, ejecuta los bloques insertados en la agrupación del **si**. En caso de que esta condición no sea verdadera, ejecuta los bloques insertados en la agrupación del **si no**.

Esperar hasta que <condición>: el script marca una pausa hasta que la condición sea verdadera. Esto permite, por ejemplo, esperar la pulsación en una tecla para continuar la ejecución del programa.

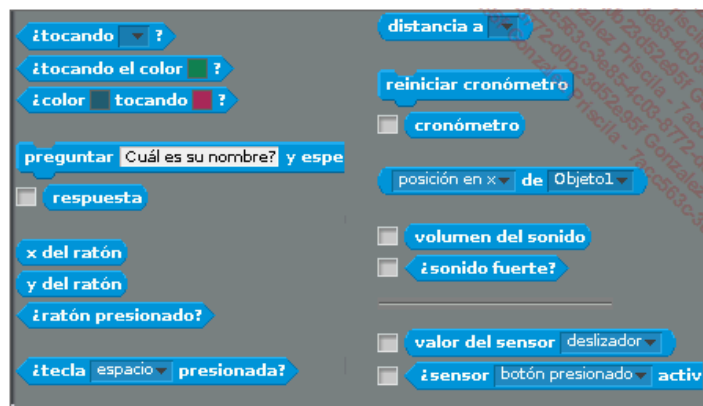
Repetir hasta que <condición>: repite los bloques insertados en la agrupación hasta que la condición sea verdadera. Cuando la condición es verdadera, detiene la ejecución.

Detener programa: cuando se alcanza este bloque, la ejecución del script que contiene el bloque se detiene.

Detener todo: todos los scripts se paran, este bloque equivale a una pulsación en el botón rojo situado en la parte superior de la escena.

5. Los sensores

Los bloques de control de la sección anterior toman decisiones durante la ejecución de los scripts. Estas decisiones derivan del valor de una condición. Los sensores proporcionan la información para que el bloque de control pueda decidir la marcha que se debe seguir. Los sensores con los extremos redondeados devuelven un valor, mientras que los sensores con los extremos puntiagudos devuelven verdadero o falso según el resultado de la verificación indicada por un signo de interrogación.



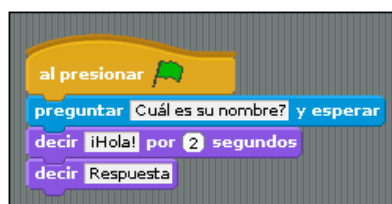
¿Tocando?: esta condición se convierte en verdadera si el duende toca uno de los objetos seleccionados en la lista desplegable. Los objetos presentes en la lista son: el puntero del ratón, el borde de la escena o los demás duendes.

¿Tocando el color []?: esta condición se convierte en verdadera cuando el duende toca un objeto del color indicado. Para cambiar el color, es necesario pulsar en el cuadrado de color. El cursor se transforma en un cuentagotas, con el que basta con pulsar sobre el color del objeto presente en la escena. Entonces el cuadrado toma este color, que será el desencadenante de la condición cuando duende lo toque.

¿Color [] tocando []?: esta condición se convierte en verdadera cuando el primer color toca el segundo color. La elección de los colores se hace como se ha explicado anteriormente, pulsando en el cuadrado de color y después seleccionando cada color con el cuentagotas. Esta condición permite tener más precisión que la anterior. De hecho, esta vez no es todo el duende el que desencadena la condición, sino una pequeña parte del duende, cuyo color sea el seleccionado.

Preguntar [su nombre] y esperar: este bloque hace que el duende hable, como el bloque de apariencia **decir**, pero muestra una zona de texto en la escena y espera a que el usuario responda y valide su respuesta para continuar el desarrollo del script.

Respuesta: un clic en la opción situada a la izquierda de **respuesta** muestra una zona en la escena en la que aparece la respuesta escrita por el usuario. Esta función se puede utilizar durante la puesta a punto de los scripts. También se puede utilizar el sensor de respuesta, como en el siguiente ejemplo:



Esta vez, el sensor **respuesta** se ha deslizado en el bloque de apariencia **decir**. Cuando se pulsa la bandera verde, el duende pregunta al usuario, solicitándole su nombre y esperando su respuesta (por ejemplo Ángel). Después de la validación de la respuesta, dice *hola* durante 2 segundos y a continuación muestra en un bocadillo el texto de la respuesta: *Ángel*.

X del ratón: este sensor devuelve la coordenada x del ratón. Es posible usarla para desencadenar las acciones cuando el ratón pasa por encima de una coordenada particular de la escena.

Y del ratón: este sensor devuelve la coordenada y del ratón.

¿Ratón presionado?: este sensor devuelve **verdadero** si se pulsa el botón del ratón en cualquier parte de la pantalla.

¿Tecla [Espacio] presionada?: si se pulsa la tecla [Espacio], el sensor devuelve verdadero. La lista desplegable permite definir qué tecla se comparará. Están disponibles todas las teclas alfanuméricas y numéricas, así como las cuatro flechas de dirección.

Distancia a []: este sensor devuelve la distancia entre el duende y el puntero del ratón o alguno de los demás duendes presentes en la escena.

Reiniciar cronómetro: pone el cronómetro a cero.

Cronómetro: este sensor devuelve el valor actual del cronómetro. Comprobando este valor en una estructura de control es posible, por

ejemplo, limitar la duración de un juego o aumentar su dificultad a lo largo de una determinada franja de tiempo. Si se marca la opción, el valor del cronómetro se muestra en la escena.

[Posición en x] de [Objeto1]: este sensor devuelve la información de las propiedades de un objeto. La lista permite seleccionar entre los valores posición *x*, posición *y*, dirección, aspecto nº, tamaño y volumen. Entre los objetos también figura la escena, por lo que podemos recuperar el número mostrado en segundo plano y el volumen.

Volumen del sonido: este sensor devuelve el valor del nivel de volumen de un sonido captado por el micrófono (en un PC). El valor está comprendido entre 1 y 100. Para que este sensor devuelva un valor, el ordenador debe estar provisto de un micrófono que funcione. Si la opción está marcada, se muestra en la escena el valor del volumen del sonido recibido por el altavoz, lo que puede ser útil para ajustar el programa.

¿Sonido fuerte?: este sensor devuelve **verdadero** si el micrófono recibe un sonido fuerte. Si no hay ruido o si el nivel sonoro es insuficiente, devuelve **falso**. A diferencia del sensor volumen de sonido, que indica el nivel del sonido, este indica únicamente la presencia de un sonido fuerte. Para que este sensor devuelva un valor, el ordenador debe estar provisto de un micrófono que funcione.

Valor del sensor "deslizador": este bloque envía al programa el valor de un sensor analógico o digital, conectado a la GPIO o a una tarjeta de adquisición (Scratch Sensor, Lego, PiFace, etc...). Consulte los capítulos La GPIO de la Raspberry Pi y Los periféricos, para acceder a ejemplos de uso.

¿Sensor "botón presionado" activado?: detecta la pulsación en un botón de la tarjeta Scratch Sensor.

6. Los sonidos

Scratch ofrece la posibilidad de jugar con los sonidos. Pueden ser propios de un duende (maullido de un gato, ladrido de un perro, etc.) o desencadenarse durante una acción (choque de dos duendes, partida perdida, etc.). Algunos duendes tienen uno o varios sonidos asociados. Otros no disponen de esta posibilidad. La pestaña **Sonidos** en la zona central de la pantalla muestra las propiedades de los sonidos asociados a un duende, si estos sonidos existen.

El usuario tiene la posibilidad de registrar un sonido desde el micrófono del ordenador o importar un sonido (.wav únicamente). La importación se hace pulsando el botón **Importar**. Hay disponible una librería de sonidos variados.

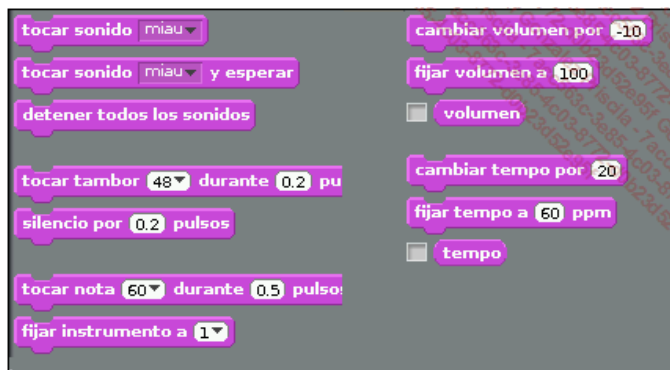


El sonido se puede seleccionar de entre los que hay disponibles en la carpeta **Sonidos** o bien descargarlos y después importarlos desde una carpeta cualquiera del ordenador a la carpeta personal del usuario *pi* o al escritorio.



La prueba de un sonido se realiza pulsando en el botón de reproducción o en el altavoz. Es necesario que el sonido esté operativo tanto en una pantalla HDMI como en unos cascos o un altavoz amplificado conectados a la salida de audio analógica.

Los siguientes bloques permiten intervenir sobre los sonidos de un duende.



Tocar sonido [miau]: el duende seleccionado es el gato. El sonido asociado al gato se llama **miau**. Cuando el script pasa por este bloque, desencadena la lectura del sonido **miau** y después el script pasa inmediatamente al siguiente bloque.

Tocar sonido [miau] y esperar: a diferencia del bloque anterior, el script se detiene y espera a que el sonido se haya leído completamente antes de pasar al siguiente bloque.

Detener todos los sonidos: al pasar por este bloque, todos los sonidos que se están ejecutando se detienen. Por ejemplo, este bloque se puede utilizar al final de una partida para detener todos los sonidos antes de reproducir la música de fin.

Tocar tambor [48] durante 0.2 pulsos: el sonido se reproduce durante 0.2 pulsos. Se trata de un sonido de percusión. Desafortunadamente, los diferentes instrumentos ofrecidos en la lista se definen para un sintetizador MIDI que existe en las tarjetas de sonido de un PC, pero no en la Raspberry Pi. Por tanto, Scratch opta por una solución de seguridad y únicamente se emite un ruido de percusión corto, independientemente de los ajustes seleccionados. Sin embargo, la nota se respeta.

Silencio por 0.2 pulsos: cuando se están reproduciendo varias notas, este bloque introduce una pausa de duración igual a 0.2 pulsos.

Tocar nota [60] durante 0.5 pulsos: cuando se pasa por este bloque, se reproduce la nota seleccionada durante la duración indicada.

Fijar instrumento a [1]: permite seleccionar el instrumento que va a reproducir las notas. No funciona en la Raspberry Pi porque falta un sintetizador MIDI. Todas las notas se reproducen de la misma manera.

Cambiar volumen por -10: este bloque cambia el volumen de los sonidos reproducidos o leídos. Es posible aumentar o reducir el volumen. Los valores límite son 0 y 100.

Fijar volumen a 100: este bloque provoca que el volumen pase al valor máximo.

Volumen: permite recuperar el valor del volumen para usarlo en los scripts. Si la opción está marcada, el valor del volumen se muestra en la escena.

Cambiar tempo por 20: este bloque modifica el tempo, es decir la velocidad a la que Scratch reproduce las notas. El tempo se expresa en ppm: pulsos por minuto. Por tanto, un pulso (o tiempo) dura 60 segundos/tempo en ppm. Una nota reproducida durante 0,5 pulsos durará 0,5 segundos. La modificación del tempo cambia la duración de la nota:

Tempo	Duración de la nota
60 ppm	0,5 s
180 ppm	0,16 s

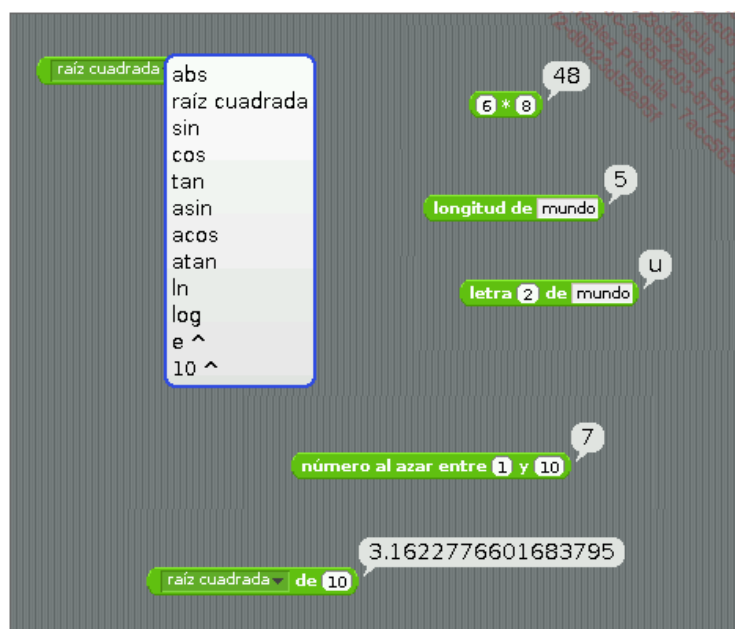
Fijar tempo a 60 ppm: este bloque permite cambiar el tempo, es decir, la velocidad a la que Scratch va a reproducir las notas.

Tempo: este bloque permite recuperar el valor del tempo. Si la opción está marcada, el valor del tempo se muestra en la escena.

7. Los operadores

En todo script, como en cualquier programa informático, llega un momento donde el uso de operadores se convierte en obligatorio. Los operadores aritméticos devuelven un valor numérico positivo o negativo. Tienen extremos redondeados. Los operadores lógicos devuelvan verdadero o falso. Sus extremos son en forma de ángulo. La forma de los operadores hace que solo se puedan insertar en una zona con la forma correspondiente en otro bloque.

a. Operadores aritméticos



Los operadores aritméticos (extremos redondeados) devuelven el resultado de una operación. Pueden recibir un único argumento, como la raíz o el seno, o dos argumentos como la multiplicación. Basta con rellenar la o las zonas del operador y pulsar en el bloque para que aparezca el resultado. Cuando se está ejecutando un script, la operación se realiza y su resultado se envía al bloque en el que se encuentre el operador.

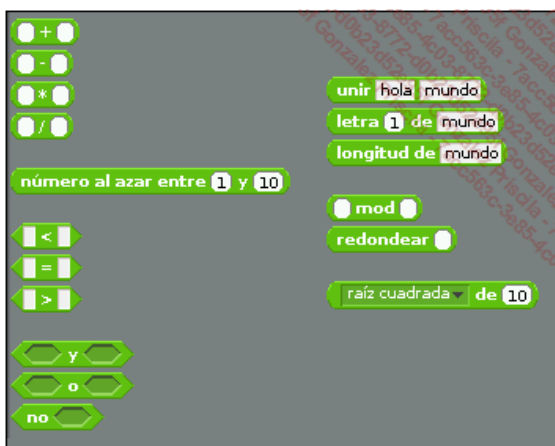
Los operadores aritméticos solo aceptan cifras en las zonas de introducción de datos.

b. Operadores lógicos



Los operadores lógicos (extremos con forma de ángulo) reciben dos argumentos, salvo el operador **no**. Realizan la operación solicitada y devuelven un resultado: verdadero o falso.

La siguiente captura de pantalla agrupa los bloques aritméticos (+, -, *, /, generador de número aleatorio, módulo, redondeo y raíz), los bloques lógicos (no, y, o) así como los bloques que operan con cadenas de caracteres.



Operaciones aritméticas: las cuatro operaciones están disponibles (+ - * /). Necesitan dos argumentos que pueden ser valores fijos introducidos durante la creación del script o un valor proporcionado por un sensor que se deslizará a la zona de argumento correspondiente.

Número al azar entre 1 y 10: el operador devuelve un número entero seleccionado aleatoriamente entre dos límites modificables, que forman parte de los números que se pueden extraer.

Comparaciones: los operadores de comparación menor que (<), igual (=) y mayor que (>) realizan la comparación entre los dos valores presentes en las zonas derecha e izquierda del signo. Devuelven verdadero o falso, según sea el resultado de la operación.

Los operadores de comparación aceptan letras y cifras en las zonas de introducción de datos. En el caso de las letras, su código ASCII se usa para la comparación. Por tanto, hay que respetar las mayúsculas y minúsculas. Los operadores de comparación pueden trabajar con cadenas de caracteres. <valor = valor> devolverá *verdadero*, lo que permite, por ejemplo, comprobar un valor introducido por el teclado para verificar una respuesta.

Operaciones lógicas: las operaciones lógicas permiten combinar dos valores. El operador **y** devuelve *verdadero* solo si los dos valores presentes en las zonas derecha e izquierda del **y** son verdaderos. Si uno de los dos valores es falso, devuelve *falso*. También si los dos valores son falsos, devuelve *falso*.

El operador **o** devuelve *verdadero* si uno de los dos valores es verdadero. También devuelve *verdadero* si los dos valores son verdaderos.

El operador **no** devuelve el inverso del valor. Por tanto, devuelve *verdadero* si el valor es *falso* y *falso* si el valor es *verdadero*.

Las siguientes tablas, también llamadas "tablas de verdad", representan todos los casos posibles para los operadores **y**, **o** y **no**.

Y		
valor 1	valor 2	resultado
FALSO	FALSO	FALSO
FALSO	VERDADERO	FALSO
VERDADERO	FALSO	FALSO
VERDADERO	VERDADERO	VERDADERO

O		
valor 1	valor 2	resultado
FALSO	FALSO	FALSO
FALSO	VERDADERO	VERDADERO
VERDADERO	FALSO	VERDADERO
VERDADERO	VERDADERO	VERDADERO

NO	
valor 1	resultado
FALSO	VERDADERO
VERDADERO	FALSO

Unir [hola] [mundo]: este operador realiza una operación llamada concatenación de las cadenas de caracteres, presentes en las zonas de introducción de datos. La concatenación consiste en crear una sola cadena de caracteres a partir de varias cadenas, poniéndolas una detrás de otra.

Esta operación no añade espacios entre las cadenas, sino que se deberá prever durante la escritura del script.

Las posibilidades pueden parecer limitadas con solo las dos zonas pudiendo recibir las cadenas de caracteres. Sin embargo, es posible ubicar en cada una de las zonas otra **agrupación** y aumentar de esta manera las posibilidades de este operador. Una zona puede recibir una palabra, frase o un sensor. En el siguiente ejemplo, una primera **agrupación** crea la cadena *hola a+todo*, una segunda la cadena *el +mundo* y, para terminar, la **agrupación** que recibe las dos anteriores crea la cadena final: *hola a todo +el mundo*.



Letra 1 de [mundo]: este operador devuelve la *n*-ésima letra de una cadena de caracteres. Si el número es superior a la longitud de la cadena, no se indica ningún error, pero no se devuelve ninguna letra.

Longitud de [mundo]: devuelve la longitud de la cadena, espacios incluidos.

Mod: este operador devuelve el resto de la división entera. Por ejemplo, para 110 módulo 12, la respuesta será 2. En efecto 110/12 = 9 y

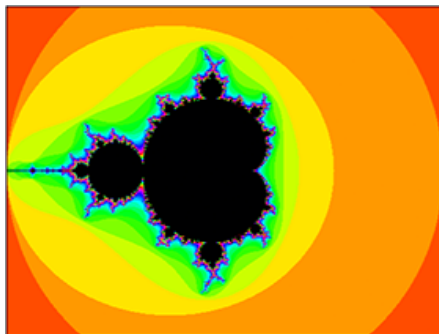
restan 2.

Redondear: este operador redondea el valor que recibe como argumento al entero más cercano. Por ejemplo 3,4 dará 3 mientras que 3,5 o 3,7 darán 4.

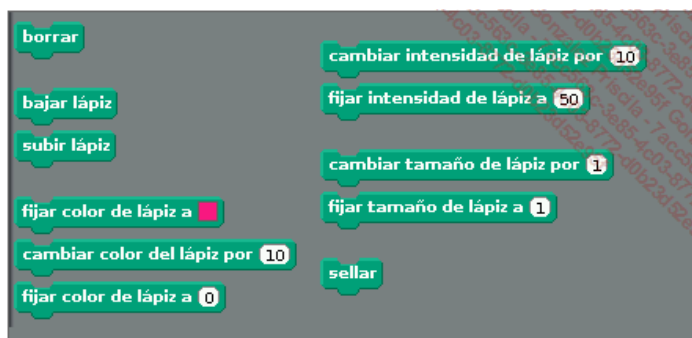
[Raíz cuadrada] de: este operador permite seleccionar de una lista desplegable la operación a realizar. Es posible elegir entre valor absoluto (abs), raíz cuadrada, seno (sin), coseno (cos), tangente (tan), arcoseno (asin), arcocoseno (acos), arcotangente (atan), logaritmo neperiano (ln), logaritmo decimal (log), exponencial (e[^]) y potencia de 10 (10[^]). Da acceso a los cálculos matemáticos en los scripts.

8. El lápiz

Los duendes pueden estar provistos de un lápiz que va a dejar una huella por el lugar donde pasan los duendes. El color y el tamaño del lápiz se definen de manera fija. Se pueden modificar por el programa en tiempo de ejecución del script.



Conjugando la posición y el color del lápiz, es posible realizar diseños complicados como este fractal de Mandelbrot (Mandelbrot Set Fractal remix - Pete D. - <http://scratch.mit.edu/projects/11453062/>).

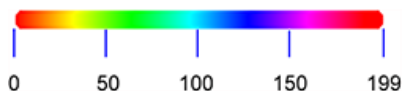


Borrar: elimina todas las trazas dejadas por el lápiz de todos los duendes en la escena.

Bajar lápiz: a partir de este bloque, el lápiz comienza a dejar una traza en la escena.

Subir lápiz: a partir de este bloque, el lápiz no deja más traza en la escena.

Fijar color de lápiz a : fija el color del lápiz. El color es un número entre 0 y 199. La paleta de colores disponibles va del rojo (0) al rojo (199).



Los colores están en el orden de los colores del arco iris (rojo, naranja, amarillo, verde...). 0 corresponde al rojo, 70 al verde, 130 al azul, 170 al magenta, etc.

Cambiar color del lápiz por 10: cambia el color del lápiz en el valor indicado (positivo o negativo).

Fijar color de lápiz a 0: fija el color del lápiz a un valor dado.

Cambiar intensidad de lápiz por 10: modifica la intensidad del color. El color no se modifica, sino que cambia su luminosidad. La intensidad puede variar de 0 a 100. Con el valor cero, el color se convierte en negro, con 50 el color es normal y con 100 el color se convierte en blanco.

Fijar intensidad de lápiz a 50: fija la intensidad del lápiz a un valor predefinido.

Cambiar tamaño de lápiz por 1: aumenta el tamaño del lápiz en 1 píxel.

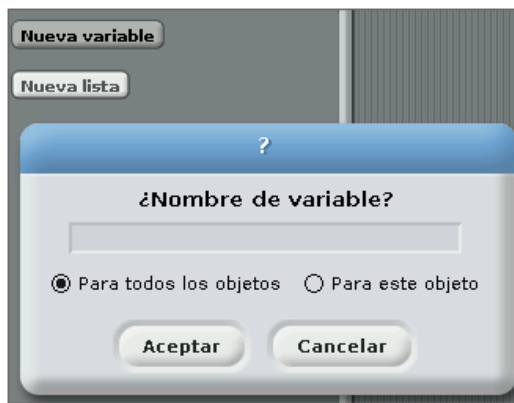
Fijar tamaño de lápiz a 1: fija el tamaño del lápiz a un valor predeterminado. Aquí el lápiz tendrá una longitud de 1 píxel.

Sellar: deja una copia del aspecto del duende en el lugar donde este bloque se ejecuta en el script.

9. Las variables

Además de los bloques presentes originalmente en el juego de instrucciones de Scratch, el programador puede necesitar crear sus propias variables. En un juego, por ejemplo, puede ser necesario un contador: un duende debe evitar los objetos; cada vez que toca uno, el contador se incrementa. Después de tocar cinco objetos, el juego termina. El contador no existe en los bloques disponibles. Es el programador el que debe crearlo.

También es posible crear una lista. Una lista está formada de un conjunto de palabras. Puede servir para un duende que debe recoger objetos. Cada vez que recoge un objeto, este desaparece de la pantalla y su nombre se añade a la lista.



La creación de una variable o de una lista se hace pulsando en **Nueva variable** o en **Nueva lista**. Las ventanas de creación de una variable o de una lista son idénticas. Hay que introducir el nombre de la variable o el de la lista e indicar si esta será accesible por todos los duendes (llamada global) o únicamente por el duende para la que se crea (llamada local).

a. Creación de una variable



Durante la creación de una nueva variable (aquí la variable **contador**), aparecen bloques adicionales en la paleta de bloques.

Contador: es la variable creada. Este bloque permite usarla en otros bloques para comprobar su valor con los operadores de comparación, por ejemplo. Si la opción está marcada, la variable se muestra en la escena.

Fijar contador a 0: fija un valor a la variable. Aquí **contador** se inicializa a cero.

Cambiar contador por 1: aumenta el valor de la variable en la cantidad indicada. Aquí el valor de **contador** se aumenta en 1. También se dice que el contador se incrementa.

Mostrar variable contador: al pasar por este bloque, la variable se muestra en la escena y permanece visible mientras que el bloque **esconder variable contador** no se ejecute. La opción que hay delante de la variable **contador** se marca automáticamente cuando este bloque se ejecuta.

Esconder variable contador: durante la ejecución de este bloque, la variable contador desaparece de la pantalla. La opción ubicada delante de la variable contador se desmarca automáticamente cuando se ejecuta este bloque.

Atención: la variable permanece oculta mientras no se ejecute el bloque **mostrar variable contador**. Esto puede provocar un funcionamiento que puede parecer incorrecto cuando se prueban los programas. Es mejor tener la costumbre de indicar explícitamente en los scripts los momentos donde una variable se debe mostrar.

Por ejemplo, al comienzo de un juego, si el contador debe aparecer en la escena, es necesario insertar un bloque **mostrar variable contador** para forzar esta visualización. En caso contrario, en algunos casos donde la variable se ha ocultado anteriormente no aparecerá cuando se inicie el juego.

b. Creación de una lista

Igual que sucedía con la creación de una variable, la creación de una lista (aquí **cesta**) implica la aparición de nuevos bloques.



Cesta: este bloque devuelve la lista que tiene ese nombre. Si la opción delante del bloque está marcada, aparece en la escena.



Aquí la lista cesta contiene los objetos: *estrella*, *pedra* y *flor*. La longitud de la lista se indica en la parte inferior de la ventana.

Añade cosa a cesta: añade un ítem (elemento de una lista) a la cesta. El nombre del ítem (**cosa**) se puede modificar, bien indicando otro nombre o utilizando un bloque que devuelva el nombre del ítem que se debe añadir a la lista. Si se crean varias listas, estas aparecen en la lista desplegable con la condición de que sean accesibles por el duende. Por tanto, solo aparecerán sus listas locales y las listas globales accesibles por todos los duendes. Las listas locales propias de los demás duendes no se muestran en la lista desplegable.

Borrar [1] de cesta: elimina uno de los ítems (**cosa**) de la lista. La ortografía de las dos palabras debe ser exactamente la misma. Si hay ítems por debajo de este en la lista, se recolocan para llenar el vacío creado.

Insertar cosa en [1] de cesta: inserta un ítem (**cosa**) en la posición indicada. El resto de ítems, si existen, se enviarán a la parte inferior de la lista.

Reemplazar objeto [1] de cesta: sustituye el primer elemento de la lista por **cosa**. El número del ítem, así como su nombre, se pueden modificar.

Ítem [1] de cesta: devuelve el nombre del primer ítem de la lista. El número del ítem se puede modificar.

Longitud de cesta: devuelve el número de ítems contenidos en la lista.

Cesta contiene cosa: verifica si la lista **cesta** contiene un ítem **cosa**. Devuelve *verdadero* si la lista contiene este ítem, devuelve *falso* si no la contiene. Atención, la ortografía debe ser exactamente la misma.

Crear un juego sencillo en Scratch

1. Definir el escenario

Provisto de las herramientas disponibles en Scratch, es posible crear escenas, animaciones o juegos.

Para crear un juego, en primer lugar es necesario definir un escenario que va a servir de guía durante la realización.

Proponemos realizar el siguiente juego:

El fotógrafo

Escenario: un fotógrafo está a orillas de un estanque. Los duendes van a pasar por la pantalla. El fotógrafo deber tomar fotos de los duendes, llevando el cursor del ratón al duende antes de pulsar. Los duendes pasarán 10 veces. Al final del juego, el gato aparece y anuncia el resultado.

Desarrollo: cuando el jugador pulse en la bandera verde, se muestra una pantalla de presentación. Después de un tiempo de espera para permitir la lectura, el pingüino aparece e informa que es necesario pulsar en la barra [Espacio] para jugar.

El jugador pulsa la barra [Espacio]. Aparece la escena de juego, así como el visor del aparato de fotos que sigue al ratón. Los duendes aparecen aleatoriamente y pasan por la pantalla. El jugador intenta hacerles una foto. Después del décimo paso, el juego se detiene. El gato aparece en la escena y anuncia el resultado. Informa al jugador de que, para volver a jugar debe pulsar en la barra [Espacio].

2. Definir las interacciones

Escena

La escena será el elemento central del juego. La escena es un duende particular, aunque acepta scripts como el resto de duendes. Como es el elemento central del juego, será el encargado de la gestión de las secuencias (presentación, juego, fin del juego) y del control de los duendes.

Cuando se pulsa la bandera verde, la escena muestra el segundo plano de la presentación. Pasados algunos segundos, envía un mensaje al pingüino para que aparezca en la escena.

El pingüino recibe el mensaje, aparece en la escena y anuncia al jugador que debe pulsar la barra [Espacio] para jugar, espera algunos segundos y desaparece.

Inicio del juego

Cuando el jugador pulsa en la barra [Espacio], la escena comienza inicializando las variables que se utilizarán durante el juego, en particular el contador de puntos (puntos), el número de intentos (intento) y una variable que indica que un duende está en la escena (juegoEnCurso).

La escena extrae una cifra que define el duende que debe aparecer. Envía a todos los duendes un mensaje con su valor top inicial y después espera para hacer un nuevo dibujo a que el duende haya terminado su recorrido. Solo el duende cuyo número se corresponde con el extraído inicia su camino. Configura la variable juegoEnCurso a 1 al inicio de su recorrido para indicar que está en la escena. Durante su trayecto, comprueba si el jugador ha pulsado el ratón y, en este caso, comprueba si la distancia del cursor es inferior a 25 píxeles. Si es el caso, el contador de puntos se incrementa. Cuando ha terminado su recorrido, pone a cero la variable que indica su presencia en la escena y desaparece.

Para complicar un poco el juego, uno de los duendes tendrá una velocidad de marcha que acelerará, mientras que el otro tendrá una velocidad de marcha constante.

La escena, que estaba en espera de liberación de la variable, incrementa la variable intento y después extrae un nuevo número aleatorio, hasta completarse los 10 números. Cuando los 10 duendes han pasado por la escena, envía a todos los duendes el mensaje que indica que el juego ha terminado.

Fin del juego

Cuando recibe este mensaje, el gato avanza por la escena y anuncia el marcador. Después informa al jugador de que debe pulsar la barra [Espacio] para volver a jugar.

3. Preparar la escena

Para preparar la escena son necesarias dos imágenes de 480 x 360 píxeles. Una será la imagen de presentación del juego y la otra será la escena utilizada durante el juego.

Pulse en la escena en la zona de los duendes, y después en la pestaña **Fondo**. Pulse en el botón **Importar** y seleccione la imagen a cargar. Repita la operación para la presentación y para la escena del juego. Elimine a continuación la escena vacía presentada al inicio.



➤ Las imágenes de la escena y de presentación del juego están disponibles para su descarga desde la página Información.

4. Diseñar los duendes

Para el juego, por defecto se presenta el gato en la apertura del proyecto. Hay que crear tres nuevos duendes:

- el visor del aparato de fotos.
- el logo de la Raspberry Pi.

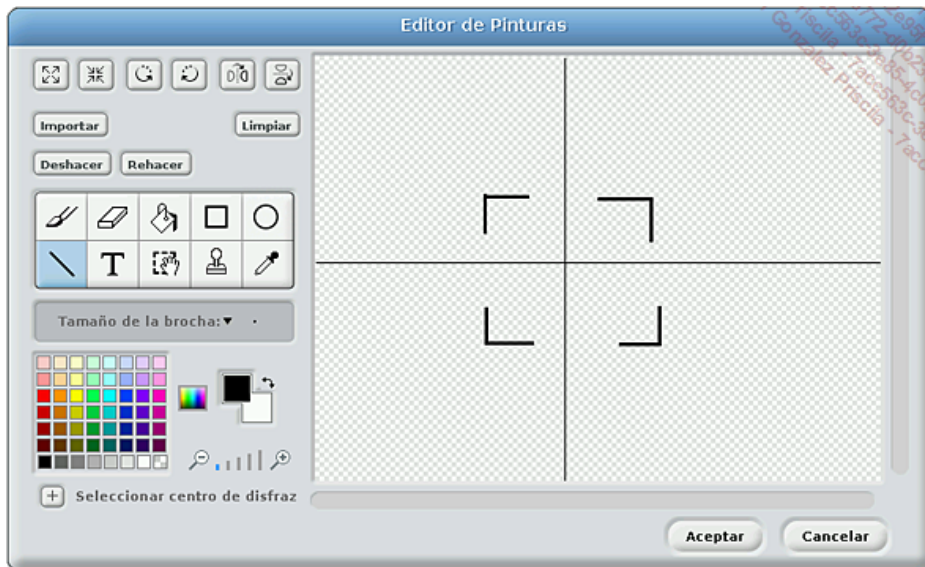
- el pingüino.

a. Visor del aparato de fotos



Pulse el botón **Diseñar un nuevo objeto** en la zona de los duendes (una estrella con un pincel). El editor gráfico se abre sobre un fondo transparente (el tapiz blanco y gris). Seleccione el color amarillo en el selector de colores y el pincel en la zona de herramientas. Trace un ángulo recto utilizando la tecla [Shift] para forzar los trazos horizontales y verticales.

Pulse en el cursor y seleccione su ángulo. Repita el diseño con el cursor hasta tener cuatro ángulos. Seleccione la herramienta de selección (la mano delante de un cuadrado de puntos) y seleccione uno de los cuatro diseños. Pulse los botones **Espejo Horizontal** y **Espejo Vertical** (los dos botones que hay encima del botón **Limpiar**) para obtener la posición deseada. Sitúe sus ángulos para obtener una imagen parecida a la siguiente.



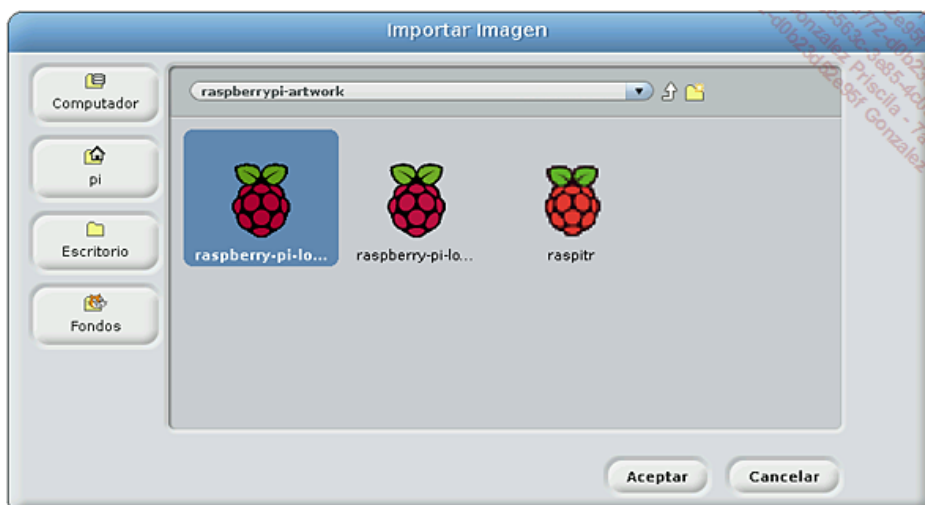
Pulse el botón **Definir el centro de la escena** en la parte inferior izquierda del editor gráfico. El centro de la escena se situará en el lugar donde pulse en la zona de diseño. Intente que coincidan el punto central de la cruz con el centro de su visor.

Termine pulsando en el botón **Aceptar**.

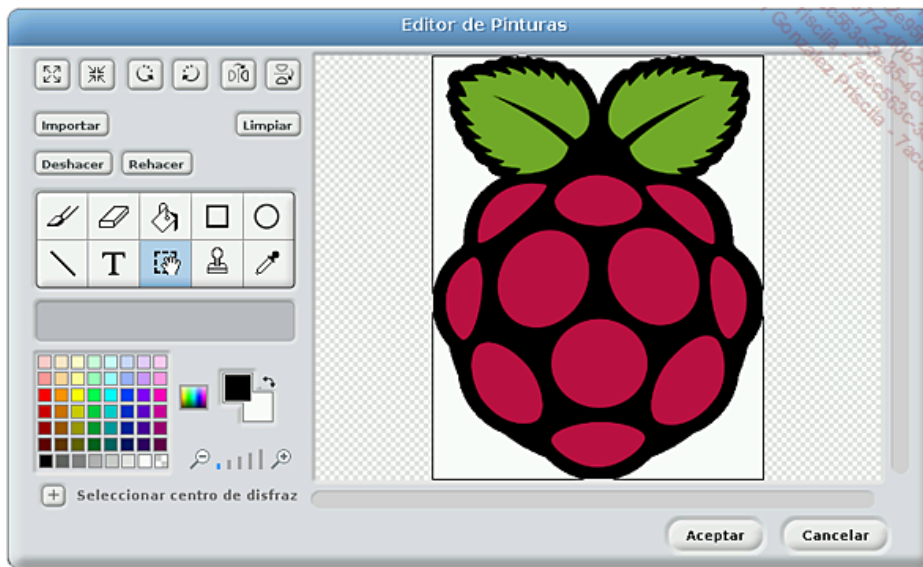
b. El logo Raspberry Pi

En la zona de los duendes, pulse en el botón **Diseñar nuevo objeto** (botón en forma de estrella sobre la que figura un pincel). Se abre el editor gráfico. Pulse en el botón **Importar**. Recorra las carpetas de la Raspberry hasta encontrar el lugar donde se ha almacenado el logo y pulse sobre él.

- Los logos e imágenes usados para el juego (logo Raspberry Pi, Tux...) están disponibles para su descarga desde la página Información.



El logo aparece en la zona de diseño del editor gráfico.



El logo aparece sobre un fondo blanco de forma rectangular. Es así como se verá en la escena: enmarcado por un rectángulo blanco.

Para eliminar este fondo molesto, pulse en el bote de pintura y seleccione el color transparente (última casilla de la parte inferior en la parte derecha del selector de colores). Haga clic con el extremo del bote de pintura que se desborda en la zona blanca. Esta debería llenarse de cuadrados blancos y grises indicando su transparencia.

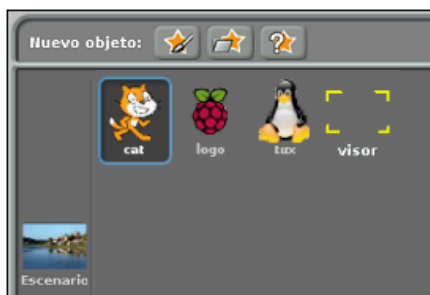
Pulse en el botón **Aceptar** para guardar el duende.

Eventualmente, si la importación de imágenes genera duendes sobredimensionados o infradimensionados, los dos botones situados encima del botón **Importar** permiten ajustar su tamaño. Pulse en [Shift] antes de hacer clic para abrir una ventana que permite ajustar el porcentaje del tamaño, de manera mucho más fina que un simple clic en uno de los botones.

c. El pingüino

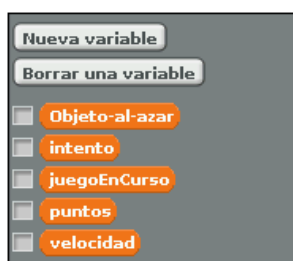
La creación del duende del pingüino se hace de la misma manera que para el logo de la Raspberry Pi, a partir de una imagen de Tux que debe haber descargado. Aquí será necesario tener la precaución de hacer transparentes las zonas de sombra debajo del pingüino.

Tras la creación de los duendes, la zona de los duendes se parecerá a la siguiente:



5. Crear las variables

Ahora que la escena y los duendes están definidos, es necesario definir las variables que se utilizarán durante el juego.

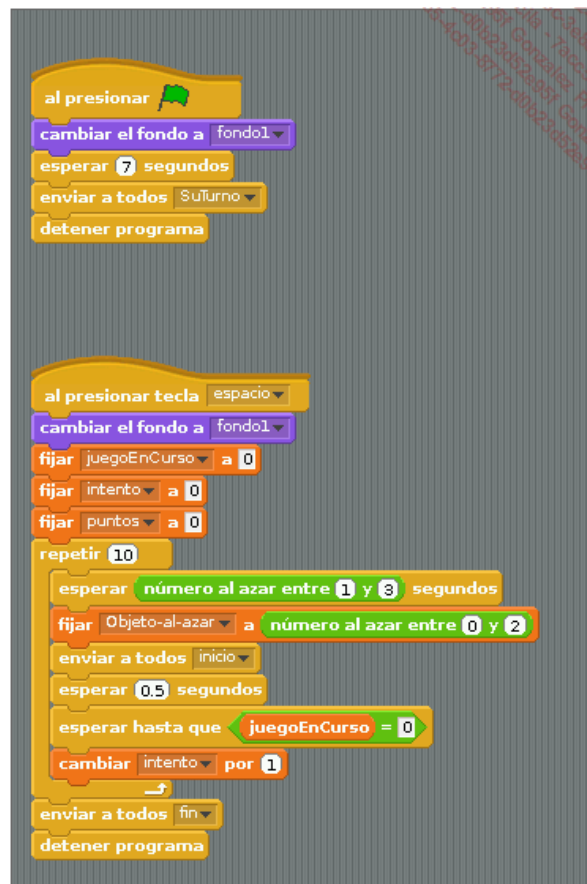


- Objeto-al-azar: variable global - contiene el número aleatorio extraído por la escena. Los duendes leen esta variable para saber si es su turno para pasar a escena. 0 representa el gato, 1 representa el logo de la Raspberry Pi y 2 representa el pingüino.
- intento: variable global - memoriza el número de duendes que han atravesado la escena. Esta variable se ha utilizado durante la puesta a punto del programa. Se podría eliminar.
- juegoEnCurso: variable global - indica que un duende actualmente está presente en la escena.
- puntos: variable global - cuenta los puntos conseguidos por el jugador.
- velocidad: variable local - utilizada únicamente por el logo Raspberry Pi para aumentar su velocidad de marcha.

6. Animar el juego

Ahora cada duende debe recibir el o los scripts que debe ejecutar. Pruebe cada script individualmente. Cuando un script funciona, se puede integrar al conjunto. La escritura simultánea de un gran número de scripts interactuando los unos con los otros habitualmente hace que sea difícil la búsqueda de las causas de funcionamiento erróneo.

a. Los scripts de la escena



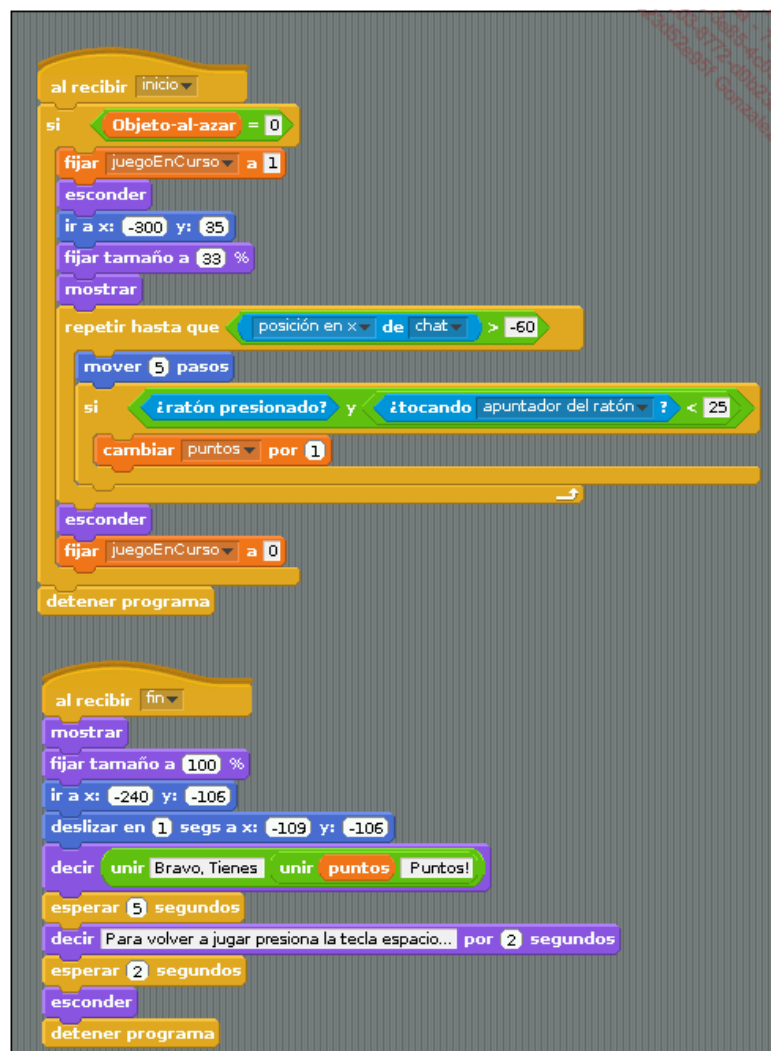
La escena recibe dos scripts:

- Un script de presentación que se activa cuando el jugador pulsa en la bandera verde.
- Un script de juego que se lanza cuando se pulsa la barra de espacio.

El primer script solo muestra el fondo de la presentación durante 7 segundos, antes de activar el script del pingüino con el mensaje *SuTurno*.

El segundo script arranca cuando el jugador pulsa en la barra de espacio. Muestra la escena del juego e inicializa las variables. A continuación, para las 10 pruebas del jugador, el script espera un tiempo entre 1 y 3 segundos. Se extrae un número al azar y se almacena en la variable *Objeto-al-azar*. Envía el mensaje *inicio* a todos los duendes y hace una pausa de 0,5 segundos para permitir al script del duende arrancar y posicionar *juegoEnCurso* a 1. Espera a que el duende devuelva el control volviendo a poner a cero *juegoEnCurso* para continuar. Cuando las 10 pruebas terminan, envía el mensaje de fin a todos los duendes y se detiene.

b. Los scripts del gato



El gato recibe dos scripts: el primero gestiona su movimiento en la escena durante el juego, el segundo gestiona el anuncio del resultado.

Primer script

El primer script arranca cuando la escena envía el mensaje *inicio* a todos los duendes. El script comprueba si se ha extraído el número del gato. Si es así, arranca, en caso contrario se detiene.

Cuando el script arranca, posiciona la variable *juegoEnCurso* para indicar que está en funcionamiento, el gato se sitúa fuera de la escena, su tamaño se reduce al 33 % para simular que se aleja. A continuación avanza hasta -60, lugar donde desaparece detrás de los árboles de la escena.

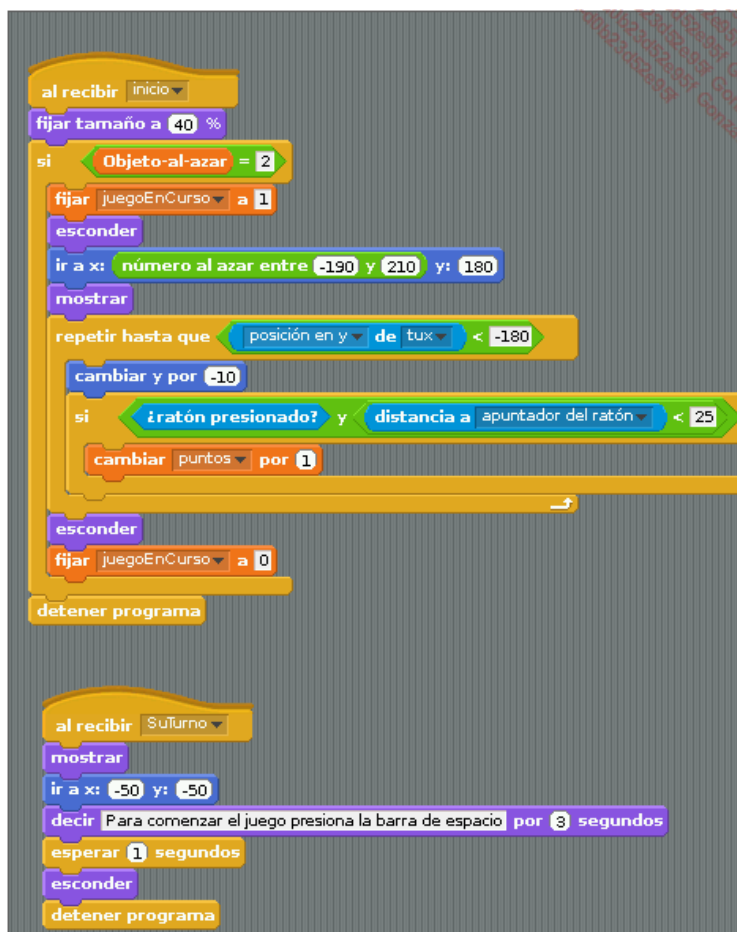
El gato avanza en grupos de 5 pasos, comprobando si el jugador ha pulsado el ratón cerca (25 píxeles). Si es el caso, el contador de puntos se incrementa. Si el jugador mantiene el ratón pulsado, el contador se incrementa en cada bucle, lo que permite aumentar el número de puntos conseguidos.

Cuando llega a su destino, el gato desaparece y después el script libera la variable *juegoEnCurso* para indicar a la escena que puede empezar otro duende. El script se detiene.

Segundo script

El segundo script arranca cuando la escena envía el mensaje *terminado*. El gato aparece en la escena con su tamaño normal y después indica el número de puntos obtenidos y la manera de volver a jugar.

c. Los scripts del pingüino



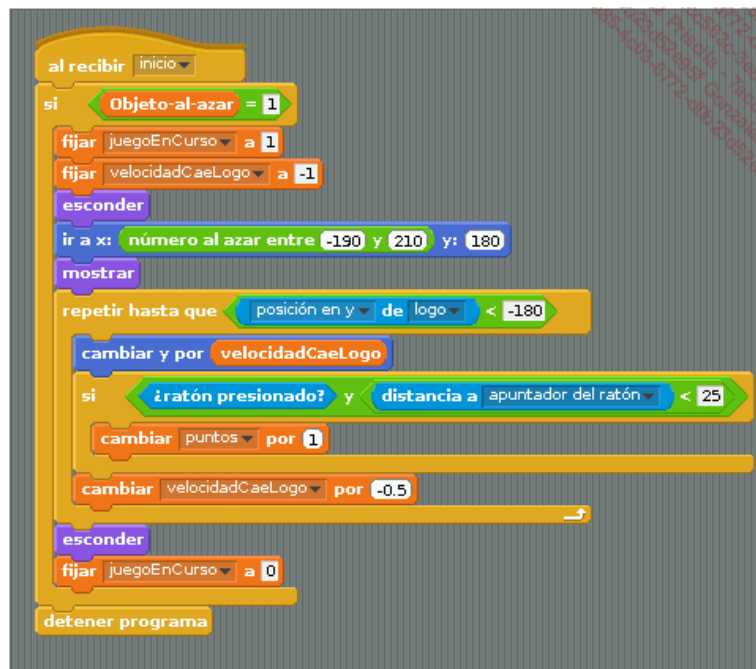
Primer script

El primer script del pingüino es prácticamente idéntico al que gestiona el movimiento del gato. Como el pingüino llega desde la parte superior de la escena el script determina, en primer lugar, la coordenada x extrayendo un número entre -190 y 210. A continuación, el duende desciende a una velocidad de 10 píxeles por bucle hasta la parte inferior de la escena (-180).

Segundo script

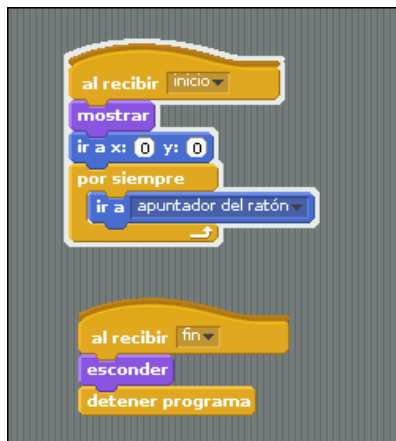
El segunda script arranca cuando la escena envía el mensaje *SuTurno*. El pingüino aparece en la escena e indica al jugador cómo iniciar el juego.

d. El script del logo



El script es idéntico al del pingüino. La velocidad de marcha se fija al inicio a -1 y aumenta en -0.5 con cada paso del bucle para provocar una aceleración de la marcha.

e. Los scripts del visor



Primer script

El primer script se ejecuta cuando la escena envía el mensaje *inicio* a todos los duendes. El visor se sitúa en el centro de la escena y después sigue los movimientos del puntero del ratón.

Segundo script

Cuando la escena envía el mensaje de finalización a todos los duendes, el segundo script arranca. El visor desaparece y el script se detiene.

f. La evolución del juego

Todos los elementos del juego, así como el proyecto, están disponibles para su descarga.

No dude en modificar los argumentos del juego, añadiendo otros duendes, por ejemplo. También puede añadir sonidos para que el desarrollo del juego sea más ameno. No hay nada que se pueda "romper" en la Raspberry Pi. En el peor de los casos, el juego no funcionará como desea o no funcionará en absoluto. Siempre es posible volver atrás.

El aprendizaje de la programación pasa por estas etapas de prueba-error, que enriquecen mucho la experiencia.

Conclusión

Scratch en la Raspberry Pi sigue en versión 1.4, pero la potencia de la Raspberry Pi 3 y las numerosas mejoras y añadidos al programa, lo hacen mucho más agradable de utilizar.

La versión aparecida a finales de 2015 con Jessie se ha mejorado mucho, en particular con la adopción de una máquina virtual Squeak sobre la que se ejecuta Scratch, sustituyendo el antiguo motor de Scratch. En la misma Raspberry Pi, el rendimiento de Scratch se multiplica por entre 3 y 10 veces, según la prueba realizada. El número de tramos mostrados por segundo se ha doblado, pasando de 15 a 30 para una mayor fluidez de las animaciones.

La introducción de un servidor GPIO ofrece un medio sencillo de acceder a las entradas/salidas y a las tarjetas adicionales. Este servidor también gestiona la cámara y da acceso a la dirección IP, así como a la fecha y hora de la máquina. También es posible imaginar un programa Scratch que tome una foto al pulsar en un pulsador e imprima la fecha y hora de la toma de la imagen en la foto.

En el momento de escribir estas líneas se soportan las tarjetas Sense-HAT, Pibrella, Explorer HAT, PiFace, PiLite y Ryantek motor Board. Hay otras tarjetas que se deberán añadir a la lista.

Originalmente visto como una herramienta destinada al aprendizaje de la programación por los niños, Scratch se presenta en la actualidad como una solución que permite interactuar de manera sencilla con el mundo real.

Historia de Python

Desarrollado en una semana por Guido van Rosum (según la leyenda), el lenguaje Python debe su nombre a los Monty Python, de los que Guido es admirador. Guido explica en su blog (<http://python-history.blogspot.com.es/2009/01/personal-history-part-1-cwi.html>) que, recién salido de la universidad, en 1982 entra en CWI como programador. CWI (*Centrum Wiskunde & Informatica* = centro de matemáticas e informática) es el centro de investigación holandés de matemáticas e informática, situado en Amsterdam (Países Bajos). Trabajó en un proyecto llamado ABC, que se abandonó por falta de éxito. En ese momento, se une al grupo que desarrolla Amoeba, un sistema distribuido basado en un micro núcleo desarrollado por CWI y la universidad de Amsterdam. Finalmente trabaja en el grupo multimedia. El lenguaje Python es el producto de esta experiencia en CWI. El lenguaje ABC fue la fuente de inspiración para Python, Amoeba la motivación y el grupo multimedia favoreció su desarrollo. Aunque CWI nunca invirtió en Python, se ha convertido en una herramienta importante utilizada tanto por el grupo Amoeba como por el equipo multimedia.

Iniciado a finales del año 1989, el desarrollo de Python produce una versión operativa al cabo de pocos meses. Un determinado número de personas utilizó Python en CWI. Los compañeros de Guido contribuyeron a una parte de su desarrollo.

En febrero de 1991, Python ya se distribuía con el número de versión 0.9.0 y bajo una licencia parecida a la del MIT. Por tanto, Python era open source antes de la invención de este término en 1997.



Después de muchos estímulos y del numeroso feedback recibido, Guido se ve animado para mantener actualizado el producto durante algunos años. En 1993 se crea un newsgroup Python con el aval de Guido, pero sin que él se implique. En el verano de 1994, Guido es investigador invitado durante dos meses por el NIST (*US National Institute for Standards and Technology* = instituto nacional de normalización y tecnología), que cuenta con numerosos usuarios de Python. Los forma en el uso del lenguaje y adapta Python para responder mejor a sus necesidades.

Durante su estancia en NIST, dirige un taller de Python que le vale una proposición de trabajo en CNRI (*Corporation for National Research Initiatives* = asociación de promoción de la investigación nacional), un laboratorio de investigación sin ánimo de lucro, indirectamente financiado por la DARPA (*Defense Advanced Research Projectes Agency* = agencia para los proyectos de investigación avanzada de defensa). La agencia subvenciona numerosos proyectos utilizando Python, pero no soporta directamente el desarrollo del lenguaje.

En CNRI, Guido participa en la creación de un sistema de agente móvil en Python. El equipo del proyecto le ayuda a mantener la infraestructura de la comunidad Python, como el sitio python.org, el servidor CVS (*Concurrent Versions System* = sistema de gestión de versiones) de seguimiento de las versiones o incluso la mailing list. Las versiones 1.3 a 1.6 datan de esta época. Se produce un intento de creación de una "Fundación Python", que no va más allá de la redacción del reglamento.

La CNRI intenta sin éxito financiar el desarrollo de Python con fondos de la DARPA. A continuación crea un consorcio que atrae a pocos miembros. La licencia de la versión 1.6, inicialmente dirigida a los abogados de la CNRI, se modifica después de la distribución por Richard Stallman para responder a los criterios de la Free Software Foundation y a los de la CNRI. Se convierte en una licencia open source (licencia CNRI).

En 2000, antes del estallido de la burbuja de Internet, Guido y una parte del equipo dejan la CNRI para unirse a BeOpen.com, una start-up que recluta a desarrolladores open source.

Como muchas start-up de la época, BeOpen desaparece de manera espectacular, dejando tras ella una deuda enorme y a los desarrolladores completamente desilusionados.

La licencia de Python evoluciona para reflejar la sucesión de propietarios, en primer lugar BeOpen, después la Python Software Foundation (creada en 2001). Guido y su equipo PythonLabs son contratados por Digital Creations, una de las principales empresas que usan Python, que rápidamente toma el nombre de su principal producto open source: Zope.

El lenguaje evoluciona a su versión 2 hasta diciembre de 2008, fecha en la se publica Python 3.0. Esta versión rompe con la compatibilidad hacia atrás de Python, lo que explica la coexistencia actual de las dos ramas 2.x y 3.x.

Python no ha dejado de evolucionar y mejorar su rendimiento a lo largo de estos años.

Particularidades de Python

En su origen, Python se diseñó para ser muy legible. Permite a un principiante poder escribir rápidamente un programa que funcione. Es así como los estudiantes o investigadores adoptan fácilmente el lenguaje y lo utilizan para sus trabajos. Encontramos Python tanto en la universidad como en el mundo industrial.

Los desarrolladores Python tienen a su disposición una librería muy extensa. Python está presente en muchos dominios, desde la programación de sistemas de desarrollo web, hasta aplicaciones gráficas de sistemas embebidos e incluso en el dominio del cálculo científico. Se puede comunicar con bases de datos clásicas o con directorios LDAP.

Python es un lenguaje interpretado. Se ejecuta en una máquina virtual que arranca cuando se lanza Python.

Python en modo consola

Python se puede utilizar de varias maneras. La más directa es el modo consola, que solo necesita una consola en modo texto y la escritura de comandos directamente en el prompt Python. La segunda posibilidad es el uso de un entorno de desarrollo llamado IDLE. Por último, es posible escribir programas en Python que se ejecutarán por uno u otro entorno.

Las dos ramas de Python están disponibles en la Raspberry Pi. En este capítulo se utilizará Python 3.

El objetivo de este capítulo es permitirle leer y modificar un programa en Python. El uso de programas demostrará las posibilidades del lenguaje, y será un primer paso para descubrir el lenguaje.

1. Inicio de Python

El lanzamiento de Python en una consola en modo texto se hace con el comando `python` para ejecutar Python 2.7 y con el comando `python3` para ejecutar Python 3.4.

```
pi@raspberrypi ~ $ python3
Python 3.4.2 (default, Oct 19 2014, 13:31:11)
[GCC 4.9.1] on linux
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

El prompt con el que Python le indica que espera un comando es `>>>`. Para salir de Python y volver a Linux, escriba `exit()` seguido de [Intro].

Sin entrar en las técnicas de programación, ya es posible usar Python como una calculadora (avanzada).

2. Utilizar Python como una calculadora

a. Operaciones básicas

Estos operadores realizan el cálculo de la expresión y muestran el resultado. Las `#` presentes en las líneas indican un comentario. Python ignora todo lo que sigue a la `#` en la línea.

```
>>> 2 + 3      # Suma
5
>>> 12 - 7     # Resta
5
>>> 6 * 4      # Multiplicación
24
>>> 24 / 6     # División
4.0
>>> 137 % 9    # Módulo: 137/9=15 restan 2
2
>>> 7 ** 2     # Elevar a una potencia
49
>>> 11 // 2    # División entera, los decimales se eliminan
5
>>> 3 * (5 + 2) # Calcula en primer lugar el contenido de los paréntesis
21
>>> 3 * 5 + 2  # El signo * es prioritario
17
```

b. Operadores de comparación

Estos operadores evalúan la expresión. Si es verdadera, devuelven `True`. En caso contrario devuelven `False`.

```
>>> 8 == 4     # Igualdad entre ambos valores
False
>>> 6 != 5     # Ambos valores son diferentes
True
>>> 4 > 2      # El número de la izquierda es más grande
True
>>> 5 < 27     # El número de la izquierda es más pequeño
True
>>> 8 >= 11    # El número de la izquierda es mayor o igual
False
>>> 45 <= 66  # El número de la izquierda es menor o igual
True
```

c. Operadores de asignación

Los operadores de asignación dan al operando situado a la izquierda el valor de la expresión situada a la derecha del operador. El comando `print(x)` muestra el valor de `x` en la pantalla. Es posible mostrar una variable por línea de comandos simplemente introduciendo su nombre, pero es mejor tener la costumbre de usar `print()`, que será obligatorio en los programas.

```
>>> x = 7      # x toma el valor 7
>>> x          # Mostrar el valor de x en la línea de comandos
7
>>> x = 24     # x toma el valor 24
>>> print(x)   # Mostrar el valor de x
24
>>> x += 4     # x = x + 4
>>> print(x)
28
>>> x -= 7     # x = x - 7
>>> print(x)
21
>>> x *= 2     # x = x * 2
>>> print(x)
42
>>> x /= 2     # x = x / 2
>>> print(x)
21.0
>>> x %= 6     # x = x modulo 6 = resto de la división
>>> print(x)
3.0
>>> x **= 3    # x = x elevado a 3
>>> print(x)
```

```

27.0
>>> x //= 2    # x = resultado entero de x / 2
>>> print(x)
13.0

```

d. Operadores sobre los bits

Los operadores sobre los bits intervienen en los números expresados en binario.

```

>>> x = 6      # x = 6, es decir, 0110 en binario
>>> y = 3      # y = 3, es decir, 0011 en binario
>>> print(x&y) # 0110 y 0011 = 0010
2
>>> print(x|y) # 0110 o 0011 = 0111
7
>>> print(x^y) # 0110 o exclusivo 0011 = 0101
5
>>> print(~x)  # 0110 complemento a 2 de x = 1001
-7
>>> print(~y)  # 0011 complemento a 2 de y = 1100
-4
>>> print(x << 2) # Desplazamiento de 2 bits a la izquierda
                        # equivale a multiplicar por 4
24
>>> print(x >> 2) # Desplazamiento de 2 bits a la derecha
1

```

3. Escribir una cadena de caracteres

Una cadena de caracteres es una sucesión de caracteres entre comillas.

```

>>> print("Hola a todos")
Hola a todos

```

El comando `print()`, utilizado anteriormente para mostrar el resultado de las operaciones, también permite mostrar las cadenas de caracteres.

4. Las variables

Cuando hemos escrito `x = 24` en el párrafo de las operaciones hemos reservado un espacio en la memoria para almacenar el valor 24, y Python ha dado el nombre `x` a este espacio. Por tanto, la variable `x` es un espacio memoria que puede recibir valores numéricos o caracteres.

En Python, basta con usar una variable para que se cree automáticamente.

```

>>> x = 24      # x se crea y contiene 24
>>> y = x + 8   # y se crea y contiene (24+8)=32
>>> z = x + y   # z se crea y contiene (24+32)=56
>>> print(x, y, z) # Muestra los valores de las 3 variables
                24 32 56
>>> x = "mi_variable" # x contiene una cadena de caracteres
>>> print(x)         # Mostrar la cadena de caracteres
mi_variable

```

El primer carácter de las variables es un carácter alfabético. Las cifras, así como los operadores, se excluyen y no pueden ser el inicio de un nombre de variable. El nombre de la variable no puede ser un palabra reservada (`and`, `if`, `print`, `try`, `return`...).

El nombre de las variables es sensible a la diferencia entre mayúsculas y minúsculas: `contador` y `Contador` son dos variables diferentes. Las variables que empiezan con el carácter subrayado `_` tienen un comportamiento particular. Las variables que empiezan con `_` en un módulo, no se exportan. Las variables que empiezan y terminan con dos caracteres de subrayado `__` se reservan para el lenguaje Python.

Tenga la costumbre de dar nombres representativos a sus variables. La puesta a punto de un programa con las variables `pepe`, `cosa`, `toto`... habitualmente es (siempre) más complicada que un programa en el que los nombres de las variables son `contador`, `temperatura_maxi` `odistanciaObstaculoDerecho`. Use los caracteres de subrayado `_` o las mayúsculas para separar las palabras y facilitar la lectura de los nombres de variables. Por último, intente respetar la misma norma de escritura para todas las variables en un programa.

5. Las listas

También existen listas en Python. Una lista es un conjunto de elementos separados por comas y entre corchetes. El primer elemento de la lista recibe el índice 0, los siguientes los índices 1, 2...

Creación de la lista y visualización del contenido:

```

>>> li = ["a", "var1", "xyz", "var2"]
>>> li
['a', 'var1', 'xyz', 'var2']

```

Muestra el elemento de índice 0:

```

>>> li[0]
'a'

```

Muestra el elemento de índice 3:

```

>>> li[3]
'var2'

```

Muestra el segundo elemento, partiendo desde el final (índice negativo):

```

>>> li[-2]
'xyz'

```

Muestra los elementos de índices 1 a 3 (elemento de índice 3 no incluido):

```

>>> li[1:3]
['var1', 'xyz']

```

Adición de un elemento al final de la lista y visualización del contenido de la lista:

```
>>> li.append(« var3 »)
>>> li
['a', 'var1', 'xyz', 'var2', 'var3']
```

Inserción de un elemento en el lugar del elemento 3, que recolocará el resto de elementos en una posición hacia el final de la lista:

```
>>> li.insert(3, « inserción »)
>>> li
['a', 'var1', 'xyz', inserción, 'var2', 'var3']
```

En los ejemplos anteriores, los elementos de la lista se muestran en pantalla. Es posible extraer los elementos de una lista para crear otra:

```
>>> lista2=li[1:3]
>>> lista2
['var1', 'xyz']
```

6. Crear un programa en Python

Todos los comandos que se escriben en modo consola o interactivo permiten descubrir rápidamente el lenguaje Python, pero son efímeros. Para volver a ejecutar la misma secuencia de comandos, será necesario introducirlos de nuevo.

Un editor de texto como *nano* en modo consola, o *gedit* en modo gráfico, permite introducir una sucesión de instrucciones en forma de script y después guardarlas en un archivo. A continuación bastará con ejecutar este script para reproducir el funcionamiento del programa sin tener que volver a escribir la totalidad de las instrucciones.

Escriba el siguiente programa en *nano* y guárdelo con el nombre de *multiplicacion.py*. Los comentarios son opcionales... aunque aconsejables.

```
# Tabla de multiplicación del 4
valor = 1 # Variable utilizada para contar
print("Tabla de multiplicación del 4")
while valor <= 10:
    # Al inicio del bloque a ejecutar
    # Observe la indentación
    print(valor, " x 4 = ",valor*4)
    valor += 1 # Incrementar el valor
print("Fin de la tabla de multiplicación")
```

➤ Cada línea y cada bloque de este programa se explican a continuación.

Detalles del programa *multiplicacion.py*

```
# Tabla de multiplicación del 4
```

Línea de comentario ignorada por Python por el # inicial. Piense en incluir en el comentario el nombre del programa y la información necesaria, por si tiene que volver a este script en el futuro (objetivos del programa, nombre del autor, versión de Python, eventuales parámetros...).

```
valor = 1 # Variable utilizada para contar
```

Inicializa la variable, que se multiplicará por 4 en cada línea de la tabla de multiplicación.

```
print("Tabla de multiplicación del 4")
```

Muestra un mensaje por pantalla para informar al usuario.

```
while valor <= 10:
    # Al inicio del bloque a ejecutar
    # Observe la indentación
    print(valor, " x 4 = ",valor*4)
    valor += 1 # Incrementa el valor
```

El bloque `while` se va a ejecutar mientras la condición que sigue a `while` sea verdadera, por tanto, el valor deberá ser inferior a 11. El signo `:` marca el final de la línea. Su ausencia provoca un error de sintaxis.

El bloque se define solo por la indentación, es decir desplazando a la derecha las líneas que lo componen. Habitualmente se desplazan 4 caracteres para delimitar un bloque. Es posible usar la tecla tabulación, pero la tabulación no siempre se define con el mismo número de caracteres según los sistemas. De todas maneras, es absolutamente necesario evitar usar en un mismo script espacios y tabulaciones para indentar. Seleccione una manera de indentar y manténgala durante todo el programa.

La línea `print()` muestra por pantalla la variable, que sirve de contador de línea y después `x 4` y, para terminar, el resultado de la operación `valor *4`.

La última línea del bloque: `valor += 1` incrementa la variable.

Como ya no hay indentación, el bloque `while` termina y Python vuelve a la instrucción `while` para evaluar de nuevo si la condición es verdadera o falsa. El bucle se repite hasta que se alcance el valor límite.

```
print("Fin de la tabla de multiplicación")
```

La última línea del programa informa al usuario del fin del script.

7. Ejecutar un programa

Para ejecutar el programa guardado con el nombre de *multiplicacion.py*, es necesario llamar a Python desde la línea de comandos y pasarle el nombre del script como argumento.

```
pi@raspberrypi ~ $ python3 multiplicacion.py
Tabla de multiplicación del 4
1 x 4 = 4
2 x 4 = 8
3 x 4 = 12
4 x 4 = 16
5 x 4 = 20
6 x 4 = 24
```

```
7 x 4 = 28
8 x 4 = 32
9 x 4 = 36
10 x 4 = 40
Fin de la tabla de multiplicación
```

8. Utilizar el bucle for

La instrucción `for` también permite que el programa repita un bucle un determinado número de veces.

Sintaxis

```
for (x, y, z)
```

`x` indica el primer valor del rango asignado al `for`.

`y` indica el último valor del rango asignado al `for`. Cuando se alcanza este valor, el bucle `for` termina sin ejecutar el bloque de instrucciones.

`z` indica el paso entre los valores. Por defecto `z` vale `+1`.

Ejemplo 1

```
# archivo for_ejemplo.py
# Ejemplo de bucle for
# Cuenta de 1 a 10
for n in range(1,11):
    print(n, " ")
print("terminado")
```

Cuando se ejecuta, este programa muestra:

```
pi@raspberrypi ~ $ python3 for_ejemplo.py
1
2
3
4
5
6
7
8
9
10
terminado
```

Ejemplo 2

```
# archivo for_ejemplo2.py
# Ejemplo de bucle for
# Cuenta de 1 a 10, de 3 en 3
for n in range(1,11,3):
    print(n, " ")
print("terminado")
```

Este programa muestra los números de 1 a 10, pero esta vez de 3 en 3.

```
pi@raspberrypi ~ $ python3 for_ejemplo2.py
1
4
7
10
terminado
```

➤ La instrucción `range` detiene la ejecución del bucle cuando `n` alcanza el valor 11. Por tanto, el bucle se ejecuta 10 veces. El valor 1 representa el valor de inicio del bucle. El valor 3 indica el paso entre cada valor del bucle.

9. Recuperar un valor de entrada por el teclado

Es interesante poder modificar el desarrollo del programa introduciendo un valor por el teclado. Para ello, es necesario llamar a la directiva `input()`. El programa espera a continuación la escritura de una cifra por el teclado. A continuación, debe mostrar la cifra en una línea y después el doble de esta cifra en la siguiente línea.

Ejemplo 1

```
# archivo introducido.py
# Ejemplo de escritura por el teclado
# y uso en el interior del programa
numero = input("Escribir una cifra: ")
print(numero)
print(numero + numero)
```

Este script producirá el siguiente resultado si introduce 5 y pulsa [Intro].

```
pi@raspberrypi ~ $ python3 introducido.py
Escribir una cifra: 5
5
55
```

Evidentemente, el segundo `print()` produce un resultado erróneo. La explicación tiene que ver con el hecho de que `input()` devuelve sistemáticamente una cadena de caracteres. El segundo `print()` concatena dos caracteres 5 y devuelve la cadena 55. Por tanto, funciona correctamente en el sentido de Python, pero su resultado no corresponde con lo que esperamos.

Para que el cálculo en el segundo `print()` funcione, es necesario convertir el carácter recuperado con `input()` en un número entero con el comando `int()`.

Ejemplo 2

```
# archivo introducido2.py
# Ejemplo de escritura por el teclado
# y uso dentro del programa
```

```
# El valor escrito por teclado
# se transforma en entero
numero = int(input("Escribir una cifra: "))
print(numero)
print(numero + numero)
```

El uso de `int()` permite hacer funcionar el programa según nuestras necesidades.

```
pi@raspberrypi ~ $ python3 introducido2.py
Escribir una cifra: 5
5
10
```

10. Juegue usted mismo

Parta del programa *multiplicacion.py* y modifíquelo para usar un bucle `for` en lugar del `while`. Haga una copia de seguridad con el nombre *multiplicacion2.py*.

Modifique el programa *multiplicacion2.py* para que el usuario pueda introducir por teclado la tabla de multiplicación que desea mostrar. Haga una copia de seguridad con el nombre *multiplicación2a.py*.

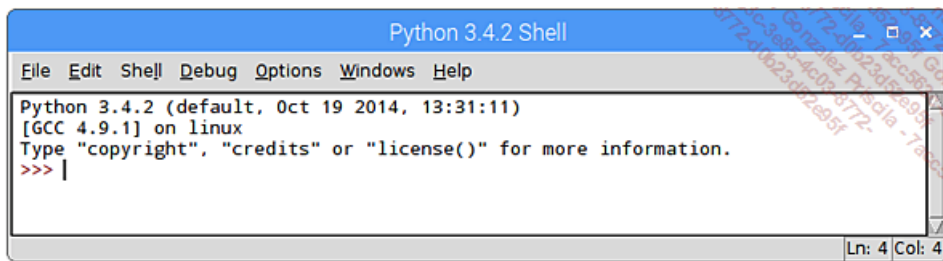
En el Anexo se ofrece una solución para estos ejercicios.

Python en entorno gráfico

El uso de un entorno de desarrollo integrado facilita la escritura y puesta a punto de los programas en Python. La Raspberry Pi ofrece un entorno para cada versión de Python. En este capítulo se utilizará IDLE de Python 3.

1. El entorno de desarrollo IDLE

Para abrir la ventana del entorno de desarrollo Python, pulse en **Menú** y después en **Programación**. Por último, pulse en **Python 3 (IDLE)**.



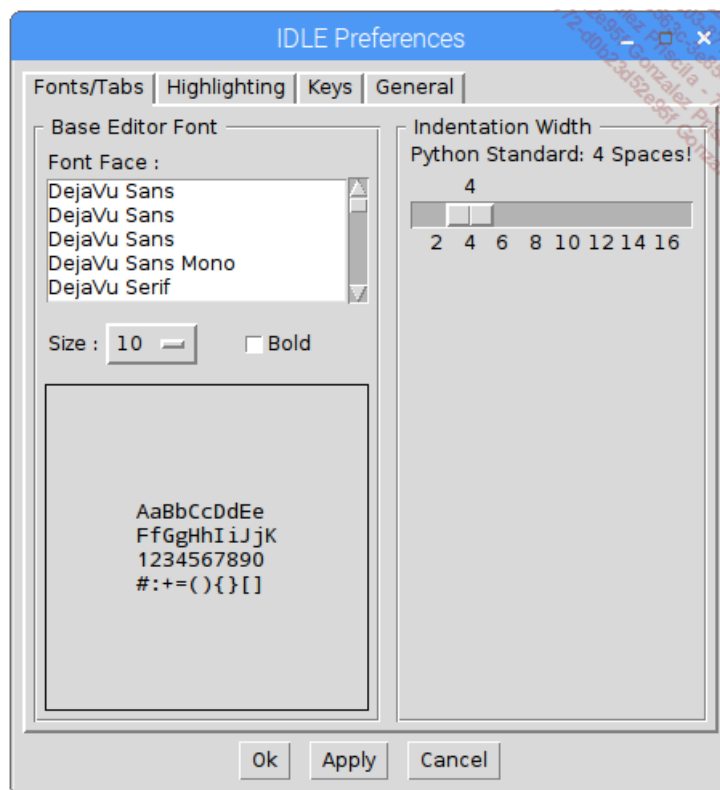
La versión de Python utilizada por el entorno IDLE, se indica en la primera línea. Aquí es la versión 3.4.2. Es posible que la versión disponible cuando utilice Python sea diferente.

El entorno IDLE da asistencia a la escritura de programas en Python con la coloración sintáctica, la indentación automática y la visualización de tooltips de ayuda (*CallTips*).

a. Ajustar el tamaño de los caracteres

Si los caracteres que se muestran en la ventana IDLE le parecen muy pequeños, puede ajustar su tamaño como desee.

En el menú **Options**, pulse en **Configure IDLE...** Se abrirá la siguiente ventana.

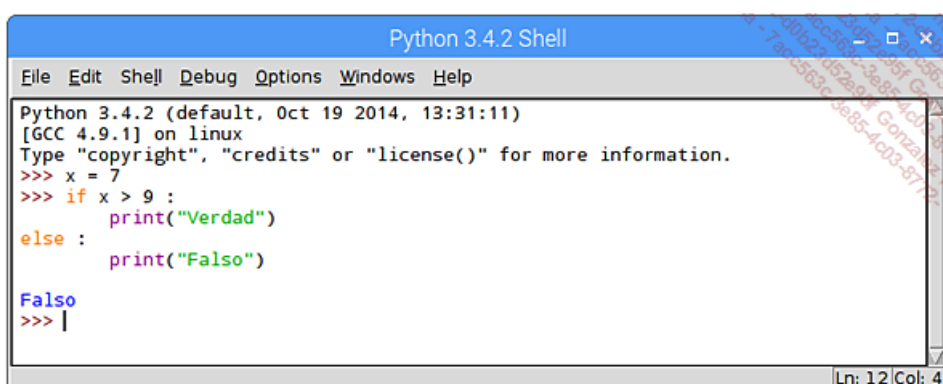


La parte izquierda le permite seleccionar el tipo de letra que desea utilizar, así como su tamaño. También puede marcar la opción **Bold** si desea que el texto esté en negrita.

La parte derecha de esta ventana ajusta la indentación automática de Python, normalmente fijada a cuatro espacios.

2. Utilizar el modo prompt

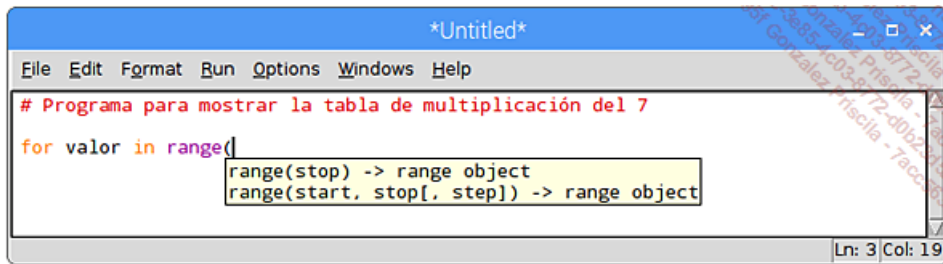
Es posible, como en modo consola, trabajar directamente desde el prompt:



Se crea la variable `x` y toma el valor 7. A continuación, una instrucción `if...else...` compara el valor con 9 y devuelve verdadero si `x` es superior, falso si es inferior. IDLE3 indenta las líneas `print()` automáticamente.

3. Crear un programa

El menú **File - New File** abre una ventana llamada *Untitled*.



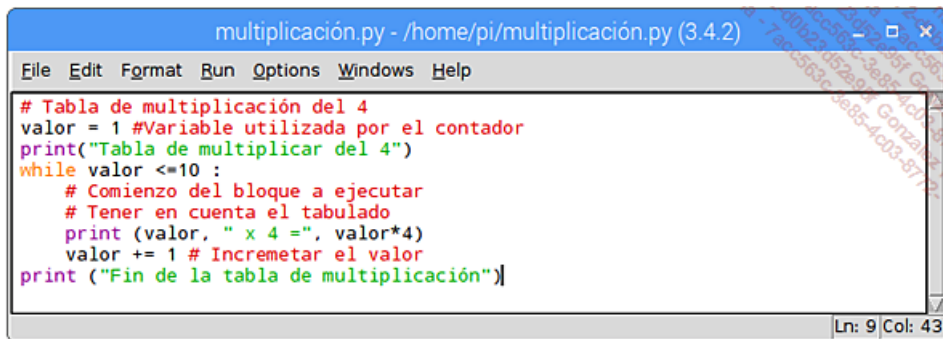
```
*Untitled*
File Edit Format Run Options Windows Help
# Programa para mostrar la tabla de multiplicación del 7
for valor in range(
range(stop) -> range object
range(start, stop[, step]) -> range object
Ln: 3 Col: 19
```

Esta ventana es un editor. La coloración sintáctica, la indentación automática y los tooltips de ayuda facilitan la escritura de programas. En el ejemplo anterior, IDLE3 recuerda automáticamente la sintaxis para dar asistencia al programador en el uso de la función `range()`.

Después de escribir el programa, el menú **File - Save As...** permite guardar el script para su uso en el futuro.

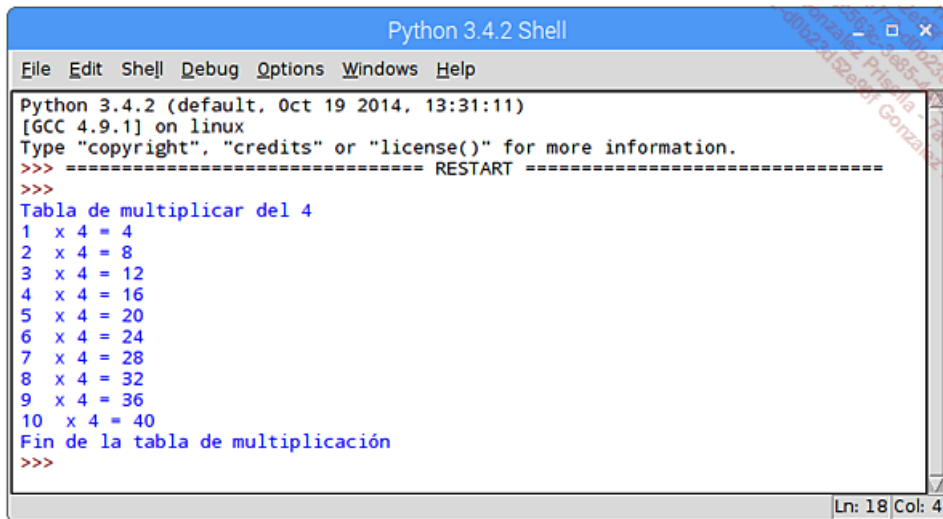
4. Ejecutar un programa

Un programa guardado se puede volver a llamar con **File - Open...** Por ejemplo, el programa *multiplicacion.py* realizado anteriormente.



```
multiplicación.py - /home/pi/multiplicación.py (3.4.2)
File Edit Format Run Options Windows Help
# Tabla de multiplicación del 4
valor = 1 #Variable utilizada por el contador
print("Tabla de multiplicar del 4")
while valor <=10 :
    # Comienzo del bloque a ejecutar
    # Tener en cuenta el tabulado
    print (valor, " x 4 =", valor*4)
    valor += 1 # Incrementar el valor
print ("Fin de la tabla de multiplicación")
Ln: 9 Col: 43
```

En la ventana del editor de script, con el menú **Run - Run Module** o pulsando la tecla [F5] se lanza la ejecución del programa en una ventana IDLE3.



```
Python 3.4.2 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.2 (default, Oct 19 2014, 13:31:11)
[GCC 4.9.1] on linux
Type "copyright", "credits" or "license()" for more information.
>>> ----- RESTART -----
>>>
Tabla de multiplicar del 4
1 x 4 = 4
2 x 4 = 8
3 x 4 = 12
4 x 4 = 16
5 x 4 = 20
6 x 4 = 24
7 x 4 = 28
8 x 4 = 32
9 x 4 = 36
10 x 4 = 40
Fin de la tabla de multiplicación
>>>
Ln: 18 Col: 4
```

La prueba y puesta a punto son posibles en IDLE3 para programas que utilizan funciones básicas de Python. Sin embargo, la ejecución de programas de tamaño importante y/o que implementan interfaces gráficas puede ralentizar mucho IDLE3, incluso provocar errores debido a la interacción entre el entorno gráfico del programa verificado y el de IDLE3. Para este tipo de programas es mejor volver al modo consola para las pruebas.

Definir una función

Los ejemplos de scripts anteriores son cortos, adaptados a una tarea única y bien definidos. El desarrollo de programas más largos hace necesaria la escritura de numerosas líneas de código. La escritura y el mantenimiento del programa se hacen cada vez más complicados.

Un buen método para abordar un programa complejo consiste en descomponerlo en múltiples tareas sencillas, de tal manera que cada una se pueda tratar por separado. Por otro lado, habitualmente comprobamos que una misma tarea aparece varias veces en un script, y es interesante no reescribir trozos de código idénticos cada vez.

En los ejemplos de scripts, las funciones `print()` e `input()` utilizadas son las funciones predefinidas del lenguaje Python. Es posible enriquecer el lenguaje definiendo sus propias funciones.

Una función se define por:

- Su nombre, que debe ser único para que el intérprete de Python pueda identificarla de manera determinada.
- Sus argumentos, que son los datos que se pasan a la función para que los trate.
- Su salida, que es el resultado que devuelve después del tratamiento de los datos.

Es posible que una función no devuelva ningún valor de retorno y solo ejecute instrucciones. En este caso, usamos el nombre de procedimiento, pero fundamentalmente su uso es idéntico al de una función.

Es obligatorio que una función se defina antes de su primer uso. En caso contrario, el intérprete de Python devuelve un error.

1. Crear una función

La directiva `def` se usa para indicar a Python que lo que sigue es una definición de función.

En el programa de la tabla de multiplicación anterior:

```
print("Tabla de multiplicación del N")
for valor in range(1,11):
    # Al inicio del bloque a ejecutar
    # Observe la indentación
    producto = valor * N
    print(valor, " x ",N," = ",producto)
print("Fin de la tabla de multiplicación")
```

Es posible definir una función que ejecute la multiplicación y devuelva el resultado:

```
# Tabla de multiplicación del N
# programa multiplicacion3.py
N = int(input("¿Qué tabla desea mostrar? "))
# input() devuelve sistemáticamente una cadena de caracteres
# es necesario convertir el valor de entrada a un número entero

# Definición de la función multipli
# Nombre: multipli
# Uso: multipli (N1,N2)
# Esta función multiplica los dos números proporcionados como argumentos
# resultado: producto de los dos números proporcionados.
def multipli(numero1, numero2):
    return numero1*numero2

print("Tabla de multiplicación de N")
for valor in range(1,11):
    # Al inicio del bloque a ejecutar
    # Observe la indentación
    producto = multipli(valor, N)
    print(valor, " x ",N," = ",producto)
print("Fin de la tabla de multiplicación")
```

La función `multipli()` se define empezando por la palabra clave `def` seguida del nombre de la función y de los argumentos que se pasan a la función, entre paréntesis. El signo `:` debe ir obligatoriamente al final de la línea.

Las instrucciones de la función están contenidas en el bloque indentado que sigue y solo tiene una línea en este caso.

La función devuelve el valor que sigue a la instrucción `return`: `numero1*numero2`, es decir, el producto de los dos números que se pasan como argumentos a la función.

Hay que documentar cada función e indicar en los comentarios: su nombre, uso, lo que hace, los argumentos necesarios y el valor que devuelve. Esto permite garantizar el mantenimiento de una función, incluso durante mucho tiempo después de su creación, sin tener que detenerse en las líneas de código para entender lo que hacen.

2. Uso de la función

Una función se utiliza de la siguiente manera:

```
variable = función(argumento1, argumento2...)
```

En el caso de la función `multipli()`, se usa en la línea:

```
producto = multipli(valor, N)
```

A la variable `producto` se le asigna el valor que devuelve la función `multipli()`, que se llama con los argumentos `valor` y `N`. Si la función se llama varias veces en el script, se hará cada vez con el valor de los argumentos.

El interés de la creación de una función en el caso de la multiplicación es mínimo, pero es necesario contemplar el uso de las funciones en el caso de scripts de varias decenas, incluso varios centenares de líneas.

Utilizar los módulos

El desarrollo de un script puede implicar la creación de numerosas funciones. La agrupación de funciones en un módulo permite almacenar juntas las funciones especializadas en módulos separados y llamarlas bajo demanda (cálculos, tratamiento de imágenes, acceso a archivos...). El uso de estas funciones se hace integrando al script el módulo que las contiene.

1. Los módulos internos

Durante la instalación de Python, se instala un determinado número de módulos al mismo tiempo que Python. Dan acceso:

- Al tratamiento de cadenas de caracteres (string, curses, etc.),
- Al sistema (sys, time, os),
- A las bases de datos (anydbm, shelve),
- A los cálculos (maths, random).

2. Crear un módulo

Un módulo es simplemente un archivo que contiene las funciones que el programador desea agrupar. Integra generalmente un programa de prueba, que permite comprobar que funcionan las funciones incluidas en el módulo. Si es necesaria la modificación de una o varias funciones, el script de prueba validará que las funciones contenidas en el módulo siguen funcionando normalmente.

El siguiente ejemplo muestra cómo crear un módulo llamado `cálculo`. Este módulo contiene la función `multipli()`, así como un programa de prueba destinado a comprobar el funcionamiento correcto de la función.

Este ejemplo utiliza los *docstrings*, que permiten crear comentarios. La cadena de caracteres que constituye un docstring se encierra entre triples comillas dobles. Los docstrings se utilizan para crear automáticamente la documentación de los módulos.

Módulo: cálculo

```
"""
    Este módulo contiene las funciones
    que ejecutan los cálculos
"""
# Definición de la función multipli
def multipli(numero1, numero2):
    """
    Devuelve un resultado igual
    al producto de los dos números
    pasados como argumentos.
    """
    return numero1*numero2

# Comprobación del módulo que permite comprobar su correcto funcionamiento
if __name__ == "__main__":
    print("7 x 3 = ",multipli(7, 3))
```

La función `multipli` es idéntica. El módulo contiene los docstrings y una prueba, cuyo detalle se describe a continuación.

Los docstrings

El primer docstring describe el destino del módulo, el segundo da las indicaciones del uso de la función. Estos docstrings son diferentes a los comentarios, que son internos a las funciones. Cuando se solicita ayuda de un módulo con la función `help(nombre_del_módulo)`, se genera una pantalla de ayuda automáticamente a partir de los docstrings presentes en el módulo. Por ejemplo, si importamos el módulo `calculo` para usarlo en un programa, el comando `help` muestra los docstrings de una manera muy profesional. El uso de docstrings evita crear archivos de ayuda separados.

```
>>> import calculo
>>> help(calculo)
```

genera la siguiente visualización:

```
Help on module calculo:

NAME
    calculo

DESCRIPTION
    Este módulo contiene las funciones
    que ejecutan los cálculos

FUNCTIONS
    multipli(numero1, numero2)
        Devuelve un resultado igual
        al producto de los dos números
        pasados como argumentos.

FILE
    /home/shares/public/python/calculo.py

(END)
```

Comprobación del módulo

La prueba del módulo utiliza las palabras reservadas `__name__` y `__main__`. Cada palabra se precede y se sigue de dos caracteres subrayados.

Si el módulo se ejecuta como script, la variable `__name__` vale `__main__`. Se ejecuta la prueba y permite comprobar que las funciones del módulo cumplen su papel.

Si el módulo se usa en un script, `__name__` no vale `__main__`: la prueba no se ejecutará y el módulo simplemente se importará.

```
pi@raspberrypi ~ $ python3 calculo.py
7 x 3 = 21
pi@raspberrypi $
```

3. Utilizar un módulo en un script

El uso de las funciones contenidas en un módulo se hace después de la importación del módulo en el script:

```
>>> import calculo
```

Para usar la función en el script, basta con preceder el nombre de la función por el nombre del módulo:

```
...
resultado = calculo.multipli(numero1, numero2)
...
```

El script de multiplicación se convierte en (con el nombre *multiplicacion4.py*):

```
# Tabla de multiplicación del N
# Programa multiplicacion4.py

# Importación del módulo calculo.py
import calculo

N = int(input("¿Qué tabla desea mostrar? "))
# input() devuelve sistemáticamente una cadena de caracteres
# es necesario convertir el valor de entrada en un número entero

print("Tabla de multiplicación del N")
for valor in range(1,11):
    # Al inicio del bloque a ejecutar
    # Observe la indentación
    producto = calculo.multipli(valor, N)
    print(valor, " x ",N," = ",producto)
print("Fin de la tabla de multiplicación")
```

También es posible importar una o varias funciones de un módulo con la directiva *from*.

```
>>> from calculo import multipli
```

A continuación, la función se puede utilizar directamente:

```
...
producto = multipli(numero1, numero2)
...
```

Pero ya no indica que la función *multipli()* es originaria del módulo *calculo*, lo cual puede resultar desconcertante para el mantenimiento de un script.

4. Alias del nombre de módulo

Cuando el nombre de un módulo es largo, la definición de un alias permite usar este módulo con su nombre abreviado.

Ejemplo

```
import miModuloConUnNombreMuyLargo as mod
```

Permite el uso en el programa del alias definido en lugar del nombre original:

```
mod.función(a,b)
```

En lugar de:

```
miModuloConUnNombreMuyLargo.función(a,b)
```

Leer y escribir en un archivo

El archivo es un medio práctico para almacenar datos cuando:

- La cantidad de información a conservar es muy importante y puede desbordar la memoria de la Raspberry Pi.
- Los datos se deben conservar después de apagar la Raspberry Pi.
- El archivo contiene datos recuperados mediante descarga (cuenta bancaria, datos del tiempo, etc.).

Por supuesto, Python ofrece mecanismos para acceder a los archivos, en modo de lectura y de escritura.

Como para consultar un libro impreso en una librería, el uso de un archivo necesita conocer su nombre para localizarlo. A continuación hay que abrir el archivo, que se puede leer secuencialmente (como una novela) o accediendo a cualquier lugar del archivo (como un libro de recetas). Cuando se termina el trabajo, el archivo, al igual que el libro, se cierra.

1. Creación de un archivo

La creación de un archivo en Python se hace abriendo en primer lugar el archivo con la función `open()` y después escribiendo los datos en este archivo con la función `write()`. La función `open()` devuelve un descriptor que permite el acceso al archivo.

- En modo escritura: si el archivo ya existe, se elimina y se sustituye por el nuevo archivo. La sintaxis es: `open("nombre_archivo", "w")`.
- En modo adición: los datos se escriben al final del archivo, sin modificar los datos presentes en el archivo. La sintaxis es: `open("nombre_archivo", "a")`.

A continuación, se cierra el archivo con `close()`.

Ejemplo de creación de archivo (creaArch.py)

```
# Creación de un archivo en modo escritura
# El archivo se crea si no existe
# El archivo se elimina si ya existe

# archivo es el descriptor que permite a Python acceder al archivo
archivo = open("data.txt", "w")
# Escritura de caracteres en el archivo
archivo.write("Datos a escribir en el archivo \n")
# Cierre del archivo
archivo.close()
```

La ejecución de este script crea el archivo `data.txt` en el directorio actual y escribe en él la frase "Datos a escribir en el archivo":

```
pi@raspberrypi ~ $ python3 creaArch.py
pi@raspberrypi ~ $ cat data.txt
Datos a escribir en el archivo
```

➤ La `\n` al final de la cadena de caracteres inserta un salto de línea.

2. Escritura en un archivo

La escritura en un archivo existente sin modificar su contenido se hace en modo adición:

Ejemplo de adición en un archivo (adicionArch.py)

```
# Adición de datos al final de un archivo
# archivo es el descriptor que permite a Python acceder al archivo
# Esta vez, open() utiliza el modo "a"
archivo = open("data.txt", "a")
# Escritura de caracteres al final del archivo
archivo.write("Datos a escribir al final del archivo \n")
# Cierre del archivo
archivo.close()
```

La ejecución de este script añade la frase "Datos a escribir al final del archivo" después de los datos ya presentes:

```
pi@raspberrypi ~ $ python3 adicionArch.py
pi@raspberrypi ~ $ cat data.txt
Datos a escribir en el archivo
Datos a escribir al final del archivo
```

3. Lectura de un archivo

La función `open()` es la encargada de leer un archivo. Esta vez se llama en modo lectura. La sintaxis es: `open("nombre_archivo", "r")`

Ejemplo de lectura de archivo (leeArch.py)

```
# Lectura del contenido de un archivo y visualización por pantalla
# archivo es el descriptor que permite a Python acceder al archivo
# Esta vez, open() utiliza el modo "r"
archivo = open("datos.txt", "r")
contenido = archivo.read()
# Muestra por pantalla el contenido del archivo
print("Muestra el archivo:\n=====\n", contenido)
# Cierre del archivo
archivo.close()
```

La ejecución de este script devuelve por pantalla el contenido del archivo:

```
pi@raspberrypi ~ $ python3 leeArch.py
Muestra el archivo:
=====
línea 1 del archivo de datos
línea 2 del archivo de datos
línea 3 del archivo de datos
línea 4 del archivo de datos
fin del archivo de datos

pi@raspberrypi ~ $
```

4. Eliminación de un archivo

Para eliminar un archivo, es necesario disponer de un archivo para las pruebas: cree un archivo `prueba.txt` que contenga algunos caracteres.

La eliminación de un archivo necesita el acceso al sistema. En Python, este acceso se proporciona con el módulo `OS`. `OS` acepta muchos comandos en forma `os.comando(argumentos)`. Para eliminar un archivo, es necesario usar `os.remove(ruta)`.

Ejemplo de script para eliminar un archivo (`supArch.py`)

```
# Eliminación del archivo prueba.txt
# Importación del módulo os que gestiona los comandos Linux
import os
# Eliminación del archivo
os.remove("prueba.txt")
```

Si el archivo no existe, el script devuelve un error:

```
pi@raspberrypi ~ $ python3 supArch.py
Traceback (most recent call last):
  File "supArch.py", line 7, in <module>
    os.remove("prueba.txt")
OSError: [Errno 2] No such file or directory: 'prueba.txt'
```

Si el archivo existe, no se devuelve ningún mensaje de error.

```
pi@raspberrypi ~ $ python3 supArch.py
pi@raspberrypi ~ $
```

5. Gestión de excepciones

Dentro de un script, es posible administrar las excepciones. En caso de eliminación del archivo, sería mejor anunciar al usuario que el archivo no existe, en lugar de mostrar un mensaje de error poco comprensible. Este mecanismo utiliza las instrucciones `try...except`. Si el archivo no existe, `os.remove` pone a 1 la variable `OSError`, utilizada por `except` para desencadenar la visualización del mensaje.

Ejemplo de script para eliminar un archivo con gestión de errores (`gesterr.py`)

```
# Eliminación del archivo prueba.txt
# Importación del módulo os que gestiona los comandos Linux
import os
# Eliminación del archivo
try:
    os.remove("prueba.txt")
except OSError:
    print("Atención: el archivo no existe.\n")
```

Si el archivo existe, no se devuelve ningún mensaje. Si el archivo no existe, se muestra el mensaje previsto en el script en lugar del mensaje de error interno.

Una primera ejecución del script no da ningún problema, porque el archivo acaba de ser creado y existe.

```
pi@raspberrypi ~ $ python3 supArch2.py
pi@raspberrypi ~ $
```

Una segunda ejecución del script, cuando el archivo se ha eliminado y ya no existe, provoca la visualización del mensaje de error incluido en el script.

```
pi@raspberrypi ~ $ python3 supArch2.py
Atención: el archivo no existe.

pi@raspberrypi ~ $
```

Ir más allá con Python

Las páginas de este capítulo sobre Python tienen como objetivo presentar el lenguaje y permitir a un usuario de la Raspberry Pi entender un programa escrito en Python, con el objetivo de aportarle modificaciones.

Si tiene que crear programas para un uso profesional con Python, puede utilizar el libro *Python 3, los fundamentos del lenguaje*. Este libro de Sébastien Chazallet, aparecido en Ediciones ENI dentro de la colección Recursos Informáticos, permite conocer cómo usar Python para resolver problemas (protocolos, servidores, imágenes...). Tiene numerosos ejemplos, en forma de tutoriales, que cubren los dominios actuales: gestión de datos, interfaces web, gráficos, etc.

Como todo lenguaje de programación, Python necesita una inversión que será largamente recompensada por la satisfacción de poder escribir sus propios programas en función de sus necesidades.

Descripción de la interfaz GPIO

El capítulo Descripción técnica presenta la GPIO (*General Purpose Input/Output* = entrada/salida de uso general), así como sus revisiones. Es importante referirse a este capítulo antes de conectar la Raspberry Pi a elementos externos.

La GPIO ha evolucionado considerablemente desde su primera versión. Algunas clavijas han cambiado de función a lo largo de las versiones y el número de clavijas ha pasado de 26 a 40. Las 26 primeras clavijas de la GPIO de las Raspberry Pi Zero y Raspberry Pi 3 son estrictamente idénticas a las de la Raspberry Pi modelo B Rev.2. Esto garantiza la compatibilidad entre estos modelos y permite usar tarjetas adicionales de la generación anterior.

Consulte la tabla que presenta la distribución de las clavijas de la GPIO en el capítulo Descripción técnica.

1. Funciones disponibles

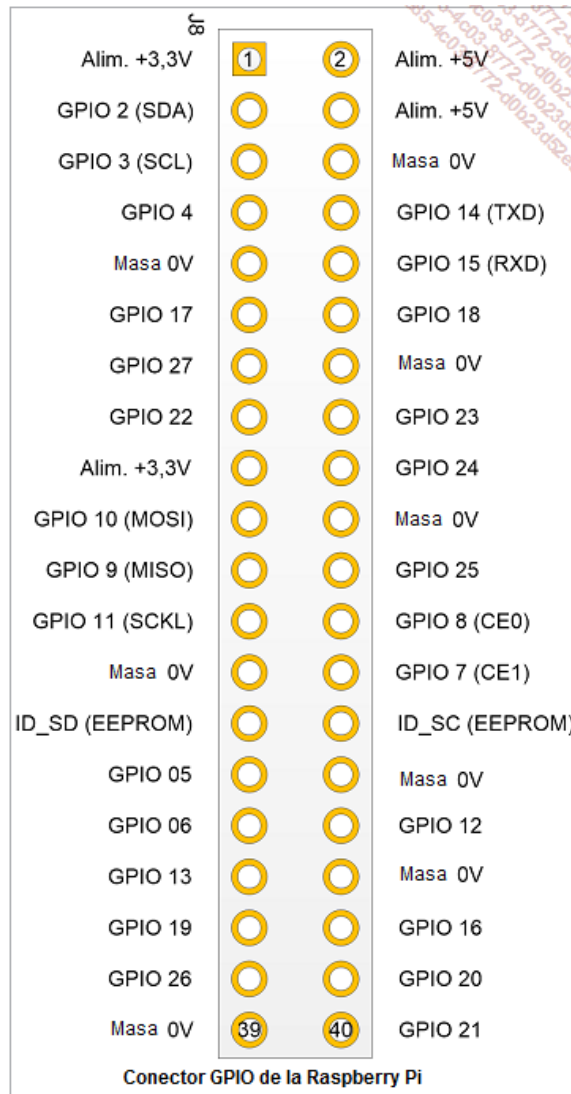
Las clavijas GPIO de la Raspberry Pi están conectadas a un conector de 40 puntos J8, situado en un borde de la tarjeta. Las clavijas del conector están repartidas a lo largo de dos franjas de 20 clavijas. Hay 26 clavijas GPIO accesibles (GPIO 2 a GPIO 27).

Se corresponden con las entradas/salidas digitales, capaces de enviar y recibir señales digitales 1 y 0, en forma de tensión 0 y 3,3 voltios.

Algunas clavijas se pueden utilizar para ofrecer indistintamente un bus I2C, un bus SPI o una E/S UART. En este caso, las clavijas correspondientes no se pueden utilizar más como entradas-salidas digitales.

La distribución de las clavijas del conector GPIO se define según el siguiente esquema. El número de la GPIO en los SoC BCM2835 y BCM2837 figura en las clavijas de los extremos del conector. Las clavijas impares están a la izquierda y las pares en la parte derecha.

➤ La GPIO es idéntico en las BCM2835 y BCM2837. A lo largo de este capítulo el SoC se designará como BCM283x.

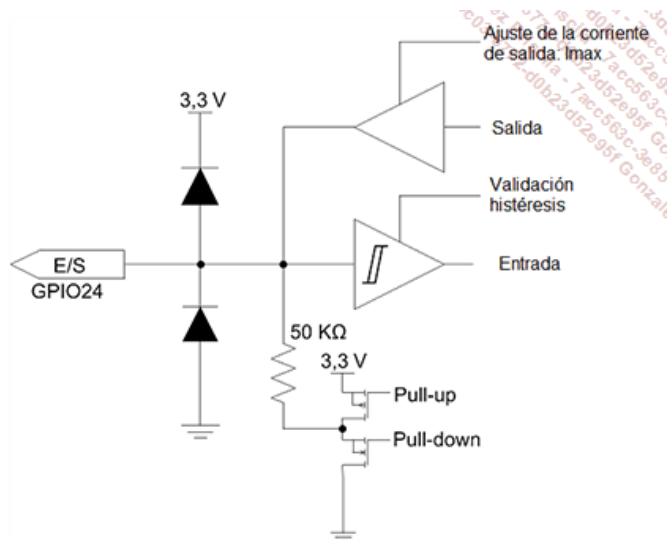


2. Valores límite (V e I)

Como en todo circuito electrónico, los puertos de la GPIO tienen limitaciones de tensión (V) y de corriente (I), que hay que conocer e integrar para usar la Raspberry Pi con toda seguridad.

a. Esquema de la interfaz GPIO

El siguiente esquema es una representación simplificada de la salida de una clavija GPIO del SoC Broadcom BCM283x.



La clavija de entrada/salida (E/S GPIO24) está a la izquierda del esquema. A la derecha de este punto de acceso físico, los componentes representados están en el SoC. Todas las E/S GPIO se basan en el mismo circuito.

Es posible ajustar la corriente de salida de 2 mA a 16 mA, en pasos de 2 mA, seleccionar *slew rate* o velocidad de escaneo de la señal y poner en servicio la histéresis en la entrada. Estas opciones son válidas para el conjunto de clavijas de la GPIO. Es imposible modificar por separado cada clavija.

- La histéresis permite formatear una señal utilizando umbrales diferentes en los extremos positivo y negativo. Se usa para eliminar el ruido de una señal, por ejemplo.

b. Umbrales de la señal de entrada

La puerta de entrada acepta tensiones entre 0 y 3,3 voltios. Decide si la señal de entrada es baja o alta, comparándola con un umbral de tensión. Por defecto, el umbral vale alrededor de 1,8 voltios pero no está garantizado. El umbral se puede situar en cualquier punto entre la tensión máxima, definida por un 0 y la mínima, definida por un 1, es decir entre 0,8 y 2 voltios.

c. Histéresis

Es posible poner en servicio la histéresis (la puerta de entrada toma el nombre de trigger de Schmitt) en las puertas de entrada. Si la histéresis está activada, los umbrales son diferentes para el extremo positivo y para el negativo de la señal. Esto permite eliminar eventuales perturbaciones presentes en las señales. Broadcom ya no publica los valores de los umbrales para la BCM283X.

d. Limitación de la corriente de salida

La puerta situada en la salida puede ofrecer un nivel bajo (0 voltios) o alto (3,3 voltios). Solo se representa una salida en el esquema simplificado. En realidad, son varias puertas las que hay ubicadas en paralelo y se pueden activar por software. En función del número de puertas en servicio, la corriente de salida puede variar de 2 mA a 16 mA, en pasos de 2 mA. Esta corriente está disponible tanto cuando la puerta emite corriente (nivel alto), como cuando la absorbe (nivel bajo).

Atención: la limitación de corriente no corresponde a la puesta en servicio de un dispositivo cualquiera, protegiendo la salida del SoC BCM283X. Si la corriente de salida está limitada a 16 mA, esto significa que más allá de este valor de corriente el nivel no se garantiza. Por ejemplo, si la puerta de salida tiene un nivel alto con una limitación de corriente de 16 mA, y la corriente alcanza 20 mA, la tensión de salida va a caer por debajo del umbral, asegurando que la señal se ve como un nivel alto por los siguientes circuitos. Además, la corriente de salida de la puerta va a sobrepasar el límite autorizado por el fabricante. Corre el riesgo de deteriorarse definitivamente.

e. Resistencia pull-up

La resistencia pull-up conecta la entrada al raíl +3,3 V o a la masa de 50 k Ω en el esquema. Broadcom no ofrece su valor, pero está entre 40 k Ω y 100 k Ω . Los transistores que conectan la resistencia al raíl +3,3 voltios (*pull-up*) o a la masa (*pull-down*) contribuyen a esta indeterminación: su resistencia interna varía en función de la tensión presente en la entrada.

f. Configuraciones peligrosas

Corriente máxima de salida

Recuerde que el valor de la corriente de salida especifica únicamente la corriente máxima por debajo de la cual se respetan las especificaciones del nivel de salida. Esta no es ni la corriente que la puerta es capaz de entregar ni una limitación de corriente que impedirá a la puerta ser destruida.

La puerta de salida es un generador de tensión. En el nivel alto, hace todo lo posible por alcanzar la salida a 3,3 voltios. En el nivel bajo, intenta alcanzar la salida a 0 voltios.

El hecho de que llegue (o no) depende únicamente de lo que esté conectado exteriormente a la clavija GPIO.

Si hay un cortocircuito en la clavija GPIO con la masa (0 voltios), la puerta de salida será incapaz de garantizar la presencia de un nivel alto en la clavija. De hecho, intentará al menos enviar el máximo de corriente. El valor de la corriente solo estará limitado por la resistencia interna de la puerta, que será destruida casi al instante.

El mismo razonamiento se aplica si la puerta intenta entregar un nivel bajo y está conectada por un cortocircuito al raíl 3,3 voltios.

Por tanto, hay que estar muy atento cuando se conectan circuitos externos, porque un error puede deteriorar irremediablemente el SoC.

Un consumo importante de corriente en las salidas puede implicar una disipación térmica excesiva del SoC. Hay que evitar que las salidas absorban o generen más corriente de la prevista. Si la corriente máxima de la puerta se fija a 2 mA, la corriente máxima utilizada debe permanecer inferior a este valor.

Corriente total en las salidas

La alimentación a 3,3 voltios de la Raspberry Pi se ha diseñado para una corriente máxima de 3 mA para la salida GPIO. Si cada una de las 26 clavijas de la GPIO debe salir a 16 mA, la corriente alcanzará 26 x 16 mA, es decir, 416 mA. Para una corriente como esta, la tensión proporcionada por la alimentación cae y los condensadores de desacoplamiento repartidos en la tarjeta se utilizan para generar el

excedente de corriente.

Tales picos de corriente pueden ocurrir cuando todas las salidas GPIO conmutan al mismo tiempo. Son susceptibles de provocar bajadas de tensión que pueden afectar incluso a la memoria RAM o la tarjeta micro SD. Puede provocar paradas o reinicios de la Raspberry Pi.

Sobrepasar la tensión

La Raspberry Pi trabaja bajo una tensión de 3,3 voltios. La conexión de una salida GPIO a una tensión superior a 3,3 voltios o inferior a 0 voltios puede dañar al instante los diodos de protección de esta salida.

Si la salida se debe someter a tensiones más allá de la franja 0 - 3,3 voltios, es necesario prever una resistencia de protección externa, calculada para limitar la corriente proporcionada o absorbida por la salida a 0,5 mA.

Cargas capacitivas

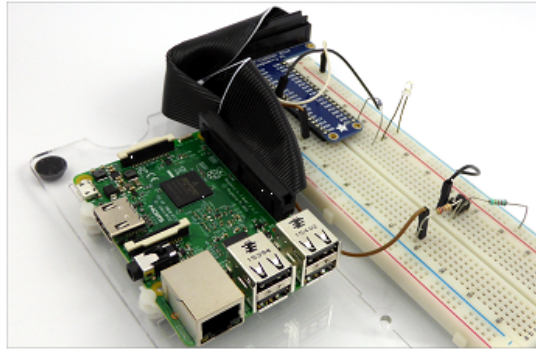
Las cargas capacitivas (condensadores o componentes que tienen un comportamiento parecido) demandan una corriente importante durante el establecimiento de la tensión. No conecte una carga capacitiva directamente a una salida GPIO. Una resistencia exterior en serie con la carga debe limitar la corriente al valor máximo permitido para la salida GPIO. Por ejemplo, para 16 mA, se debe insertar una resistencia como mínimo de $3,3 \text{ voltios} / 16 \text{ mA} = 200 \ \Omega$.

Uso de la GPIO

En la Raspberry Pi, el acceso a la GPIO se puede hacer directamente por línea de comandos o utilizando librerías o interfaces desarrolladas para este propósito. Para visualizar el estado de una salida GPIO, el diodo LED (*Light Emitting Diode* = diodo electroluminiscente) es un medio sencillo y fácil de implementar.

1. Configuración utilizada

Las pruebas se realizan con una Raspberry Pi 3, que funciona con Raspbian. La Raspberry Pi está conectada a una tarjeta de pruebas mediante un cable Pi Cobbler.

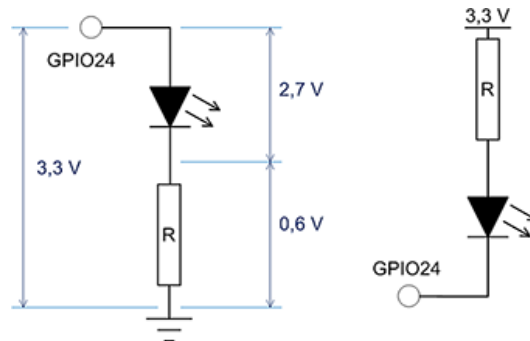


2. Encender un diodo LED

Los programadores tienen una costumbre: el primer programa que hacen en un lenguaje de programación muestra "Hello world" u "Hola mundo". Esta costumbre se ha transmitido al mundo de los sistemas embebidos: el primer programa consiste en provocar un destello de un LED.



El LED utilizado es un diodo de luz blanca que emite luz cuando lo recorre una corriente en sentido directo. La localización de las pestañas de un LED se representa en el anterior esquema. El cátodo es la pestaña más corta del componente. También se corresponde con la parte plana del cuello del LED.



El diodo LED blanco se puede conectar a la clavija GPIO 24 de dos maneras.

- El primer método (a la izquierda del esquema anterior), enciende el LED cuando el nivel de salida de la clavija GPIO 24 es alto (3,3 voltios). El ánodo está conectado a la clavija GPIO 24 y el cátodo está unido a la masa a través de una resistencia destinada a limitar la corriente.
- El segundo método (a la derecha del esquema anterior), enciende el LED cuando el nivel de salida de la clavija GPIO 24 es bajo (0 voltios). El cátodo está conectado a la clavija GPIO 24. El ánodo recibe los 3,3 voltios a través de una resistencia.

En ambos casos, la tensión presente en los bornes del conjunto diodo-resistencia, se reparte entre los dos componentes. La curva característica del LED impone una tensión de 2,7 voltios a sus bornes. La permanencia de la tensión disponible $3,3 \text{ voltios} - 2,7 \text{ voltios} = 0,6 \text{ voltios}$ aparece en los bornes de la resistencia.

Las características del LED utilizado indica una corriente de 20 mA, más allá de las posibilidades de la salida GPIO (16 mA). El LED emite luz a partir de una corriente inferior a 1 mA, con una luminosidad suficiente para las pruebas. Este valor de corriente no hace correr ningún riesgo a la salida GPIO 24 en la que está conectado el LED.

Para las aplicaciones en las que se debe obtener la luminosidad máxima del LED será necesario prever un buffer capaz de ofrecer la intensidad requerida (transistor, ULN2803...).

👉 El ULN2803 es un circuito integrado que contiene ocho transistores capaces de suministrar una carga de 500 mA bajo 50 voltios.

El valor de la resistencia se determina por $R = U/I$, con $U = 0,6 \text{ voltios}$ e $I = 1 \text{ mA}$. Un valor cercano a $600 \ \Omega$ valdrá (560 o $680 \ \Omega$).

En los ejemplos que siguen, el LED se conecta a la GPIO 24 de acuerdo con el esquema izquierdo: ánodo conectado a la GPIO 24 y cátodo conectado a la resistencia que se une a la masa.

3. Leer un nivel de entrada

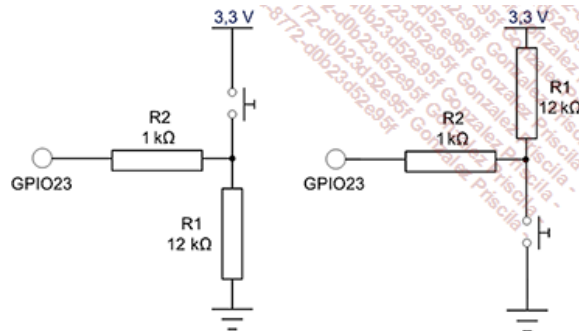
Las clavijas de la GPIO de la Raspberry Pi también se pueden configurar como entrada. Por tanto, pueden detectar si el nivel presente en la entrada es alto o bajo. Configurada como entrada, la clavija GPIO es flotante, es decir, no tiene nivel de tensión definido. Para garantizar el funcionamiento correcto de la entrada será necesario fijar su nivel con una resistencia pull-up.

RECORDATORIO IMPORTANTE: las clavijas GPIO de la Raspberry Pi no están protegidas. No se deben someter a una tensión inversa, ni a una tensión superior a 3,3 voltios. En caso contrario, corre el riesgo de destruirlas. Es obligatorio comprobar detenidamente los circuitos antes de

enchufarlos.

Cuando sea necesario desencadenar una acción en la Raspberry Pi, normalmente la solución que se adopta es un interruptor conectado a una entrada de la GPIO.

Según la necesidad, se puede conectar un botón de accionamiento al raíl 3,3 voltios o a la masa. La resistencia R1 de 12 kΩ limita la corriente a 0,3 mA, mientras se pulsa el accionador. La resistencia R2 de 1 kΩ no tiene efecto si GPIO23 está configurado como entrada. Debido a la superior impedancia de la entrada, la corriente que circula en R2 solo provoca una caída de tensión mínima, sin afectar al funcionamiento de la entrada.



Por el contrario, en el esquema izquierdo, en caso de que el puerto GPIO23 esté configurado como salida y pase a 0, en ausencia de resistencia R2 la salida está conectada directamente al raíl +3,3 voltios si se pulsa el accionador. La salida se destruye por esta conexión directa a la alimentación. La resistencia R2 sirve de protección limitando la corriente en la salida a 3,3 voltios / 1 kΩ = 3 mA. La GPIO está protegida.

La función de la resistencia R2 es idéntica en el esquema derecho: cuando la GPIO23 se configura como salida con un nivel alto y se acciona el conmutador, R2 limita la corriente a unos 3 mA.

Para los scripts de lectura del puerto GPIO23 se utiliza el esquema izquierdo con una resistencia pull-up conectada a la masa.

4. Administrar la GPIO por línea de comandos

En los sistemas Linux, todo es un archivo. Por ejemplo, la carpeta `/dev` mencionada en el capítulo Usar la línea de comandos contiene los archivos que son el medio de acceso del sistema a los periféricos, tanto en modo escritura como lectura (discos duros, terminales, impresoras, etc.). Este enfoque estandariza los accesos y simplifica la tarea de los desarrolladores.

La GPIO es accesible por línea de comandos a través de un sistema de archivos disponible en el espacio del usuario (*userspace*). En este espacio, los comandos o aplicaciones no pueden acceder, ni siquiera de manera accidental, a una zona memoria que no les pertenezca.

Solo `root` puede intervenir en el estado de las entradas/salidas de la Raspberry Pi.

a. Acceso a la GPIO

Las clavijas de la GPIO están accesibles en `/sys/class/gpio`. Esta carpeta contiene:

```
pi@raspberrypi:/sys/class/gpio $ ls -al
total 0
drwxrwx--- 2 root gpio 0 jul. 8 18:51 .
drwxr-xr-x 51 root root 0 ene. 1 1970 ..
-rwxrwx--- 1 root gpio 4096 jul. 8 18:51 export
lrwxrwxrwx 1 root gpio 0 jul. 8 18:51 gpiochip0 -> ../../
devices/platform/soc/3f200000.gpio/gpio/gpiochip0
lrwxrwxrwx 1 root gpio 0 jul. 8 18:51 gpiochip100 -> ../../
devices/platform/soc/soc:virtgpio/gpio/gpiochip100
-rwxrwx--- 1 root gpio 4096 jul. 8 18:51 unexport
```

Los archivos `export` y `unexport` se utilizan para crear un acceso a una clavija de la GPIO. `gpiochip0` contiene la información de la GPIO.

- `export`: archivo en modo solo escritura que permite solicitar al núcleo que exporte el control de una clavija de la GPIO al espacio de usuario. Basta con escribir el número de la clavija de la GPIO en la exportación con una redirección para realizar esta operación. Atención, se trata del número de la clavija de la GPIO en el SoC y no del número de la clavija del conector de salida GPIO, situado en la tarjeta del circuito impreso de la Raspberry Pi.
- `gpiochip0`: el controlador de la GPIO se accede con `gpiochip0`. En esta carpeta figuran, en particular, los archivos en modo solo lectura que permiten leer la información en la GPIO:

```
pi@raspberrypi:/sys/class/gpio/gpiochip0 $ ls -al
total 0
drwxrwx--- 3 root gpio 0 jul. 8 18:51 .
drwxr-xr-x 3 root root 0 jul. 8 18:51 ..
-rwxrwx--- 1 root gpio 4096 jul. 8 18:51 base
lrwxrwxrwx 1 root gpio 0 jul. 8 18:51 device -> ../../../../
3f200000.gpio
-rwxrwx--- 1 root gpio 4096 jul. 8 18:51 label
-rwxrwx--- 1 root gpio 4096 jul. 8 18:51 ngpio
drwxrwx--- 2 root gpio 0 jul. 8 18:51 power
lrwxrwxrwx 1 root gpio 0 jul. 8 18:51 subsystem -> ../../../../
../../class/gpio
-rwxrwx--- 1 root gpio 4096 jul. 8 18:51 uevent
```

Los archivos más útiles de esta carpeta son:

- `base` que devuelve el número del controlador de la GPIO.
- `label` que devuelve el nombre asignado a la GPIO.
- `ngpio` que indica el número de clavijas de la GPIO en el SoC. Es interesante observar que `ngpio` devuelve 54, mientras que solo 26 clavijas de la GPIO son accesibles. No todas las clavijas son accesibles, o se utilizan para controlar otras partes de Raspberry Pi.
- `unexport`: archivo en modo solo escritura, permite informar al núcleo que debe eliminar el acceso a una clavija de la GPIO. Hay que escribir el número de la clavija cuyo acceso es necesario eliminar en `unexport` con una redirección.

b. Creación de un acceso a una clavija GPIO

Por defecto, no existe ningún acceso a una clavija de la GPIO. Para poder usar una clavija, en primer lugar es necesario solicitar que se cree el acceso:

Una nueva carpeta `gpio24` aparece en la carpeta `/sys/class/gpio`. Es el punto de entrada para configurar la clavija GPIO 24 del SoC:

```
pi@raspberrypi:/sys/class/gpio# $ echo 24 > export
```

```
pi@raspberrypi:/sys/class/gpio $ ls -al gpio24
lrwxrwxrwx 1 root gpio 0 jul.  8 19:23 gpio24 -> ../../devices/
platform/soc/3f200000.gpio/gpio/gpio24
```

Para crear accesos a otras clavijas de la GPIO es necesario repetir la operación, indicando cada vez el número de la clavija a la que se desea acceder.

c. Eliminación del acceso a una clavija GPIO

De la misma manera que el archivo *export* informa al núcleo que debe crear un acceso a una clavija de la GPIO, el archivo *unexport* le va a indicar que debe eliminar un acceso. Para eliminar el acceso a la GPIO 24:

```
pi@raspberrypi:/sys/class/gpio $ ls
export gpio24 gpiochip0 gpiochip100 unexport
pi@raspberrypi:/sys/class/gpio $ echo 24> unexport
```

Después de escribir el número de la clavija (aquí 24) en el archivo *unexport*, la carpeta *gpio24* desaparece:

```
pi@raspberrypi:/sys/class/gpio $ ls
export gpiochip0 gpiochip100 unexport
```

d. Recuperar la información de una clavija de la GPIO

Es posible recuperar la información de dirección (clavija como entrada o como salida) y el valor existente actualmente en la clavija de la GPIO:

```
pi@raspberrypi:/sys/class/gpio $ echo 24 > export
pi@raspberrypi:/sys/class/gpio $ ls
export gpio24 gpiochip0 gpiochip100 unexport
pi@raspberrypi:/sys/class/gpio $ cd gpio24
pi@raspberrypi:/sys/class/gpio/gpio24 $ ls
active_low device direction edge power subsystem uevent value
pi@raspberrypi:/sys/class/gpio/gpio24 $ cat direction
in
pi@raspberrypi:/sys/class/gpio/gpio24 $ echo out > direction
pi@raspberrypi:/sys/class/gpio/gpio24 $ echo 1 > value
pi@raspberrypi:/sys/class/gpio/gpio24 $ cat value
1
pi@raspberrypi:/sys/class/gpio/gpio24 $ echo 0 > value
```

Aquí la clavija GPIO 24 está configurada como salida (out) y el valor de salida es 0 (0 voltios). La escritura de un 1 en el archivo *value* hace que la salida pase a 1 (3,3 voltios), lo que se confirma durante la nueva lectura de este archivo. El LED conectado a la GPIO 24 se enciende.

Cuando la clavija GPIO 24 se configura como entrada, la lectura de *value* da el nivel presente en la clavija.

e. Parpadeo de un LED en shell

Utilizando los comandos anteriores, encienda y apague sucesivamente un LED conectado a la GPIO 24. El programa *led_parpadea.sh*, en el anexo, ofrece una solución de script para automatizar el parpadeo.

5. Administrar la GPIO en Python

El acceso a la GPIO en Python se facilita por la existencia de un módulo especializado: RPi.GPIO, que ofrece la manera de controlar fácilmente los puertos GPIO de la Raspberry Pi. La versión disponible de RPi GPIO es la versión 0.6.2 en el momento de la escritura de estas líneas (<https://pypi.python.org/pypi/RPi.GPIO>).

Este módulo no está previsto para aplicaciones en tiempo real:

- Por un lado porque es imposible prever cuándo Python desencadena su recolector de basura (*garbage collector*).
- Por otro porque se ejecuta por el núcleo Linux que no está adaptado actualmente. Se puede asignar otro proceso de manera prioritaria en la CPU, provocando retrasos imprevistos en su programa.

➤ Las versiones de RPi GPIO para Python 2 y Python 3 se instalan por defecto en Raspbian.

Comprobar la instalación de RPi GPIO

Cree un archivo llamado *test_rpigpio.py* con nano, y agregue las siguientes líneas:

```
#!/usr/bin/python3
import RPi.GPIO as GPIO
print(GPIO.VERSION)
print(GPIO.RPI_INFO)
```

Convierta el script en ejecutable con el comando

```
chmod 755 test_rpigpio.py
```

Por último, ejecute el script:

```
pi@raspberrypi:~ $ ./test_rpigpio.py
0.6.2
{'RAM': '1024M', 'REVISION': 'a22082', 'P1_REVISION': 3, 'TYPE':
'Pi 3 Model B', 'PROCESSOR': 'BCM2837', 'MANUFACTURER': 'Embest'}
```

El script devuelve la versión de RPi GPIO (0.6.2), así como una serie de información de la tarjeta utilizada: type Pi 3 Model B, 1024 MB de RAM...

Si aparece esta información, RPi GPIO está instalado en su Raspberry Pi y funciona normalmente. Puede continuar con el resto del capítulo.

a. Controlar un LED

El control de un LED en Python pasa por la carga inicial del módulo *RPi.GPIO* y del módulo *time*.

```

# Script led_parpadea.py

# Importar el módulo de control de la GPIO
# y el módulo time que ofrece las funciones
# relativas a la gestión del tiempo
import RPi.GPIO as GPIO
import time

# indicar que es necesario usar los números de puerto GPIO
GPIO.setmode(GPIO.BCM)
# configuración de las clavijas GPIO a usar
GPIO.setup(24, GPIO.OUT)

while True:
    # Poner la salida al nivel 1
    GPIO.output(24, GPIO.HIGH)
    # Esperar medio segundo
    time.sleep(0.5)
    # Poner la salida al nivel 0

    GPIO.output(24, GPIO.LOW)
    # Esperar medio segundo
    time.sleep(0.5)

```

Hay que indicar a `RPi.GPIO` si los números de clavijas que se usan en el programa son números de clavijas físicas (de 1 a 40) o números GPIO del SoC:

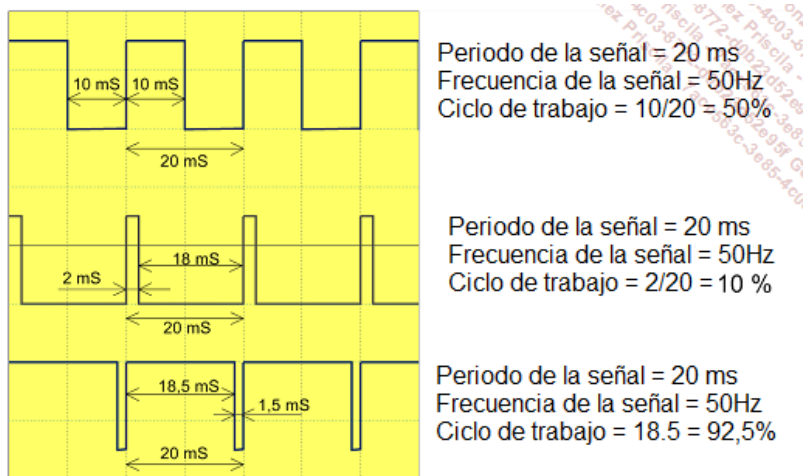
```
GPIO.setmode(GPIO.BCM)
```

Aquí se utilizarán los números GPIO del SoC. A continuación, la salida 24 se pondrá alternativamente al nivel alto (`GPIO.HIGH`) y después al nivel bajo (`GPIO.LOW`), con un tiempo de pausa de 0,5 segundos.

No está prevista la salida de este programa, que se ejecuta indefinidamente. Pulse las teclas [Ctrl] C para interrumpirlo.

Variación de la luminosidad

Desde la versión 0.5.2a, `RPi.GPIO`, dispone de una función de gestión del PWM (*Pulse Width Modulation*= modulación del ancho del pulso). El PWM consiste en hacer variar la relación cíclica de una señal. Esta variación aplicada a un LED modifica la luminosidad resultante, el ojo integra los impulsos y da la sensación de una variación en la intensidad de la luminosidad.



La relación cíclica de una señal rectangular puede variar del 0 % al 100 %. En el ejemplo anterior, la frecuencia es de 50 Hz, el periodo de 20 ms. La relación cíclica es la relación entre el tiempo durante el que la señal está en el estado alto y la duración del periodo.

Cuando la relación cíclica vale 0 %, el LED está completamente apagado. Cuando vale 50 %, el LED está encendido la mitad del tiempo. Cuando la relación cíclica es del 10 %, el LED solo recibe cortos impulsos durante 2 ms cada 20 ms.

El siguiente programa `led_pwm.py` enciende y apaga el LED progresivamente utilizando el PWM.

```

# Script led_pwm.py
# Importar el módulo de control de la GPIO
# y el módulo time que ofrece las funciones
# relativas a la gestión del tiempo
import RPi.GPIO as GPIO
import time

# indicar que es necesario usar los números de puerto GPIO del SoC
GPIO.setmode(GPIO.BCM)
# configuración de las clavijas GPIO a usar
# uso de la clavija GPIO 24 en salida
GPIO.setup(24, GPIO.OUT)

# creación de un objeto para el PWM en el puerto 24
# a una frecuencia de 50 Hertz
led = GPIO.PWM(24,50)

# preparar la variación de ciclo del PWM
cycle = 10
print("Enciende el LED, relación cíclica del 10%")
# iniciar el PWM con un ciclo del 10%
led.start(cycle)
# aumentar el ciclo hasta el 100% por pasos del 10%
for cycle in range (10, 101, 10):
    # mostrar la relación cíclica en pantalla
    print("Relación cíclica = ",cycle)
    # modificar la señal aplicada al LED
    led.ChangeDutyCycle(cycle)
    # esperar medio segundo
    time.sleep(0.5)
# cuando el LED se enciende al 100%, esperar 2 segundos
time.sleep(2)
# reducir el ciclo hasta el 10%
for cycle in range (100, 0, -10):

```

```

# mostrar la relación cíclica en pantalla
print("Relación cíclica = ",cycle)
# modificar la señal aplicada al LED
led.ChangeDutyCycle(cycle)
# esperar medio segundo
time.sleep(0.5)
# Cuando el LED se apaga, esperar 2 segundos
time.sleep(2)

# salir correctamente del programa
led.stop()
GPIO.cleanup()
print("Apagar el LED y parar el programa")

```

El programa tiene dos bucles *for*: el primero para aumentar la luminosidad del LED y el segundo para reducir la luminosidad. Los comandos `led.start()` y `led.stop()` arrancan y detienen el PWM en la clavija 24. El comando `led.ChangeDutyCycle()` modifica la relación cíclica de la señal PWM.

Servomotores

Los servomotores de radio control reciben el mismo tipo de señal. La frecuencia también es de 50 Hz, pero los impulsos solo varían de 1 a 2 ms y provocan una rotación de 180 grados. Para un impulso de 1 ms, el servomotor actúa en un sentido, para 2 ms en el otro sentido y para 1,5 ms está en posición central.

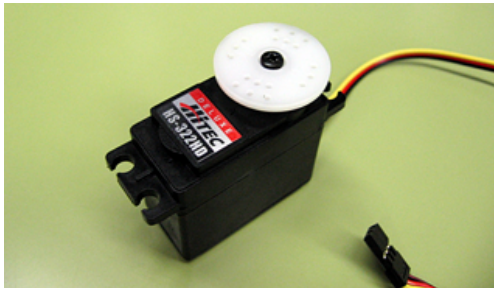


Photo J.L.Galvez - Licencia CC2.5

Un servomotor se puede controlar por la Raspberry Pi. El sistema operativo de la Raspberry Pi no es un sistema en tiempo real. De modo que hay que aceptar las variaciones de posición o manifestaciones de los sensores del servomotor causadas por señales variables, provocadas por la prioridad de algunos proceso respecto a otros.

b. Leer el nivel de una entrada

La lectura de una entrada GPIO necesita el uso del módulo `RPi.GPIO`. Este módulo se importa al inicio del script `lectura_gpio.py`. Se usa un bucle `while` para mostrar permanentemente el estado del botón de activación. Para salir del bucle, use [Ctrl] C. En este script, que usa la clavija GPIO 23, será necesario activarla inicialmente.

Recuerde que el esquema utilizado tiene una resistencia pull-up conectada a la masa. Cuando se suelta el botón, el nivel de entrada en GPIO 23 es de 0 voltios.

```

# Script lectura_gpio.py
# Importar el módulo de control de la GPIO
import RPi.GPIO as GPIO

# indicar que es necesario usar los números de puerto GPIO del SoC
GPIO.setmode(GPIO.BCM)

# configuración de las clavijas GPIO a usar
# usar la clavija GPIO 23 como entrada
GPIO.setup(23, GPIO.IN)

# lectura permanente del estado del botón
while True:
    if (GPIO.input(23)):
        print("Botón de activación pulsado")

```

El script utiliza un bucle infinito `while True:` para leer la entrada GPIO 23. `setmode` precisa el uso de los números de puerto del SoC. `setup` configura la clavija GPIO como entrada. `input` lee la entrada GPIO 23.

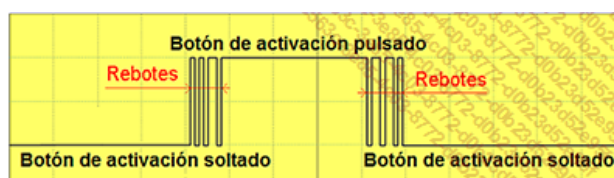
La ejecución de este script seguida de una acción muy breve en el botón de activación provoca la escritura por pantalla de muchas líneas:

```

pi@raspberrypi ~ $ sudo python3 lectura_gpio.py
Botón de activación pulsado
Botón de activación pulsado
Botón de activación pulsado
Botón de activación pulsado
Botón de activación pulsado
Botón de activación pulsado
...

```

El botón de activación está en el origen de estas numerosas detecciones. La sustitución del botón de activación por un producto de mejor calidad no tiene efecto. Durante la pulsación en el botón, se establece el contacto pero se producen rebotes, provocando una sucesión de cambios en el estado de la entrada GPIO. El programa detecta todos estos cambios y muestra una línea para cualquier valor 1 detectado en la entrada.



Además, la pulsación en el botón de activación, incluso aunque parezca rápido para un humano, deja al script el tiempo suficiente para hacer muchos bucles. Durante el bucle, el script muestra el texto *Botón de activación pulsado*.

Es posible añadir una pieza de hardware anti-rebotes (báscula, circuito RC o trigger de Schmitt), pero esto complica la implantación de una sencilla activación. La eliminación de los rebotes mediante software es más elegante (y menos cara).

El anti-rebote basado en software se desencadena por la detección del cambio de estado de la entrada GPIO. Cuando se produce el cambio, se tiene en cuenta y se muestra por pantalla. Un temporizador deja tiempo a los rebotes para terminar, antes de probar de nuevo el estado del botón de activación.

```
# Script antirebote.py
# Condición de inicio: el botón de activación se debe soltar
#
# Importar el módulo de control de la GPIO
import RPi.GPIO as GPIO
# Importar el módulo time para el temporizador
import time

# crear la variable para recordar el estado de la salida
# de activación cuando se suelta, entrada_inicial vale 0
entrada_inicial = 0

# indicar que es necesario usar los números de puertos GPIO del SoC
GPIO.setmode(GPIO.BCM)

# configuración de las clavijas GPIO a usar
# uso de la clavija GPIO 23 en entrada
GPIO.setup(23, GPIO.IN)

# lectura del estado del botón
while True:
    entrada = GPIO.input(23)
    # si ha habido un cambio en el estado, mostrar
    # por pantalla el texto que indica la pulsación del botón
    if (not(entrada_inicial) and entrada):
        print("Botón de activación pulsado")
    # guardar el estado actual en la memoria
    entrada_inicial = entrada
    # esperar a que los rebotes terminen
    time.sleep(0.1)
```

La comprobación se realiza sobre el valor anterior del nivel en la entrada GPIO 23. Al inicio, el valor de `entrada_inicial` es 0. Mientras el botón de activación permanece en reposo, el nivel de entrada de GPIO 23 permanece a 0. En este caso:

`entrada_inicial = 0` y `entrada = 0`, por tanto, `(not entrada_inicial) = 1`

y `((not entrada_inicial) and entrada)` vale `(1 and 0)` es decir 0: la condición del `if` es falsa.

Durante una pulsación en el botón de activación, la señal de entrada pasa a 1. Tan pronto como la señal es 1, `((not entrada_inicial) and entrada)` vale `(1 and 1)` es decir 1: la condición del `if` se convierte en verdadera. Se muestra en pantalla el texto **Botón de activación pulsado**. La variable `entrada_inicial` se convierte en 1 y el script marca una pausa de 100 ms con `time.sleep(0.1)`. Este lapso de tiempo permite esperar a que los rebotes terminen antes de continuar el programa.

Esta manera de proceder puede ralentizar la activación de una acción controlada por una pulsación en el botón de activación, pero es suficiente la mayor parte del tiempo. Otro enfoque consiste en detectar un cambio de estado, sea cual sea, y esperar a que el estado de la entrada se estabilice. Cuando tres o cuatro lecturas sucesivas devuelvan el mismo resultado, se puede considerar que los rebotes han terminado y se transmite el valor al resto del programa.

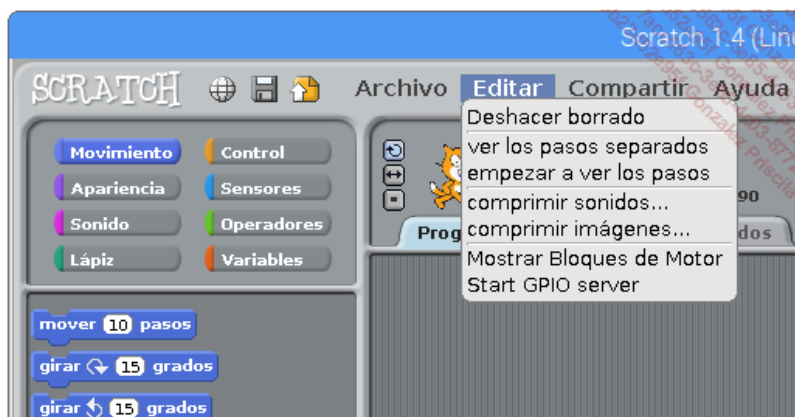
Gracias al anti-rebote basado en software, cada pulsación en el botón de activación solo provoca una visualización de la frase **Botón de activación pulsado**.

6. Acceder a la GPIO desde Scratch

Desde septiembre de 2015, la versión de Scratch incluida en Raspbian Jessie integra un servidor GPIO que permite controlar los LED, los buzzers, leer el estado de los botones, etc. Con este servidor, Scratch también es capaz de gestionar un determinado número de tarjetas HAT:

- Piglow, una tarjeta que tiene 18 LED RGB, que se controlan de manera individual.
- PiFace una tarjeta que ofrece ocho entradas y ocho salidas, de las cuales dos tienen relés.
- PiBrella, una tarjeta muy sencilla pero muy bien hecha para aprender a utilizar la GPIO: tiene un botón, tres LED (uno rojo, otro naranja y uno último verde) y un vibrador.
- Explorer HAT Pro, más compleja de implementar, comprende los componentes relacionados directamente con la GPIO y otros al bus I2C. Tiene cuatro LED, cuatro entradas con buffer, dos controladores de motor en H, cuatro entradas analógicas y cuatro teclas capacitivas.
- Sense HAT, la tarjeta que el astronauta Timothy PEAKE ha incorporado y utilizado en el ISS durante los seis primeros meses de 2016. Está equipada con muchos sensores (temperatura, humedad, presión, acelerómetro, giroscopio, magnómetro, compás magnético, etc) y de un mini joystick. Una matriz de LED RGB de formato 8x8 permite visualizar mensajes o imágenes.
- PiLite, una matriz de 126 LED organizados en 14x9. Permite ver los mensajes, jugar, etc... Hay disponible un emulador de la tarjeta PiLite para la Raspberry Pi (https://github.com/CisecoPlc/PiLite/tree/master/Python_Examples).
- Ryantek, Polulu, Camjam Edukit 3 MOTOR CONTROLLER son tarjetas de control de motores para la robótica. Aunque estas tarjetas tienen un diseño diferente, su uso con Scratch tiene exactamente los mismos controles.

Por defecto, el servidor no está activo durante el arranque de Scratch. Es posible iniciarlo con el menú **Editar - Start GPIO server** (siguiente imagen).



Pulse en la última línea del menú **Start GPIO server** para permitir a Scratch acceder a la GPIO.

También es posible arrancar el servidor dentro de un programa, enviando el mensaje `gpioserveron` a todos.

Scratch puede acceder a las clavijas de la GPIO en modo entrada, salida todo o nada o en PWM para generar una tensión pseudo analógica (ver *RPi GPIO: Variación de la luminosidad*). También podrá controlar tarjetas adicionales.

➤ Para más información acerca del uso de Scratch, vaya al capítulo Programar en Scratch.

a. Configurar los puertos de la GPIO

La primera etapa antes de utilizar los puertos de la GPIO con Scratch consiste en arrancar el servidor y configurar las entradas/salidas que se van a utilizar.



En el ejemplo anterior, un clic en la bandera verde (arranque del programa) provoca el envío de mensajes a todos (broadcast), que van en el orden:

- Iniciar el servidor GPIO (**gpioserveron**)
- Configurar el puerto 24 en modo salida (**config24out**)
- Configurar el puerto 18 en modo salida PWM (**config18outputpwm**)
- Configurar el puerto 22 en modo entrada (**config22in**)

➤ El número de puertos GPIO indicado aquí no es el número físico de la clavija del conector, sino el del puerto en el SoC BCM2835/BCM2837.

Sintaxis de configuración de las clavijas GPIO

Sintaxis:

config + N° de puerto + <configuración>

<configuración>	Ajuste realizado.
in input inpullup inputpullup	Configura un puerto en modo entrada, con las resistencias a +3,3v conectadas (pull-up).
inpulldown inputpulldown inpullnone	Configura un puerto en modo entrada sin las resistencias a +3,3v conectadas.
out output	Configura un puerto GPIO en modo salida.
Outpwm	Configura un puerto GPIO en modo salida PWM.

Ejemplo:

config12in configura el puerto 12 de la GPIO en modo entrada con las resistencias.

Utilización de las clavijas GPIO

Sintaxis:

gpio + N° de puerto + <valor>

<valor>	Valor a aplicar en la salida.
on high	Pasa la salida a 1 (3,3v).
off low	Pasa la salida a 0 (0v).

Ejemplo:

gpio17on pone el puerto 17 de la GPIO a 1 (3,3v)

Salida PWM

gpio + N° de puerto + pwm + <valor>

<valor>	Valor de la salida PWM entre 0 y 1024.
---------	--

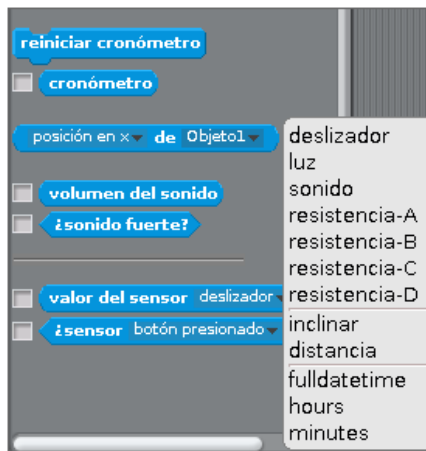
Ejemplo:

gpio22pwm522 pone el puerto 22 de la GPIO a una tensión media de 1,68v (3,3v * 522 / 1024)

Lectura de la hora

Enviar un mensaje a todos conteniendo *gettime*.

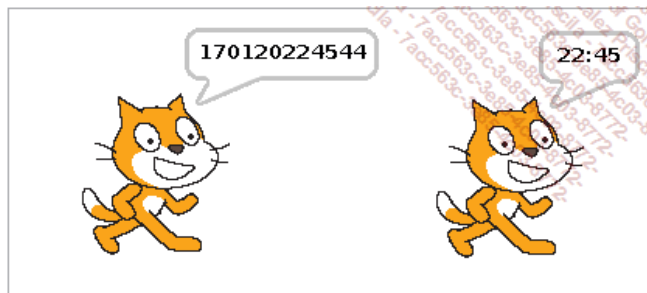
La hora está disponible en la sección de los sensores. Muestre los sensores. Acceda a la lista de sensores disponibles del bloque **valor del sensor**, pulsando en la flecha situada a la derecha del deslizador.



Seleccione **fulldatetime** para mostrar la fecha y la hora en formato AAMDDHHMMSS, poco legible, u opte por la visualización de la hora y los minutos (**hours+minutes**), que habría que concatenar con el operador **unir**.



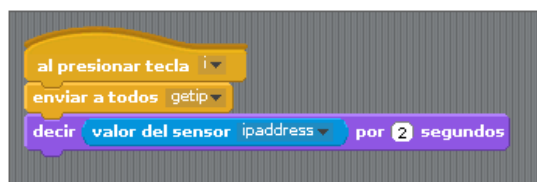
Por ejemplo, su duende le dará la hora de esta manera:



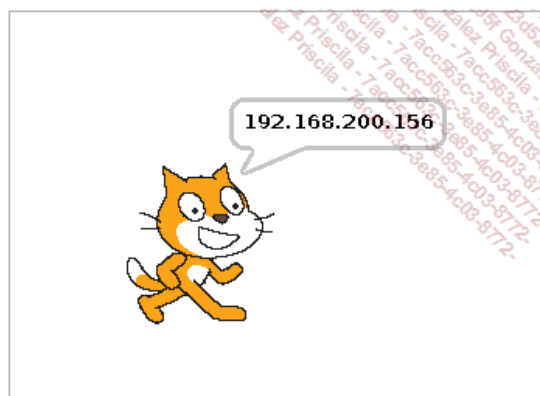
Visualización de la dirección IP

Enviar un mensaje a todos conteniendo **getip**.

La dirección IP de la Raspberry Pi está disponible en la sección de los sensores. Muestre los sensores. Acceda a la lista de sensores disponibles del bloque **valor del sensor**, pulsando en la flecha situada a la derecha del deslizador. Seleccione **getip** e integre este bloque a un programa:



Su duende ahora puede anunciar la dirección IP de la Raspberry Pi.



b. Utilizar una tarjeta adicional

Para utilizar una tarjeta adicional, basta con crear una variable llamada **AddOn** y dar a esta variable el valor definido para la tarjeta seleccionada. La tarjeta *PiFace* se presenta en este libro (consulte el capítulo Los periféricos).

Comando de la tarjeta PiFace

Sintaxis:

output + N° de salida + <valor>

<valor>	Valor a aplicar a la
---------	----------------------

	salida.
on high	Pasa la salida a 1 (3,3v).
off low	Pasa la salida a 0 (0v).

Ejemplo:

output6on pone la salida 6 de la tarjeta PiFace a 1.

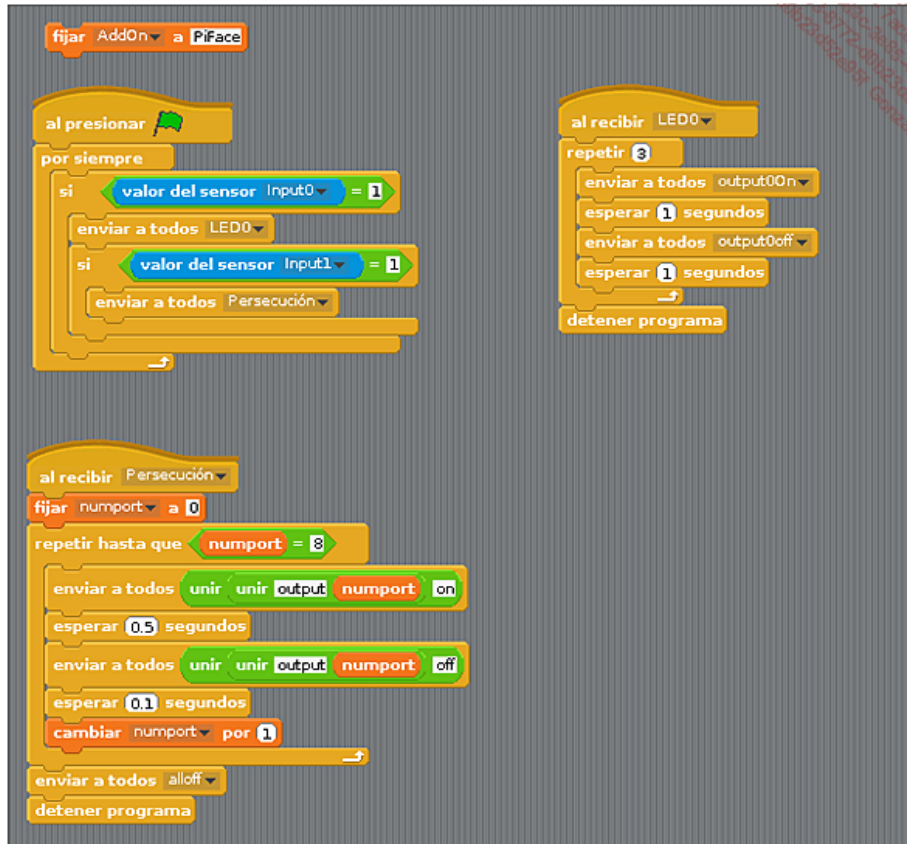
Sintaxis:

all + <valor>

<valor>	Valor a aplicar a la salida.
on	Pone todas las salidas de la tarjeta PiFace a 1 (enciende todos los LED).
off	Pone todas las salidas de la tarjeta PiFace a 0 (apaga todos los LED).

Crea una variable AddOn. En los bloques de esta variable introduzca a **AddOn asignar PiFace** y arrastre este bloque a la zona del programa. El valor de la variable para agregar la tarjeta PiFace es **PiFace**. Esta declaración de tarjeta adicional se encuentra en la parte superior izquierda de la siguiente captura de pantalla.

La adición de este bloque hace que aparezca en la sección **Sensores - valor del sensor** la lista de entradas de la tarjeta PiFace (Input0... Input7).



El programa anterior está disponible descargando *scratch_piface.sb*. Contiene varias secciones:

Cuando Al presionar verde: cuando se pulsa la bandera verde, esta parte del programa entra en un bucle indefinido. Lee los botones 0 y 1 de la tarjeta PiFace. Si se pulsa el botón 0, envía el mensaje **LEDO**, si se pulsa el botón 1, envía el mensaje **Persecución**.

Cuando enviar a todos LED0 : este programa hace parpadear tres veces el LED 0 de la tarjeta PiFace. El relé que tiene asociado parpadea rítmicamente, lo que permite comprobar fácilmente que el programa funciona.

Cuando enviar a todos Persecución: este programa aumenta el número de puerto GPIO de 0 a 7. Enciende durante medio segundo el LED correspondiente y lo apaga. El comando enviado a la tarjeta PiFace se crea concatenando los elementos necesarios con **unir**. Al final del script, el comando **aloff** apaga todos los LED (incluso si no es necesario).

Como ejercicio, añade a este programa la posibilidad de mover el duende por la escena, utilizando los botones S2 y S3 de la tarjeta PiFace. Se propone una solución en los Anexos, disponible para su descarga: *scratch_piface2.sb*.

c. Scratch y el aprendizaje del código

Con la posibilidad de controlar los puertos de la GPIO y gestionar las tarjetas de extensión, Scratch se convierte en una herramienta fundamental para el aprendizaje del código. Ofrece la posibilidad de que un programa interactúe fácilmente con el mundo real. Para los profesores, esta posibilidad abre perspectivas interesantes para que los alumnos se interesen por la realización de programas interactivos.

d. Conclusión

Accesibles por numerosos medios, los puertos de la GPIO abren el acceso de la GPIO al mundo físico. Son relativamente fáciles de configurar, tanto en modo lectura como escritura. El número restringido de puertos GPIO disponibles y sus limitaciones en tensión e intensidad han favorecido el desarrollo de muchas tarjetas de extensión que ofrecen:

- numerosas entradas/salidas.
- entradas/salidas analógicas.
- la posibilidad de controlar potencias más importantes.
- el uso de los buses SPI, I²C o 1-Wire.

El siguiente capítulo presenta alguna de las tarjetas más conocidas.

El módulo cámara

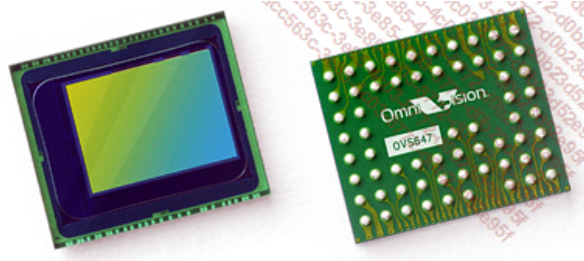
En mayo de 2013, el módulo cámara anunciado por la Fundación Raspberry Pi estaba disponible. Resultado de la tecnología utilizada en los smartphones y adaptada a la Raspberry Pi, el módulo cámara presenta características interesantes por un precio de 27 €.

En abril de 2016, la Fundación puso en el mercado una cámara con mejores características, a un precio equivalente.

1. Características del módulo Cámara V1.3

El sensor

El primer módulo cámara está equipado de un sensor OmniVision OV5647. Toma fotos con una resolución de 5 megapíxeles (2592 x 1944 píxeles) y vídeos en 720p (1280 x 720 píxeles) a 60 fotogramas por segundo. En este caso, la imagen se capta en la totalidad del sensor, los píxeles están agrupados en grupos de 2 x 2 píxeles para aumentar la sensibilidad. El sensor también puede tomar vídeo en 1080p (1920 x 1080 píxeles) a 30 imágenes por segundo.



Cada píxel tiene una dimensión de 1,4 x 1,4 micras. El sensor mide alrededor de 2,74 x 3,67 mm, es decir, casi diez veces menos que un formato "normal" de fotografía (24 x 36 mm). El sensor está equipado con un objetivo focal de 3,6 mm para una apertura cercana a f/3. Esto equivale a un objetivo de 35 mm (un gran angular) en fotografía clásica. El objetivo está enroscado y ajustado para una claridad infinita. A continuación se pega. La distancia de claridad va de 1 metro al infinito.

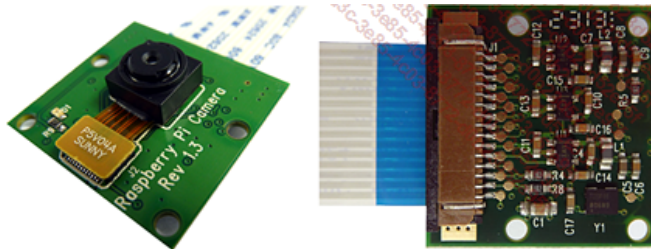
El sensor tiene de fábrica un determinado número de funciones automáticas:

- Control de exposición
- Balanceo de blancos
- Filtro de banda
- Detección de iluminación de 50 Hz y 60 Hz
- Ajuste del nivel de negro

El sensor puede ofrecer la imagen en bruto (modo RAW) para un tratamiento posterior.

El circuito cámara

La Fundación Raspberry Pi ha utilizado el sensor OV5647 para diseñar el módulo cámara de la Raspberry Pi. Tiene interfaz con el bus CSI (consulte el capítulo Descripción técnica).



El módulo mide tan solo 25 x 20 x 10 mm. Se conecta a la Raspberry con un cable plano relativamente flexible de 15 cm. Es posible sustituir este cable por un modelo más largo.

El sitio de la Fundación Raspberry Pi insiste en la fragilidad de este módulo, en particular en su sensibilidad a la electricidad estática. Por tanto, es conveniente no manipular inútilmente y sin precaución la cámara, cuya cara posterior deja a la vista componentes y pistas de circuito impreso.

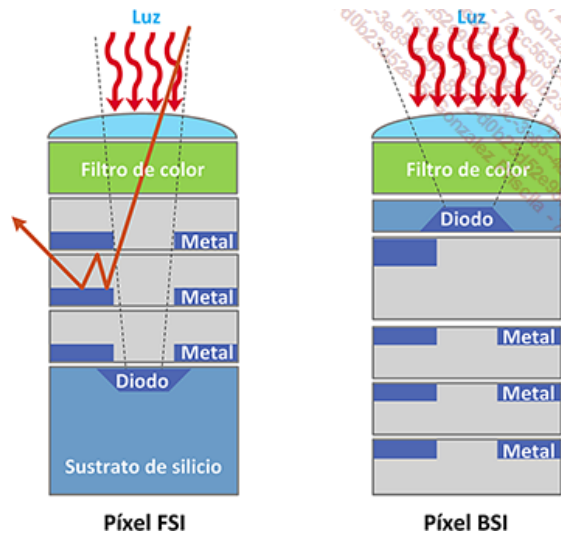
2. Características del módulo cámara V2

El sensor OV5647 llega al final de su ciclo de vida. La Fundación ha sustituido el módulo cámara V1.3 por un nuevo modelo V2. Esta cámara está equipada con un sensor de 8 mega píxeles Sony de tipo IMX219. La cámara puede registrar vídeos en alta definición 1080p30, 720p60 y dispone de un modo VGA90 en 640x480 píxeles a 90 frames por segundo. Las imágenes fijas se pueden capturar con una definición de 3280x2464 píxeles.

Las dos versiones son totalmente compatibles con toda la gama de Raspberry Pi, provistas de un puerto CSI.

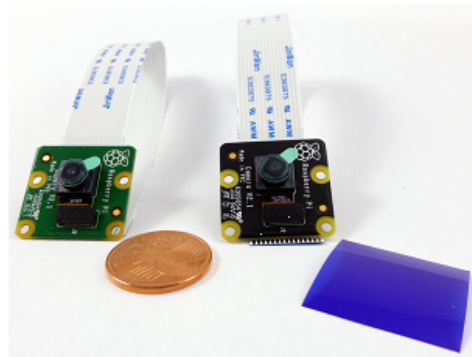
a. Una nueva tecnología de sensor

En el sensor de la primera versión de la cámara, la luz seguía el camino tradicionalmente utilizado en los sensores. Los rayos luminosos llegan a una micro lente antes de atravesar un filtro coloreado (rojo, verde o azul) y después atraviesa varias capas que tienen uniones metálicas destinadas a enlazar los fotodiodos utilizados para recibir la luz. Estos sensores FSI (*Front Side Illumination* = Iluminación frontal) han equipado generaciones de aparatos de fotografía digital.



Con este sistema FSI (a la izquierda en el esquema anterior), el ángulo de apertura se reduce por la presencia de conductores metálicos en las capas situadas entre la lente y el diodo. Además, algunos rayos se pueden reflejar por los conductores metálicos, lo que implica una reducción de la sensibilidad, una debilitación del contraste y diafonía con los píxeles contiguos a causa de las reflexiones.

La tecnología BSI (*Back Side Illumination* = Iluminación trasera) consiste en montar el sensor de imagen al revés y disponer los filtros de color y las micro lentes en la cara posterior de los píxeles, de manera que la luz se reciba por la cara posterior del sensor. En el sensor BSI, la disposición de las capas está invertida. El metal y las capas dieléctricas se encuentran debajo de la red de sensores, ofreciendo un camino mucho más directo a la luz para que alcance el píxel. Esto optimiza el factor de relleno (tamaño de los píxeles mayor) para ofrecer una mejor sensibilidad con baja luminosidad. El rendimiento del sensor es mejor con baja luminosidad y los colores están más próximos a la realidad.



La cámara Pi modelo V2 está disponible en dos versiones: normal o infrarroja. La versión normal tiene delante del sensor un filtro que elimina el infrarrojo, al que el sensor es particularmente sensible. La versión PiNoIR (a la derecha en la imagen anterior) se distingue del modelo normal por su circuito de color negro. No tiene filtro infrarrojo. Por tanto, el sensor recibe las longitudes de onda correspondientes. En particular, esto permite filmar por la noche aclarando la escena con LED infrarrojos invisibles para el ser humano.

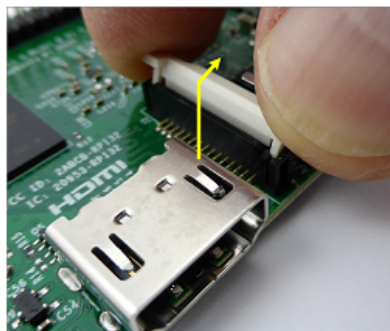
Sin filtro, el sensor recibe la gama completa de colores. Para los botánicos, que desean tener una idea de la actividad de la fotosíntesis de las plantas, es interesante. Las plantas aparecen verdes porque reflejan el color verde. El resto, azul y ultravioleta de un lado de la banda y rojo más infrarrojo, se utilizan por la clorofila para la fotosíntesis. Un botánico puede tener una indicación del lugar en el que la fotosíntesis es más activa en las plantas, intercalando el filtro azul entregado con la cámara. El sensor solo recibirá el infrarrojo y las partes más claras de las plantas, cuando la luz se filtre, serán aquellas en las que la fotosíntesis es más activa.

b. Tabla comparativa

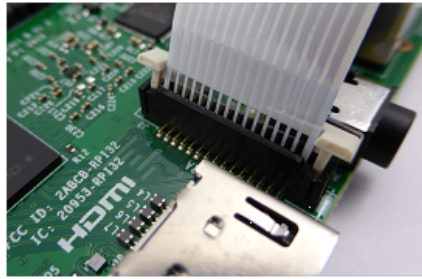
Comparativo Cámara Pi		
	V1	V2
Sensor	5 Mpixeles	8 Mpixeles
Resolución foto	2592x1944	3280x2464
Vídeo maxi	1080p	1080p
Tamaño del módulo	20x25x10 mm	25x23x9 mm

3. Conexión a la Raspberry Pi

La conexión del módulo cámara al bus CSI no presenta ningún problema particular. Hay que abrir el conector levantando delicadamente el sistema de bloqueo e inclinándolo hacia el puerto Ethernet.

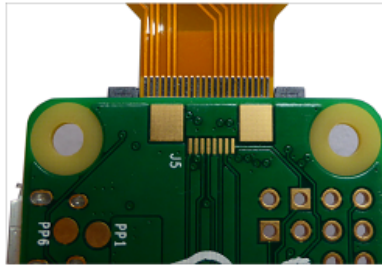


Hay que poner los contactos metálicos del cable plano del lado de la toma HDMI, deslizar a continuación el cable plano de la cámara en el conector y bajar el sistema de bloqueo para bloquear el cable.



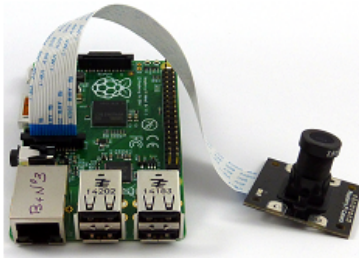
La Raspberry Pi 3 está equipada con su módulo cámara, ahora puede guardar imágenes fijas o vídeos.

La conexión de una cámara en la Raspberry Pi Zero usa el mismo tipo de toma equipada con un sistema de bloqueo. El cable plano adaptador utilizado en función del tamaño reducido del conector de la Raspberry Pi Zero se debe insertar con los contactos metálicos girados hacia la tarjeta del circuito impreso, como se muestra a continuación.



El módulo cámara dispone de cuatro agujeros que se podrán utilizar para fijarla a un soporte que garantice la protección mecánica y eléctrica del circuito.

4. Uso de cámaras compatibles



La foto anterior muestra una tarjeta cámara compatible con la cámara original. Está equipada con un objetivo M12 x 0,5. También existen cámaras equipadas con objetivos de tipo CS, de un diámetro más importante. Estas cámaras compatibles pueden estar equipadas con sensores normales o infrarrojos (PiNoIR).



Las cámaras anteriores, equipadas con un objetivo CS, se usan para la vídeo-vigilancia. Los objetivos CS están muy extendidos para este tipo de aplicación. Su precio es muy variable: algunos euros para los objetivos básicos, cuarenta para un zoom como el 5-50 mm y hasta varios cientos de euros para zooms motorizados.

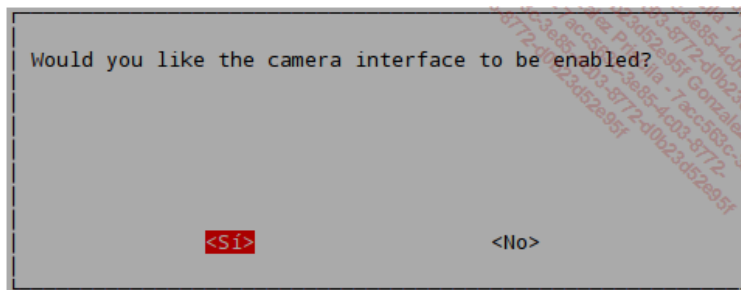
5. Activación de la cámara

a. Activación en modo texto

Para activar la cámara, inicie raspi-config y seleccione la línea 5 **Enable camera**.

```
pi@raspberrypi ~ $ sudo raspi-config
```

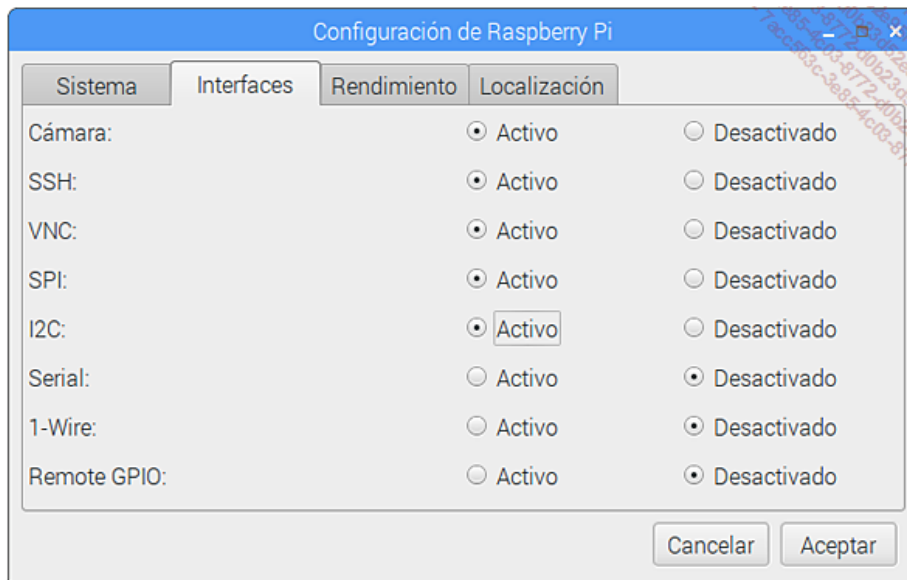
En la ventana que aparece, seleccione **<Sí>** con la tecla [Tab] y valide. La cámara se activará la próxima vez que reinicie el sistema.



b. Activación en modo gráfico

Desde el escritorio de Raspbian, pulse en el botón que da acceso al menú principal.

Menú - Preferencias - Configuración de Raspberry Pi



Pulse en la pestaña **Interfaces** y después en la opción **Activo** en la línea **Cámara**. Confirme su selección pulsando en el botón **Aceptar**. Una ventana le informará entonces que la Raspberry Pi se debe volver a iniciar para que la modificación tenga efecto. Seleccione la opción que le convenga, un reinicio inmediato o reconocer la cámara durante el siguiente inicio de la Raspberry Pi.

6. Captura de una imagen fija

El programa *raspistill* se usa para tomar una foto.

```
pi@raspberrypi ~ $ raspistill -o test_camera_01.jpg -t 5000
```

La imagen aparece a pantalla completa, y transcurrido el tiempo fijado (-t 5000 = 5000 ms) desaparece. El directorio actual contiene ahora una imagen *test_camera_01.jpg*.

7. Registro de un vídeo

Para el registro de vídeos se usa el programa *raspivid*.

```
pi@raspberrypi ~ $ raspivid -o video_01.h264 -t 5000
```

La línea de comandos anterior guarda un vídeo de una duración de 5 segundos (5000 ms). El registro recibe el nombre *video_01.h264*.

Es posible visualizar el vídeo en un PC con VLC, o en la Raspberry Pi con *omxplayer*, instalado de manera nativa con Raspbian.

```
pi@raspberrypi ~ $ omxplayer video_01.h264
```

8. Timelapse

El timelapse es un vídeo realizado tomando imágenes a intervalos regulares y juntando después estas imágenes en un vídeo.

Cree un directorio para almacenar las imágenes capturadas:

```
pi@raspberrypi:~ $ mkdir timelapse
```

Muévase al directorio que acaba de crear:

```
pi@raspberrypi:~ $ cd timelapse
```

En un primer lugar, *raspistill* captura las imágenes:

```
pi@raspberrypi:~/timelapse $
raspistill -vf -hf -w 1920 -h 1080 -o test_%04d.jpg -t1 5000 -t
250000
```

-vf (*Vertical Flip*) pivota la imagen verticalmente.

-hf (*Horizontal Flip*) pivota la imagen horizontalmente.

Estas dos opciones permiten poner la imagen en la dirección correcta cuando la cámara esté montada al revés.

-o indica el nombre del archivo de salida (*output*); %04d va a numerar las imágenes con 4 cifras.

-t1 (*TimeLapse*) indica el espacio entre dos tomas de imagen en milisegundos: aquí 5.000 milisegundos, es decir, una imagen cada 5 segundos.

-t indica el tiempo de toma de imágenes total, también en milisegundos. 250.000 de milisegundos representan 250 segundos, es decir, poco más de cuatro minutos.

-w y -h indican respectivamente la anchura y altura de la imagen, en píxeles.

Después de haber obtenido las 50 imágenes individuales, hay que ensamblarlas. Para esto, hay que usar un programa llamado *avconv*, presente en el paquete *libav-tools*, que hay que instalar:

```
pi@raspberrypi:~/timelapse $ sudo apt-get install libav-tools
```

Por último, *avconv* nos va a permitir crear un vídeo:

```
pi@raspberrypi:~/test_t1 $ avconv -r 10 -i test_%04d.jpg -r 10 -  
vcodec libx264 -crf 20 -g 15 -vf scale=1920:1080 timelapse.mp4
```

Serán necesarios cerca de cuarenta segundos de Raspberry Pi para crear el vídeo *timelapse.mp4*. La tasa de ocupación del procesador será cercana al 100 %.

avconv crea el vídeo a partir de las imágenes .jpg presentes. El número de imágenes por segundo está limitado a 10 (-r 10), lo que es suficiente para un timelapse. *avconv* usa el codec gratuito libx264 (*vcodec=libx264*) para la codificación, con una resolución de 1920x1080 px (*scale=1920:1080*).

Para visualizar el vídeo:

```
pi@raspberrypi:~/test_t1 $ omxplayer timelapse.mp4
```

La película se muestra en la pantalla de la Raspberry Pi, superpuesta a todo lo que encuentre, esté en modo texto o en modo gráfico. Los cuatro minutos de timelapse se mostrarán en cinco segundos.

Las tarjetas de extensión

La GPIO abierta de la Raspberry Pi ha provocado el mismo entusiasmo que Arduino en su momento. Aparecieron muchas tarjetas de extensión, generalistas o muy especializadas. A continuación se presenta algunos ejemplos con sus características y las posibilidades que ofrecen. La elección no tiene objetivos comerciales, representa un panel de tarjetas normalmente necesarias para acompañar aquellas realizaciones que se hacen con la Raspberry Pi.

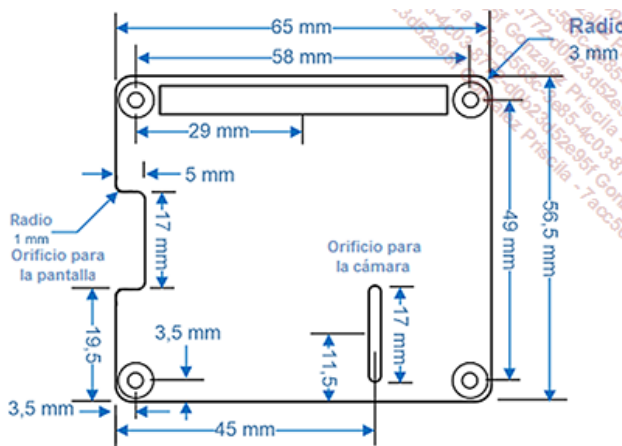
1. Las tarjetas HAT

a. Presentación

Con la Raspberry Pi B+ la Fundación introdujo el concepto de tarjeta HAT (*Hardware Attached on Top* = hardware fijado en la parte superior), inspirado en lo que existe en los ecosistemas de Arduino con los Shields y en BeagleBone con los Capes.

La tarjeta HAT está provista de una EEPROM que se detecta al inicio del sistema por el *firmware*. Las clavijas de la GPIO 27 y 28 se reservan exclusivamente para el acceso a la EEPROM a través del bus I²C y no se deben utilizar o conectar a ninguna otra cosa. Esta prohibición es válida para la Raspberry Pi 3 y para la Raspberry Pi Zero.

b. Dimensiones



Las dimensiones de las tarjetas HAT se han definido tomando como base los agujeros de fijación previstos en la Raspberry Pi B+ y todas sus sucesoras. Están perfectamente adaptadas para un uso con la Raspberry Pi 3. Esta disposición en rectángulo de los agujeros permite una fijación eficaz y sólida de las tarjetas adicionales.

El formato HAT prevé unas muescas para pasar los cables CSI para la cámara y DSI para una pantalla.

Los espaciadores entre la Raspberry Pi y la tarjeta HAT deben medir 10 o 12 mm, en función del conector GPIO.

- Una familia de tarjetas de extensión con dimensiones adaptadas a la Raspberry Pi Zero está cerca de ver la luz. La diferencia de factor de forma entre la Raspberry Pi Zero y la Raspberry Pi 3 hace que no puedan estar en formato HAT.

c. Funcionamiento

Las tarjetas HAT se detectan durante el inicio de la Raspberry Pi B+. Tienen una EEPROM que contiene la siguiente información:

- Un **UUID** (*Universal Unique Identifier* = identificador universal único) de 128 bits que permite identificar cada tarjeta de manera única.
- El identificador del producto (**PID** = *Product ID* = identificador del producto) y su versión (**PVER** = *Product Version* = versión del producto).
- El nombre del fabricante (**VSTR** = *ASCII Vendor String* = nombre del fabricante en forma de cadena de caracteres ASCII).
- El nombre del producto (**PSTR** = *ASCII Product String* = nombre del producto en forma de cadena de caracteres ASCII).
- Una tabla de configuración de los puertos GPIO que permite un funcionamiento "plug-and-play".
- Una descripción de la arborescencia de hardware (*Device Tree*) que se va a integrar en la arborescencia de hardware general y permitirá la configuración automática de los componentes de la tarjeta siempre y cuando exista algún controlador disponible para estos componentes.

- Solo se permite una tarjeta HAT a la vez en la Raspberry Pi B+. Por tanto, la dirección de la EEPROM en el bus I²C puede ser fija. Esta limitación se debe a que incluso si se soportaran múltiples EEPROM, podría haber conflictos entre la programación de los puertos GPIO de las diferentes tarjetas. Por ejemplo, si dos tarjetas apiladas usan el mismo puerto GPIO. No se prevé ningún mecanismo para detectar estos conflictos, el formato HAT obliga a que solo se utilice una tarjeta HAT al mismo tiempo.

d. Conclusión

Con el formato HAT, la Fundación ha pretendido facilitar la puesta en servicio de tarjetas adicionales para los usuarios que no son necesariamente especialistas en GNU/Linux. Es responsabilidad de los creadores de tarjetas HAT respetar las especificaciones para que sus productos se puedan implantar de la manera más sencilla posible.

2. La tarjeta de sonido HiFiBerry DAC+

a. Presentación de la tarjeta HiFiBerry DAC+

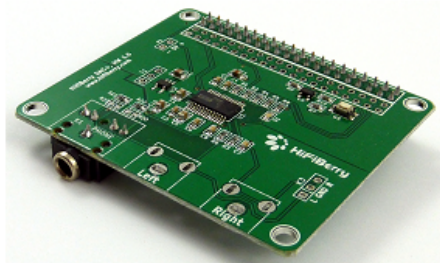
Bautizada HiFiBerry DAC+, esta tarjeta destinada a la Raspberry Pi 3 funciona también con las Raspberry Pi A+, B+ y con la Raspberry Pi 2 modelo B. Está equipada con un doble convertidor digital Analógico PCM5122 de la marca Burr-Brown. Este circuito previsto para funcionar con 3 voltios se controla por los buses SPI o I²C. Convierte en esta tarjeta los datos recibidos en 24 bits a la frecuencia de 192kHz, proporcionando un sonido de una calidad muy por encima del de la Raspberry Pi.

La tarjeta fue diseñada por la empresa suiza Model 9.

La tarjeta es compatible con el estándar HAT de la Raspberry Pi. Integra una EEPROM - protegida en modo escritura - para la configuración automática de la Raspberry Pi. Se entrega con cuatro espaciadores que permiten una fijación sólida de la tarjeta de sonido en la Raspberry Pi.

Hay disponibles dos versiones: una con tomas RCA (CINCH) y otra con un jack de 3,5 mm. Las dos versiones cuestan alrededor de 30 €.

La tarjeta HiFiBerry no puede atacar directamente los altavoces. Será necesario prever el uso de un amplificador o de altavoces amplificados, como los que se utilizan en un PC. En este último caso, la calidad del sonido se verá reducida de manera importante por la calidad de los altavoces utilizados. Para mi uso, dispongo de amplificadores provistos de un conector jack 3,5 mm, de modo que es esta la versión que he seleccionado.



b. Puesta en servicio de la tarjeta HiFiBerry DAC +

El sistema recomendado para arrancar con la tarjeta HiFiBerry es Raspbian. Antes de la puesta en servicio de la tarjeta HiFiBerry DAC+, el sistema se debe actualizar, así como los módulos del núcleo.

```
sudo rpi-update
sync
sudo reboot
```

A continuación, es necesario configurar los controladores para que los buses I2C y SPI funcionen. Es posible hacer esto en modo gráfico usando el menú principal del escritorio de Raspbian Jessie:

Menú - Preferencias - Configuración de la Raspberry Pi

Haga clic en la pestaña **Interfaces** y después en la opción **Activo** de las líneas SPI y I2C. Confirme su selección haciendo clic en el botón **Aceptar**.

En modo texto, edite el archivo `/boot/config.txt` para que las líneas de texto relacionadas con los buses SPI y I2C se activen (elimine la # al principio de cada línea):

```
# Uncomment some or all of these to enable the optional hardware
interfaces
dtparam=i2c_arm=on
#dtparam=i2s=on
dtparam=spi=on
```

Desactive el sonido original de la Raspberry Pi :

```
# Enable audio (loads snd_bcm2835)
dtparam=audio=off
```

Agregue el controlador HiFiBerry DAC+ al final del archivo `/boot/config.txt`. Durante el siguiente arranque, el controlador de la tarjeta se integrará automáticamente al nodo Linux.

```
dtoverlay=hifiberry-dacplus
```

Tras el inicio, puede comprobar con el comando `aplay -l` que el controlador se ha instalado correctamente y que la tarjeta HiFiBerry DAC+ es la tarjeta de sonido por defecto:

```
pi@raspberrypi:/ $ aplay -l
**** lista de los Periféricos de hardware PLAYBACK ****
tarjeta 0: sndrpihifiberry [snd_rpi_hifiberry_dacplus],
periférico 0: HiFiBerry DAC+ HiFi pcm512x-hifi-0 []
  Sub-periféricos: 0/1
  Sub-periférico #0: subdevice #0
```

Para probar el funcionamiento correcto de la tarjeta, conecte los altavoces amplificados o el sistema de audio a la salida de su tarjeta.

Copie un archivo `.wav` en la tarjeta micro SD o descargue el archivo de prueba disponible desde la página Información o en framboise314 (breve extracto de Happyrock por Bensound - Licencia CC):

```
pi@raspberrypi:~ $ cd /home/pi/Music
pi@raspberrypi:~/Music $ wget http://www.framboise314.fr/docs/test.wav
--2016-07-06 18:25:21-- http://www.framboise314.fr/docs/test.wav
Resolución de www.framboise314.fr (www.framboise314.fr)... 185.49.20.101
Conexión a www.framboise314.fr
(www.framboise314.fr)|185.49.20.101|:80... conectado.
consulta HTTP transmitida, esperando respuesta... 200 OK
Tamaño: 2598156 (2,5M) [audio/x-wav]
Copiado en: «test.wav»
test.wav.1 100%[=====] 2,48M 418KB/s
ds 6,1s
2016-07-06 18:25:28 (417 KB/s) - « test.wav » copiado
[2598156/2598156]
```

Seleccione un sonido y reproduzca con el comando `aplay nombre_del_archivo_wav`:

```
pi@raspberrypi:~/Music $ aplay test.wav
Lectura WAVE 'test.wav' : Firma 16 bit Little Endian,
Frecuencia 44100 Hz, Estereo
```

Esto le permite comprobar que toda la cadena funciona. Los altavoces deben reproducir algunos segundos de música.

Si no hay sonido, consulte al foro HiFiBerry. En primer lugar, compruebe que el mezclador digital está ajustado correctamente.

Escriba `amixer` en la línea de comandos. Busque información sobre **Simple mixer control 'Digital'**. Es posible que el mezclador esté configurado a 0 o a un valor demasiado bajo. Las siguientes líneas lo indican:

```
Front Left: Playback 166 [80%] [-20.50dB] [on]
Front Right: Playback 166 [80%] [-20.50dB] [on]
```

Si el valor del ajuste es demasiado bajo, configure el parámetro de control del volumen al 80% con el comando:

```
amixer sset Digital 80%
```

Después compruebe que el parámetro se haya aplicado correctamente con el comando `amixer`:

```
Simple mixer control 'Digital',0
Capabilities: pvolume pswitch
Playback channels: Front Left - Front Right
Limits: Playback 0 - 207
Mono:
Front Left: Playback 166 [80%] [-20.50dB] [on]
Front Right: Playback 166 [80%] [-20.50dB] [on]
```

➤ Hay que evitar subir el volumen por encima de un valor que pueda provocar la deformación y/o recortar la señal. 80% es un buen valor de partida.

c. Uso de mpg123

El uso de `aplay` limita la escucha a archivos de tipo `voc`, `wav`, `raw` o `au`. Para poder escuchar archivos `mp3` es necesario usar otras aplicaciones, como por ejemplo `mpg123`, un lector de archivos `mp3` para Linux.

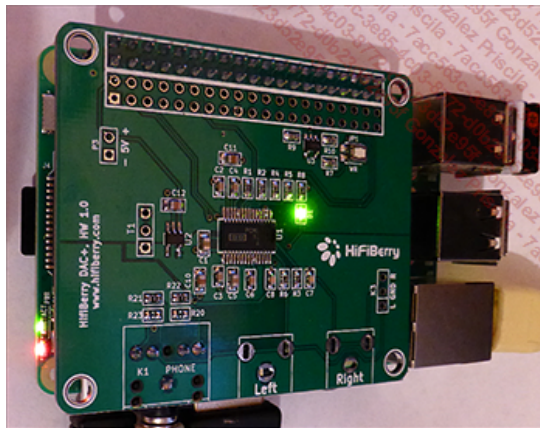
Empiece instalando **mpg123**:

```
pi@raspberrypi $ sudo apt-get install mpg123
```

Ahora es posible oír archivos `mp3`. Para conocer los parámetros disponibles, escriba `mpg123` en la línea de comandos y valide. Se muestra por pantalla el archivo de ayuda.

Para empezar, transfiera archivos `mp3` al directorio `/home/pi/Music` o descargue un archivo `mp3` en línea (por ejemplo, desde la página <http://www.bensound.com/royalty-free-music/track/happy-rock>).

```
pi@raspberrypi:~/Music $ mpg123 bensound-happyrock.mp3
High Performance MPEG 1.0/2.0/2.5 Audio Player for Layers 1, 2 and 3
  version 1.20.1; written and copyright by Michael Hipp and others
  free software (LGPL) without any warranty but with best wishes
Playing MPEG stream 1 of 1: bensound-happyrock.mp3 ...
MPEG 1.0 layer III, 112 kbit/s, 44100 Hz joint-stereo
[1:45] Decoding of bensound-happyrock.mp3 finished.
```



Durante la lectura de un fragmento de música, se ilumina el LED D1 situado en la tarjeta HiFiBerry DAC+ (imagen anterior).

Conclusión

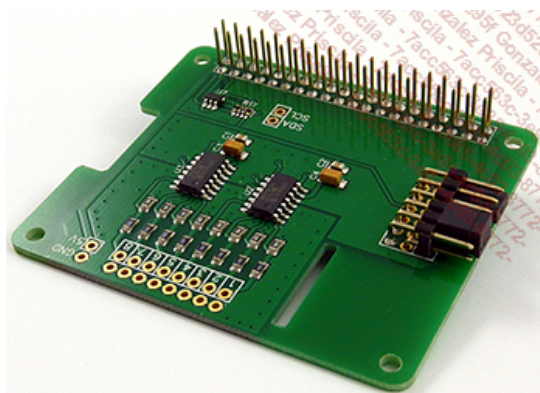
La tarjeta HiFiBerry permite dotar a la Raspberry Pi de una tarjeta de sonido de calidad y obtener un conjunto que sea capaz de servir de punto de partida para una instalación de audio de alto rendimiento.

El uso de esta tarjeta con otras distribuciones diferentes a Raspbian no es inmediato y necesitará una adaptación, cuyos detalles están disponibles en línea. OpenElec, uno de los centros multimedia más utilizados en la Raspberry Pi, integra controladores para HiFiBerry (<https://www.hifiberry.com/2014/12/guide-use-the-hifiberry-ampamp-with-xbmc-kodi/>) así como distribuciones como Volumio o RuneAudio.

3. La tarjeta ADC Pi Plus

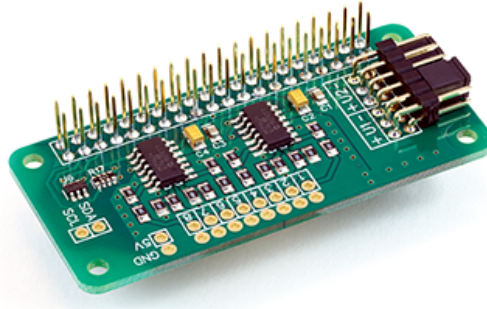
a. Presentación de la tarjeta ADC Pi Plus

La tarjeta ADC Pi Plus es una tarjeta HAT que se puede usar en la Raspberry Pi 3 incorporando así la posibilidad de leer información analógica. La tarjeta tiene ocho entradas analógicas que aceptan tensiones de 0 a 5 voltios. La conversión analógico/digital se puede hacer en 11 bits (240 muestras por segundo), 13 bits (60 muestras/s), 15 bits (15 muestras/s) o 17 bits (3,75 muestras/s). La elección del muestreo se hará en función de las necesidades de la aplicación, de la precisión de la medida o de la frecuencia de adquisición de la información.

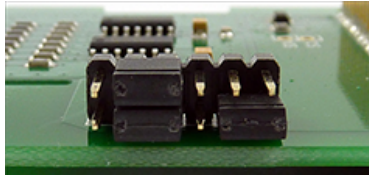


La tarjeta ADC Pi Plus se gestiona por la Raspberry Pi en el bus I²C. Es posible apilar hasta cuatro tarjetas para obtener un máximo de 32 entradas analógicas. Para distinguir las tarjetas en el bus I²C, cada una debe tener una dirección propia.

El modelo ADC Pi Zero (siguiente imagen) es el equivalente a la tarjeta ADC Pi Plus, pero se ha diseñado para adaptarse a la Raspberry Pi Zero.



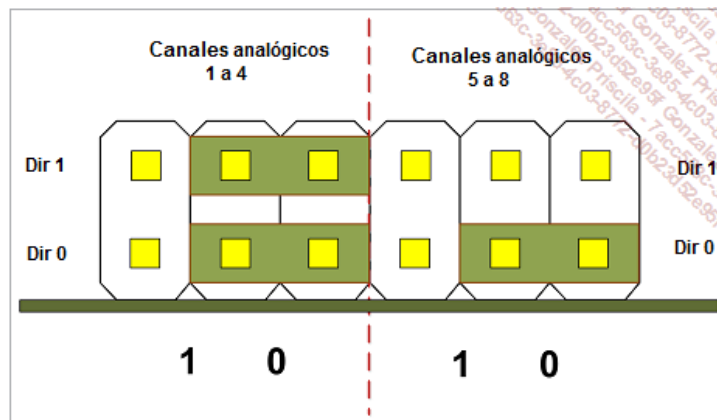
La tarjeta ADC Pi Plus está equipada con dos circuitos integrados MCP3424 que gestionan cada uno cuatro entradas analógicas. Cada circuito dispone de su propia dirección. Por tanto, la tarjeta es accesible a través de dos direcciones correspondientes cada una a cuatro canales. La configuración de la dirección se realiza con jumpers.



Los dos bits de dirección Dir 0 y Dir 1 pueden tomar los valores **0**, **1** o **flotante**. Este último valor se obtiene cuando no hay ningún puente (entrada de dirección se deja "al aire"). Por defecto, la tarjeta se entrega configurada en las direcciones 0x68 y 0x69. Los símbolos 0x indican que el valor de la dirección se expresa en hexadecimal.

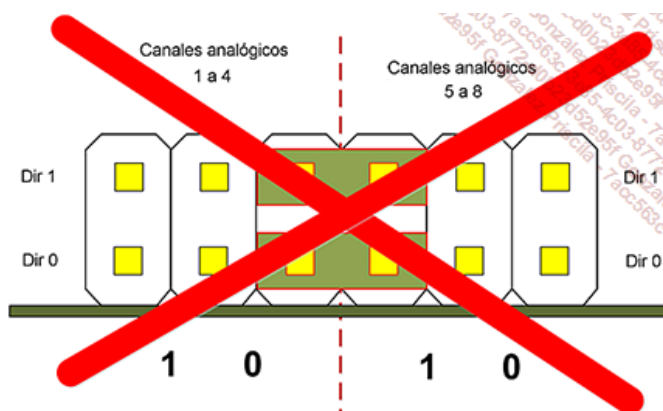
Direccionamiento ADC Pi Plus		
Dir 0	Dir 1	Dirección
0	0	0x68
flotante	flotante	0x68
0	flotante	0x69
0	1	0x6A
1	0	0x6B
1	flotante	0x6C
1	1	0x6D
flotante	0	0x6E
flotante	1	0x6F

La dirección 0x68 se puede obtener de dos maneras: dejando las dos entradas al aire (flotantes) o configurando los dos jumpers a 0.



El esquema anterior muestra la configuración por defecto. Los canales 1 a 4 se acceden en la dirección 0x68 (Dir 0 = 0 y Dir 1 = 0), y los canales 5 a 8 en la dirección 0x69 (Dir 0 = 0 y Dir 1 = flotante).

Para simplificar el acceso si se utilizan varias tarjetas es mejor usar pares de direcciones sucesivas: 0x68 y 0x69, 0x6A y 0x6B, 0x6C y 0x6D, 0x6E y 0x6F.



Las clavijas situadas a ambos lados de la línea punteada, dando forma a los dos bloques en el esquema anterior, se conectan respectivamente a la masa y a 5V. Nunca deben configurarse los jumpers en la línea punteada, porque se corre el riesgo de dañar la tarjeta y/o la Raspberry Pi.


```
Recepción de objetos: 100% (177/177), 62.60 KiB | 0 bytes/s, hecho.
Resolución de los deltas: 100% (99/99), terminado.
Comprobación de la conectividad... terminada.
```

La operación solo lleva algunos instantes e importa los programas a una carpeta llamada *ABElectronics_Python3_Libraries*:

```
pi@raspberrypi:~/ABElectronics_Python3_Libraries $ ls
ADCDACPi      ADCPi      ExpanderPi  LICENSE    RTCPi
smbusmodule.c
ADCDifferentialPi  DeltaSigmaPi  IOPI      README.md  ServoPi
```

Los programas para la tarjeta ADC Pi Plus están en el directorio *ADCPi*:

```
pi@raspberrypi:~/ABElectronics_Python3_Libraries/ADCPi $ ls
ABE_ADCPi.py  demo-acs712-30.py  demo-readvoltage.py
ABE_helpers.py  demo-logvoltage.py  README.md
```

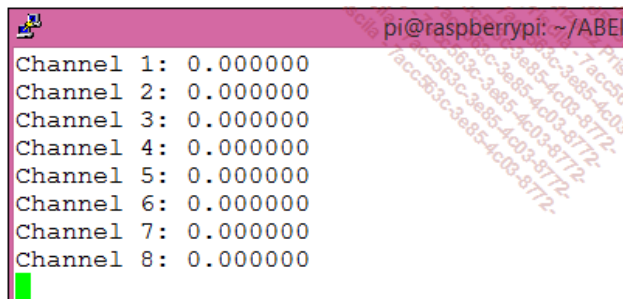
Para usar los programas, se necesita el módulo Python *smbus*. Hay que instalarlo previamente:

```
pi@raspberrypi ~/ABElectronics_Python3_Libraries $ sudo apt-get
install python3-smbus
Lectura de las listas de paquetes... Terminado
Construcción del árbol de dependencias
Lectura de la información de estado... Terminado
Se instalarán los siguientes paquetes nuevos:
python3-smbus
```

A continuación es posible ejecutar el programa de demostración *demo-readvoltage.py*:

```
pi@raspberrypi
~/ABElectronics_Python3_Libraries/ABElectronics_ADCPi $ sudo
python3 demo-readvoltage.py
```

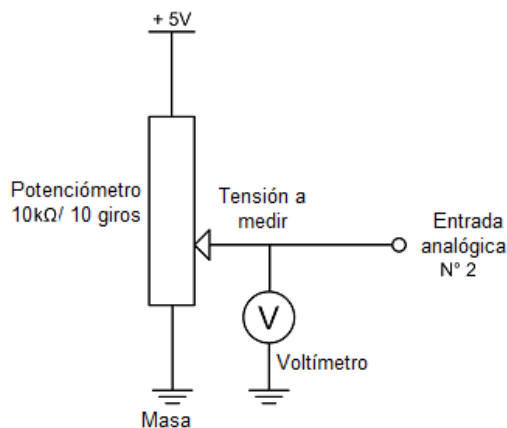
No se aplica ninguna señal a las entradas. El resultado debe ser una serie de valores nulos. Sin embargo, esta prueba permite comprobar que el conjunto hardware + software está bien instalado.



```
pi@raspberrypi: ~/ABEle
Channel 1: 0.000000
Channel 2: 0.000000
Channel 3: 0.000000
Channel 4: 0.000000
Channel 5: 0.000000
Channel 6: 0.000000
Channel 7: 0.000000
Channel 8: 0.000000
```

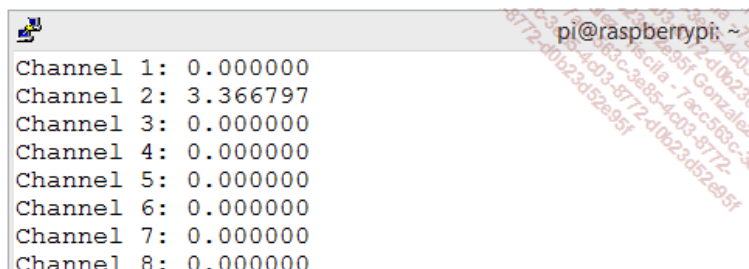
Medida de una tensión

Las pruebas de la tarjeta se han realizado utilizando un potenciómetro de 10 giros y un multímetro.



La tensión a medir se aplica a la entrada analógica n°2.

El programa *demo-readvoltage.py* produce el siguiente resultado:



```
pi@raspberrypi: ~
Channel 1: 0.000000
Channel 2: 3.366797
Channel 3: 0.000000
Channel 4: 0.000000
Channel 5: 0.000000
Channel 6: 0.000000
Channel 7: 0.000000
Channel 8: 0.000000
```

La tensión medida por el voltímetro es 3,40 V. La diferencia entre las dos medidas es inferior a 1 %.

- La tensión aplicada a las entradas analógicas no debe exceder las tensiones de alimentación más de 0,4 V. En caso contrario, se puede deteriorar el circuito integrado. Debe permanecer comprendida entre -0,4 V y +5,4 V.

Conclusión

Esta tarjeta añade ocho entradas analógicas a la Raspberry Pi. Esto abre la vía para la adquisición de datos inaccesibles de otra manera. Los

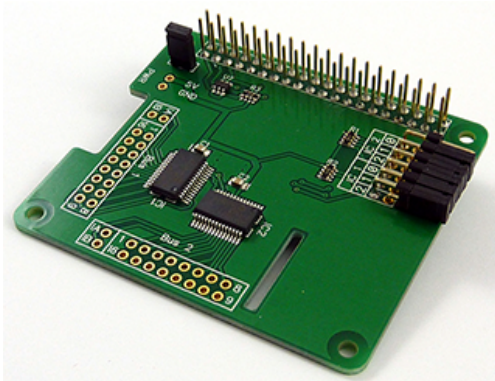
programas de demostración constituyen un buen punto de inicio para el desarrollo de aplicaciones más ambiciosas.

4. La tarjeta IO Pi Plus

a. Presentación de la tarjeta IO Pi Plus

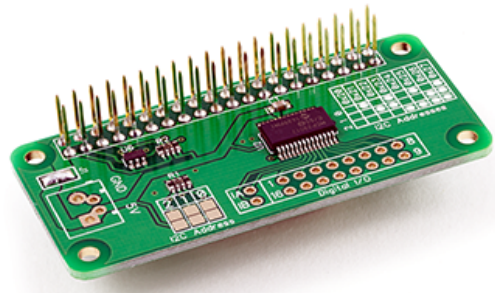
La tarjeta IO Pi Plus es una tarjeta HAT que se puede usar en la Raspberry Pi 3, a la que se agrega 32 entradas/salidas digitales. Esto permite aumentar mucho las posibilidades de entrada/salida de la Raspberry Pi.

La tarjeta IO Pi Plus se gestiona por la Raspberry Pi en el bus I²C. Es posible apilar hasta cuatro tarjetas para obtener un máximo de 128 entradas analógicas. Para distinguir las tarjetas en el bus I²C, cada una debe disponer de una dirección propia.

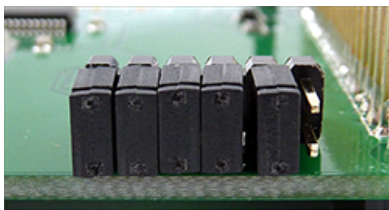


La tarjeta IO Pi Plus está equipada con dos circuitos integrados MCP23017, cada uno gestionando 16 entradas/salidas digitales. Es posible configurar de manera separada cada clavija como entrada o salida. También es posible la gestión de las clavijas en bloques de 8. Hay dos bloques por circuito.

Hay una versión de esta tarjeta destinada a la Raspberry Pi Zero, que le añade 16 entradas/salidas digitales, la tarjeta IO Pi Zero (siguiente imagen). Solo tiene un circuito MCP23017, pero su uso es idéntico al de la versión para la Raspberry Pi 3, vaya al sitio web de ABElectronics para obtener más información.



Cada circuito MCP23017 dispone de su propia dirección en el bus I²C. Por tanto, la tarjeta es accesible por dos direcciones correspondientes cada una a 16 E/S. La configuración de dirección se realiza mediante jumpers.

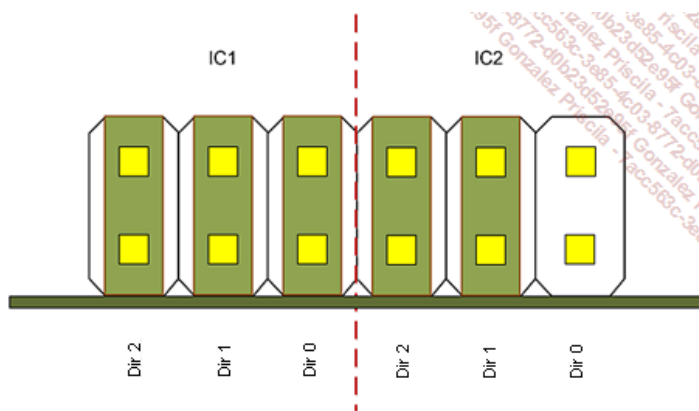


Durante su entrega, la tarjeta se configura con las direcciones 0x20 y 0x21 (fotografía anterior). La tabla y esquema siguientes permiten encontrar las direcciones en función de la posición de los jumpers.

Para usar varias tarjetas apiladas y aumentar todavía más el número de entradas/salidas es necesario modificar el direccionamiento de las tarjetas para que cada una de las direcciones sea diferente. La siguiente tabla indica las configuraciones que pueden utilizarse.

Direccionamiento IO Pi Plus			
Dir 2	Dir 1	Dir 0	Dirección
X	X	X	0x20
X	X		0x21
X		X	0x22
X			0x23
	X	X	0x24
	X		0x25
		X	0x26
			0x27

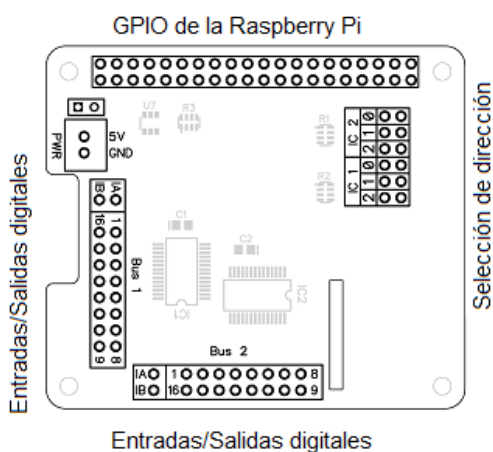
La X corresponde a un puente presente en las clavijas. Las clavijas de configuración de dirección se separan en dos bloques: los tres conectores de la izquierda corresponden a IC1 y los tres de la derecha a IC2.



El esquema anterior muestra la configuración por defecto de la tarjeta IO Pi Plus. IC1 recibe la dirección 0x20 e IC2 recibe la dirección 0x21.

- Todas las operaciones de los jumpers de direccionamiento se deben hacer obligatoriamente en modo desenchufado. También es necesario para la inserción o retirada de la tarjeta en la Raspberry Pi.

b. Las entradas digitales



En la tarjeta, las entradas/salidas se reparten en dos bloques llamados Bus 1 y Bus 2. Cada bloque tiene 16 entradas. Cada entrada es capaz de controlar una corriente de 25 mA (de entrada o de salida). Por el contrario, la corriente total máxima aceptable por un bus está limitada a 125 mA.

El puente situado junto a las pastillas marcadas PWR (alimentación) permite aislar la tarjeta de la alimentación de la Raspberry Pi. Si se quita el puente, es necesario alimentar la tarjeta en los bornes PWR: 5V y GND. Esto disminuye la carga en la alimentación de la Raspberry Pi.

c. Instalación de la tarjeta

La instalación de la tarjeta se hace en modo desenchufado. Compruebe detenidamente que el conector de 40 clavijas de la GPIO está bien configurado. A continuación, ubique los espaciadores que se entregan con la tarjeta.

La tarjeta utiliza el bus I²C, es necesario activar la gestión del bus. El párrafo anterior (tarjeta ADC Pi Plus) da las indicaciones para la activación del bus I²C.

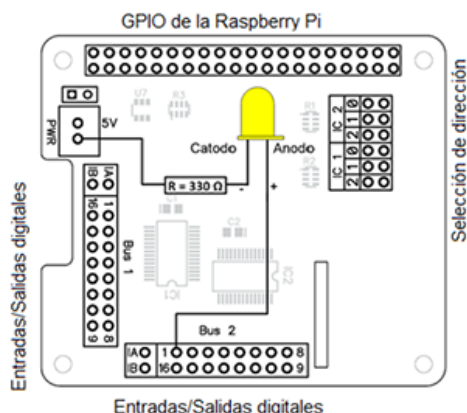
Después de situar la tarjeta, arranque la Raspberry Pi y compruebe que se detecta correctamente en el bus I²C:

```
pi@raspberrypi:~$ sudo i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20: 20 21  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Los dos circuitos I2C se detectan correctamente en sus respectivas direcciones: 0x20 para el primero y 0x21 para el segundo.

d. Pruebas de la tarjeta

Conecte un LED en las conexiones correspondientes a la clavija 1 del bus 2 y a la masa. Intercale una resistencia de 330 ohms destinada a limitar la corriente.



La tarjeta IO Pi Plus, como la tarjeta ADC Pi Plus, se proporciona con las librerías para Python. Se descargan con el comando `git`, disponible en Raspbian:

```
git clone
https://github.com/abelectronicsuk/ABElectronics_Python_Libraries
.git
```

Vaya a la carpeta `ABElectronics_Python3_Libraries/ABElectronics_IOPi`. Los programas de demostración para IOPi están en esta carpeta.

```
pi@raspberrypi:~/ABElectronics_Python3_Libraries/IOPi $ ls
ABE_helpers.py      demo-iopiread.py      __pycache__          tutorial1.py
ABE_IoPi.py         demo-iopireadwrite.py  README.md
tutorial2.py
demo-iointerrupts.py demo-iopiwrite.py     tutorial1a.py
```

Ahora es posible ejecutar una de las demostraciones, `tutorial1.py`, que hace pasar alternativamente la salida 1 del bus (al que está conectado el LED) de 1 a 0, cada segundo. El LED parpadea al ritmo del cambio de este valor.

```
pi@raspberrypi:~/ABElectronics_Python3_Libraries/IOPi $ sudo
python3 tutorial1.py
```

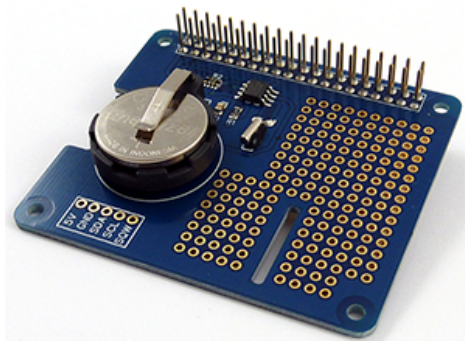
Conclusión

Añadiendo 32 entradas/salidas a la Raspberry Pi, la tarjeta IO Pi Plus permite controlar una gran cantidad de relés o detectar fallos y hacer que el sistema reaccione en consecuencia. La presencia de programas de demostración constituye un punto de partida para las pruebas.

5. La tarjeta RTC Pi Plus

a. Presentación de la tarjeta RTC Pi Plus

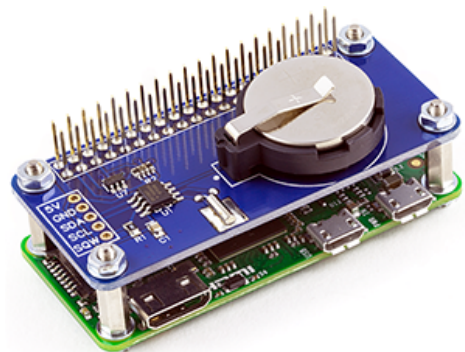
La tarjeta RTC Pi Plus es una tarjeta HAT que se puede utilizar en la Raspberry 3. Permite añadir un reloj a la Raspberry Pi. Cuando se conectada a Internet, utiliza un servidor de tiempo para recuperar la hora. En las aplicaciones embebidas, cuando la Raspberry Pi se usa sin conexión a Internet (registro de datos, estación meteorológica, etc.) esta tarjeta aporta la solución.



La tarjeta RTC Pi Plus está equipada de un circuito RTC DS1307 conectado al bus I²C. La dirección del reloj es 0x68 en el bus I²C y no se puede modificar. Si la tarjeta RTC Pi Plus se usa con otra tarjeta conectada al bus I²C (IO Pi Plus, ADC Pi Plus...), será necesario vigilar que ninguna de estas tarjetas use la misma dirección.

De manera previa a la puesta en servicio de la RTC Pi Plus, será necesario insertar una pila CR2032 en el soporte previsto para este efecto. Esta es la pila que va a alimentar el reloj cuando la alimentación de la Raspberry Pi se desconecte.

Existe una versión RTC Pi Zero (más abajo) diseñada para la Raspberry Pi Zero.



- El fabricante recomienda no conectar la tarjeta a la Raspberry Pi sin que la pila CR2032 esté instalada en su soporte. Se corre el riesgo de dañar el circuito DS1307 presente en la tarjeta. Durante la carga de pila, hay que desconectar la Raspberry Pi.

La tarjeta RTC Pi Plus dispone de una zona libre, llena de patillas para soldar, que puede servir para cablear prototipos de conexiones electrónicas.

b. Puesta en hora de la Raspberry Pi

En su origen, la Raspberry Pi está equipada de un reloj RTC, también llamado "guard time". Por defecto, se conecta a un servidor de tiempo que le ofrece una hora precisa y a continuación es el sistema el que mantiene este reloj "de sistema" en hora.

Si la Raspberry Pi no tiene acceso a Internet, el reloj de sistema se debe poner en hora manualmente. En caso contrario, la hora del sistema es falsa:

```
pi@raspberrypi:~ $ sudo date -s "6 jul 2016 19:17:05"
miércoles 6 julio de 2016, 19:17:05 (UTC+0200)
```

La visualización de la fecha y la hora del sistema se hace con el comando `date`:

```
pi@raspberrypi:~ $ date
miércoles 6 julio de 2016, 19:17:53 (UTC+0200)
```

c. Instalación de la tarjeta

La instalación de la tarjeta se hace en modo desenchufado. Compruebe detenidamente que el conector de 40 clavijas de la GPIO está bien posicionado. A continuación, ubique los espaciadores que se entregan con la tarjeta.

La tarjeta utiliza el bus I²C, es necesario activar la gestión del bus. Lea los párrafos anteriores (ADC Pi Plus y IO Pi Plus) para la activación del bus I²C y de la instalación de i2c-tools.

Después de situar la tarjeta, arranque la Raspberry Pi y compruebe que se detecta correctamente en el bus I²C:

```
pi@raspberrypi:~ $ sudo i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  -- UU --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

La tarjeta RTC se detecta correctamente en la dirección 68.

d. Configuración del reloj RTC

Ahora hay que permitir al sistema acceder al reloj. El procedimiento de instalación se describe en la página <https://www.abelectronics.co.uk/kb/article/30/rtc-pi-on-a-raspberry-pi-raspbian-jessie>

Edite el archivo `/boot/config.txt` y agregue el overlay de gestión del reloj RTC de tipo DS1307 para integrar el control al inicio del nodo Linux:

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

Al final del archivo, añada la línea:

```
dtoverlay=i2c-rtc,ds1307
```

Guarde el archivo ([Ctrl] X) y agregue el módulo `rtc-ds1307` al archivo `/etc/modules`.

```
sudo nano /etc/modules
```

Agregue la siguiente línea al final del archivo:

```
rtc-ds1307
```

Guarde el archivo ([Ctrl] X).

Modifique ahora el archivo `/lib/udev/hwclock-set`

```
sudo nano /lib/udev/hwclock-set
```

Ponga una `#` delante de las siguientes líneas:

```
if [ -e /run/systemd/system ] ; then
exit 0
fi
```

Que deben quedar así:

```
#if [ -e /run/systemd/system ] ; then
#exit 0
#fi
```

Guarde el archivo ([Ctrl] X) y a continuación inicie de nuevo la Raspberry Pi para que se tengan en cuenta estas modificaciones.

```
sudo reboot
```

e. Uso del reloj RTC DS1307

Durante el primer uso de la tarjeta, la fecha que se muestra es el 1 de enero de 2000.

La lectura del reloj o su puesta en hora con la Raspberry Pi necesita el comando `hwclock` :

Sintaxis

```
hwclock [opciones]
```

- r Lee el reloj RTC y muestra el resultado por la salida estándar.
- s Pone en hora el reloj del sistema a partir del reloj RTC.
- w Pone en hora el reloj RTC a partir del reloj del sistema.

Para leer el contenido del reloj RTC y mostrar el resultado:

```
pi@raspberrypi:~ $ sudo hwclock -r
sab. 01 de ene. 2000 01:01:07 CET -0. 240807 segundos
```

El reloj RTC se puede poner en hora a partir del reloj del sistema, si está en hora, gracias a una conexión a Internet:

```
pi@raspberrypi ~ $ sudo hwclock -w
```

Como siempre ocurre con Linux, la ausencia de mensaje de error indica que la operación se desarrolla correctamente. El reloj RTC ahora se

sincroniza con el reloj del sistema.

Durante los siguientes inicios, el sistema recuperará automáticamente la hora del reloj guardada. Por tanto, es posible usar una Raspberry Pi no conectada a la red pero con la hora de sistema correcta.

f. Uso de varias tarjetas ABElectronics



La foto anterior presenta el uso de las tarjetas presentes en los párrafos anteriores antes de situar los espaciadores y configurar las direcciones.

Conclusión

Para el control de proyectos complejos que necesiten entradas analógicas y numerosas salidas digitales, estas tarjetas proporcionan una solución compacta, gracias a su apilamiento en la GPIO de la Raspberry Pi. La existencia de la tarjeta RTC Pi Plus permite abordar la realización de un sistema autónomo capaz de permanecer sincronizado con la hora real.

6. PiFace Digital 2

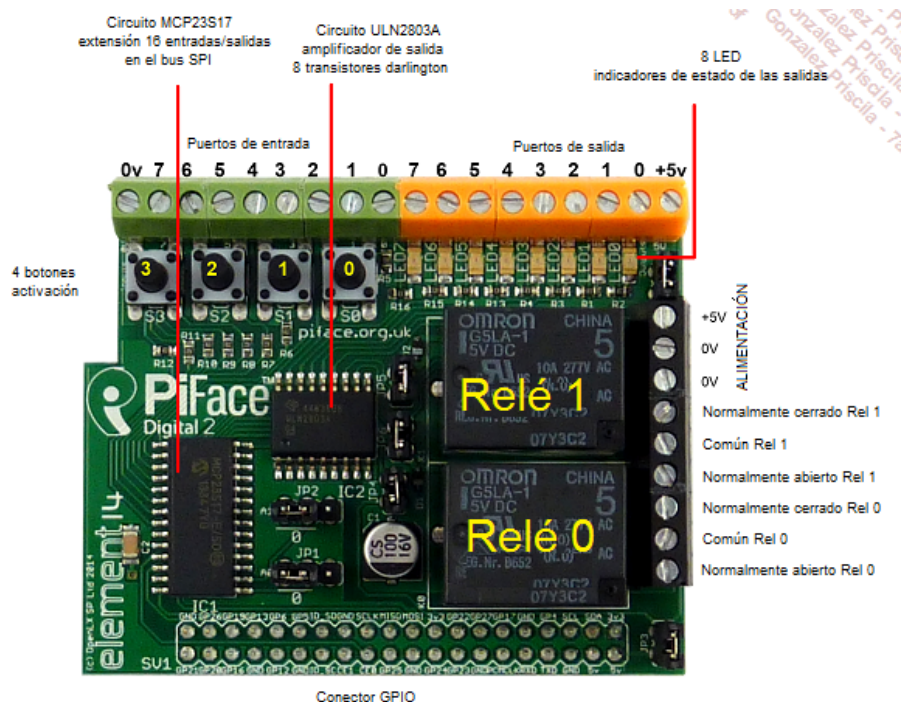
La tarjeta PiFace Digital 2 se ha puesta a punto por la School of Computer Science (escuela de ciencia de la computación) de la universidad de Manchester.

a. Presentación de la tarjeta PiFace Digital 2

La tarjeta PiFace Digital 2 se entrega lista para su uso. Sus dimensiones casi respetan el formato HAT (57 x 68 mm). Se conecta al puerto GPIO de la Raspberry Pi y un tope de caucho pegado debajo de la tarjeta se apoya en la toma HDMI para estabilizar la tarjeta e impedir eventuales cortocircuitos.

Está provista de bornes enroscados para todas las entradas y salidas, lo que facilita la conexión de los cables y evita deterioros que podrían aparecer a la larga en otro tipo de conectores (soldaduras, bornes elásticos, etc.).

La tarjeta PiFace Digital 2 pone a disposición del usuario ocho entradas y ocho salidas digitales. Cuatro de las entradas están provistas de botones de activación que se pueden utilizar para las pruebas. Todas las salidas se equipan con un diodo que indica el estado de cada salida. Por último, dos de las salidas controlan cada una un relé. También es posible acceder directamente a los contactos del relé mediante un conjunto de conectores. Esto aumenta considerablemente la potencia que puede conmutar la tarjeta PiFace.

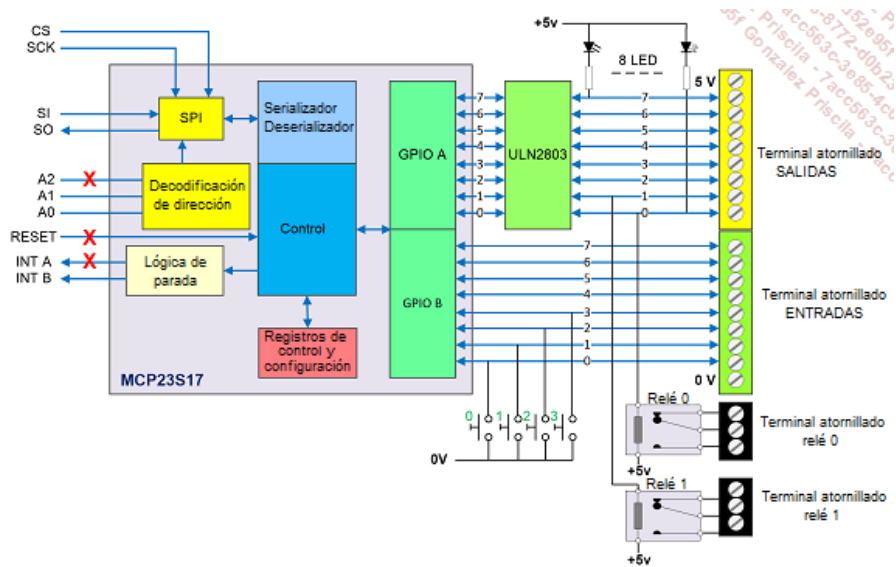


Esta tarjeta, que se vende por 35 € aproximadamente, es una variación de la PiFace original para la Raspberry Pi 3. No responde a las especificaciones HAT porque no hay agujeros de fijación, sin duda por falta de espacio, así como los huecos para los cables planos CSI de la cámara y DSI de la pantalla.

b. Esquema de la tarjeta

El esquema de la tarjeta PiFace Digital 2 no está disponible. Es posible reconstruirlo a partir de la diversa documentación disponible en Internet.

La tarjeta PiFace utiliza el bus SPI para comunicarse con la Raspberry Pi. Esto deja libres y disponibles los puertos de la GPIO para otros usos.



El núcleo de la tarjeta PiFace es un circuito MCP23S17. Este circuito existe en versión I²C (MCP23017) y en versión SPI. Esta última es la que se usa en la PiFace 2. La MCP23S17 ofrece 16 entradas/salidas digitales repartidas en dos puertos GPIO A y GPIO B, que ofrecen 8 E/S cada uno. Cada clavija del circuito se puede configurar como entrada o salida individualmente. En la tarjeta PiFace 2, los puertos son dedicados: GPIO A se configura como salida y ataca a una red de transistores Darlington capaces de controlar cada uno una carga de 500 mA bajo 50 voltios. Cada salida está equipada con un diodo LED que indica el estado de la salida. Cuando la salida está activada, el LED correspondiente se enciende.

Las salidas 0 y 1 de la GPIO A atacan también un relé. Cada relé tiene una potencia de corte de 10 A bajo 250 voltios. Sin embargo, la proximidad de la tarjeta PiFace 2 con la Raspberry Pi hace que debamos ser prudentes y se desaconseja la aplicación de la tensión del sector en estos relés. Por otro lado, las tarjetas generalmente no están protegidas mediante una cubierta durante las pruebas y es posible el contacto con un punto que tenga corriente. El contacto de alguna parte del cuerpo con el sector puede ser mortal.

Es obligatoria la prudencia más extrema: es necesario absolutamente limitar la tensión conmutada por los relés a 24 voltios como máximo, para evitar cualquier accidente.

Las entradas están provistas cada una de una resistencia pull-up a +5 voltios y se activan cuando el nivel en el borne de entrada pasa a 0. Las cuatro primeras entradas están numeradas con 0, 1, 2 y 3, están equipadas con un botón de activación. Esta disposición es muy útil para las pruebas.

Del lado de la GPIO, la tarjeta está conectada al bus SPI. Las señales SCLK, MOSI y MISO unen las clavijas correspondientes de la MCP23S17. Cada tarjeta está dotada de una dirección configurada por los jumpers JP1 y JP2 que pone a 0 o a 1 la línea de dirección A0 y A1.

En las últimas versiones de la tarjeta PiFace, el cable de interrupción INT B conecta la clavija 22 de la GPIO de la Raspberry Pi (GPIO 25). No se utiliza por el software proporcionado con la tarjeta actualmente.

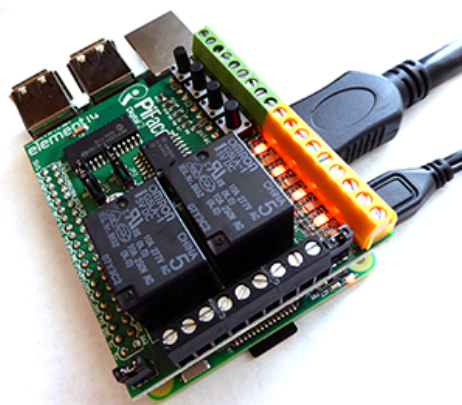
➤ Una interrupción es una señal que provoca la parada del programa actual para ejecutar un programa destinado a tratar la interrupción, así como la causa que la provoca. Esto evita, por ejemplo, tener que leer sistemáticamente las entradas para comprobar si se ha pulsado un botón. Con las interrupciones, es la pulsación del botón la que desencadena la ejecución del programa encargado de tratar la información "botón pulsado".

Los cables marcados con una cruz en el esquema de la tarjeta no se conectan a las clavijas de la GPIO (Reset, A2, INT A).

c. Conexión de la tarjeta

La tarjeta PiFace se conecta directamente a la GPIO de la Raspberry Pi. Por tanto, las precauciones que deben tenerse en cuenta son las mismas que las mencionadas para cualquiera de las tarjetas descritas anteriormente: control visual de la GPIO, restablecimiento de las clavijas desplazadas e inserción cautelosa del conector en la GPIO.

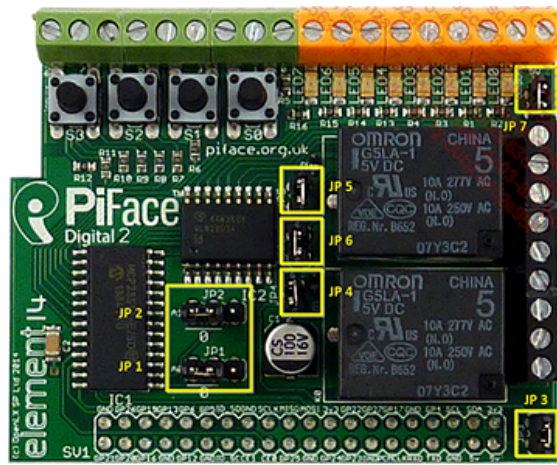
Realice un control visual después de la inserción. No debe haber ningún desplazamiento entre el conector hembra y la GPIO de la Raspberry Pi. No debe ser visible ninguna clavija.



La tarjeta PiFace se adapta perfectamente a la Raspberry Pi, incluso aunque no cumple el formato HAT especificado por la Fundación Raspberry Pi.









d. Configuración de la tarjeta

La tarjeta PiFace Digital 2 tiene siete conectores provistos de jumpers que permiten configurarla.



JP1 y JP2

Los jumpers JP1 y JP2 se correspondan con las direcciones A0 y A1 del bus SPI. Permiten asignar una dirección entre las cuatro disponibles en la tarjeta PiFace. Por defecto, la tarjeta PiFace 2 se configura en la dirección 0.

Dirección	JP1 - A0	JP2 - A1
0	 0	 0
1	 0	 0
2	 0	 0
3	 0	 0

JP3

JP3 está situado junto al conector GPIO. Cuando está presente, conecta la alimentación de 5 voltios de la tarjeta PiFace 2 con la alimentación de 5 voltios disponible en el conector GPIO de la Raspberry Pi. La tarjeta PiFace se alimenta con los 5 voltios proporcionados por la Raspberry Pi. Por el contrario, si la alimentación de 5 voltios se aplica al borne de la tarjeta PiFace 2, es esta última la que alimenta a la Raspberry Pi.

JP4

JP4 se sitúa entre los relés y los jumpers JP1 y JP2. Pone en marcha los diodos de protección o diodos de vía libre, que protegen los transistores de salida del circuito ULN2803A. Estos diodos eliminan la sobretensión generada cuando se conecta una carga inductiva. Como la tarjeta tiene dos relés, es prudente dejar JP4 en su lugar para garantizar la protección de los transistores Darlington de salida.

JP5 y JP6

JP5 y JP6 se ubican entre los relés y el circuito ULN2803, poniendo en marcha los relés. Si desea desactivar los relés, elimine estos jumpers.

JP7

JP7 se sitúa en un ángulo de la tarjeta, entre los terminales. Desconecta la alimentación de todas las salidas de la tarjeta PiFace. Si elimina este jumper, los LED y los relés dejan de funcionar. Evite manipular este puente cuando la Raspberry Pi esté en funcionamiento. La subida de corriente que corre el riesgo de producirse cuando la pone en su lugar puede provocar un reinicio de la Raspberry Pi.

e. Comprobaciones de la tarjeta

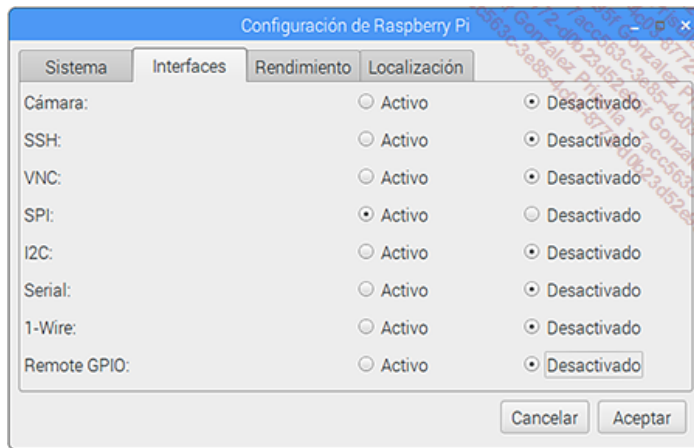
La tarjeta PiFace utiliza el bus SPI para comunicarse con la Raspberry Pi. Por defecto el bus SPI está desactivado, porque hay muchos usuarios de la Raspberry que no lo necesitan.

Activación del bus SPI

La primera etapa de la puesta en marcha de la tarjeta es la activación del bus SPI.

En modo gráfico, active el bus SPI con el menú principal del escritorio de Raspbian Jessie:

Menú - Preferencias - Configuración de la Raspberry Pi - pestaña Interfaces



Para activar el bus SPI, pulse en el botón **Activo** de la línea **SPI**. El bus SPI se activará durante el siguiente inicio de la Raspberry Pi.

En modo texto, edite el archivo `/boot/config.txt`:

```
pi@raspberrypi:~ $ sudo nano /boot/config.txt
```

Modifique la línea que permite activar el bus SPI. Este se activará durante el siguiente inicio de la Raspberry Pi.

```
# Uncomment some or all of these to enable the optional hardware
interfaces
dtparam=spion
```

Reinicie la Raspberry Pi.

Instalación de las librerías de la PiFace

La segunda etapa es la instalación de las librerías de gestión de la tarjeta PiFace. Las librerías de la tarjeta PiFace están integradas en la Raspbian. La instalación se hace a partir de los repositorios de la distribución.

En primer lugar, actualice su distribución.

```
pi@raspberrypi ~ $ sudo apt-get update
pi@raspberrypi ~ $ sudo apt-get upgrade
```

Instale las librerías de gestión de la tarjeta PiFace, así como el emulador en modo gráfico:

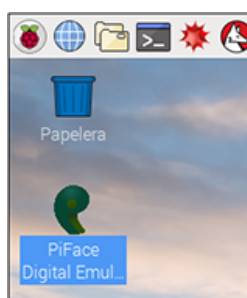
```
[style code] pi@raspberrypi ~ $ sudo apt-get install
python3-pifacedigitalio python-pifacedigitalio
```

Si desea instalar el emulador y usar la tarjeta PiFace con Scratch:

```
pi@raspberrypi ~ $ sudo apt-get install
python3-pifacedigital-scratch-handler
pi@raspberrypi ~ $ sudo apt-get install
python3-pifacedigital-emulator
```

En función de su versión de Raspbian, se añadirá o actualizará un determinado número de archivos. También es posible que su versión de Raspbian ya integre los archivos necesarios. En este caso, un mensaje le indica que ya dispone de la versión más reciente disponible.

Como consecuencia del procedimiento de instalación, aparece un nuevo icono en el escritorio de su máquina. Este icono abre el emulador gráfico de la tarjeta PiFace.

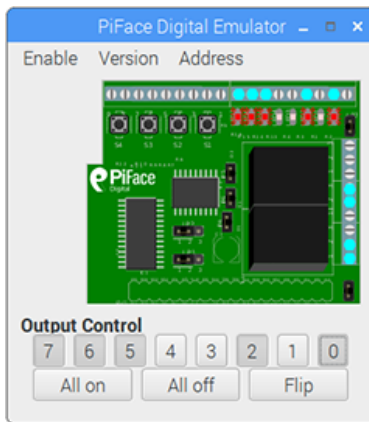


Inicio de la interfaz gráfica

Con un doble clic en el icono, se abre el emulador gráfico de la tarjeta PiFace.

➤ En función de las evoluciones del emulador, la interfaz puede diferir ligeramente de la que se presenta aquí. Esta versión es la que estaba disponible mientras se redactaba este libro.

Se abre una ventana gráfica en la que figura una representación de la tarjeta PiFace. En la parte inferior de la ventana hay botones que permiten modificar los valores aplicados a las salidas de la tarjeta. Durante la primera apertura de la interfaz, se muestra la versión 2 de la tarjeta PiFace. Si tiene la tarjeta PiFace Digital primera versión, pulse en **Versión** en el menú situado en la parte superior de la ventana y después seleccione **PiFace Digital** en la lista. A partir de este momento la interfaz mostrará la tarjeta PiFace Digital.



Menu de la interfaz gráfica PiFace

Enable

- **Output Control:** cuando se selecciona esta opción, lo que se indica con una casilla de selección, están habilitados los botones de comando de las salidas. Cuando esta opción no está seleccionada, los botones están deshabilitados, las salidas se ponen a 0 y es imposible modificar su estado.
- **Input Pullups:** cuando se selecciona esta opción, las entradas están conectadas al riel de alimentación 5V por una resistencia pull-up de 100k Ω .

Version

- **PiFace Digital:** muestra la interfaz de la tarjeta PiFace para el modelo B (GPIO 26 clavijas).
- **PiFace Digital 2:** muestra la interfaz de la tarjeta PiFace para la Raspberry Pi 3 modelo B (y las Raspberry Pi que tienen una GPIO de 40 clavijas).

Address

- Permite seleccionar la dirección de la tarjeta a la que se aplican los comandos (**0, 1, 2, 3**).

Los tres botones ubicados debajo de los comandos de las salidas permiten:

- **All on:** activar todas las salidas.
- **All off:** desactivar todas las salidas.
- **Flip:** invertir el estado de las salidas. Las salidas activadas se desactivan, y a la inversa.

Mostrar los valores de las entradas/salidas

Cuando una salida está activada, un círculo azul sustituye el diseño del tornillo correspondiente en el borne y se ilumina el LED rojo. En los terminales de los relés, los bornes en contacto con cada relé también se representan con un círculo, que sustituye al tornillo.

En las entradas, el círculo azul indica una entrada activada. Los botones de activación S1 a S4 (se corresponden respectivamente con las entradas 0 a 3) son de color amarillo cuando se mantienen pulsados. La entrada correspondiente se indica con un círculo azul. En la imagen anterior, se pulsa el botón S2 y se activa la entrada 1.

La interfaz gráfica permite realizar una prueba rápida, fiable y visual de la tarjeta PiFace.

Programas en Python

Si bien todos los lenguajes de programación pueden acceder a la tarjeta PiFace, el lenguaje predilecto en la Raspberry Pi es Python. Los siguientes ejemplos muestran cómo acceder a las salidas, entradas y cómo usar las interrupciones. Para ejecutar uno de los scripts Python, escriba en línea el comando:

```
python3 nombre_del_script.py
```

➔ Vaya al capítulo Programar en Python para más información sobre este lenguaje.

Comprobación de parpadeo de un LED

El script *pfio_led_1.py* hace parpadear el LED 0 de la tarjeta PiFace. El relé correspondiente se activa cuando el LED se enciende. Este script está disponible para su descarga.

```
# Script pfio_led_1.py
# Parpadeo del LED 0 de la PiFace
# Con comando del relé

# Importar las funciones de la PiFace
import pifacedigitalio as pfio
# Importar la función sleep
from time import sleep

# Inicialización
pfio.init()
pf=pfio.PiFaceDigital()
while (True):
    # Poner el LED 0 en el estado ENCENDIDO (1)
    pf.leds[0].turn_on()
    # Esperar 0.5 segundos
    sleep(0.5)
    # Poner el LED 0 en el estado APAGADO (0)
    pf.leds[0].turn_off()
    # Esperar 0.5 segundos
    sleep(0.5)
```

Lectura de las entradas

El script *pfio_lit_1.py* lee el estado de la entrada 1 de la tarjeta PiFace. Si se acciona la activación S2 conectada a esta entrada, el programa enciende el LED 1 durante 0,5 segundos. Este script está disponible para su descarga.

```

# Script pfiio_lit_1.py
# Leer la entrada 1
# Enciende el LED 1 durante 0,5 segundos
# cuando la activación S2 está activada

# Importar las funciones de la PiFace
import pifacedigitalio as pfiio
# Importar la función sleep
from time import sleep

# Inicialización
pfiio.init()
pf=pfiio.PiFaceDigital()
while (True):
    # Leer la entrada 1
    lectura=pf.input_pins[1].value
    if(lectura==1):
        pf.leds[1].turn_on()
        sleep(0.5)
        pf.leds[1].turn_of()

```

Uso de las interrupciones

El script *pfiio_int_1.py* implementa un listener que monitoriza los puertos de entrada y desencadena el tratamiento de esta interrupción, únicamente cuando se produce el evento. Esto evita monopolizar los recursos del procesador para hacer *polling* (escaneo permanente) de las entradas. Este script está disponible para su descarga.

```

# Script pfiio_int_1.py
# Lectura de las entradas por interrupción

# Importar el módulo pifacedigitalio
import pifacedigitalio
pifacedigitalio.init()

pfd=pifacedigitalio.PiFaceDigital()
# Definir la función a ejecutar
# que se pasará como argumento al listener
# también llamada función de callback o función de rellamada
def toggle_led0(event):
    pfd.leds[0].toggle()

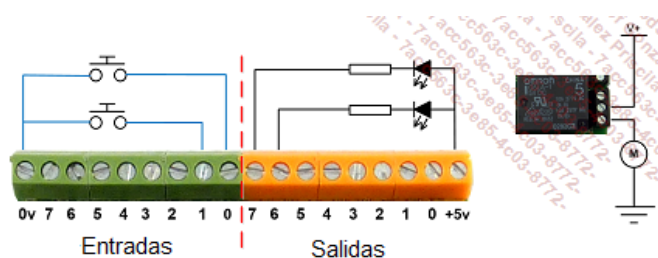
# Monitorizar los eventos en los puertos de entrada
# y ejecutar la función de callback
listener=pifacedigitalio.InputEventListener()
listener.register(0, pifacedigitalio.IODIR_ON, toggle_led0)
listener.activate()

```

Cada pulsación en el botón de activación S1 provoca alternativamente el encendido y el apagado del LED 0. La desactivación de la interrupción se hace con `listener.deactivate()`, la última instrucción del script.



Conexión de elementos exteriores



Los esquemas anteriores indican algunas conexiones posibles en la PiFace. De izquierda a derecha tenemos:

- Dos botones de activación, conectados a las entradas 0 y 1.
- Dos diodos LED conectados a las salidas 6 y 7.
- Un motor conectado a una salida relé que se alimentará cuando la salida correspondiente se active.

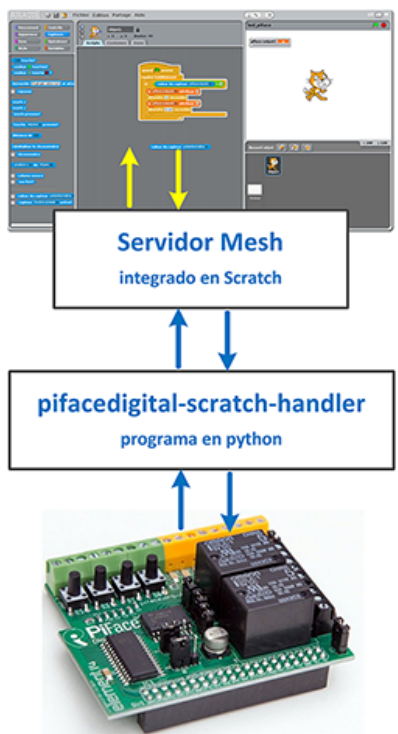
f. Control de la tarjeta PiFace en Scratch

Las versiones recientes de Raspbian integran las herramientas de gestión de la tarjeta PiFace y Scratch dispone de bloques que permiten gestionar las entradas/salidas de la tarjeta. Esto abre la puerta al uso de la tarjeta PiFace en unión con Scratch y permite considerar las interacciones sencillas entre el software Scratch y el mundo exterior.

➔ Vaya al capítulo Programar en Scratch para descubrir las nociones básicas del uso de Scratch.

Principio de funcionamiento

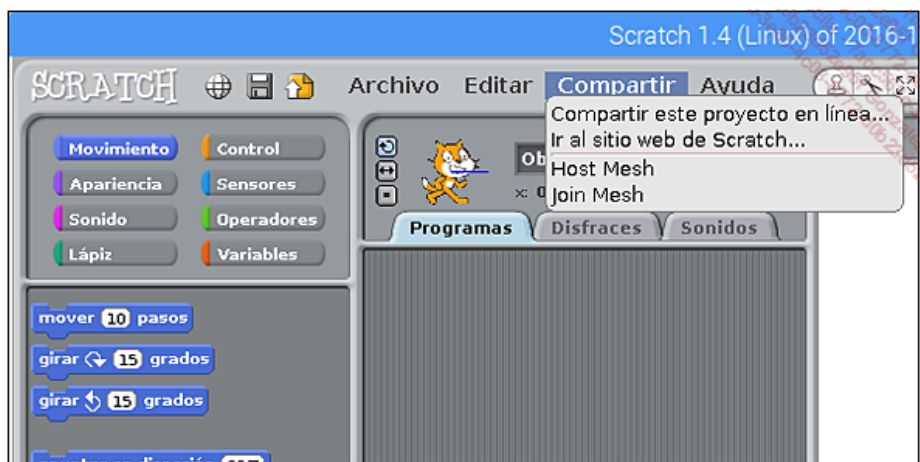
Monte la tarjeta PiFace Digital 2 en la Raspberry Pi 3. Scratch funciona por defecto de manera autónoma en la Raspberry Pi y no se puede comunicar con la tarjeta PiFace. Para que pueda comunicarse con el mundo exterior, es necesario implementar un servidor llamado Mesh. Originalmente se creó para permitir a varios proyectos Scratch interactuar, incluso si se ejecutan en ordenadores diferentes, Mesh permite a los proyectos compartir variables y mensajes. Esto también ofrece la oportunidad de crear programas que se comunican con el mundo real.



El esquema anterior explica la unión entre los diferentes elementos. Cuando Scratch arranca, es necesario activar de arriba hacia abajo los programas que permiten a Scratch comunicarse con la tarjeta PiFace Digital 2.

Arranque del servidor Mesh

El servidor Mesh existe en Scratch, pero no está activado por defecto. Para su arranque, lance Scratch. Con la tecla [SHIFT] pulsada, pulse en el menú **Compartir** y suelte la tecla [SHIFT].



Habitualmente, el menú **Compartir** tiene dos líneas. Pulsar la tecla [SHIFT], ha permitido mostrar dos líneas adicionales, relativas al servidor Mesh.

Pulse en **Host Mesh** para iniciar el servidor.



La apertura de una ventana con la dirección de la máquina (imagen anterior), indica que la operación se ha ejecutado correctamente y que el servidor Mesh está disponible en esta dirección.

Pulse en **Aceptar** para confirmar el arranque del servidor Mesh.

Scratch ahora está listo para hablar con los programas externos.

Arrancar el administrador de la tarjeta PiFace

El programa en Python encargado de garantizar la unión entre el servidor Mesh y la tarjeta PiFace se llama pifacedigital-scratch-handler. Para ejecutar este programa, abra un terminal y use el control /usr/bin/pifacedigital-scratch-handler:

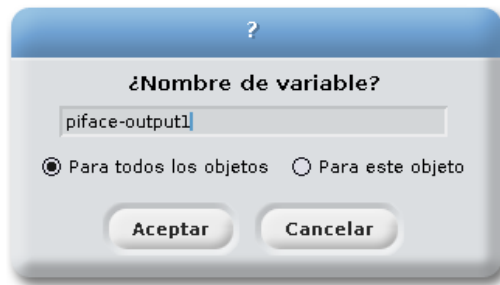
```
pi@raspberrypi:~ $ /usr/bin/pifacedigital-scratch-handler
Connecting... Connected.
sending: sensor-update "piface-input1" 0
sending: sensor-update "piface-input2" 0
sending: sensor-update "piface-input3" 0
sending: sensor-update "piface-input4" 0
sending: sensor-update "piface-input5" 0
sending: sensor-update "piface-input6" 0
```

```
sending: sensor-update "piface-input7" 0
sending: sensor-update "piface-input8" 0
ending main thread
```

El administrador de la tarjeta PiFace está en su sitio y listo para enviar información en ambos sentidos entre Scratch y la tarjeta PiFace.

Activar una salida con Scratch

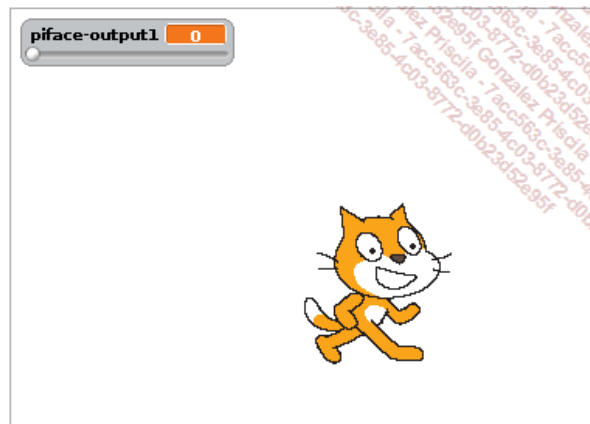
Cree una variable en Scratch. Llámela **piface-output1** respetando escrupulosamente la sintaxis, para que se reconozca por el administrador pifacedigital-scratch-handler.



Valide la creación de esta variable pulsando el botón **Aceptar**. Los bloques correspondientes a esta variable aparecen en la zona situada a la izquierda de la ventana de Scratch, como se muestra a continuación:



La variable también se muestra en la escena. Pulse dos veces en la variable para que aparezca un cursor, de la siguiente manera:



Mueva el cursor: en el terminal puede observar que el movimiento del cursor se transmite bien:

```
received sensor-update: ["piface-output1", '0', '']
received sensor-update: ["piface-output1", '1', '']
received sensor-update: ["piface-output1", '2', '']
received sensor-update: ["piface-output1", '3', '']
received sensor-update: ["piface-output1", '4', '']
received sensor-update: ["piface-output1", '5', '']
received sensor-update: ["piface-output1", '6', '']
received sensor-update: ["piface-output1", '9', '']
```

Cuando el valor de **piface-output1** es diferente de cero, el LED correspondiente a esta salida en la tarjeta PiFace se enciende y el relé se cierra. Scratch se comunica con el mundo real. Queda por probar este control desde un programa Scratch.

Hacer parpadear un LED

Cree el siguiente programa en Scratch (disponible para su descarga [led_piface.sb](#)):



Arranque el programa pulsando en la bandera verde. El LED 1 de la tarjeta PiFace parpadea una vez por segundo y el relé correspondiente se cierra.

➤ La tarjeta PiFace dispone de ocho salidas. Estas son accesibles a través de las variables piface-output1, piface-output2...

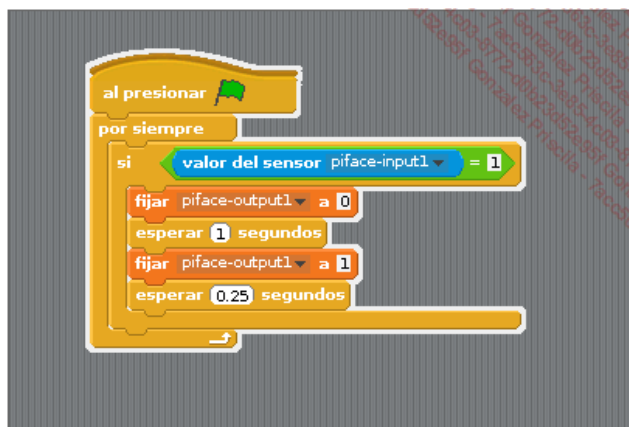
Ejecutar un programa pulsando un botón

La tarjeta Piface también dispone de ocho entradas. Las cuatro primeras entradas están equipadas de botones para facilitar las pruebas.



Las entradas de la tarjeta PiFace son accesibles en la sección **Sensores - deslizador**. La captura de pantalla anterior muestra la lista de los sensores que se muestran pulsando en la flecha situada a la derecha de **deslizador**. Los ocho elementos de la parte inferior de la lista son las entradas de la tarjeta: **piface-input1**, **piface-input2**...

Pulse en **piface-input1** e integre este sensor al programa de la siguiente manera (disponible para su descarga [led_piface_input.sb](#)):



Ejecute el programa pulsando en la bandera verde. No pasa nada mientras que no pulse el botón S0 unido a la entrada 1. Cuando el botón S0 se pulsa, el LED correspondiente a la salida1 parpadea una vez por segundo.

➤ Observe que cuando el botón se suelta, el parpadeo se detiene dejando el LED iluminado.

Modifique el programa para que el LED quede apagado (solución disponible para su descarga: [led_piface_input0.sb](#))

Conclusion

La PiFace Digital 2 es extremadamente compacta y se adapta perfectamente a la Raspberry 3. Las salidas equipadas con transistores Darlingon permiten controlar las cargas correspondientes (50 V/500 mA). Los dos relés permiten la conmutación de fuertes corrientes. Los terminales enroscados dan seguridad a las conexiones en caso de uso de la Raspberry Pi en un sistema que genera vibraciones. El emulador gráfico de la tarjeta y la disponibilidad de módulos incluidos en Raspbian y Scratch facilitan el uso de la tarjeta PiFace, particularmente para un uso escolar. Es una elección a considerar para controlar una instalación que solo necesita entradas/salidas todo o nada.

Solo le falta a esta tarjeta la posibilidad de tratar señales analógicas. Será necesario añadir a la Raspberry Pi tarjetas de conversión analógico-digital y/o digital-analógico para aprovechar esta posibilidad.

Utilizar tarjetas específicas

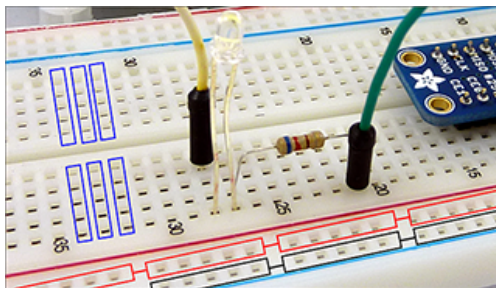
El uso de la Raspberry Pi con componentes externos implica habitualmente la puesta en marcha de una placa prototipada, también llamada breadboard.

1. Tarjeta prototipada

La tarjeta prototipada es un soporte de experimentación. Permite probar las conexiones y comprobar su funcionamiento antes de realizar una tarjeta definitiva.

Placa reutilizable

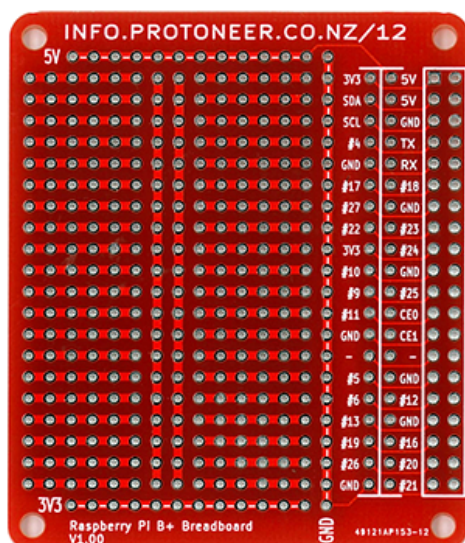
Puede ser reutilizable como la breadboard. En este caso, la placa tiene filas de agujeros conectados entre sí. Estas placas existen en varias dimensiones. Varios railes distribuyen la alimentación a lo largo de la placa. Las filas de agujeros se identifican con cifras, con el objetivo de ofrecer identificadores para la realización de una conexión.



En la foto anterior, se muestran rectángulos que rodean series de agujeros. En las dos zonas centrales, los contactos se juntan en bloques de cinco. No hay relación entre dos bloques contiguos, ni entre las dos partes de la zona central. Por el contrario, en cada lado de la tarjeta cada bloque de un riel de alimentación se conecta a sus vecinos en una misma línea.

Placa no reutilizable

Para los proyectos que utilizan componentes particulares o para entregar corrientes elevadas, los contactos de la breadboard algunas veces son demasiado frágiles. Conviene usar placas de experimentación soldadas. Estas placas son habitualmente de un único uso. Las soldaduras repetidas en las patillas terminan por deteriorarla o despegarla.



La placa anterior dispone de una ubicación para un conector compatible con la GPIO de la Raspberry Pi (a la derecha). Hay rieles de alimentación de 5V y 3,3V (en la parte superior y en la parte inferior de la placa). La masa GND se sitúa en la parte derecha del bloque de patillas y tiene dos columnas de patillas conectadas todas juntas al centro para distribuir la alimentación si es necesario. Las filas de seis patillas ofrecen una zona de cableado libre. Las señales de la GPIO están disponibles en dos filas de patillas identificadas por serigrafía. Pueden recibir conectores o un cable soldado directamente.

Conexión a la GPIO

La conexión de una placa prototipo a la Raspberry Pi se debe realizar sin correr el riesgo de falsos contactos, que podrían hacer pensar un funcionamiento incorrecto de un programa o de la propia GPIO.

El conector GPIO de la Raspberry Pi es un conector estándar de 40 clavijas, con un difusor de 2,54 mm. Hay conectores hembra de 40 puntos adaptados a la GPIO. Existen en diferentes alturas en su versión soldada a una tarjeta de circuito impreso.

También están disponibles en versión conjunto, con un cable metálico de 40 conductores. En el otro extremo hay una placa de circuito impreso provista de pinchos que garantizan la unión con la placa prototipo.

Algunas tarjetas, como estas descritas a continuación, tienen conectores de pinchos. La unión está garantizada directamente mediante conectores hembra-hembra adaptados a los conectores.

2. Control del motor PAP

Para facilitar la puesta en marcha de aplicaciones a base de motores paso a paso (PAP), existen kits con un motor paso a paso 28BYJ-48 alimentado en 5 voltios, además de su circuito de control basado en ULN2803A (ocho transistores Darlington 50 V/500 mA) por unos de 3 €.

Este conjunto se entrega listo para su uso y permite implementar rápidamente el control de un motor paso a paso basado en Raspberry Pi.

Motor paso a paso

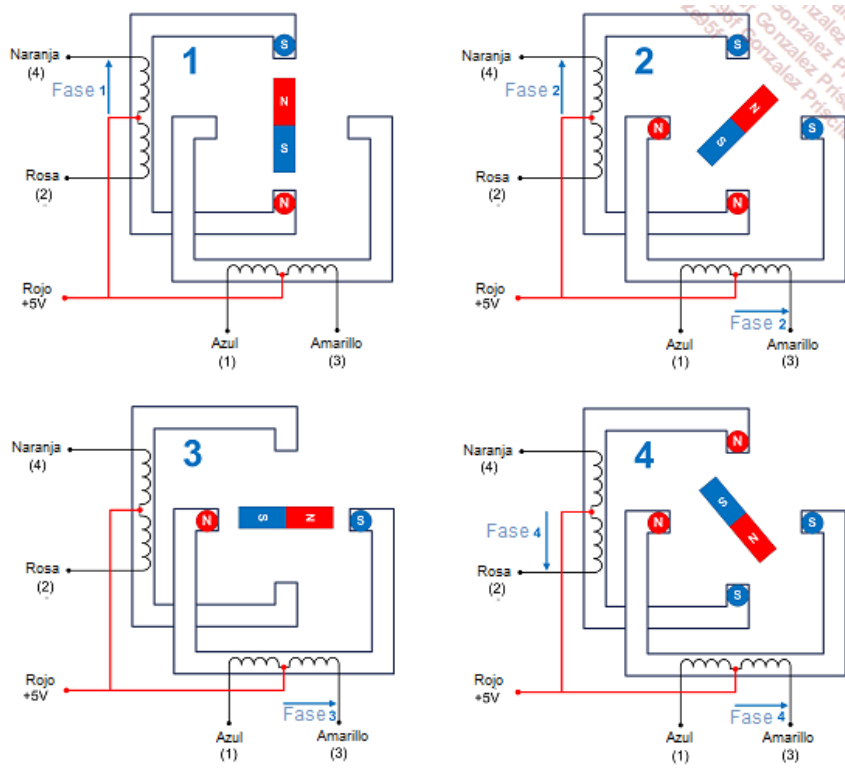
El motor 28BYJ-48 es un motor de cinco cables. Cuatro bobinas impulsan un rotor formado por un imán permanente. Están alimentadas siguiendo una secuencia de ocho fases, las cuatro primeras se representan a continuación.

- Durante la primera fase, el cable 4 (naranja) está conectado a la masa. El campo magnético inducido por la bobina atrae el imán en posición vertical.
- Durante la segunda fase, los cables 4 (naranja) y 3 (amarillo) están conectados a la masa. Esto crea dos polos sur y dos norte. El imán

se posiciona a medio camino de los polos sur y norte creados de esta manera.

- Durante de la fase 3, es solo el cable 3 (amarillo) el que está conectado a la masa. El imán se posiciona horizontalmente.
- Durante de la fase 4, los cables 2 (rosa) y 3 (amarillo) están conectados a la masa. El imán se posiciona entre los polos sur y norte creados.

Las demás fases reproducen este mecanismo hasta que el imán haya completado una vuelta completa. El ciclo se repite para la siguiente vuelta.



El rotor real es un imán de ocho polos, lo que permite obtener pasos más pequeños que los que se representan en los esquemas. La documentación del motor indica el orden de las fases que se deben respetar para una rotación correcta del motor.

	--> Sentido horario -->							
Fase	1	2	3	4	5	6	7	8
4 Naranja	●	●						●
3 Amarillo		●	●	●				
2 Rosa				●	●	●		
1 Azul						●	●	●
	<-- Sentido contrario a las agujas del reloj <--							

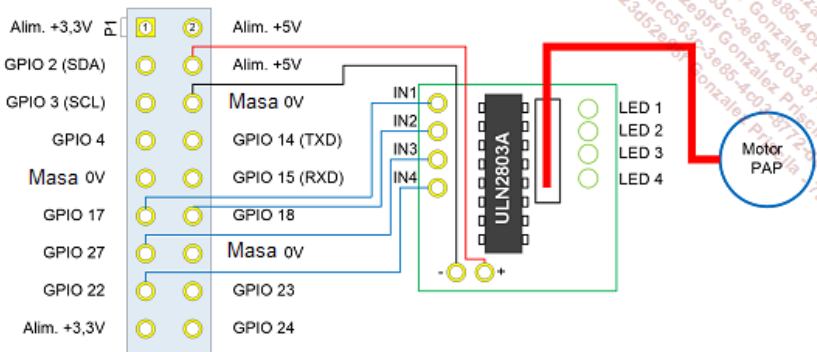
Conexión a la GPIO

Los bornes de alimentación de la tarjeta del controlador del motor paso a paso se conectan a los bornes 4 (+5 voltios) y 6 (masa).

Los bornes del controlador se conectan respectivamente a:

- IN 1: GPIO 17
- IN 2: GPIO 18
- IN 3: GPIO 27
- IN 4: GPIO 22

Es posible usar otras clavijas del conector GPIO de la Raspberry Pi B+, bastará con modificar el programa en consecuencia.



Programa de control

```

#-----
# Script pap.py
# Script de control del motor paso a paso 28BYJ-48
# El script recibe un argumento por línea de comandos
# ha o A = sentido contrario a las agujas del reloj
# h o H = sentido de las agujas del reloj
#
#-----
#!/usr/bin/env python
    
```

```

# Importación de las librerías utilizadas
import RPi.GPIO as GPIO
import time
import sys

def DetenerScript():
    print "Hay que poner un argumento después del nombre del programa:"
    print "a = sentido contrario a las agujas del reloj"
    print "h = sentido de las agujas del reloj"
    sys.exit()

# Si no hay argumento, detener el script
if (len(sys.argv) < 2):
    DetenerScript()

# Si el argumento no es válido detener el script
# Leer el argumento
SentidoDeRotacion = sys.argv[1]
# Comprobar la validez del argumento
ArgumentosValidos = ['a', 'A', 'h', 'H']
if SentidoDeRotacion not in ArgumentosValidos:
    DetenerScript()

# Determinar el sentido de rotación
if (SentidoDeRotacion == 'a') or (SentidoDeRotacion == 'A'):
    Sentido = 'contrario a las agujas del reloj'
if (SentidoDeRotacion == 'h') or (SentidoDeRotacion == 'H'):
    Sentido = 'de las agujas del reloj'

# Uso de los números de puerto GPIO del SoC
GPIO.setmode(GPIO.BCM)

# Clavijas utilizadas: GPIO17,GPIO18,GPIO27,GPIO22
Clavijas = [17,18,27,22]

# Configurar las 4 clavijas como salida
for salida in Clavijas:
    print ("Configuración de las salidas")
    print ("para rotación en el sentido %s" %(Sentido))
    GPIO.setup(salida,GPIO.OUT)
    GPIO.output(salida, False)

# Definición de las variables
ContadorDePaso = 0
TiempoDePausa = 0.5

# Secuencia de las salidas
# según la documentación del 28BYJ-48
NumeroDePaso = 8
Fase = []
Fase = range(0, NumeroDePaso)
Fase[0] = [1,0,0,0]
Fase[1] = [1,1,0,0]
Fase[2] = [0,1,0,0]
Fase[3] = [0,1,1,0]
Fase[4] = [0,0,1,0]
Fase[5] = [0,0,1,1]
Fase[6] = [0,0,0,1]
Fase[7] = [1,0,0,1]

# Apertura de un bloque try except
# para interceptar la pulsación de [Ctrl]+ C
# Esto provoca el reinicio de la GPIO
# y después la parada del programa
try:

    # Bucle principal del programa
    while True:
        # Aplica los valores a las salidas
        # en función de la fase actual
        for salida in range(0, 4):
            SalidaN = Clavijas[salida]
            if Fase[ContadorDePaso][salida]!=0:
                print " Fase %i Activación salida %i"
                %(ContadorDePaso,SalidaN)
                GPIO.output(SalidaN, True)
            else:
                GPIO.output(SalidaN, False)

        if Sentido == 'de las agujas del reloj':
            # Sentido de las agujas del reloj
            ContadorDePaso += 1
        elif Sentido == 'contrario a las agujas del reloj':
            # Sentido contrario a las agujas del reloj
            ContadorDePaso -= 1

        # Cuando se alcanza el final de la secuencia
        # volver al inicio
        if (ContadorDePaso == NumeroDePaso):
            ContadorDePaso = 0
        if (ContadorDePaso < 0):
            ContadorDePaso = NumeroDePaso-1

        # Pausa antes del paso siguiente
        time.sleep(TiempoDePausa)

except KeyboardInterrupt:
    # Reset GPIO settings
    print "Vuelta de la GPIO al estado inicial"
    GPIO.cleanup()

```

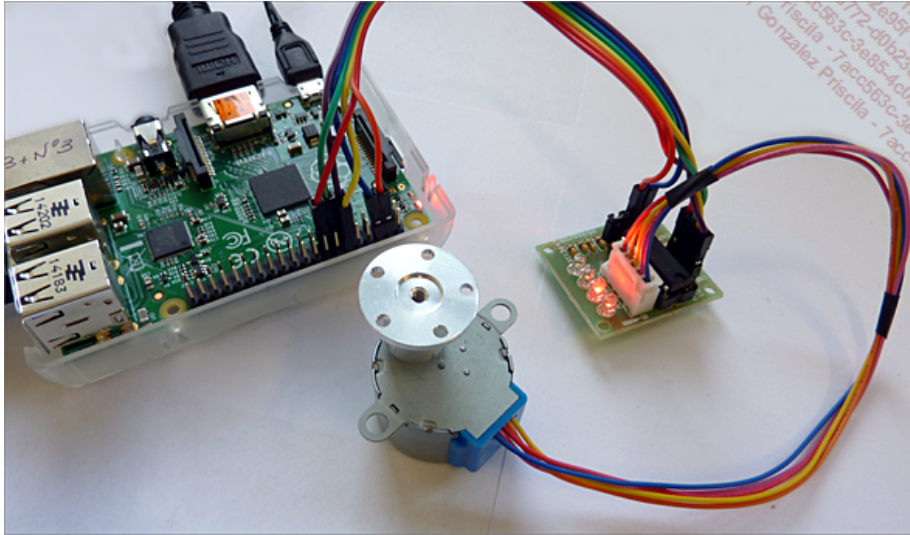
Este programa espera un argumento que indica el sentido de rotación del motor (a = antihorario, h = horario). Después de las inicializaciones, el programa envía sucesivamente comandos destinados a hacer girar el motor en los puertos GPIO 17, 18, 27 y 22.

Estas órdenes corresponden a las fases (numeradas de 1 a 8) que figuran en la documentación del motor. La tabla se escanea tanto en un

sentido como en el otro, en función del argumento que se pasa por línea de comandos.

La pulsación en [Ctrl] C detiene el programa y pone la GPIO en su estado inicial.

La modificación de la variable TiempoDePausa permite variar la velocidad del motor. Atención, como consecuencia de la importante ventaja mecánica del motor 28BYJ-48, el eje puede parecer inmóvil con un tiempo de pausa largo (0,5 segundos). Reduzca el valor de esta variable a 0,05 segundos para observar la rotación del motor.

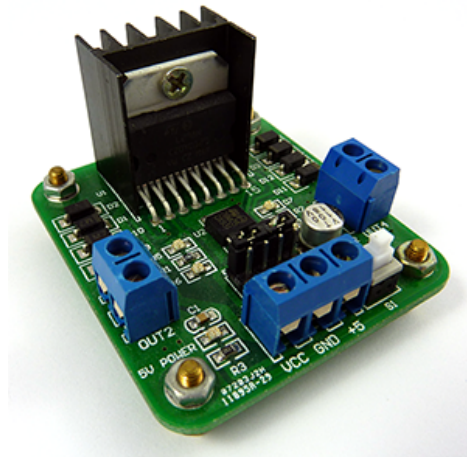


Conclusión

La puesta en marcha de un motor paso a paso con la Raspberry Pi es relativamente fácil. Para el control de varios motores, sin duda será necesario prever una tarjeta de extensión de control de múltiples motores paso a paso. El uso de motores paso a paso reutilizados o de origen indeterminado requiere cierta adaptación del sistema de control.

3. Control de un motor CC

El control de un motor de corriente continua (CC) se hace con un puente en H si el motor debe girar en ambos sentidos.



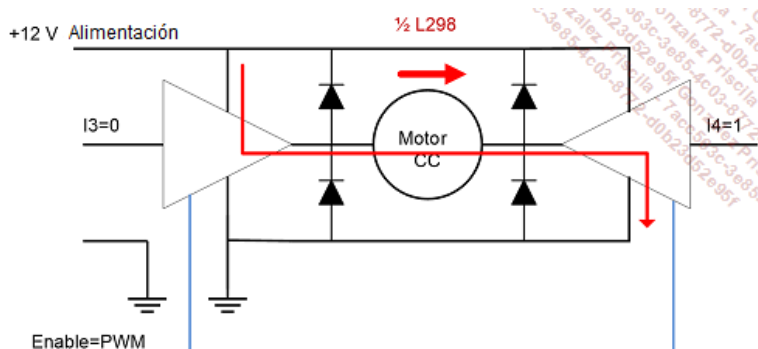
- Un puente en H es un sistema electrónico que permite hacer circular la corriente en la carga a la que se conecta bien en un sentido o en otro en función de una señal de control.

El L298 es un circuito integrado que tiene dos puentes y, por tanto, puede controlar dos motores de corriente continua. Cada puente se debe proteger obligatoriamente con diodos anti-retroceso. La tarjeta anterior, disponible en Internet por algunos euros, facilita la puesta en marcha del circuito integrado L298. Los terminales enroscados aportan seguridad a las conexiones de alimentación y a las uniones de los motores.

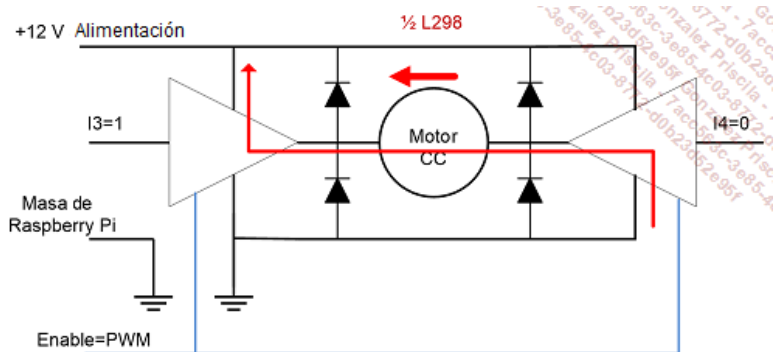
Rotación en ambos sentidos

Cada medio-L298 tiene dos amplificadores. El motor se conecta entre sus dos salidas. Cada amplificador puede conectar el borne del motor al que está conectado a la tensión positiva de alimentación o a la masa. Una entrada Enable valida el funcionamiento de cada uno de los dos amplificadores. Cada amplificador se controla por una entrada (I3 y I4) que determina el sentido de circulación de la corriente y, por tanto, el sentido de rotación del motor.

El método clásico consiste en controlar el sentido de rotación del motor con I3 e I4 y aplicar una señal PWM en la entrada Enable para administrar la velocidad de rotación:



El puente en H contiene los amplificadores, representados por triángulos. La salida del amplificador aplica +12 V o la masa en el borne del motor al que está conectada en función del valor de la entrada. La clavija Enable recibe la señal PWM y permite ajustar la velocidad del motor jugando con la relación cíclica de la señal. En este esquema, el motor gira en el sentido de las agujas del reloj.



En el esquema anterior, las entradas de los amplificadores reciben niveles inversos. La corriente circula en sentido inverso en el motor. Este gira en sentido contrario a las agujas del reloj. La clavija Enable permite ajustar la velocidad del motor, haciendo variar la relación cíclica de la señal aplicada (PWM).

La siguiente tabla indica el estado del motor en función de las diferentes señales en las entradas:

Enable	I3	I4	Motor
1	0	1	Gira en sentido de las agujas del reloj
1	1	0	Gira en sentido contrario a las agujas del reloj
1	0	0	Detenido
1	1	1	Detenido
0	No aplica	No aplica	Detenido

Vemos que si se impone un sentido de rotación a través de I3 e I4, la aplicación del PWM en Enable permite administrar el tiempo durante el que el motor se alimenta.

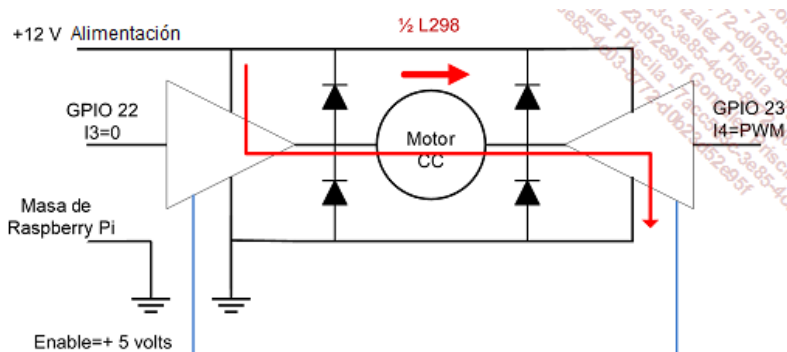
Rotación en un único sentido

En caso de que el motor solo deba girar en un sentido, el enfoque se simplifica. La tarjeta utilizada para esta aplicación se vende por algunos euros en varias tiendas online. La L298 se monta sobre un radiador. La tarjeta gestiona dos motores que consumen 2 A, con una alimentación comprendida entre +7,5 y +35 voltios.

Un interruptor permite poner la tarjeta en modo enchufado/desenchufado y un regulador de 5 voltios (78M05) ofrece esta tensión a la tarjeta. Su corriente de salida se limita a 500 mA y no permite alimentar más de una Raspberry Pi. La tensión de +5 voltios está disponible en un borne atornillado.

Se enciende un LED cuando hay 5 voltios y dos LED por motor muestran el estado de las entradas (I1, I2 y I3, I4). Los diodos de protección están presentes en cada salida.

Un puente permite conectar la clavija Enable a los +5 voltios. El control del motor solo se hace con I3 e I4. Una de las clavijas recibe 0, la otra el PWM. El motor gira en un solo sentido a la velocidad marcada por PWM.



En esta configuración, la clavija Enable recibe una tensión fija de + 5 voltios y permite el funcionamiento permanente de los amplificadores.

Las entradas de los amplificadores reciben una de ellas una señal de parada/encendido y la otra una señal PWM que permite ajustar la velocidad de rotación del motor. En contrapartida a la simplificación del control (dos clavijas en lugar de las tres que hay cuando el motor puede girar en ambos sentidos), el motor solo puede girar en el sentido de las agujas del reloj.

Se utilizan las clavijas GPIO 22 y 23 (BCM). El siguiente programa provoca una aceleración y después una deceleración, y termina parando el motor.

```
# Script cc.py
# Script de control de motor de corriente continua
# Las salidas 22 y 23 (BCM) se utilizan para el control

# Importar las librerías RPi.GPIO y time
import RPi.GPIO as GPIO
import time

# Utilizar la numeración de las clavijas del SoC BCM
GPIO.setmode(GPIO.BCM)

# Configurar la GPIO 23 como salida
GPIO.setup(23, GPIO.OUT)
# Configurar la GPIO 22 como salida
GPIO.setup(22, GPIO.OUT)
print "Configuración de los puertos"

# Crear un objeto I3 para enviar el PWM a 50 Hertz al puerto 22
I3 = GPIO.PWM(22, 50)
# Arrancar el PWM con una relación cíclica de 0 (parada)
I3.start(0)
# Activar el puerto 23 = I4 de la tarjeta de control motor
GPIO.output(23, GPIO.LOW)
print "Arrancar PWM - 23 LOW"

# Esperar un segundo
```

```
time.sleep(1)

# Acelera progresivamente el motor de 0 a 100%
for n in range (10, 105, 5):
    I3.ChangeDutyCycle(n)
    print "Duty ",n
    time.sleep(0.5)

# Decelerar progresivamente el motor de 100% a 0
for n in range (100, 0, -5):
    I3.ChangeDutyCycle(n)
    print "Duty", n
    time.sleep(0.5)

# Para el motor
GPIO.output(23, GPIO.LOW)
print "23 LOW"

# Poner todas las clavijas GPIO en modo entrada
GPIO.cleanup()
```

Conclusión

La variación de tensión entre niveles lógicos de la GPIO (3,3 voltios) y de L298 (5 voltios) no provoca un funcionamiento incorrecto. Sin embargo, para garantizar un funcionamiento estable, se aconseja intercalar un convertor de tensión entre las salidas de la Raspberry Pi y las entradas del L298. De la misma manera, con el objetivo de evitar retrocesos de tensión que pudieran estar causados por cargas inductivas, es necesario prever la protección de las clavijas de la GPIO. Una simple resistencia y un diodo Zener de 3,3 voltios permiten limitar los riesgos. Pero para una protección seria o para aplicaciones industriales es mejor incorporar un diodo Transil o Transzorb.

- Un diodo Zener es un componente diseñado para mantener una tensión constante en sus bornes. Si la tensión aumenta, el diodo Zener se convierte en conductor y evita un aumento de tensión, que sería destructiva para la entrada de la GPIO.

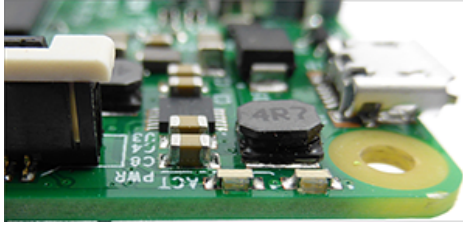
Un diodo Transil o Transzorb es un diodo especial, limitador de voltaje transitorio, utilizado para proteger las conexiones electrónicas delicadas, contactos de relés, etc.

Fiabilidad de la Raspberry Pi

En fabricación, la tasa de Raspberry Pi defectuosas está cercana a 1 por cada 16.000 tarjetas producidas. Cada Raspberry Pi se prueba individualmente antes de salir de la cadena de fabricación. Preguntado sobre esto en agosto de 2016, Kubii (www.kubii.fr/), el distribuidor francés de la Raspberry Pi, afirma que hay una tasa de devoluciones del 0,5 %. Un primer filtro por teléfono apuesta por una reinstalación de NOOBS o la sustitución de la alimentación. Estas dos operaciones resuelven la mayor parte de los errores encontrados.

Por tanto, la Raspberry Pi es un producto fiable. La mayor parte de los problemas clásicos encontrados durante la puesta en marcha y el uso de la Raspberry Pi se clasifican en la página http://elinux.org/R-Pi_Troubleshooting.

Los LED de la Raspberry 3



Los LED de la Raspberry Pi 3 son visibles en la foto anterior. A la derecha, se encuentra el LED rojo de alimentación PWR y, a la izquierda, el LED verde de actividad ACT.

1. LED ACT de la Raspberry Pi 3

a. LED ACT apagado

El LED ACT permanece apagado en ausencia de tarjeta micro SD o si introduce en el conector una tarjeta micro SD virgen (nueva) o defectuosa. Si solo se muestra la pantalla de color es porque hay archivos corruptos, en general *kernel7.img*. Vuelva a crear la tarjeta de nuevo completamente.

En un funcionamiento normal, el LED ACT permanece apagado. Produce destellos de luz cuando el sistema operativo accede a la tarjeta micro SD.

b. LED ACT encendido de manera fija

Si el archivo *start.elf* existe pero está vacío o corrupto, el LED ACT permanece encendido de manera fija. Si sucede esto, intente formatear la tarjeta con la herramienta oficial SDformatter (consulte la sección Formateo de la tarjeta micro SD en el capítulo Preparar la tarjeta micro SD). Esta herramienta se recomienda para las tarjetas SD, SDHC y/o las tarjetas SDXC. Diferentes sistemas operativos y otras herramientas pueden formatear una tarjeta micro SD de manera que esta no pueda ser reconocida correctamente por la Raspberry Pi. Por ejemplo, algunas herramientas solo pueden formatear la primera partición o usar formatos no compatibles como FAT, NTFS o exFAT.

Por el contrario, SDformatter borra todo el contenido de la tarjeta y crea una única partición formateada en FAT32, que es el único formato accesible para la Raspberry Pi.

c. LED ACT parpadeando cuatro veces

Con una tarjeta preparada normalmente (win32DiskImager) si el archivo *start.elf* no está, el LED ACT se enciende pero se apaga cuatro veces rápidamente. Se enciende de nuevo dos segundos y comienza de nuevo a parpadear cuatro veces. No se muestra nada en la pantalla, que permanece en negro.

El LED ACT permanece iluminado de manera fija si la tarjeta micro SD es defectuosa, se ha desenganchado o contiene archivos corruptos. Parpadea durante los accesos a la tarjeta micro SD.

d. LED ACT parpadea siete veces

En la versión Raspbian Jessie, si *start.elf* es correcto, la pantalla muestra el cuadrado coloreado que se ve más abajo, indicando que la GPU ha arrancado. En Raspbian PIXEL, esta pantalla ya no se muestra y la pantalla muestra directamente la o las frambuesas que indican el número de núcleos de la tarjeta utilizada.



Si el nodo *kernel.img* o *kernel7.img* no se pueden cargar porque no están, el LED ACT se enciende siete veces rápidamente y después permanece encendido dos segundos antes de volver a parpadear. Vuelva a copiar un *kernel.img* desde otra tarjeta o vuelva a crear completamente la tarjeta micro SD.

2. LED PWR de la Raspberry Pi 3

El LED PWR se enciende cuando se aplican 5V de tensión a la tarjeta. Si la tensión de alimentación de la tarjeta por debajo de los 4,6 voltios, un detector enciende el LED verde PWR.

a. Parpadeo del LED PWR

Esta caída de tensión se puede deber a una alimentación defectuosa o incapaz de entregar la corriente solicitada por la Raspberry Pi y los periféricos conectados a sus puertos USB.

El cable USB que conecta la alimentación a la toma micro SD de la Raspberry Pi también puede ser el motivo de una caída de tensión importante si es de mala calidad o está defectuoso. Es conveniente comprobarlo en caso de duda.

Es importante vigilar los parpadeos del LED rojo PWR porque las caídas de tensión, incluso si son breves, pueden provocar defectos de escritura en la tarjeta micro SD y comprometer el correcto funcionamiento del sistema.

b. LED PWR apagado

Si el LED PWR permanece apagado, la alimentación no está conectada correctamente o está fallando. Compruebe en primer lugar si hay tensión en la toma donde está enchufada la alimentación. Tenga cuidado, en particular si utiliza una toma múltiple.

Desconecte todos los periféricos conectados en la o las tomas USB.

Si hay tensión, el problema puede estar en la alimentación. Puede estar fuera de servicio o estar en modo seguro a causa de un consumo de

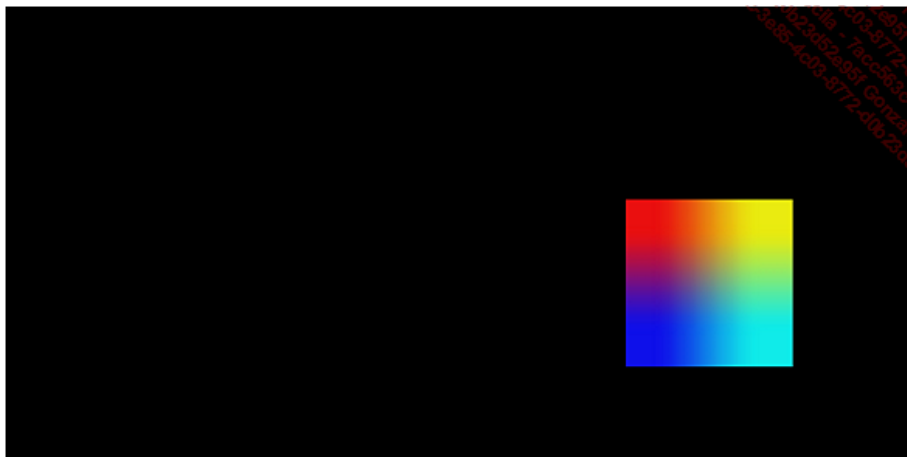
corriente demasiado importante en su salida. En este caso, algunas alimentaciones se protegen. Una alimentación/cargador utilizado sin problema con un teléfono móvil corre el riesgo de ser insuficiente para la Raspberry Pi y se pone en modo seguro. La Fundación recomienda una alimentación 5V / 2,5A para la Raspberry Pi 3.

Compruebe cuidadosamente el cable que une la alimentación a la toma micro USB macho. No debe tener ningún corte ni estar dañado. Controle visualmente el estado de la toma micro USB en el extremo del cable. En caso de duda, cambie la alimentación. Incluso si el cable no presenta daños, puede estar cortado por debajo del recubrimiento en la salida de la alimentación o de la toma USB. Esto se produce a la larga en cables enrollados de manera apretada alrededor de la alimentación.

Controle el estado de la toma hembra micro USB soldada a la tarjeta de la Raspberry Pi. No debe ceder a la presión de un dedo. Una toma que se mueve implica la posibilidad de un deterioro de las soldaduras que la mantienen unida a la placa del circuito impreso.

3. Indicador de tensión baja en la pantalla

Las versiones actuales del firmware de la Raspberry Pi integran un detector de tensión baja. Este programa se ejecuta por la GPU y muestra en el ángulo superior derecho de la pantalla un indicador que cambia según la versión del sistema operativo utilizado. Con las versiones anteriores a PIXEL, es un pequeño cuadrado multicolor que se parece al gran cuadrado de color que cubre la pantalla durante el inicio de la Raspberry Pi.



Raspbian PIXEL introduce un símbolo más elocuente, y la sobre-tensión se indica por un rayo amarillo superpuesto en la imagen mostrada.



Este símbolo aparece a veces brevemente durante los primeros segundos del boot. En caso de tensión insuficiente (inferior a 4,6V), se superpone a la imagen normalmente mostrada en la pantalla.

Como sucedía anteriormente, hay que buscar la causa de esta advertencia en el entorno de la alimentación.

Es posible (aunque no está recomendado) eliminar esta visualización (así como la de la temperatura máxima a continuación) añadiendo `avoid_warnings=1` en el archivo `config.txt`.

4. Indicador de temperatura máxima en la pantalla

Las versiones recientes del firmware de la Raspberry Pi también integran un detector para el exceso de temperatura máxima del SoC. Este programa se ejecuta por la GPU y esta vez muestra en el ángulo superior derecho de la pantalla un pequeño cuadrado rojo para las versiones anteriores a PIXEL.

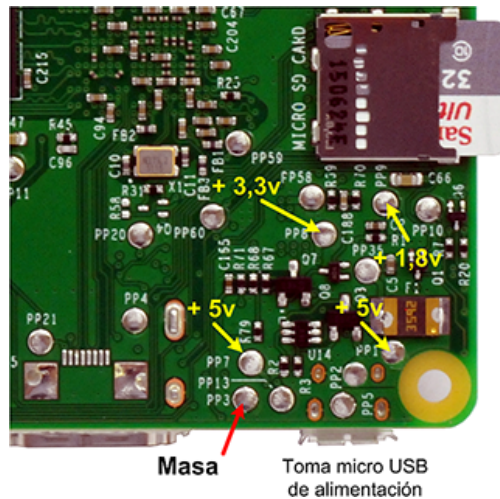


Con PIXEL aparece un termómetro en caso de temperatura excesiva. Este símbolo se superpone a la imagen normalmente mostrada en la pantalla. Por defecto, la temperatura de inicio se fija en 85°C. Es posible modificar este umbral para observar el indicador en la pantalla, añadiendo `temp_limit = 50` en el archivo `config.txt`. El símbolo que indica una temperatura demasiado elevada aparecerá a partir de 50°C.

5. Medida de la tensión en la Raspberry Pi 3

La medida de la tensión en la tarjeta de la Raspberry Pi se puede hacer en el conector GPIO de 40 puntos, o en algunos de los puntos de comprobación marcados como PPx ubicados debajo de la tarjeta.

Para controlar la tensión de 5 voltios, conecte un voltímetro entre las clavijas 2 (+ 5V) y 6 (Masa) tomando todas las precauciones necesarias para que las sondas de medición no provoquen un cortocircuito.



Hay puntos de comprobación previstos debajo de la tarjeta de la Raspberry Pi 3 (imagen anterior). Los 5V aplicados al puerto micro USB se miden en PP1 y el valor de la tensión de 5V realmente disponible para la tarjeta (después del polyfuse) en PP7. Los 3,3V son accesibles en el punto de comprobación PP8 y los 1,8V en PP9.

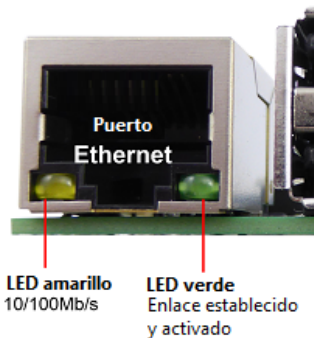
El cable de referencia (masa) del aparato de medida puede estar unido a PP3 o al blindaje (envoltorio metálico) de las tomas USB, RJ45 o al conector micro SD.

Sitúe el voltímetro en la gama de los 20 voltios. La tensión media debe estar entre 4,75 y 5,25 voltios. Una tensión cercana a 4,75 voltios puede provocar reinicios o fallos de la Raspberry Pi. Principalmente es el caso cuando se enchufa un periférico USB que consume mucha corriente, como por ejemplo una llave Wi-Fi, un teclado USB o un disco duro externo. La subida de tensión provocada por el consumo del periférico implica un reseteo del sistema. Una tensión superior a 5 voltios es garantía de estabilidad.

Si es necesario, cambie la alimentación por un modelo de potencia suficiente para alimentar la Raspberry Pi. Una alimentación conveniente para la Raspberry Pi debería entregar más de 5 voltios y ser capaz de ofrecer, al menos, 2 A sin provocar ningún colapso.

6. LED del puerto Ethernet

El puerto Ethernet de la Raspberry Pi 3 está equipado con dos LED.



El LED amarillo situado en la parte inferior izquierda en la imagen anterior indica el establecimiento de una conexión de 100 Mb/s cuando se enciende. Se indica una conexión de 10 Mb/s cuando se apaga este LED amarillo.

El LED verde de la parte derecha del conector se enciende cuando se establece el enlace físico. Parpadea cuando hay actividad en el puerto Ethernet.

El puerto Ethernet de la Raspberry Pi es Auto-MDIX. Se adapta automáticamente al hardware al que está conectado y no necesita el uso de un cable de red directo o cruzado, según el caso. Si encuentra problemas de conexión a la red, en primer lugar compruebe que el LED verde está encendido. Si no es el caso, compruebe la correcta inserción del cable en la toma Ethernet del lado de la Raspberry Pi, pero también en el otro extremo.

Si el cable está bien apretado y bloqueado por ambos lados, es el propio cable el que puede estar afectado. Intente sustituirlo por otro cable de calidad.

Si el defecto no está provocado por el cable, pregúntese por el puerto al que está conectado el otro extremo del cable (switch, router, PC, Box...).

Si no se enciende ningún LED, sospeche del cable Ethernet que ha conectado. Compruebe que está bien apretado en los dos lados y que los clips de bloqueo bloquean correctamente ambas tomas. Intente cambiar el cable de Ethernet por otro que sepa que funciona.

La conexión se hace a 10 Mb/s

Con los equipos de red actuales, la conexión se hace a 100 Mbits/s. Solo los antiguos switches o routers usan todavía una velocidad de conexión a 10 Mbit/s.

Puede comprobar la velocidad de conexión del puerto Ethernet consultando el contenido del archivo *speed*:

```
pi@raspberrypi ~ $ cat /sys/class/net/eth0/speed
100
```

Este comando confirma que la Raspberry Pi está bien conectada a 100 Mbit/s.

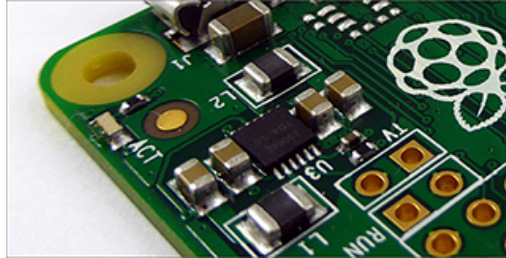
Si la velocidad indicada es 10 Mbits/s compruebe el equipamiento al que está conectada la Raspberry Pi.

El LED ACT de la Raspberry Pi Zero

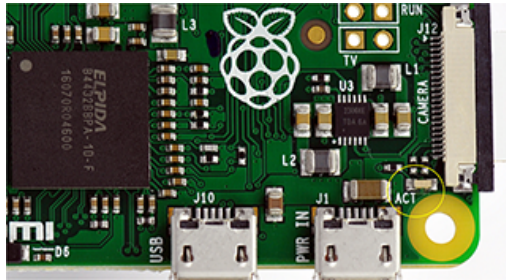
La Raspberry Pi Zero está equipada con un único LED, debido a su tamaño reducido. El LED que se ha implementado en la Raspberry Pi Zero es el LED ACT.

1. Posición del LED ACT

En función de la versión, se sitúa en un lugar u otro:



En la primera versión de la Raspberry Pi, se sitúa en el borde de la tarjeta (imagen anterior).



En la segunda versión, la presencia de un conector para la cámara ha obligado a los diseñadores a mover el LED ACT. Ya no está en el borde de tarjeta sino ligeramente desplazado respecto al ángulo del conector.

2. Estados del LED ACT de la Raspberry Pi Zero

En ausencia del LED PWR es el único led, el LED ACT, el encargado de proporcionar información acerca del estado de la Raspberry Pi Zero. Por defecto, el LED está apagado. Es el caso en particular si conecta la alimentación de la Raspberry Pi sin haber insertado antes la tarjeta micro SD.

➤ Piense siempre en desenchufar la toma micro USB de alimentación antes de insertar o retirar la tarjeta micro SD.

a. El LED de la Raspberry Pi Zero parpadea cuatro veces

Cuando conecta la toma micro USB de alimentación con una tarjeta micro SD insertada en su conector, el LED ACT se enciende. Si el archivo *start.elf* no está, el LED ACT se apaga cuatro veces, se queda iluminado un momento y se apaga de nuevo...

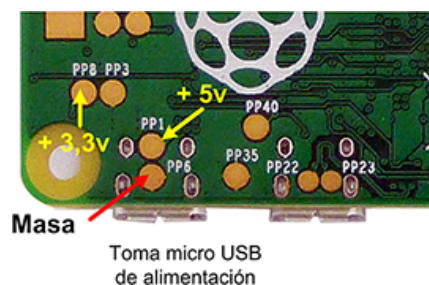
b. El LED de la Raspberry Pi Zero parpadea siete veces

Cuando conecta la toma micro USB de alimentación con una tarjeta micro SD insertada en su conector, el LED ACT se enciende. Si el archivo *kernel.img* no está o está corrupto, el LED ACT se enciende siete veces, se queda apagado un momento y se enciende de nuevo siete veces... La pantalla muestra el cuadrado coloreado que indica que la GPU ha arrancado.

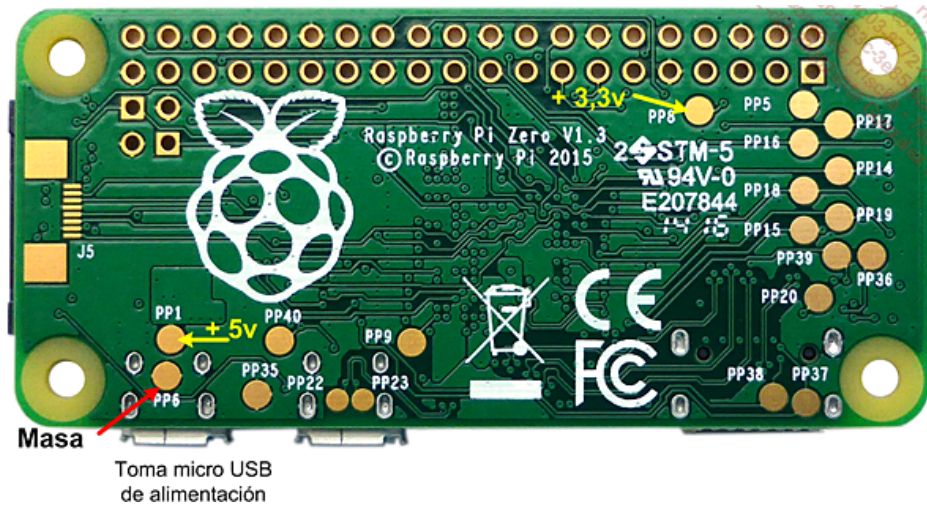
3. Medir la tensión de la Raspberry Pi Zero

La GPIO de la Raspberry Pi Zero, incluso si no está equipada del conector de 40 clavijas, es idéntica a todas las GPIO de la gama Raspberry Pi.

Para controlar la tensión de 5 voltios, conecte un voltímetro entre las clavijas 2 (+ 5V) y 6 (Masa), tomando todas las precauciones necesarias para que las sondas de medida no provoquen un corto-circuito. La medición también se puede hacer en algunos puntos probados marcados PPx, situados en la tarjeta.



En esta primera versión de la Raspberry Pi Zero, también hay puntos de comprobación previstos en la tarjeta (imagen anterior). Los 5V aplicados al puerto micro USB se miden en PP1. Los 3,3V son accesibles en el punto de comprobación PP8. El cable de referencia (masa) del aparato de medición puede estar unido a PP6 o al blindaje (envoltorio metálico) de la toma micro USB o en el conector micro SD.



En la Raspberry Pi Zero V1.3 de segunda generación, el punto de comprobación PP8 se ha movido hacia el conector GPIO.

La Raspberry Pi no arranca

Los problemas de inicio de la Raspberry Pi se relacionan principalmente con problemas de alimentación o de la tarjeta micro SD.

Observe los LED y consulte a las secciones anteriores si observa un comportamiento anormal de los LED ACT y PWR en particular. Compruebe que la tensión de alimentación es suficiente (superior a 5 voltios). Compruebe el inicio utilizando una alimentación que entregue una corriente mayor. Algunas alimentaciones baratas, previstas para la carga de smartphones, dan una corriente máxima de 1 o 2 A, pero ven su tensión de salida caer cuando entregan esta corriente.

Incluso si la Raspberry Pi arranca, muestra la pantalla de color, después el texto y arranca de nuevo a continuación, es necesario suponer que se trata de un problema de alimentación. El inicio en modo gráfico interviene después de la visualización del texto en modo consola. El modo gráfico consume más corriente de alimentación. Con ciertas alimentaciones, la puesta en marcha del modo gráfico provoca una caída de tensión de la alimentación, lo que implica el reinicio de la Raspberry Pi. Si los reinicios se suceden regularmente, en ese caso, sustituya la alimentación por un modelo más potente.

Controle la tarjeta micro SD y su soporte (tipo de tarjeta micro SD, presencia de los archivos en la tarjeta, integridad de los archivos, estado del soporte de la tarjeta micro SD, etc). Intente con otra tarjeta (otra marca, de clase 10) para despejar la duda respecto a la compatibilidad de la tarjeta micro SD. No dude en comprar una tarjeta de marca, ligeramente más cara pero normalmente de mejor calidad.

Problema de conexión a la red

Su Raspberry Pi 3 se conecta físicamente y los LED situados en la toma Ethernet se encienden. Pese a esto, la conexión a la red se pierde cuando utiliza algunos periféricos USB o las transferencias de datos en la red son importantes. Los siguientes párrafos aportan elementos de respuesta.

1. Un periférico USB deja colgada la red

La conexión de un periférico USB bloquea el acceso a la red. La causa reside habitualmente en la mala calidad de la alimentación. Una alimentación un poco justa para alimentar la Raspberry Pi se puede colapsar durante la conexión de un periférico USB. Compruebe la calidad de la alimentación como se ha explicado anteriormente (use los puntos de comprobación PPx). Intente usar una alimentación con mayor rendimiento.

Algunos periféricos USB consumen mucha corriente (más de 100 mA) y se deben conectar a un hub USB alimentado. Sin embargo, algunos hubs USB de mala calidad consumen corriente en el puerto USB de la Raspberry Pi, incluso cuando se conectan a una alimentación principal. Compruebe la compatibilidad del hub con la Raspberry Pi (http://elinux.org/RPi_Powered_USB_Hubs).

Algunos periféricos USB (ratón y teclado) conformes a la norma USB 1 provocan bloqueos del puerto Ethernet cuando se conectan a un hub USB alimentado. Si consumen menos de 100 mA, intente conectarlos directamente a los puertos USB de la Raspberry Pi o sustitúyalos por modelos más recientes conformes a la norma USB 2.

2. Mover la tarjeta micro SD bloquea la red

Cuando un usuario tiene varias Raspberry Pi, resulta práctico tomar la tarjeta micro SD de una para usarla en otra.

Algunas distribuciones memorizan la dirección MAC (dirección de hardware) de la Raspberry Pi en la que se instalan. En este caso existe un archivo `/etc/udev/rules.d/70-persistent-net.rules` en el que el sistema operativo memoriza la dirección MAC asociada a `eth0`. Los siguientes dispositivos de red, cuando se detecten, se llamarán `eth1`, `eth2`... porque tienen direcciones MAC diferentes.

Este problema se puede resolver editando el archivo `/etc/udev/rules.d/70-persistent-net.rules` y eliminando los ajustes inapropiados, por ejemplo:

```
ACTION=="add", SUBSYSTEM=="net",
SYSFS{address}=="00:e0:4c:12:34:56", NAME="realtek"
```

Durante el siguiente inicio, el sistema reconocerá el dispositivo con normalidad.

3. Bloqueo de la red bajo una fuerte carga

El núcleo reserva memoria para el control USB (que también gestiona el puerto Ethernet). Sin embargo, durante un tráfico importante en el puerto Ethernet (descarga, uso de torrent...) la memoria se puede saturar, provocando bloqueos. Compruebe que el archivo `/etc/sysctl.conf` contiene la línea:

```
vm.min_free_kbytes = 8192
```

Esta línea indica al núcleo que debe conservar permanentemente 8 KB de RAM disponibles en todo momento. En primer lugar, intente disminuir este valor a 1634 (edite el archivo con el comando `sudo nano /etc/sysctl.conf`).

Si esta modificación no es eficaz, añada las siguientes líneas en `/boot/cmdline.txt` con `nano`:

```
smsc95xx.turbo_mode=N
dwc_otg.dma_enable=1
dwc_otg.dma_burst_size=256
```

La primera línea indica que la conexión Ethernet no debe usar el modo turbo. Esto no provoca generalmente cambios visibles en el rendimiento.

Las dos líneas siguientes especifican el uso de la DMA (*Direct Memory Access* = acceso directo a la memoria) para el acceso a los archivos de espera FIFO (*First In First Out* = primero en entrar, primero en salir) en los que se almacenan los datos que llegan del puerto Ethernet. La última línea fija el tamaño de los bloques de datos transferidos a 256 bytes (por defecto vale 32 bytes).

Si estas modificaciones no tienen efecto para prevenir la ralentización o los bloqueos, intente limitar el ancho de banda con `wondershaper` (disponible para Raspbian), en particular la velocidad de subida:

```
pi@raspberrypi ~ $ sudo apt-get install wondershaper
Lectura de las listas de paquetes... Terminado
Construcción del árbol de dependencias
Lectura de la información de estado... Terminado
Se instalarán los siguientes paquetes nuevos:
wondershaper

... / ...

Configuración de wondershaper (1.1a-6) ...
```

Sintaxis

```
sudo wondershaper <nombre_interfaz_red> <velocidad_descarga>
<velocidad_subida>
```

Las velocidades de descarga y de subida se expresan en Kbits/s.

```
pi@raspberrypi ~ $ sudo wondershaper eth0 10000 1500
```

Los valores de descarga (10 000 Kbits/s) y de subida (1 500 Kbits/s) seleccionados en este ejemplo fijan el límite de débito de la tarjeta `eth0`. Si constata un funcionamiento lento, intente reducir progresivamente el valor de cada uno de estos argumentos.

Para finalizar la acción de `wondershaper`, utilice el siguiente comando:

```
sudo wondershaper clear <nombre_interfaz_red>
```

```
pi@raspberrypi ~ $ sudo wondershaper clear eth0
Wondershaper queues have been cleared.
```

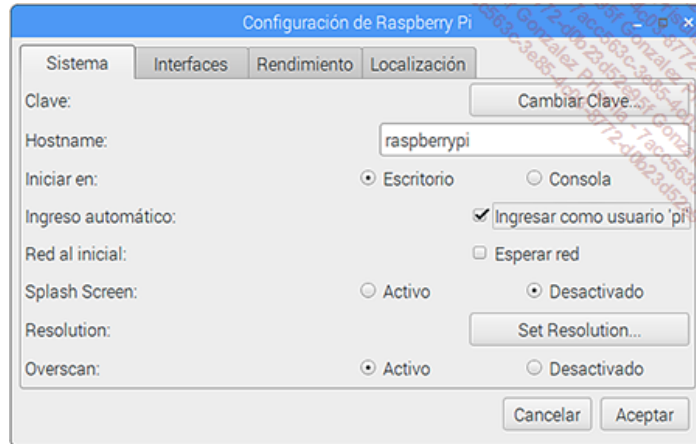
Visualización anormal en la pantalla

Por defecto, la Raspberry Pi se configura con el modo overscan, lo que provoca en general la aparición de franjas negras antiestéticas alrededor de la imagen. Una alimentación un poco justa o un cable HDMI de calidad mediocre también puede comprometer la calidad de la imagen proporcionada por la Raspberry Pi.

1. Imagen rodeada de bordes negros

Por defecto, la imagen proporcionada por la Raspberry Pi está rodeada de bordes negros. Al principio, estos bordes estaban destinados a dejar tiempo a los televisores con tubo catódico para hacer volver el spot luminoso al borde de la pantalla o hasta la parte superior de la imagen. Con las pantallas digitales actuales esta precaución es inútil.

En modo gráfico, a partir del **Menú** vaya a **Preferencias** y después a **Configuración de Raspberry Pi**:



En la ventana que aparece (imagen anterior), pulse en **Desactivado** en la línea **Overscan**. La modificación se tendrá en cuenta después del reinicio de la Raspberry Pi.

En modo texto, durante la configuración inicial en el menú de *raspi-config* seleccione **Advanced Options** y **Overscan**. La ventana que se abre le pide usar el *overscan*. Seleccione **<Desactivado>** y después arranque de nuevo la Raspberry Pi. Las franjas negras habrán desaparecido.

El ajuste de este argumento también puede realizarse en el archivo */boot/config.txt*. Edite este archivo con *sudo nano* y busque la línea que contiene *disable_overscan*. Ponga el valor a 1:

```
disable_overscan=1
```

Si la línea no existe, agréguela la al final del archivo. Reinicie la Raspberry Pi para tener en cuenta esta modificación.

2. Imagen que se desborda de la pantalla

Es posible ajustar con precisión el tamaño de la imagen si esta es más pequeña que la superficie de la pantalla o, por el contrario, si desborda la pantalla.

En el archivo */boot/config.txt*, copie las siguientes líneas:

```
# uncomment the following to adjust overscan. Use positive
#numbers if console goes off screen, and negative
#if there is too much border
#overscan_left=16
#overscan_right=16
#overscan_top=16
#overscan_bottom=16
```

Elimine los signos # delante de las líneas que empiezan por *overscan_* y sustituya el valor 16 por un valor conveniente para su configuración. Ajuste de manera aproximada el *overscan* modificando el número de píxeles a 10 para afinar el ajuste y después afine para ajustar la imagen al tamaño exacto de su pantalla. Si quedan bordes con un valor ajustado a cero, indique valores negativos para reducir todavía más la longitud de los bordes negros. Los cuatro valores no tienen por qué ser necesariamente idénticos. Ajuste cada valor para que la imagen ocupe la totalidad de la pantalla.

Anote estos valores de *overscan*, servirán para configurar la Raspberry Pi la próxima vez que instale un sistema operativo.

3. Puntos de colores e interferencias

Un cable HDMI demasiado largo o de calidad insuficiente puede provocar pérdidas de señal entre la Raspberry Pi y la pantalla. En general, la pérdida de señal se traduce en la desaparición total de la imagen. Ocasionalmente, las perturbaciones provocadas por estas pérdidas aparecen en forma de puntos blancos o de colores en la imagen.

Intente sustituir el cable HDMI por otro de mejor calidad. Los cables HDMI más caros no funcionan necesariamente mejor que el resto.

La potencia de la señal enviada por la Raspberry Pi al bus HDMI puede ajustarse en */boot/config.txt*.

```
# uncomment to increase signal to HDMI, if you have interference,
# blanking, or no display
#config_hdmi_boost=4
```

Elimine la # al comienzo de la línea *config_hdmi_boost=4* que fija el nivel de salida en el puerto HDMI. El valor 1 es adecuado para cables muy cortos y el valor 7 para cables HDMI largos o durante el uso de adaptadores HDMI a DVI, que pueden atenuar la señal. Permanezca entre 1 y 7 si es posible. El valor máximo es 11, pero no es adecuado para algunos monitores cuya entrada HDMI se puede deteriorar por señales fuertes.

Una alimentación de 5 voltios demasiado baja también puede crear este defecto. Compruebe la calidad de la alimentación (use los puntos de comprobación PPx). Intente usar una alimentación con mejor rendimiento.

4. No hay visualización en la pantalla

La ausencia total de visualización en la pantalla puede ser el resultado de un arranque de la Raspberry Pi con la toma HDMI desconectada o el

monitor desenchufado. La Raspberry Pi comprueba el puerto HDMI al inicio y activa la salida de vídeo compuesta si el puerto HDMI no responde. Conecte el cable HDMI a la Raspberry Pi y al monitor. Enchufe el monitor y, si es necesario, seleccione la entrada HDMI en el menú. Arranque a continuación la Raspberry Pi. La imagen debería aparecer en la pantalla.

Forzar el modo HDMI

Si el monitor HDMI no muestra ninguna imagen cuando arranca, el inicio se ha realizado con el cable HDMI en su lugar y el monitor está enchufado, es necesario forzar el inicio en HDMI.

En el archivo `/boot/config.txt`, añada:

```
hdmi_safe=1
```

La Raspberry Pi arranca con la compatibilidad máxima en HDMI y el monitor debería mostrar una imagen.

5. La resolución no es la adecuada

Si la resolución de su pantalla no es satisfactoria, puede forzar la resolución de su elección en el archivo `/boot/config.txt`. Seleccione el grupo (1 para un televisor, 2 para un monitor) y el modo de visualización. Aquí `hdmi_mode=16` fuerza la resolución a 1024 x 768 píxeles (consulte el capítulo Arrancar Raspbian):

```
# uncomment to force a specific HDMI mode (this will force VGA)
# Fuerza el vídeo en 1024x768 px
hdmi_group=2
hdmi_mode=16
```

Durante el próximo reinicio de la Raspberry Pi, la resolución se adaptará a sus ajustes.

El sonido no funciona

1. Sin sonido en un monitor HDMI

Algunos monitores dotados de una toma de vídeo HDMI conmutan la entrada en modo DVI, incluso cuando la entrada HDMI está conectada a la Raspberry Pi. Otros aparatos HDMI pueden funcionar perfectamente con este monitor HDMI.

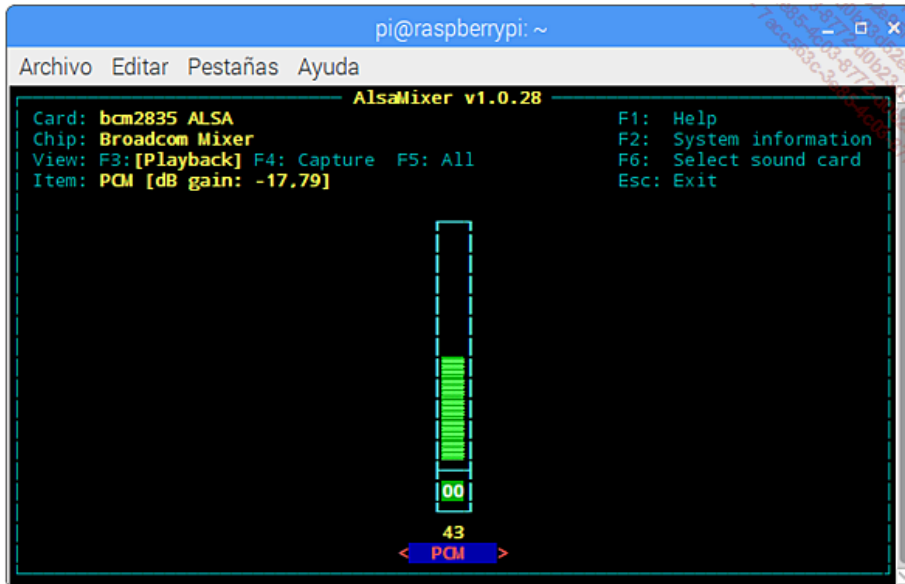
Edite el archivo `/boot/config.txt` y active la siguiente línea, eliminando la `#` que la precede:

```
hdmi_drive=2
```

Esta línea fuerza la elección del modo HDMI.

2. Sin sonido completamente

En modo texto, abra `alsamixer`. Use el cursor (flecha hacia arriba) para aumentar el volumen. Compruebe que el mute no esté activado (tecla M).



Con Raspbian, el módulo `snd_bcm2835` se carga por defecto. El comando `aplay` permite probar la salida de sonido analógico, si no se ha detectado un monitor que acepte el sonido HDMI:

```
pi@raspberrypi ~ $ aplay /usr/share/sounds/alsa/Front_Center.wav
Lectura WAVE '/usr/share/sounds/alsa/Front_Center.wav': Signed
16 bit Little Endian, Frecuencia 48000 Hz, Mono
```

Una voz femenina dice "Front center". Por defecto, la elección de la salida utilizada para el sonido es automática: HDMI si el monitor HDMI soporta el audio, analógica en el resto de casos. Puede forzar la salida analógica utilizada por la Raspberry Pi:

```
pi@raspberrypi ~ $ amixer cset numid=3 <n>
```

`<n>` puede recibir los valores:

- 0 para la selección automática.
- 1 para la salida de sonido analógica.
- 2 para la salida de sonido HDMI.

Por ejemplo, para forzar la salida de sonido por la toma jack (analógico), utilice el comando: `amixer cset numid=3 1`

Las versiones recientes de `raspi-config` permiten configurar la salida de sonido por la salida analógica o hacia la salida HDMI (opción 9, opción A8).

El teclado es QWERTY

A pesar de la elección del teclado realizada en el menú gráfico de configuración o en *raspi-config*, sucede que el teclado no está configurado correctamente en modo gráfico o en modo consola. Este funcionamiento incorrecto ya no debería suceder en las versiones más recientes de los sistemas operativos de la Raspberry Pi.

1. Teclados AZERTY y QWERTY

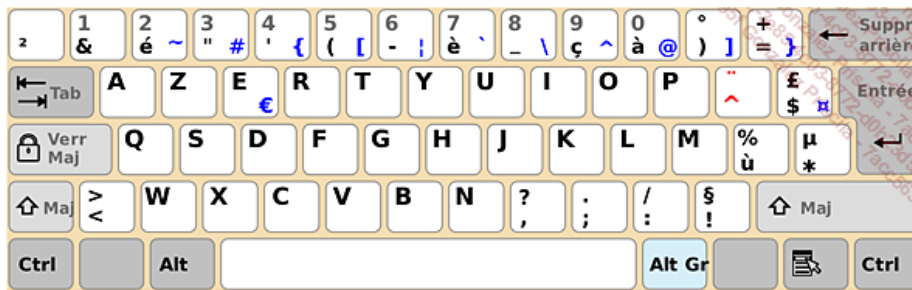
El teclado QWERTY es el utilizado normalmente en ESPAÑA.

Todos los teclados, YA SEAN AZERTY, QWERTY (países angloparlantes, España, norte de Europa, etc.) o QWERTZ (países germano parlantes y de Europa Central), no envían el código ASCII correspondiente al carácter que figura en la tecla que se pulsa. Envían un código correspondiente a la posición de la tecla en una matriz líneas/columnas. Cada pulsación en una tecla envía el código del teclado correspondiente a la tecla. Al soltar la tecla, se envía el mismo código al que el micro controlador del teclado añade 128 (el bit más significativo del byte se pone a 1).

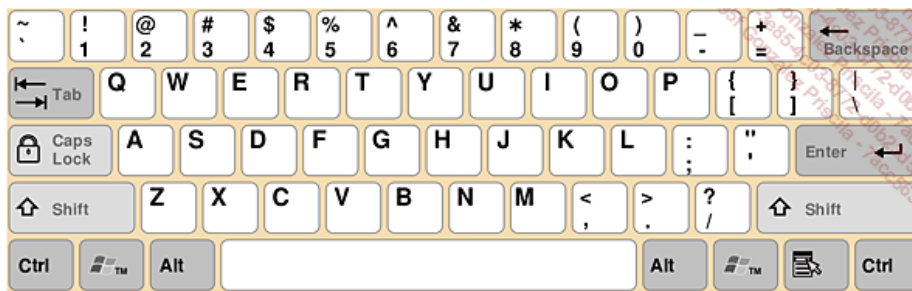
Al pulsar la tecla A de un teclado AZERTY o Q de un teclado QWERTY se produce el envío del código binario 00010101 (0x15 en hexadecimal). Es a nivel del sistema operativo donde el código se traduce en código ASCII, en función de los argumentos seleccionados por el usuario.

Por lo tanto, es a nivel del sistema donde es necesario intervenir para corregir la traducción incorrecta de los códigos de teclado en códigos ASCII.

Las siguientes disposiciones de teclados le permitirán localizar, si es necesario, las ubicaciones de las teclas que debe pulsar si su teclado está mal configurado (las imágenes de los siguientes teclados tienen licencia Creative Commons: Asignación - Compartición en las mismas condiciones 3.0 España - Autor: Michka B).



Disposición de las teclas del teclado AZERTY



Disposición de las teclas del teclado QWERTY

En modo gráfico, vaya al **Menú - Preferencias - Configuración de Raspberry Pi**.

En modo texto, inicie *raspi-config* y configure el código de letra a *es_ES.UTF-8*. En las opciones de internacionalización (**Internationalization Options**) seleccione **I3 Change keyboard Layout**. Seleccione a continuación **Teclado genérico PC 105 teclas y Español - variante Latin 9**. Reinicie la Raspberry Pi.

Normalmente, en las consolas accesibles por [Alt][F1], [Alt][F2]... el teclado está en QWERTY. El teclado QWERTY es el utilizado normalmente en España.

Por el contrario, es posible que en un terminal abierto en modo gráfico, el teclado esté incluso en AZERTY.

En este caso, es necesario informar al servidor X del tipo de teclado utilizado:

```
pi@raspberrypi ~ $ sudo setxkbmap es
```

Esta modificación solo es válida para la sesión actual. El defecto va a reaparecer después de reiniciar la Raspberry Pi.

Para guardar de manera definitiva esta modificación, abra con *nano* el archivo oculto *.bashrc*, que se encuentra en su directorio personal (en principio */home/pi*). Agregue una línea que contenga `sudo setxkbmap es` al final del archivo. Esta línea se ejecutará cada vez que se inicie una sesión y fuerza el teclado español.

Monitorización de la Raspberry Pi

Puede ser interesante conocer, en un momento dado, el estado general de la Raspberry Pi. Hay muchas herramientas de supervisión gráfica que presentan los datos en páginas HTML. Esta información, mostrada en forma de curvas, permite monitorizar la evolución del sistema a lo largo del tiempo.

Para un uso básico, una herramienta de monitorización en modo texto, poco consumidora de recursos, normalmente da las primera indicaciones en caso de funcionamiento incorrecto. EZ Server Monitor es una de ellas. Es un script fácil de instalar y ejecutar en modo remoto para obtener rápidamente información acerca del estado de un sistema. También hay una versión web de esta herramienta.

1. Instalación de EZ Server Monitor

En primer lugar, hay que descargar el script con el comando `wget` desde el sitio web de origen, <http://www.ezservermonitor.com>:

```
pi@raspberrypi:~ $ wget --content-disposition
http://www.ezservermonitor.com/esm-sh/downloads/version/2.2
--2016-08-16 15:39:13--
http://www.ezservermonitor.com/esm-sh/downloads/version/2.2
Resolución de www.ezservermonitor.com (www.ezservermonitor.com)...
213.186.33.17
Conexión a www.ezservermonitor.com
(www.ezservermonitor.com)|213.186.33.17|:80... conectado.
petición HTTP enviada, esperando respuesta... 200 OK
Tamaño: 3993 (3,9K) [application/zip]
Copiado en: « ezservermonitor-sh_v2.2.zip »

ezservermonitor-sh_ 100%[=====>] 3,90K --.-KB/s
ds 0,006s

2016-08-16 15:39:14 (600 KB/s) - «ezservermonitor-sh_v2.2.zip»
copiado [3993/3993]
```

El archivo `ezservermonitor-sh_v2.2.zip` se guarda en el directorio actual.

Descomprima el archivo para extraer el script con el comando `unzip`:

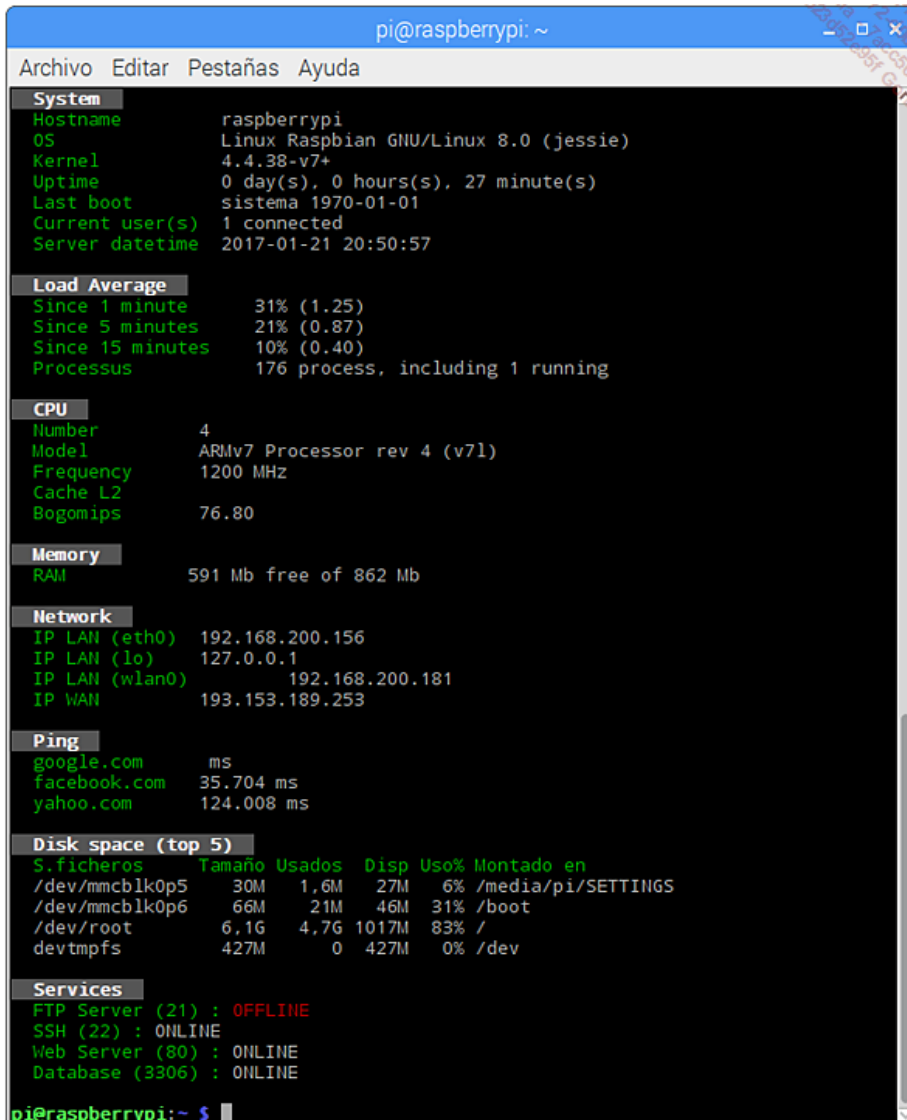
```
pi@raspberrypi:~ $ unzip ezservermonitor-sh_v2.2.zip
Archive:  ezservermonitor-sh_v2.2.zip
  inflating: eZServerMonitor.sh
```

Convierta el script en ejecutable (respete las mayúsculas y minúsculas):

```
pi@raspberrypi:~ $ chmod u+x eZServerMonitor.sh
```

Ejecute el script en local o en remoto con la opción `-a` para mostrar toda la información (respete las mayúsculas y minúsculas):

```
pi@raspberrypi:~ $ ./eZServerMonitor.sh -a
```



Se agrupa en una única pantalla la información del sistema.

Conclusión

Intervenir en la Raspberry Pi en caso de problemas hace necesario un buen conocimiento de los principios de funcionamiento de un sistema informático. Incluso si parece sencillo reducir el tamaño de la Raspberry Pi, hay que tener en cuenta que es un sistema informático completo.

El uso de una herramienta como EZ Server Monitor ofrece las primeras indicaciones que pueden ayudar al usuario a localizar la causa de un funcionamiento incorrecto.

Sea cual sea, la solución de problemas de la Raspberry Pi solo se puede hacer siendo riguroso. Cada hipótesis de error se debe comprobar individualmente.

Con su experiencia sobre los distribuidores de la Raspberry Pi, la mayoría de los errores se deben a una tarjeta micro SD defectuosa o corrupta y a un defecto en la alimentación (bloqueo sector o cable). Empiece siempre haciendo estas dos comprobaciones.

Respuestas al ejercicio del capítulo Arrancar sobre un disco externo

Monte manualmente la segunda partición en una carpeta `/mnt/data`.

```
pi@raspberrypi ~ $ cd /
pi@raspberrypi / $ sudo mkdir data
pi@raspberrypi / $ sudo mount /dev/sda2 /data
```

Modifique `/etc/fstab` para que la partición `sda2` esté conectada automáticamente al inicio del sistema.

```
proc /proc proc defaults 0 0
/dev/mmcblk0p1 /boot vfat defaults 0 2
#/dev/mmcblk0p2 / ext4 defaults,noatime 0 1
/dev/sda1 / ext4 defaults,noatime 0 1
/dev/sda2 /data vfat defaults,noatime 0 2
```

Lo que da:

```
pi@raspberrypi ~ $ df
Sis.arch. 1K-blocks Util. Disponible Uti% Montada en
rootfs 309637120 1568392 295471036 1% /
/dev/root 309637120 1568392 295471036 1% /
devtmpfs 183620 0 183620 0% /dev
tmpfs 38376 260 38116 1% /run
tmpfs 5120 0 5120 0% /run/lock
tmpfs 76740 0 76740 0% /run/shm
/dev/mmcblk0p1 57288 34768 22520 61% /boot
/dev/sda2 661865536 16 661865520 1% /data
```

Respuestas a los ejercicios del capítulo Programar en Python

Parta del programa *multiplicacion.py* y modifíquelo para usar un bucle `for` en lugar del `while`. Haga una copia de seguridad con el nombre *multiplicacion2.py*.

```
# Tabla de multiplicación por 4

valor = 1 # Variable utilizada para contar
print("Tabla de multiplicación por 4")
for valor in range(1,11):
    # Al inicio del bloque a ejecutar
    # Observe la indentación
    print(valor, " x 4 = ",valor*4)
print("Fin de la tabla de multiplicación")
```

Modifique el programa *multiplicacion2.py* para que el usuario pueda introducir por teclado la tabla de multiplicación que desea mostrar. Haga una copia de seguridad con el nombre *multiplicacion2a.py*.

```
# Tabla de multiplicación por N
# La instrucción input() muestra en pantalla el mensaje que se pasa
# como argumento y devuelve la cadena de caracteres escrita por
# teclado. La instrucción int() transforma la cadena
# de caracteres en un número entero
N = int(input("¿Qué tabla desea mostrar? "))

# La variable llamada contador se usa para contar
# el número de bucles realizados
# Se inicializa a 1, que es su valor de inicio.
contador = 1
# La instrucción print() muestra el texto pasado como argumento,
# así como el valor de la variable N
print("Tabla de multiplicación por ", N)
# El bucle for va a ejecutar el bloque que sigue a la instrucción for
# para el rango de valores proporcionado en la directiva range()
# El primer valor será 1
# El último valor será 10 (11-1)
# En ausencia del tercer argumento, el valor de
# la variable contador se incrementa (aumenta en 1)
# con cada paso del bucle
for contador in range(1,11):
    # Al inicio del bloque a ejecutar
    # Observe la indentación
    # La siguiente instrucción realiza la multiplicación
    # y guarda el resultado en producto
    producto = contador * N
    # Muestra en pantalla una línea
    # de la tabla de multiplicación
    print(contador, " x ",N," = ",producto)
# El print() final informa al usuario del final del programa
print("Final de la tabla de multiplicación")
```

Ejecute el programa con Python:

```
pi@raspberrypi ~ $ python3 multiplicacion2a.py
```

Respuestas al ejercicio del capítulo La GPIO de la Raspberry Pi

Utilizando los comandos anteriores, escriba un script que haga parpadear un LED conectado al puerto GPIO 24.

Abra el editor de texto nano:

```
pi@raspberrypi ~ $ nano led_parpadea.sh
```

Escriba el script destinado a hacer parpadear un LED conectado al puerto GPIO 24.

```
#!/bin/bash
# Script led_parpadea.sh
# Parpadeo de un LED en línea de comandos
# Este script no tiene prueba destinada a la interrupción
# Se detiene pulsando CTRL C

# Haga que el directorio /sys/class/gpio esté activo
cd /sys/class/gpio

# Cree el acceso al puerto GPIO 24
echo 24 > export

# Haga el directorio gpio24 activo
cd gpio24/

# Configure el puerto GPIO 24 en salida
echo out > direction

# Inicializar la variable x a 1
x="1"

# Bucle while: este bucle se ejecuta mientras
# se cumple la condición
# aquí la condición siempre es verdadera:
# el bucle nunca se interrumpe
while[ $x -gt 0 ]

# Se ejecuta el bloque incluido entre do y done
# por el bucle while

# Al inicio del bloque de instrucciones
do
  # La instrucción echo envía el valor 1 al archivo value
  # Esto tiene como efecto encender el LED
  echo 1 > value

  # Un mensaje en pantalla informa al usuario
  # del encendido del LED
  echo "LED encendido"

  # El comando sleep suspende la ejecución del programa
  # durante la duración especificada (en segundos)
  sleep 2

  # La instrucción echo envía el valor 0 al archivo value
  # Esto tiene como efecto apagar el LED
  echo 0 > value

  # Un mensaje en pantalla informa al usuario
  # del apagado del LED
  echo "LED apagado"

  # El comando sleep suspende la ejecución del programa
  # durante la duración especificada (en segundos)
  sleep 2

# Fin del bloque de instrucciones
done
```

Haga el script ejecutable con el comando `chmod`:

```
pi@raspberrypi ~ $ sudo chmod +x led_parpadea.sh
```

Ejecute el script. El uso de `sudo` es obligatorio para acceder a la GPIO.

```
pi@raspberrypi ~ $ sudo ./led_parpadea.sh
```

El script ejecuta un bucle infinito. No está previsto interrumpirlo pulsando en una tecla del teclado. Para detener la ejecución de `led_parpadea.sh` pulse la secuencia de teclas [Ctrl] C.

Puede modificar la velocidad de parpadeo cambiando el valor que sigue a la instrucción `sleep`.

Sonic Pi

```
Frère Jacques
2.times do
  play 60
  sleep 0.5
  play 62
  sleep 0.5
  play 64
  sleep 0.5
  play 60
  sleep 0.5
end
2.times do
  play 64
  sleep 0.5
  play 65
  sleep 0.5
  play 67
  sleep 1
end
2.times do
  play 67
  sleep 0.25
  play 69
  sleep 0.25
  play 67
  sleep 0.25
  play 65
  sleep 0.25
  play 64
  sleep 0.5
  play 60
  sleep 0.5
end
2.times do
  play 60
  sleep 0.5
  play 55
  sleep 0.5
  play 60
  sleep 1
end
```

Tabla de configuración del vídeo

Valores hdmi-mode CEA - Televisión

Esta tabla ofrece los valores de hdmi-mode que permiten configurar una Raspberry Pi conectada a un televisor. Los modos que utilizan el formato 16:9 se detectan por la letra H en la columna Formato.

hdmi_mode: valores utilizables en modo CEA: hdmi_group=1				
Nº del modo	Modo	Frecuencia de imagen	Agrupación de píxeles	Formato
hdmi_mode=1	VGA			
hdmi_mode=2	480p	60 Hz		
hdmi_mode=3	480p	60 Hz		H
hdmi_mode=4	720p	60 Hz		
hdmi_mode=5	1080i	60 Hz		
hdmi_mode=6	480i	60 Hz		
hdmi_mode=7	480i	60 Hz		H
hdmi_mode=8	240p	60 Hz		
hdmi_mode=9	240p	60 Hz		H
hdmi_mode=10	480i	60 Hz	4x	
hdmi_mode=11	480i	60 Hz	4x	H
hdmi_mode=12	240p	60 Hz	4x	
hdmi_mode=13	240p	60 Hz	4x	H
hdmi_mode=14	480p	60 Hz	2x	
hdmi_mode=15	480p	60 Hz	2x	H
hdmi_mode=16	1080p	60 Hz		
hdmi_mode=17	576p	50 Hz		
hdmi_mode=18	576p	50 Hz		H
hdmi_mode=19	720p	50 Hz		
hdmi_mode=20	1080i	50 Hz		
hdmi_mode=21	576i	50 Hz		
hdmi_mode=22	576i	50 Hz		H
hdmi_mode=23	288p	50 Hz		
hdmi_mode=24	288p	50 Hz		H
hdmi_mode=25	576i	50 Hz	4x	
hdmi_mode=26	576i	50 Hz	4x	H
hdmi_mode=27	288p	50 Hz	4x	
hdmi_mode=28	288p	50 Hz	4x	H
hdmi_mode=29	576p	50 Hz	2x	
hdmi_mode=30	576p	50 Hz	2x	H
hdmi_mode=31	1080p	50 Hz		
hdmi_mode=32	1080p	24 Hz		
hdmi_mode=33	1080p	25 Hz		
hdmi_mode=34	1080p	30 Hz		
hdmi_mode=35	480p	60 Hz	4x	
hdmi_mode=36	480p	60 Hz	4x	H
hdmi_mode=37	576p	50 Hz	4x	
hdmi_mode=38	576p	50 Hz	4x	H
hdmi_mode=39	1080i	50 Hz	Borrado reducido	
hdmi_mode=40	1080i	100 Hz		
hdmi_mode=41	720p	100 Hz		
hdmi_mode=42	576p	100 Hz		
hdmi_mode=43	576p	100 Hz		H
hdmi_mode=44	576i	100 Hz		
hdmi_mode=45	576i	100 Hz		H
hdmi_mode=46	1080i	120 Hz		
hdmi_mode=47	720p	120 Hz		
hdmi_mode=48	480p	120 Hz		
hdmi_mode=49	480p	120 Hz		H
hdmi_mode=50	480i	120 Hz		
hdmi_mode=51	480i	120 Hz		H
hdmi_mode=52	576p	200 Hz		
hdmi_mode=53	576p	200 Hz		H
hdmi_mode=54	576i	200 Hz		
hdmi_mode=55	576i	200 Hz		H

hdmi_mode=56	480p	240 Hz		
hdmi_mode=57	480p	240 Hz		H
hdmi_mode=58	480i	240 Hz		
hdmi_mode=59	480i	240 Hz		H

La letra H indica un formato de imagen de 16:9, variación de un formato normalmente en 4:3.

2x indica multiplicar por dos los píxeles: cada píxel se repite dos veces.

4x indica multiplicar por cuatro los píxeles: cada píxel se repite cuatro veces.

Los modos que utilizan el doble y el cuádruple de la imagen aumentan artificialmente el número de píxeles. Provocan una degradación de la imagen.

En las pantallas con tubo catódico, la eliminación corresponde al retroceso del haz de electrones después del escaneo de una línea o de una imagen. En las pantallas digitales actuales, el tiempo de retroceso no existe. Por tanto, la duración correspondiente a la eliminación se puede reducir.

Valores hdmi-mode DMT - Monitores informáticos

Esta tabla ofrece los valores de hdmi-mode que permiten configurar una Raspberry Pi conectada a un monitor informático. Para algunos modos, la frecuencia de imagen no está disponible. En la Raspberry Pi, el modo más elevado que se acepta corresponde a 1920 x 1200 píxeles @ 60 Hz, con eliminación reducida. Esta limitación se impone por la frecuencia de píxeles seleccionada.

hdmi_mode: valores utilizables en modo DMT: hdmi_group=2				
Nº del modo	Resolución	Frecuencia de imagen		
hdmi_mode=1	640x350	85 Hz		
hdmi_mode=2	640x400	85 Hz		
hdmi_mode=3	720x400	85 Hz		
hdmi_mode=4	640x480	60 Hz		
hdmi_mode=5	640x480	72 Hz		
hdmi_mode=6	640x480	75 Hz		
hdmi_mode=7	640x480	85 Hz		
hdmi_mode=8	800x600	56 Hz		
hdmi_mode=9	800x600	60 Hz		
hdmi_mode=10	800x600	72 Hz		
hdmi_mode=11	800x600	75 Hz		
hdmi_mode=12	800x600	85 Hz		
hdmi_mode=13	800x600	120 Hz		
hdmi_mode=14	848x480	60 Hz		
hdmi_mode=15	1024x768	43 Hz	No usar, incompatible con Raspberry Pi	
hdmi_mode=16	1024x768	60 Hz		
hdmi_mode=17	1024x768	70 Hz		
hdmi_mode=18	1024x768	75 Hz		
hdmi_mode=19	1024x768	85 Hz		
hdmi_mode=20	1024x768	120 Hz		
hdmi_mode=21	1152x864	75 Hz		
hdmi_mode=22	1280x768		Borrado reducido	
hdmi_mode=23	1280x768	60 Hz		
hdmi_mode=24	1280x768	75 Hz		
hdmi_mode=25	1280x768	85 Hz		
hdmi_mode=26	1280x768	120 Hz		
hdmi_mode=27	1280x800		Borrado reducido	
hdmi_mode=28	1280x800	60 Hz		
hdmi_mode=29	1280x800	75 Hz		
hdmi_mode=30	1280x800	85 Hz		
hdmi_mode=31	1280x800	120 Hz	Borrado reducido	
hdmi_mode=32	1280x960	60 Hz		
hdmi_mode=33	1280x960	85 Hz		
hdmi_mode=34	1280x960	120 Hz	Borrado reducido	
hdmi_mode=35	1280x1024	60 Hz		
hdmi_mode=36	1280x1024	75 Hz		
hdmi_mode=37	1280x1024	85 Hz		
hdmi_mode=38	1280x1024	120 Hz	Borrado reducido	
hdmi_mode=39	1360x768	60 Hz		
hdmi_mode=40	1360x768	120 Hz	Borrado reducido	
hdmi_mode=41	1400x1050		Borrado reducido	
hdmi_mode=42	1400x1050	60 Hz		
hdmi_mode=43	1400x1050	75 Hz		
hdmi_mode=44	1400x1050	85 Hz		

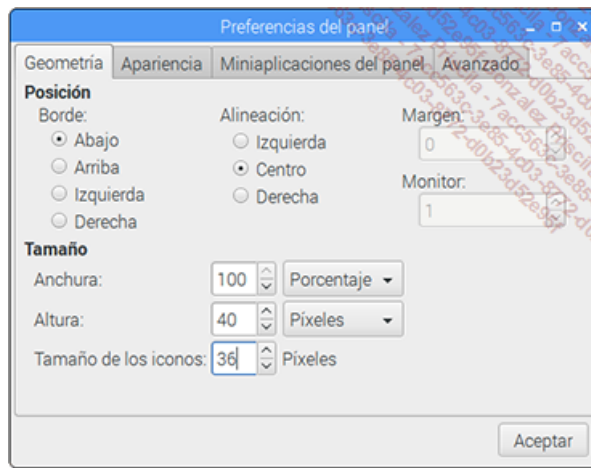
hdmi_mode=45	1400x1050	120 Hz	Borrado reducido	
hdmi_mode=46	1440x900		Borrado reducido	
hdmi_mode=47	1440x900	60 Hz		
hdmi_mode=48	1440x900	75 Hz		
hdmi_mode=49	1440x900	85 Hz		
hdmi_mode=50	1440x900	120 Hz	Borrado reducido	
hdmi_mode=51	1600x1200	60 Hz		
hdmi_mode=52	1600x1200	65 Hz		
hdmi_mode=53	1600x1200	70 Hz		
hdmi_mode=54	1600x1200	75 Hz		
hdmi_mode=55	1600x1200	85 Hz		
hdmi_mode=56	1600x1200	120 Hz	Borrado reducido	
hdmi_mode=57	1680x1050		Borrado reducido	
hdmi_mode=58	1680x1050	60 Hz		
hdmi_mode=59	1680x1050	75 Hz		
hdmi_mode=60	1680x1050	85 Hz		
hdmi_mode=61	1680x1050	120 Hz	Borrado reducido	
hdmi_mode=62	1792x1344	60 Hz		
hdmi_mode=63	1792x1344	75 Hz		
hdmi_mode=64	1792x1344	120 Hz	Borrado reducido	
hdmi_mode=65	1856x1392	60 Hz		
hdmi_mode=66	1856x1392	75 Hz		
hdmi_mode=67	1856x1392	120 Hz	Borrado reducido	
hdmi_mode=68	1920x1200		Borrado reducido	
hdmi_mode=69	1920x1200	60 Hz		
hdmi_mode=70	1920x1200	75 Hz		
hdmi_mode=71	1920x1200	85 Hz		
hdmi_mode=72	1920x1200	120 Hz	Borrado reducido	
hdmi_mode=73	1920x1440	60 Hz		
hdmi_mode=74	1920x1440	75 Hz		
hdmi_mode=75	1920x1440	120 Hz	Borrado reducido	
hdmi_mode=76	2560x1600		Borrado reducido	
hdmi_mode=77	2560x1600	60 Hz		
hdmi_mode=78	2560x1600	75 Hz		
hdmi_mode=79	2560x1600	85 Hz		
hdmi_mode=80	2560x1600	120 Hz	Borrado reducido	
hdmi_mode=81	1366x768	60 Hz		
hdmi_mode=82	1080p	60 Hz		
hdmi_mode=83	1600x900		Borrado reducido	
hdmi_mode=84	2048x1152		Borrado reducido	
hdmi_mode=85	720p	60 Hz		
hdmi_mode=86	1366x768		Borrado reducido	

Existe en la Raspberry Pi una frecuencia máxima de reloj para los píxeles, lo que limita el modo más elevado a 1920 x 1200 píxeles con una frecuencia de imagen de 60 Hz y una eliminación reducida.

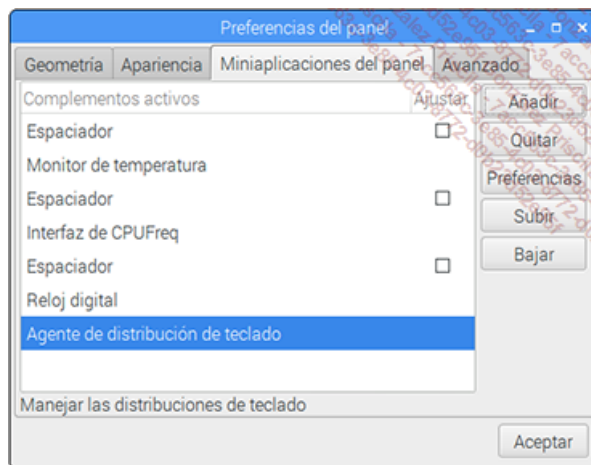
En las antiguas pantallas con tubos catódicos, la eliminación corresponde al retroceso del haz de electrones después del escaneo de una línea o una imagen. En las pantallas digitales actuales, el tiempo de retroceso no existe porque no hay escaneo por un haz de electrones. Por tanto, la duración correspondiente a la eliminación se puede reducir.

Creación de un nuevo cuadro de mandos

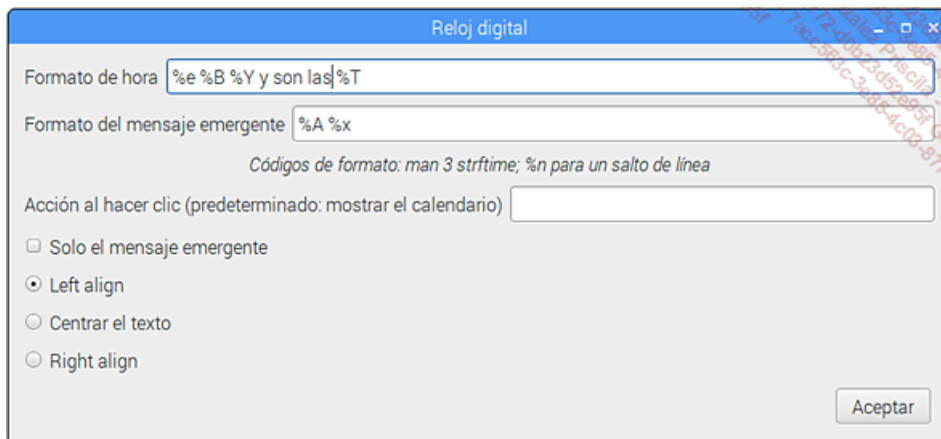
Mueva el cuadro de mandos y ajuste su tamaño como sigue.



Añada los siguientes elementos en el cuadro de mandos:



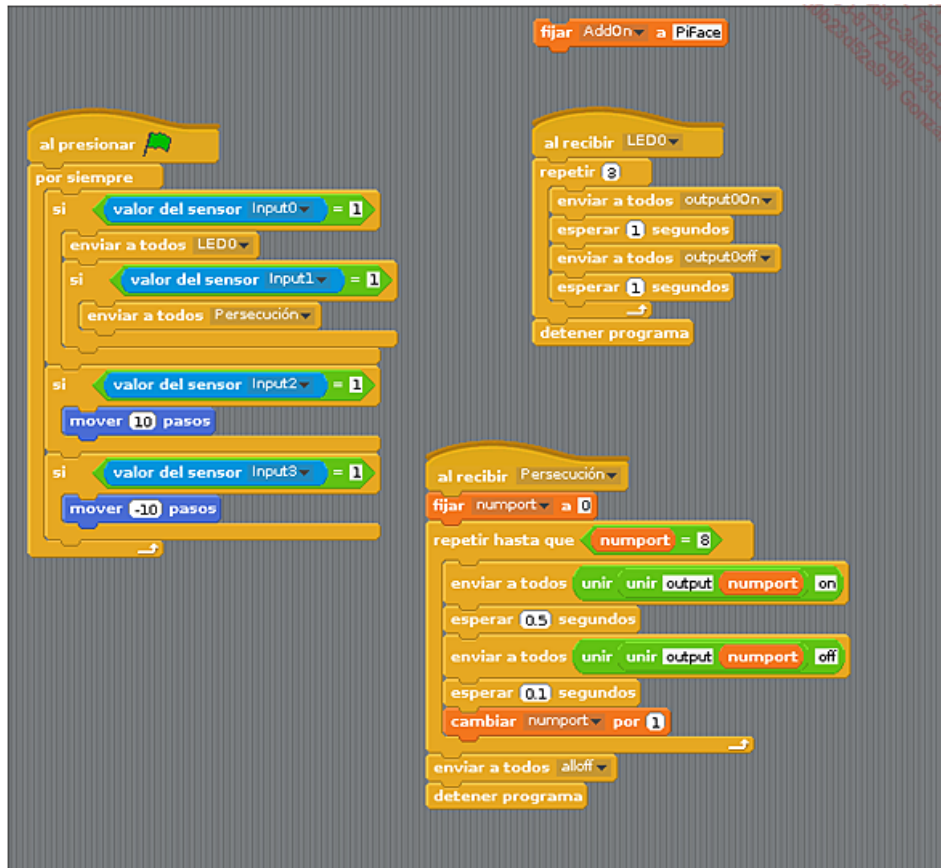
Modifique las preferencias del reloj digital. Seleccione la línea del reloj digital (clic con el botón izquierdo) y pulse en el botón **Preferencias**. Modifique la línea **Formato del reloj** para que sea idéntica a la de la siguiente imagen.



La información para personalizar la visualización del reloj se puede consultar en <https://travesuras.wordpress.com/2011/12/12/20111212-1/> y <http://php.net/manual/en/function.strftime.php>.

Controlar la tarjeta PiFace desde Scratch

Este programa añade la posibilidad de mover el duende Scratch por la escena con los botones 2 y 3:



El programa está disponible para su descarga ([scratch_piface2.sb](#)).