

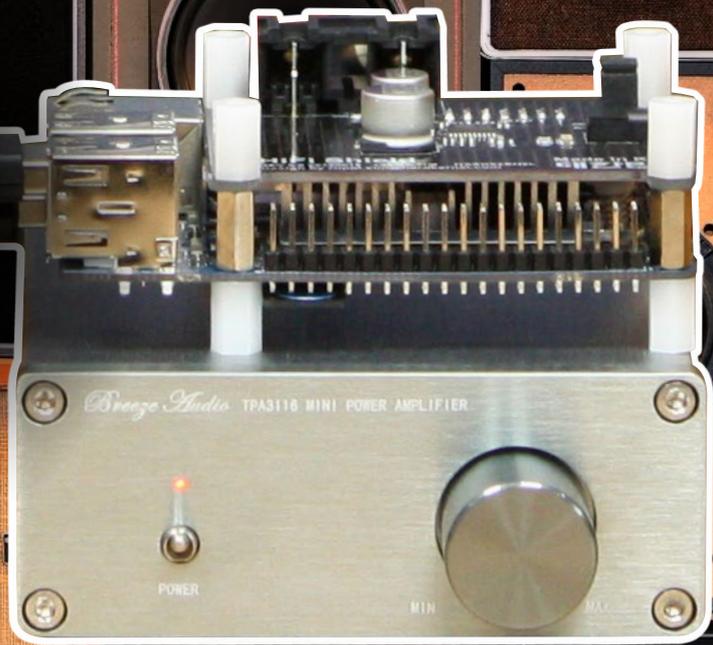
ODROID

Año dos
Num. #23
Nov 2015

Magazine

Hifi Shield

La placa con el consumo de energía más eficiente del mundo ahora te ofrece la mejor calidad de audio posible



Crea tu propio
Reproductor de
musica RuneAudio

El Gestor de Arranque
de ODROID por dentro
Das U-BOOT



Qué defendemos.

Nos esmeramos en presentar una tecnología punta, futura, joven, técnica y para la sociedad de hoy.

Nuestra filosofía se basa en los desarrolladores. Continuamente nos esforzamos por mantener estrechas relaciones con éstos en todo el mundo.

Por eso, siempre podrás confiar en la calidad y experiencia que representa la marca distintiva de nuestros productos.

Simple, moderno y único.

De modo que tienes a tu alcance lo mejor



HARDKERNEL



Ahora estamos enviando los dispositivos ODROID U3 a los países de la UE! Ven y visita nuestra tienda online!

Dirección: Max-Pollin-Straße 1
85104 Pförring Alemania

Teléfono & Fax

telf : +49 (0) 8403 / 920-920
email : service@pollin.de

Nuestros productos ODROID se pueden encontrar en: <http://bit.ly/1tXPxwe>





Tradicionalmente la reproducción de audio de alta fidelidad suponía gastar un montón de dinero, pero el nuevo **HiFi-Shield** de **Hardkernel** (de 39\$ en <http://bit.ly/IMGUICY>) hace que sea bastante rentable convertir tu **ODROID-C1+** en un reproductor de música de calidad profesional. Cuando se utiliza con algún software como **RuneAudio** o **Volumio**, es perfecto como sistema de audio integrado para el hogar y se puede gestionar desde una simple pantalla táctil. Un **C1+**, cuando se combina con la pantalla táctil de 7" (de 55\$ en <http://bit.ly/INWxDx>) y la imagen **RuneAudio** se convierte en un completo y económico sistema de audio doméstico.

Este mes, nos centramos en la administración de sistemas y hardware. **Uli** nos ofrece una visión general del gestor de arranque **ARM** llamado **Das U-Boot**, **David** continúa su serie sobre **Gestión de Volúmenes Lógicos**, **Andrew** cubre algunas técnicas en las que utilizaremos **LFTP** y **CRON** para automatizar las copias de seguridad, **Pascal** nos ayuda a evitar que el agobiante spam llene nuestro buzón de correo, **Andy** nos presenta un proyecto de servidor personal **Cloudshell**, y **Adrian** nos enseña cómo transcodificar vídeos utilizando **Handbrake**. En el lado divertido de **ODROIDs**, **Tobias** nos presenta **GLShim**, que permite ejecutar varios juegos en la plataforma **ODROID**.

ODROID Magazine, que se publica mensualmente en <http://magazine.odroid.com/>, es la fuente de todas las cosas ODROIDianas. • Hard Kernel, Ltd. • 704 Anyang K-Center, Gwanyang, Dongan, Anyang, Gyeonggi, South Korea, 431-815 • fabricantes de la familia ODROID de placas de desarrollo quad-core y la primera arquitectura ARM "big.LITTLE" del mundo basada en una única placa. Para información sobre cómo enviar artículos, contacta con odroidmagazine@gmail.com, o visita <http://bit.ly/lyplmXs>. Únete a la comunidad ODROID con miembros en más de 135 países en <http://forum.odroid.com/> y explora las nuevas tecnologías que te ofrece Hardkernel en <http://www.hardkernel.com/>.



HARDKERNEL



ameriDroid

High-Performance Embedded Computers



ODROID-XU4



ODROID-C1+

Hundreds of products available online for the professional developer and hobbyist alike

Hardkernel's EXCLUSIVE North American Distributor

ODROID

Magazine



**Rob Roy,
Editor Jefe**

Soy un programador informático que vive y trabaja en San Francisco, CA, en el diseño y desarrollo de aplicaciones web para clients locales sobre mi cluster ODROID. Mis principales lenguajes son jQuery, angular JS y HTML5/CSS3. También desarrollo SO precompilados, Kernels personalizados y aplicaciones optimizadas para ODROID basadas en las versiones oficiales de Hardkernel, por los cuales he ganado varios Premios. Utilizo mi ODROIDS para diversos fines, como centro multimedia, servidor web, desarrollo de aplicaciones, estación de trabajo y como plataforma de juegos. Puedes echar un vistazo a mi colección de 100 GB de software ODROID, kernel precompilados e imágenes en <http://bit.ly/1fsaXQs>.



**Robert Cleere,
Editor**

Soy un diseñador de hardware y software que actualmente vive en Huntsville, Alabama. Aunque semi-retirado del diseño de los sistemas integrados, incluyendo más de una década trabajando en el programa del transbordador espacial, continúo diseñando productos de software y hardware, y me interesa la producción de audio/video y las obras de arte. Mis lenguajes de programación son Java, C y C ++, y tengo experiencia con bastantes sistemas operativos integrados. Actualmente, mis proyectos principales son los sistemas navales de seguimiento y control, monitoreo ambiental y la energía solar. Actualmente estoy trabajando con varios procesadores ARM Cortex, pero mi ODROID-C1 es en gran medida el más poderoso de todos



**Bruno Doiche,
Senior
Art Editor**

Nuestro siempre confiado Editor Artístico tuvo que hacer este mes de pluri-empleado con unos clientes de TI un poco pirados, ¡pero con su buen humor supo devolver todos los limones que le fueron lanzados con el más brasileño de los caipiriñas!



**Nicole Scott,
Editor
Artístico**

Soy una experta en Producción Transmedia y Estrategia Digital especializada en la optimización online y estrategias de marketing, administración de medios sociales y producción multimedia impresa, web, vídeo y cine. Gestiono múltiples cuentas con agencias y productores de cine, desde Analytics y Adwords a la edición de vídeo y maquetación DVD. Tengo un ODROID-U3 que utilizo para ejecutar un servidor web sandbox. Vivo en el área de la Bahía de California, y disfruta haciendo senderismo, acampada y tocando música. Visita mi web en <http://www.nicolecscott.com>.



**James
LeFevour,
Editor
Artístico**

Soy un especialista en medios digitales que disfruta trabajando como freelance en marketing de redes sociales y administración de sitios web. Cuanto más aprendo sobre las posibilidades de ODROID más me ilusiona probar cosas nuevas con él. Me traslade a San Diego desde el Medio Oeste de los EE.UU. Todavía estoy bastante enamorado de muchos aspectos que la mayoría de la gente de la Costa Oeste ya da por sentado. Vivo con mi encantadora esposa y nuestro adorable conejo mascota; el cual mantiene mis libros y material informático en constante peligro.



**Manuel
Adamuz,
Editor
Español**

Tengo 31 años y vivo en Sevilla, España, aunque nací en Granada. Estoy casado con una mujer maravillosa y tengo un hijo. Hace unos años trabajé como técnico informático y programador, pero mi trabajo actual está relacionado con la gestión de calidad y las tecnologías de la información: ISO 9001, ISO 27001, ISO 20000 Soy un apasionado de la informática, especialmente de los microordenadores como el ODROID, Raspberry Pi, etc. Me encanta experimentar con estos equipos y traducir ODROID Magazine. Mi esposa dice que estoy loco porque sólo pienso en ODROID. Mi otra afición es la bicicleta de montaña, a veces participo en competiciones semiprofesionales.

INDICE



GESTOR DE ARRANQUE ARM - 6



GIGALOMANIA - 14



MANTENIMIENTO ODROID - 15



LVM - 17



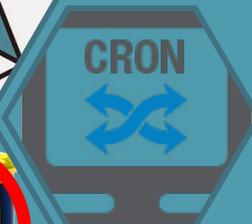
JUEGOS ANDROID: TIKI TAKA - 17



JUEGOS LINUX - 18



WIKI DE LA COMUNIDAD - 22



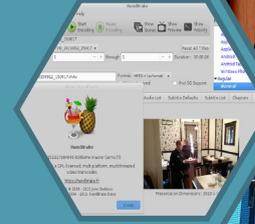
LFTP Y CRON - 23



FILTRADO DE SPAM - 27



HI-FI SHIELD - 30



HANDBRAKE - 32



SERVIDOR DOMESTICO CLOUDSHELL - 34



JUEGOS ANDROID: DUNGEON BOSS - 37



RUNE AUDIO MUSIC PLAYER - 38



PANTALLA MULTITACTIL CI - 43



CONOCIENDO A UN ODROIDIAN - 46

SACARLE MAS PROVECHO AL DAS U-BOOT

EL GESTOR DE ARRANQUE DE ODROID POR DENTRO

por Uli Middelberg

La mayoría de los nuevos dispositivos basados en ARM utilizan “Das U-Boot” (u-boot) para cargar e iniciar el kernel de Linux. Muchos proveedores se empeñan en crear configuraciones u-boot amigables para el usuario final. Si necesitas más flexibilidad, tal vez merezca la pena analizar con más detalle la estructura interna del u-boot.

¿Qué es u-boot?

U-boot es un gestor de arranque similar a lilo (<http://bit.ly/1W69Dg7>) y grub (<http://bit.ly/1PAFY5>), pero que está diseñado específicamente para dispositivos embebidos. U-boot esta mantenido por @denx (<http://bit.ly/1LFIOGm>), y se publica bajo GNU General Public License v2 (GPL-2.0+). Comparado con grub, u-boot no es tan configurable y amigable, pero su tamaño es bastante menor.

La forma más adecuada de interactuar con u-boot es a través de la consola serial. Muchos dispositivos ARM ofrecen acceso serial a través del USB, lo cual facilita el acceso a la consola. Tan sólo tienes que conectar la consola serial por USB a otra máquina Linux y tendrás acceso remoto a tu dispositivo ARM. Para ello, inicia sesión en tu máquina Linux por ssh y abre la conexión serial mediante minicom:

```
$ sudo minicom -b 115200 -D /dev/ttyUSB0
```

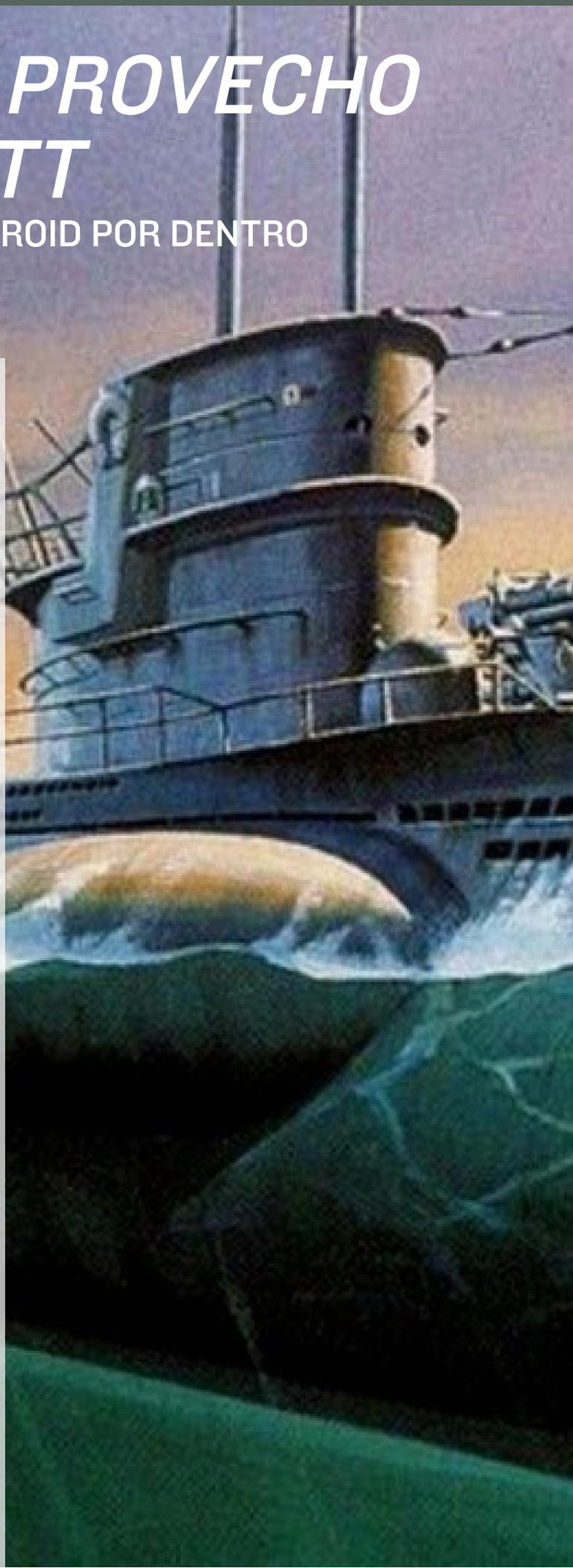
Pulsa “Ctrl-A z” para abrir la pantalla de ayuda/configuración interactiva, y pulsa “Ctrl-A x” para salir de la sesión de minicom. Minicom no es el único emulador de terminal para acceso serial. Screen y picocom son otras alternativas muy utilizadas:

```
$ sudo picocom -b 115200 /dev/ttyUSB0
```

Cuando hayas terminado de usar picocom, presiona “Ctrl-A Ctrl-X” para cerrar la sesión.

Proceso de arranque

Cuando encendemos un dispositivo, éste tiene que pasar por varias fases antes de que veas la pantalla de inicio de se-



sión o el escritorio del sistema operativo. La primera fase del proceso de arranque es el Secondary Program Loader (SPL). Esta parte de código preliminar es responsable de inicializar la placa, cargar el binario u-boot (“programa secundario”) y gestionar el flujo de control del programa principal u-boot. Es específico de cada dispositivo y suele presentarse como un bloque de código binario cerrado por el proveedor del SoC.

Secondary Program Loader

El Secondary Program Loader (SPL) y el binario u-boot residen en una región especial de la memoria flash de la placa o en los primeros sectores del disco de arranque. Los dispositivos ODROID utilizan módulos eMMC o microSD para almacenar el binario u-boot y el SPL. En el apartado de este artículo sobre las Particiones y Formato del disco encontraras información de la distribución de las áreas del disco.

U-Boot

En la segunda fase del proceso, se ejecuta el programa principal u-boot. U-boot primero busca un entorno personalizado almacenado en un espacio reservado de la tarjeta microSD o módulo eMMC, o vuelve al entorno por defecto de la compilación si fuera necesario. Es en este momento, cuando puedes interrumpir el proceso de arranque automático pulsando una tecla en la consola serial, iniciando un intérprete de comandos u-boot interactivo. La variable u-boot llamada bootdelay especifica el número de segundos que se espera a que se pulse una tecla antes de continuar con el proceso de arranque automático.

```
CPU : AMLogic S805
MEM : 1024MB (DDR3@792MHz)
BID : HKC13C0001
S/N : HKC1CC037EBCBFA4
0x0000009f
Loading U-boot...success.
```

```
U-boot-00000-geb22ea4-dirty(odroidc@eb22ea4b) (Jul 28
2015 - 22:16:46)
```

```
DRAM: 1 GiB
relocation Offset is: 2ff1c000
MMC: eMMC: 0, SDCARD: 1
IR init is done!
vpu clk_level = 3
set vpu clk: 182150000Hz, readback:
182150000Hz(0x701)
mode = 6 vic = 4
set HDMI vic: 4
mode is: 6
viu chan = 1
config HPLL
config HPLL done
reconfig packet setting done
MMC read: dev # 0, block # 33984, count 12288 ...
12288 blocks read: OK
There is no valid bmp file at the given address
=====
```

```
Vendor: Man 450100 Snr 01172c20 Rev: 4.7 Prod: SDW16
Type: Removable Hard Disk
```

```
Capacity: 15028.0 MB = 14.6 GB (30777344
x 512)
-----
Partition      Start Sector      Num Sectors      Type
  1              3072              524288           b
  2             527360           14680064         83
  3            15207424           15569920         83
=====
Net:   Meson_Ethernet
init suspend firmware done. (ret:0)
Hit Enter key to stop autoboot -- : 3 tstc enter

exit abortboot: 1
odroidc#
```

El proceso de inicio automático ejecuta una macro u-boot especial, llamada bootcmd que carga y ejecuta los siguientes procesos:

- (opcional) un entorno u-boot a medida, como uEnv.txt**
- (opcional) una macro u-boot precompilada, como la imagen del kernel boot.scr, como ulmage**
- (opcional) el binario en árbol del dispositivo, como meson8b_odroidc.dtb**
- (opcional) el disco RAM inicial, como ulnitrd**

kernel de Linux

La tercera fase es la carga del kernel de Linux. Sin embargo, antes de que el kernel de Linux tome el control, u-boot pasa una línea de comandos al kernel con una serie de parámetros esenciales. Estos parámetros se pueden ver una vez que el sistema operativo haya arrancado escribiendo en una ventana de terminal:

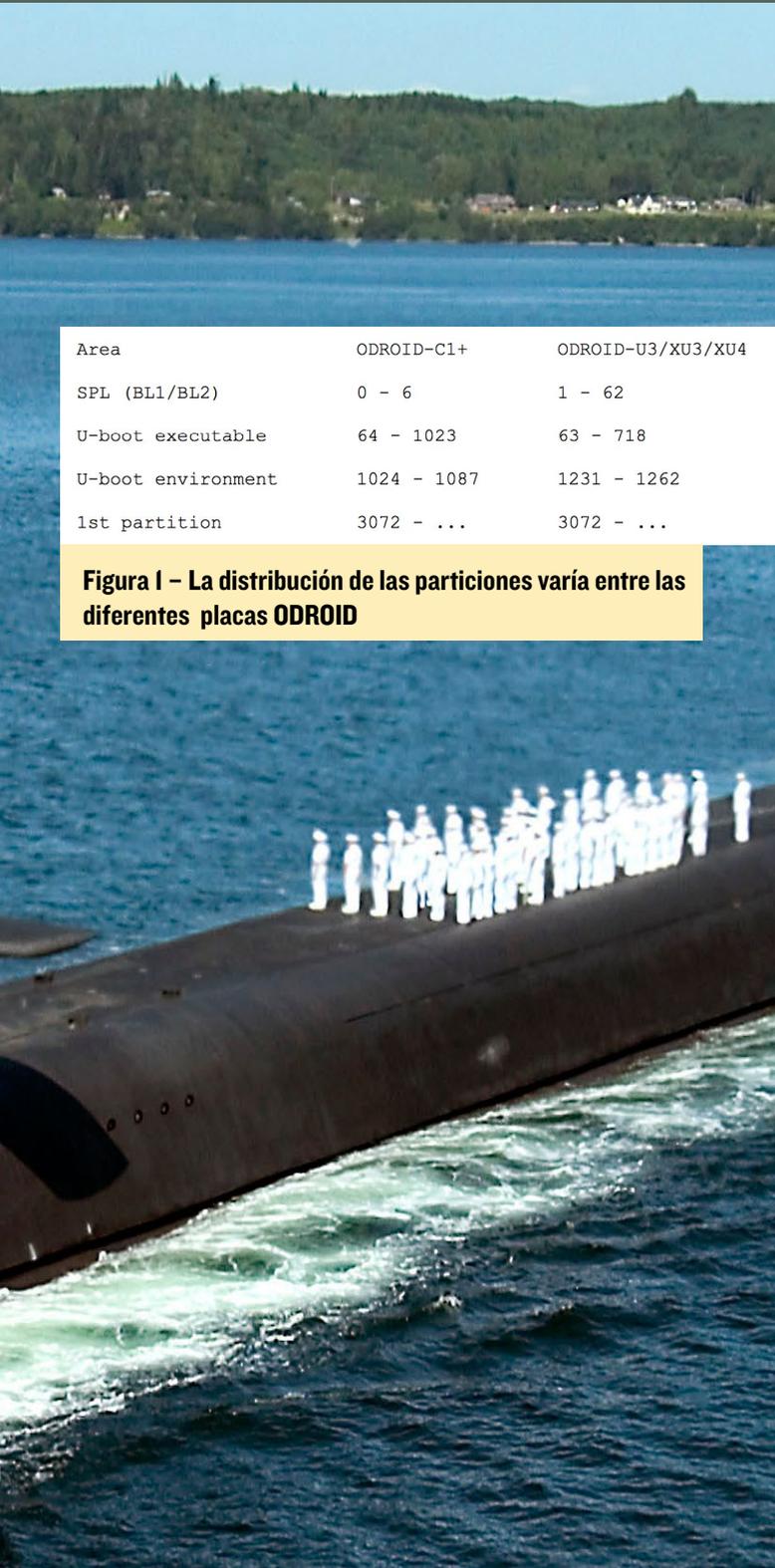
```
$ cat /proc/cmdline
root=/dev/mmcblk0p2 rootwait rw
console=ttyS0,115200n8 console=tty0 no_console_suspend
vdacfcfg=0xa000 logo=osd1,loaded,0x7900000,720p,full
dmfc=3 cvbsmode=576cvbs hdmimode=1080p m_bpp=32
vout=hdm1 disablehpd=true
```

El kernel activa el hardware, monta el sistema de archivos root en función del parámetro del kernel "root =", luego pasa el control a /sbin/init.

Atajo de Hardkernel

Hardkernel hace uso de un comando u-boot especial llamado cfgload, que pasa por alto el proceso de arranque normal y ofrece la posibilidad de configurar u-boot de una forma más simplificada en un único archivo llamado boot.ini. Los cambios de configuración se puede hacer fácilmente editando el archivo boot.ini, en lugar de tener que modificar el entorno u-boot, aunque esta sistema no permite acceder al intérprete de comandos u-boot interactivo.





Area	ODROID-C1+	ODROID-U3/XU3/XU4
SPL (BL1/BL2)	0 - 6	1 - 62
U-boot executable	64 - 1023	63 - 718
U-boot environment	1024 - 1087	1231 - 1262
1st partition	3072 - ...	3072 - ...

Figura 1 – La distribución de las particiones varía entre las diferentes placas ODROID

Sólo puedes arrancar una configuración cada vez por defecto. Si un cambio de configuración concreto hace que el sistema se bloquee durante el arranque, tendrás que quitar la microSD o módulo eMMC de tu dispositivo y revertir el cambio editando el archivo boot.ini desde tu portátil o PC.

Particiones y formato del disco

Muchas placas ARM incluyendo las ODROIDS, utilizan una microSD o módulo eMMC para almacenar los diferentes componentes del u-boot: SPL, el ejecutable del u-boot y el entorno del u-boot. A diferencia de los dispositivos que utilizan memoria de almacenamiento flash para esta finalidad, como son las tablets y los smartphones, no puedes hacer que tu dispositivo no se inicie con una actualización de firmware fallida.

Los componentes de u-boot se almacenan en áreas reservadas del disco previas a la primera partición. Como se muestra en la Figura 1, no existe una distribución común para los componentes U-Boot en las diferentes placas. Si deseas modificar las particiones con herramientas como fdisk, debe tener en cuenta que fdisk siempre intenta crear las nuevas particiones a partir del sector 2048 por defecto.

U-boot intenta cargar la imagen del kernel y los archivos adicionales de la primera partición del dispositivo de arranque designado. Las versiones anteriores de u-boot sólo admitían el sistema de archivos vfat, dando lugar a la típica distribución de particiones que se muestra en la Figura 2.

Las versiones recientes de u-boot permiten la lectura de archivos desde sistemas de archivos ext4, de modo que no hay necesidad de utilizar la partición de arranque vfat. Muchas imágenes continúan utilizando vfat por comodidad para el usuario, ya que permite editar el archivo boot.ini en Windows y OSX, que no son compatibles con los sistemas de archivos ext4 por defecto.

Comandos U-boot

El intérprete de comando interactivo del u-boot ofrece una serie de comandos ejecutables, dependiendo de la versión y el nivel de actualización. Puedes obtener una lista de todos los comandos soportados escribiendo “help” o “help <comando>”. Para más información, puedes consultar la documentación oficial en <http://bit.ly/1PH80Ai>.

Entorno U-boot

El entorno de u-boot almacena una serie de variables con el formato <variable> = <valor>. Si en el disco existe un entorno guardado, u-boots inicia el entorno de trabajo con estos valores, o vuelve al entorno por defecto. Las variables estan referenciadas por \$ {variable}:

printenv: devuelve todo el conjunto de variables

device	label	filesystem	mount point
/dev/mmcblk0p1	boot	vfat	/media/boot
/dev/mmcblk0p2	rootfs	ext4	/

Figura 2 – Esquema de particiones del gestor de arranque en vfat

printenv variable: devuelve el valor de una sola variable
setenv variable valor: fija una determinada variable con el valor designado

setenv variable: elimina una variable del entorno

env -a: restablece el entorno u-boot a los valores por defecto

saveenv: almacena el entorno actual en la microSD o módulo eMMC. También puedes leer y modificar el entorno u-boot cuando el sistema haya iniciado Linux.

Macros U-boot

U-boot utiliza variables para almacenar scripts, de modo que el siguiente comando definirá una variable llamada macro que contiene una lista de comandos, delimitada por un punto y coma (;):

```
setenv macro `...;`
```

Las macros pueden activarse escribiendo el comando “run”:

```
run macro
```

La macro más importante es bootcmd, que se ejecuta por defecto. Las Macros devuelven el valor del último comando ejecutado, que luego se puede evaluar usando una clausula if/then/else. La Información adicional se devuelve modificando variables específicas. Por ejemplo, el comando “load” utiliza una variable global llamada “filesize” para devolver el número de bytes leídos desde el disco.

Típica secuencia de arranque

La macro denominada “bootcmd” pone en marcha la secuencia de arranque, que se ejecuta en modo no interactivo. Antes de ejecutar bootcmd, u-boot inicia el entorno del u-boot que contiene las variables de configuración y las macros. Los fases que tienen lugar dentro de la macro bootcmd son:

Cargar un entorno personalizado desde la partición de arranque.

La macro bootcmd busca un archivo de texto llamado uEnv.txt en la partición de arranque, lo carga y luego mezcla su contenido con el entorno u-boot existente anulando los valores de las variables existentes. Este paso es opcional.

Cargar un entorno personalizado desde la partición de arranque

La macro bootcmd buscará a continuación un archivo llamado boot.scr en la partición de arranque, lo cargará y luego ejecutará su contenido sin tener que regresar a bootcmd. El archivo boot.scr contiene comandos u-boot compilados en un formato binario. Este paso es opcional.

Cargar la imagen del kernel desde la partición de arranque





La macro `bootcmd` posteriormente buscará la imagen del kernel. La variable `kernel` del `u-boot` contiene el nombre del fichero exacto, normalmente llamado `zImage`. Puedes arrancar una imagen de kernel alternativa modificando la variable denominada “`kernel`”.

Cargar el binario en árbol del dispositivo desde la partición de arranque

El kernel de Linux ARM necesita una descripción del dispositivo de bajo nivel o árbol del dispositivo en formato binario, ya sea incluido en la imagen del kernel o como archivo independiente. Muchas plataformas prefieren cargar el árbol binario del dispositivo como un archivo independiente, proporcionando mayor flexibilidad, lo cual permite distribuir una única imagen para diferentes plataformas, o adaptar el árbol del dispositivo para diferentes usos. Algunos proveedores permiten que `bootcmd` decida que árbol binario cargar, dependiendo de la placa.

Cargar el archivo del disco RAM inicial desde la partición de arranque

La macro `bootcmd` finalmente intentará cargar el archivo de la unidad RAM inicial desde un archivo llamado `initrd`. Este archivo se crea y actualiza usando la utilidad `update-initramfs`, que normalmente se lleva a cabo cuando se instala una nueva imagen del kernel. Este paso es opcional.

Acceso Userland a u-boot

Tras arrancar el sistema, puedes modificar el entorno `u-boot`. El paquete de Ubuntu `u-boot-tools` contiene dos utilidades: `fw_printenv` y `fw_setenv`, que se pueden utilizar para acceder al entorno de `u-boot` almacenado. En primer lugar, instala el paquete con el siguiente comando:

```
$ sudo apt-get -y install u-boot-tools
```

Luego, necesitas configurar el dispositivo de almacenamiento y la información de la ubicación en el archivo `/etc/fw_env.config`. Como ejemplo, esta es la configuración para el ODROID-C1:

```
# <device> <offset> <length>
/dev/mmcblk0 0x80000 0x8000
```

Pon a prueba la configuración con el siguiente comando:

```
$ sudo fw_printenv bootdelay
```

El entorno de `u-boot` almacenado en el disco contiene una suma de verificación CRC. Si la impresión `offset` y la extensión no coinciden, recibirás un aviso:

```
Warning: Bad CRC, using default environment
```

Usos típicos

Por defecto, el ODROID-C1 está configurado para arrancar a través de cfgload con un pequeño retardo en el arranque. Aumentando la variable u-boot denominada BootDelay te da la oportunidad de interrumpir el proceso de arranque automático. Puedes utilizar el comando fw_printenv para comprobar el valor actual y el comando fw_setenv para asignar un nuevo valor a BootDelay:

```
$ sudo fw_printenv bootdelay
$ sudo fw_setenv bootdelay 3
```

Arrancar un kernel alternativo

Si compilas tu propio kernel desde la fuente, es posible que quieras probar el nuevo kernel antes de sobrescribir el actual, o guardar un kernel funcional como backup. Es posible arrancar un kernel diferente al kernel por defecto definiendo una macro u-boot, que haga referencia a diferentes archivos de la partición de arranque:

```
sudo setenv m_boot_ 'setenv bootargs "root=/dev/
mmcblk0p2 rootwait rw console=ttyS0,115200n8
console=tty0 no_console_suspend vdaccfg=0xa000
logo=osdl,loaded,0x7900000,720p,full dmfc=3
cvbsmode=576cvbs hdmimode=1080p m_bpp=32 vout=hdmi
disablehpd=true"; fatload mmc 0:1 0x21000000 _
uImage;fatload mmc 0:1 0x22000000 uInitrd; fatload
mmc 0:1 0x21800000 _meson8b_odroidc.dtb; fdt addr
21800000; fdt rm /mesonstream; fdt rm /vdec; fdt rm /
ppmgr; fdt rm /mesonfb; bootm 0x21000000 0x22000000
0x21800000'
```

Esto define la macro de u-boot “m_boot_” que hace referencia a la imagen del kernel “_uImage” y al binario del dispositivo “_meson8b_odroidc.dtb”. Desafortunadamente, u-boot no permite saltos de línea dentro de las macros, lo cual dificulta la lectura. Si insertas un salto de línea después de cada punto y coma (;), verás la misma secuencia de comandos que en boot.ini:

```
setenv bootargs "root=/dev/mmcblk0p2 root-
wait rw console=ttyS0,115200n8 no_console_suspend
vdaccfg=0xa000 logo=osdl,loaded,0x7900000,720p,fu
ll dmfc=3 cvbsmode=576cvbs hdmimode=1080p m_bpp=32
vout=hdmi disablehpd=true";
fatload mmc 0:1 0x21000000 _uImage;
fatload mmc 0:1 0x22000000 uInitrd;
fatload mmc 0:1 0x21800000 _meson8b_odroidc.dtb;
fdt addr 21800000;
```

```

fdt rm /mesonstream;
fdt rm /vdec;
fdt rm /ppmgr;
fdt rm /mesonfb;
bootm 0x21000000 0x22000000 0x21800000';

```

Esta es la secuencia de comandos correcta durante el arranque cuando configuramos boot.ini para una configuración sin nodo principal:

```

setenv vout_mode "hdmi"
setenv m_bpp "32"
setenv hpd "0"
setenv cec "0"
setenv vpu "0"
setenv hdmioutput "0"

```

Una vez que hayas definido la macro m_boot_ y copiado la imagen del kernel y el árbol binario del dispositivo a la partición de arranque con un "_" en el nombre del archivo, puede arrancar este kernel interrumpiendo u-boot y escribiendo lo siguiente en el intérprete de comandos del u-boot:

```
run m_boot_
```

Arrancar rootfs desde una partición diferente o unidad USB

Abrás observado que u-boot pasa el parámetro de kernel "root =" al kernel de Linux. El kernel intentará montar el sistema de archivos raíz (/) desde ahí. El sistema de archivos raíz se puede redirigir de diferentes formas:

```

via device node: root=/dev/sda1
via UUID filesystem identifier: root=UUID=e139ce78-
9841-40fe-8823-96a304a09859

```

Si sólo tiene un dispositivo de almacenamiento USB conectado a la placa, puedes referenciar con seguridad el sistema de ficheros raíz usando el nodo del dispositivo. Si vas a conectar o retirar dispositivos de almacenamiento adicionales, es mejor referenciar el sistema de archivos raíz utilizando el identificador UUID, de lo contrario, el kernel podría omitir el sistema de archivos raíz al "olvidar" la memoria USB durante el siguiente arranque. Puedes leer los UUIDs disponibles con el comando blkid:

```

$ sudo blkid
/dev/mmcblk0p1: SEC_TYPE="msdos" LABEL="boot"
UUID="E26F-2230" TYPE="vfat"

```

```
/dev/mmcblk0p2: LABEL="rootfs" UUID="e139ce78-9841-40fe-8823-96a304a09859" TYPE="ext4"  
/dev/mmcblk0p3: LABEL="rootfs2" UUID="e139ce78-9841-40fe-8823-96a304a09860" TYPE="ext4"  
/dev/sda1: LABEL="rootfs" UUID="e54a458d-6a66-4ed2-9394-7b22d2943ec9" TYPE="ext4"
```

El UUID se puede ajustar al mismo tiempo que se crea un sistema de archivos con el comando mkfs:

```
$ sudo mkfs.ext4 -O ^has_journal -b 4096 -L rootfs -U e54a458d-6a66-4ed2-9394-7b22d2943ec9 /dev/sda1
```

Si omites el parámetro -U, el UUID se elige al azar. Como he dicho anteriormente, puedes definir una macro u-boot adicional que luego pasara un sistema de archivos raíz alternativo al kernel:

```
sudo setenv m_boot_usb `setenv bootargs `root=/dev/sda1 rootwait rw console=ttyS0,115200n8 console=tty0 no_console_suspend vdaccfg=0xa000 logo=osd1,loaded,0x7900000,720p,full dmfc=3 cvbsmode=576cvbs hdmimode=1080p m_bpp=32 vout=hdmi disablehpd=true`; fatload mmc 0:1 0x21000000 uImage;fatload mmc 0:1 0x22000000 uInitrd; fatload mmc 0:1 0x21800000 meson8b_odroidc.dtb; fdt addr 21800000; fdt rm /meson-stream; fdt rm /vdec; fdt rm /ppmgr; fdt rm /mesonfb; bootm 0x21000000 0x22000000 0x21800000`
```

Aunque no se recomienda, si necesitas clonar el contenido de un sistema de archivos raíz montado, puedes realizar un vínculo antes de copiar:

```
$ sudo mount /dev/sda1 /media/usb  
$ sudo mount -o bind / /mnt  
$ cd /mnt  
$ sudo find . | sudo cpio -dump /media/usb  
$ cd  
$ sudo umount /mnt  
$ sudo umount /media/usb
```

Que luego te permite arrancar desde USB escribiendo lo siguiente en el intérprete de comandos u-boot:

```
run m_boot_usb
```

Para preguntas, comentarios o sugerencias, puedes visitar el post original en <http://bit.ly/1LMcDHM>.



GIGALOMANIA

PRACTICA EL ARTE DE LA GUERRA

por Tobias Schaaf

Gigalomania es un remake del viejo juego Mega lo Mania originalmente disponible para Amiga y DOS. En este juego eres una especie de dios, tienes que gestionar tu civilización, empezando en la edad de piedra hasta llegar a la era de la energía nuclear. Comienzas con tu propia fortaleza, tienen que utilizar tus recursos para inventar nuevas armas y defensas, y utilizarlas para derrotar a tus enemigos. Todo empieza recogiendo simples piedras para usarlas como armas arrojadas, hasta que logras conseguir aviones de combate y misiles nucleares. Gigalomania es muy similar, pero con gráficos más modernos. El sistema de juego sigue siendo el mismo: investigar nuevas tecnologías, reunir recursos y matar a tus enemigos. El juego es fascinante y muy divertido.

Puedes conseguir Gigalomania desde mi repositorio bien desde la lista de paquetes de jessie/main o bien desde la lista de wheezy/main, debería funcionar en cualquier sistema con soporte SDL2:

Portada original del Mega lo Mania



Es posible construir aviones a reacción



```
$ sudo apt-get install \
gigalomania-odroid
```

Si deseas instalarlo directamente, descarga el archivo .deb desde <http://bit.ly/1ZZaKOc> e instalarlo con estos comandos:

```
$ cd ~/Downloads
$ sudo apt-get install gdebi
$ sudo gdebi ./gigalomania*.deb
```

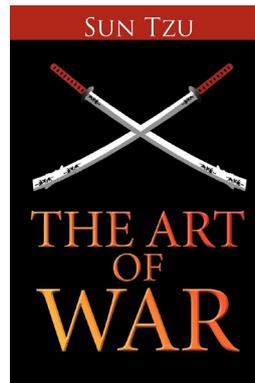
Puedes iniciar el juego escribiendo lo siguiente en una ventana de Terminal:

```
$ cd /opt/gigalomania/gigalomania
$ ./gigalomania gfx
```

Echa un vistazo a un vídeo del juego en <http://bit.ly/1LbZran>, o publicar comentarios, sugerencias y preguntas en el foro <http://bit.ly/1Z4b9yc>.



Sistema de juego (arriba y abajo)



LIMPIAR EL USB UNA RAPIDA Y SIMPLE SOLUCION PARA LOS PUERTOS USB QUE DAN FALLOS

editado por Rob Roy

Los puertos USB host de cualquier ordenador pueden ensuciarse a veces, provocando que no hagan un buen contacto con los periféricos USB, de modo que es bueno limpiarlos de vez en cuando con un bastoncillo de algodón y algo de alcohol. Las siguientes imágenes muestran cómo deben limpiarse de un modo seguro los puertos USB de un ODROID-XU4.

Hazlo con suavidad, y asegúrate de que los puertos USB están completamente secos antes de usarlos. Para comentarios, sugerencias y preguntas, puedes visitar el hilo original en <http://bit.ly/1G8sgJe>.

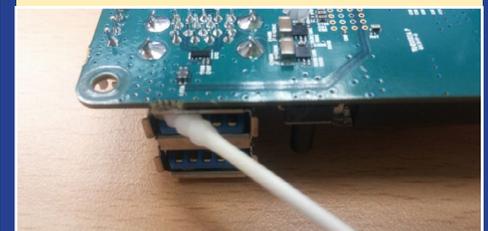
Coge un bastoncillo de algodón y alcohol y dale la vuelta al XU4 para ver las conexiones



Encontraras 5 yemas de conexión dentro de cada puerto USB



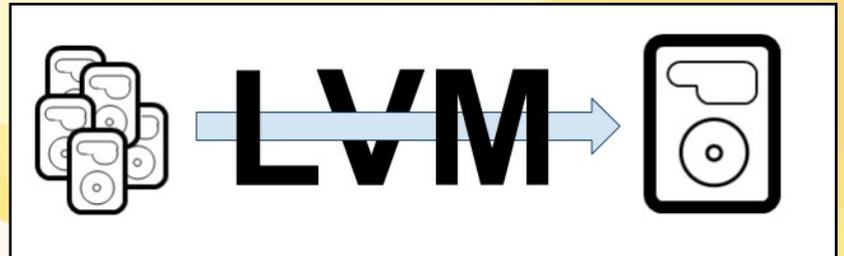
Limpia las 5 yemas con el bastoncillo de algodón y algo de alcohol



GESTION DE VOLUMENES LOGICOS

COMO MANIPULAR GRUPOS DE VOLUMENES

por David Gabriel



Tras aprender los conceptos básicos de la gestión de volúmenes lógicos (LVM), y la forma de importar y exportar tus grupos de volúmenes entre diferentes sistemas, ahora te voy a dar algunos consejos sobre cómo aprovechar los beneficios de haber configurado LVM, tanto si eres un usuario particular como si eres un administrador de sistemas de grandes empresas. Tener LVM configurado en tu sistema no sólo facilita las cosas, sino que hace también que sean más fiables y resistentes. Y esto se hace evidente cuando lo tienes en grandes entornos que están creciendo continuamente.

Añadir espacio

Supongamos que ya tienes creado tu LVM y está lleno, de modo que necesitas más espacio. Por otro lado, no tienes ninguna eMMC, tarjeta SD o disco duro externo disponible. Aunque si tiene unidades flash, así que ¿por qué no usarlas? Puedes tener tantos dispositivos como quieras en tu grupo de volúmenes. Ni siquiera tienes que crear particiones si no quieres. Los siguientes pasos preparan un disco para LVM y así podrás añadirlo al grupo de volúmenes:

```
$ pvcreate /dev/sdb
$ vgextend rootvg /dev/sdb
```

Copias de seguridad

Dispones de una base de datos muy importante que se ejecuta en tu ODROID, y perder todo su contenido sería un desastre. Así que, ¿qué puedes hacer? ¿Necesitas una copia de seguridad! Hay muchas formas de guardar tus datos de un modo seguro ante cualquier evento inesperado. Te voy a mostrar cómo hacer copias de seguridad usando “instantáneas” de LVM. Es muy simple y sólo tiene que ejecutar el siguiente comando:

```
$ lvcreate -L 20G -s -n snaplv /dev/rootvg/dblv
```

Una vez completado, puedes comprobar tu volumen lógico recién creado (LV) con el siguiente comando, que prepara el disco para LVM y lo añade al grupo de volúmenes:

```
$ lvs
```

La clave aquí es el parámetro “-s”, el cual creará una imagen congelada del volumen lógico como la instantánea que se toma en su momento. El tamaño de la instantánea no tiene porque ser igual al original, pero debería contar con suficiente espacio para los cambios que posiblemente se realicen.

Después, puedes montar el VL y copiarlo fuera del sistema de la forma que prefieras. La copia instantánea puede ser tomada con los sistemas de ficheros montados, no tienes que preocuparte de los archivos que se modificaron durante la copia de seguridad. Es muy útil en entornos de producción donde no puedes detenerlo todo sólo por una copia de seguridad, o cuando necesitas ocuparte de los archivos que no se están respaldando porque están en uso. Es bueno hacer una instantánea antes de realizar cambios en el VL, así tendrá la posibilidad de recuperarlo rápidamente todo en el caso de algo salga mal.

Retirar los discos

Imagínate la siguiente situación: Añadiste una unidad flash o un disco duro externo a tu LVM, pero ahora por alguna razón, lo necesitas en otro sitio. O lo necesitas reemplazar porque está empezando a dar fallos, o simplemente quieres cambiarlo por uno mejor y más grande. El problema está en que, una vez agregado a un grupo de volúmenes, el sistema es el que coordina cómo se guardan los datos, así que no puedes simplemente desconectarlo de la placa o te cargarás todo el LVM.

Para abordar esta cuestión, puede mover los datos entre los discos del sistema de modo que puedas vaciar uno y así retirarlo del grupo de volúmenes. Hay dos formas de hacerlo. En primer lugar, si tienes suficiente espacio disponible en el grupo de

volúmenes, utiliza el siguiente comando:

```
$ pvmove /dev/sda2
```

Ten paciencia, ya que este comando requiere tiempo para completarse. Se moverá todo el contenido de /dev/sda2 distribuyéndose a otros discos del mismo grupo de volúmenes. Tras completarse, puedes utilizar el siguiente comando para retirar el disco del grupo de volúmenes:

```
$ vgreduce rootvg /dev/sda2
```

Si no tienes espacio, tendrás que añadir un nuevo disco antes de mover los archivos. Utiliza los comandos que ya hemos visto para hacer el nuevo disco accesible a LVM y agregarlo al grupo de volúmenes. Después, puedes mover los datos del disco antiguo directamente al disco nuevo con este comando:

```
$ pvmove /dev/sda2 /dev/sdb1
```

Ten en cuenta que no importa como lo hagas, pero siempre asegúrate de contar con una backup por si aparece cualquier problema que pueda provocar la pérdida de los datos, como puede ser un corte de energía. Recuerda también que puedes hacerlo aunque los volúmenes estén activos, por lo que no es necesario desmontar ningún sistema de archivos. No obstante, recomiendo hacerlo cuando los discos no estén demasiado ocupados, de este modo el impacto sobre el rendimiento del sistema será menor.

Separar un grupo de volúmenes

Siempre es bueno tener los volúmenes bien organizados y contar con diferentes sistemas de archivos para distintas finalidades, así si hay problemas con uno no afecta al resto. Lo mismo sucede con los grupos de volúmenes: es mejor crear grupos de volúmenes independientes para los archivos del sistema, las aplicaciones y las bases de datos. Sin embargo, si no organizaste los volúmenes desde un prin-

cipio y ahora los tienes desorganizados, existe una solución. Podemos sacarle partido a la LVM una vez más y dividir los grupos de volúmenes.

No tienes que añadir nuevos discos a tu LVM si ya hay algo de espacio disponible. Para comprobarlo, utiliza el comando "pvscan" para mostrar todos los volúmenes físicos y su respectivo espacio libre. Elige un disco al que desees mover un nuevo grupo de volúmenes, luego, utiliza el comando "pvmove" para liberar espacio en él y posteriormente reorganizar los datos en los otros discos. Por último, ejecuta el siguiente comando:

```
$ vgsplit rootvg appvg /dev/sdb1
```

Esto creará un nuevo grupo de volúmenes usando /dev/sdb1. El efecto es el mismo que si hubieras utilizado el comando "vgreduce" para extraerlo del grupo volumen original (VG), luego, se utiliza el comando "vgcreate" para crear el nuevo VG. El comando "vgsplit" combina estos dos pasos en uno sólo para facilitar el proceso. Después, simplemente crea los volúmenes lógicos y los sistemas de archivos como quieras.

La gestión de volúmenes lógicos proporciona a los usuarios una gran panorámica del almacenamiento del sistema y cómo se gestiona, ofreciendo más flexibilidad y herramientas más amigables. Es más fácil de lo que parece, y cualquier persona puede utilizarlo para tener su almacenamiento bien organizado y disponer de una mayor flexibilidad a la hora de gestionar el espacio en disco.



PANTALLA TÁCTIL EN SU PLENITUD TIKI TAKA SOCCER ES EL JUEGO PERFECTO PARA LOS USUARIO DE PANTALLA TÁCTIL

por Bruno Doiche



Si has conectado a tu ODRROID una pantalla táctil para lograr que Android se ejecute en toda su gloria, no pierdas la oportunidad de descargar Tiki Taka Soccer. Es un pequeño e increíble juego que utiliza lo mejor que puede ofrecer un juego de fútbol con comandos táctiles, te hará olvidar el resto de juegos derivados del PES o FIFA. ¡Realmente es un juego de fútbol muy bien hecho!

<https://play.google.com/store/apps/details?id=com.timconstant.tikitakasoccer&hl=en>



Es un juego de fútbol muy divertido que hará un uso bastante intenso de tu pantalla táctil

SKILL	ATT:20	DEF:22	OVR:21	ENG PREM	FIXTURES
ENERGY				1 Wigan	2 2 0 0 8 1 7 6
MONEY 21.367.095				2 Liverpool	2 2 0 0 6 1 5 6
MANAGER				3 Burnly	2 1 1 0 5 2 3 4
Mr. Pixel				4 Pixel Rangers	2 1 1 0 2 1 1 4
LEVEL 25				5 Man R	2 1 0 1 2 1 1 3
STAFF				6 Norwich	2 1 0 1 4 3 3 5
15% POST MATCH ENERGY				7 Sunderland	2 1 0 1 4 3 1 5
Skills				8 WBA	2 1 0 1 4 4 0 5
+2 PASSING, +2 SHOOTIN				9 Chels	2 1 0 1 4 4 0 5
				10 Hull C	2 1 0 1 3 3 0 5
				11 Forest	2 1 0 1 2 2 0 3
				12 Burton	2 1 0 1 2 6 -4 3
				13 Arsenal	2 0 2 0 3 3 0 2
				14 Newcastle	2 0 2 0 2 2 0 2
				15 Southampton	2 0 2 0 2 2 0 2

Como otros juegos de fútbol, tendrás que gestionar tu equipo para ganar la champion

JUEGOS LINUX

COMPATIBILIDAD OPENGL USANDO GLSHIM

por Tobias Schaaf



Como todos sabemos (o deberíamos saber), los dispositivos ODROID no soportan OpenGL pero si OpenGL ES, que es un subgrupo optimizado de OpenGL diseñado para sistemas embebidos. Esto significa que muchos programas y juegos no se pueden ejecutar en ODROID, ya que necesitan la librería OpenGL. Esto resulta muy frustrante, ya que los dispositivos ODROID y OpenGL ES podrían ejecutar estos programas muy bien, pero muchos desarrolladores no se molestan en exportar sus programas a OpenGL ES y en su lugar deciden migrarlos a Android o iOS. Algunos programas son lo suficientemente maduros como para que su desarrollo ya se haya completado, y OpenGL ES nunca fue prioritario para estos programas. Hay un buen número de programas y juegos que perderemos por esta incompatibilidad. Pero no te preocupes, todavía existe una solución para ello y se llama GLshim.

¿Qué es GLshim?

GLshim es un empaquetador OpenGL para OpenGL ES, lo que significa que es posible traducir ciertas funciones OpenGL en funciones OpenGL ES, permitiendo a OpenGL utilizar la aceleración OpenGL ES para hacer las mismas (o similares) tareas. GLshim efectivamente permite ejecutar programas y juegos OpenGL con aceleración OpenGL ES en ODROID.

¿Cómo funciona GLshim?

Los programas OpenGL requieren de la librería libGL.so, que normalmente deriva bien del proveedor de la tarjeta gráfica o bien del proyecto Mesa, si los programas son compatibles con Mesa. Los programas cargan esta librería y utilizan las funciones declaradas en ésta para crear determinados efectos gráficos. La librería GLshim también se empaqueta como un archivo libGL.so, que proporciona algunas de las funciones que otros archivos libGL.so ofrecen.

En lugar de cargar el libGL.so normal como ocurre en Mesa, a los programas se les dice que carguen la librería GLshim. Esto se puede hacer bien precargando la librería, o diciéndole al programa donde está instalada la librería. También se puede hacer cuando compilas un programa.

Instalación

Si estas usando mis imágenes o repositorios, y tienes instalado mi paquete libgl-odroid, encontrarás la librería necesaria en /local/lib/usr/. De lo contrario, puede instalar GLshim en tu sistema añadiendo mi repositorio a tu imagen:

```
$ cd /etc/apt/sources.list.d/
$ wget http://oph.mdrjr.net/me-
veric/sources.lists/meveric-all-
testing.list
$ wget -O- http://oph.mdrjr.net/
meveric/meveric.asc | apt-key
```

```
add -
$ sudo apt-get update
$ sudo apt-get install libgl-
odroid libglues-odroid
```

O instala los paquetes glshim manualmente con los siguientes comandos:

```
$ cd ~/Downloads
$ mkdir glshim
$ cd glshim
$ wget http://oph.mdrjr.net/
meveric/other/freeorion/libgl-
odroid_20150922-1_armhf.deb
$ wget http://oph.mdrjr.net/me-
veric/other/freeorion/libglues-
odroid_20140903-1_armhf.deb
$ sudo apt-get install gdebi
$ sudo gdebi libglues-*.deb
$ sudo gdebi libgl-*.deb
```

Por último, vincula los drivers Mali (en el XU3 y XU4, utiliza libmali.so en lugar de libMali.so):

```
$ ln -sf /usr/lib/arm-linux-gnue-
abihf/mali-egl/libMali.so /usr/
lib/arm-linux-gnueabihf/libEGL.so
$ ln -sf /usr/lib/arm-linux-
gnueabihf/mali-egl/libMali.so /
usr/lib/arm-linux-gnueabihf/lib-
GLESv1_CM.so
$ ln -sf /usr/lib/arm-linux-
gnueabihf/mali-egl/libMali.so /
usr/lib/arm-linux-gnueabihf/lib-
GLESv2.so
```

Uso general

A modo de ejemplo, el siguiente comando ejecuta una demo con gráficos llamada “glxgears” utilizando GLshim:

```
$ LD_LIBRARY_PATH=/usr/local/lib
glxgears -fullscreen
```

Al incluir la variable `LD_LIBRARY_PATH`, puedes decirle al sistema dónde buscar las librerías necesarias. Normalmente esta es la mejor forma de usar GLshim. Se le dice a un programa que primero debe buscar todas las librerías requeridas en la carpeta `/usr/local/lib`, y si no logra encontrar una librería concreta, utilizará otras rutas definidas por las librerías. De esta manera, un programa puede cargar la librería GLshim desde `/usr/local/lib` y posteriormente cargar las otras librerías que necesita desde las rutas comunes del sistema.

Precargar GLshim

Con la variable `LD_PRELOAD`, puedes indicar al sistema que cargue una librería específica (o más de una) antes de iniciar una aplicación. De este modo, si un programa necesita una librería o determinadas funciones, utiliza la función proporcionada por la librería precargada en lugar de una librería enlazada. Esto es muy útil si el programa está codificado para utilizar una librería de una ruta específica y no carga cualquier otra ruta.

En el caso de que no funcione el método anterior de la variable `LD_LIBRARY_PATH`, esta es la mejor alternativa para cargar GLshim. El siguiente comando ejecuta la demo de gráficos “glxgears” precargando GLshim:

```
$ LD_PRELOAD=/usr/local/lib/lib-
GL.so glxgears -fullscreen
```

Compilar con GLshim

Cuando compilas un programa, éste se vincula a las librerías que necesita. Una de las librerías para OpenGL es libGL.so, como ya hemos mencionado. Esta se

incluye normalmente con el parámetro `-lGL`, que vincula la librería libGL.so instalada en tu sistema. En la mayoría de los casos en la plataforma ODROID, esta es la biblioteca libGL.so de MESA. Hay algunas formas de cambiar esto.

En primer lugar, puedes sustituir la entrada `-lGL` por la ruta de acceso completa de la librería GLshim (`/usr/local/lib/libGL.so`), lo cual funciona en la mayoría de las ocasiones. Otro método consiste en cambiar la ruta donde se supone que el programa busca las librerías. En los parámetros de vinculación, deberías encontrar algo similar a esto:

```
-Wl,-rpath,/usr/lib/arm-linux-
glibcabihf
```

Estos parámetros indican al programa que debe buscar las librerías en la carpeta estándar de tu sistema Linux, que es donde se encuentran los drivers Mesa. Puedes decirle que en lugar de buscar las librerías primero en `/usr/local/lib`, intente buscar antes en otras rutas. De este modo localizará la librería GLshim antes de que encuentre la librería Mesa, para ello cambia la entrada por lo siguiente:

```
-Wl,-rpath,/usr/local/lib
```

Otra opción es intentar vincular estáticamente libGL.a (ubicado en `/usr/local/lib`) en lugar de vincular dinámicamente libGL.so, de forma que todas las funciones que necesitas para ejecutar tu programa OpenGL formarán parte del programa en sí. La ventaja de esto es que si un programa funciona con una determinada versión de GLshim, y vinculas estáticamente esa versión, puedes estar seguro de que ese programa siempre va a funcionar sin importar la versión de GLshim instalada en el sistema.

Sin embargo, este método también tiene algunos inconvenientes. El tamaño de tu binario aumentará, lo cual no debería suponer un problema, pero cuantas más librerías vincules estáticamente, más grande será el binario resultante.

Para algunos programas, esto supone crear binarios con un tamaño de varios cientos de megabytes. Además, si se vincula GLshim estáticamente y una nueva versión de GLshim funciona mejor con tu programa, siempre se usará la versión más antigua, ya que está integrada en el programa, a pesar de contar con una versión más reciente de GLshim.

Además, este método sólo funciona si todas las funciones para OpenGL están incluidas en el código y no se activan a través de otras librerías como SDL. Las funciones que son utilizadas solo serán extraídas de la librería si la vinculas estáticamente, y si una función no está declarada en el programa, aunque sea generada dinámicamente desde SDL, el enlace estático no funcionará.

Compatibilidad

No se pueden ejecutar todos los programas OpenGL únicamente utilizando GLshim. Por ahora está limitado a OpenGL 1.x, mientras que OpenGL ha alcanzado la 4.5 en este momento (<http://bit.ly/1jUtj51>). Existen dos versiones disponibles con GLshim. Una es la del desarrollador original @lunixbochs, que empezó el trabajo sobre GLshim. Su trabajo se puede encontrar en su repositorio git en <http://bit.ly/1jcznWy>. Otro desarrollador llamado @ptitSeb ha mejorado la versión original de GLshim con su propia adaptación <http://bit.ly/1KAOBur>.

Ambas versiones funcionan con ODROID, pero son diferentes entre sí. @lunixbochs aspira a ser muy preciso, siguiendo el estándar OpenGL y la definición MESA, su trabajo tiene la intención de ser la versión más estable de GLshim. @ptitSeb intenta tener tantas funciones disponibles como le sea posible, de modo que tiene soporte para más de juego, pero al hacerlo, deja a un lado algunos otros juegos que si que funcionan con la versión de @lunixbochs

Para ayudar a decidir qué versión usar, @lunixbochs ha creado un documento que muestra sus avances en el desarrollo de GLshim, con las funciones

que ha ido implementando, <http://bit.ly/1Gqof2M>. Esta hoja de cálculo muestra que su versión está muy avanzada en lo que respecta al soporte OpenGL 1.x.

Las limitaciones de GLshim son evidentes, ya que sólo admiten funciones OpenGL 1.x en este momento y ni siquiera todas están disponibles aún. Esto significa que las funciones de OpenGL 2.x 3.x 4.x no son compatibles, y es en estas versiones donde se puede apreciar la verdadera “magia”, como son los sombreados y otros efectos “de fantasía”. GLshim está todavía limitado, y no permite ejecutar todos los programas

Figuras 1 y 2 - FreedroidRPG un juego RPG de estilo isométrico similar a Fallout o Baldurs Gate



Figuras 3 y 4 - Secret Mاريو Chronicles es un curioso clon de Super Mario que usa OpenGL para muchos efectos como la caída de las hojas



Figuras 5 y 6 - Hedgewars es un buen clon de Worms con exclusivas armas y muchos extras. Puedes incluso jugar en red o por Internet con tus amigos



Figuras 7 y 8 - Duke Nukem 3D sobre ODROID en alta definición (arriba), incluso con modelos poligonales 3D (abajo) se ejecuta muy bien, gracias a GLshim

OpenGL en tu sistema. Pero si un programa utiliza principalmente funciones OpenGL 1.x, hay una alta probabilidad de que pueda ejecutarse bajo GLshim y por lo tanto en ODROID.

Juegos clásicos

Puedes encontrar una selección de títulos que ya han sido publicados en los foros en <http://bit.ly/1RWWX5n> (tienes que iniciar sesión para verlos). Algunos ejemplos son OpenCPN, un software de cartografía y navegación GPS. Además de juegos como FreedroidRPG, Super Mario Chronicles y Hedgewars. La mayoría son títulos muy buenos.

Estos juegos se ejecutan muy rápido, y se puede apreciar que la aceleración 3D funciona bien para producir buenos gráficos. Los efectos de fondo, como la caída de las hojas, la nieve y los diferentes efectos de iluminación hacen que estos juegos sean muy divertidos. Otros juegos a parte de los creados para Linux también se ejecutan perfectamente. Puedes jugar a remakes de clásicos juegos de DOS y Windows que fueron exportados a OpenGL en ODROID, como Duke Nukem 3D, Shadow Warrior, e incluso Quake 2. Mientras que juegos como Quake 2 y Duke Nukem 3D llevan al límite a los dispositivos con la represen-



Figura 9 - Quake 2 con texturas de alta definición sobre un ODROID



Figuras 10 y 11 - Los juegos como Valyria Tear, aunque totalmente en 2D tiene impresionantes gráficos y efectos especiales, que son posibles gracias a OpenGL



Figuras 12 y 13 - Más efectos en Valyria Tear, remolinos de partículas (arriba) y brillantes efectos en ventanas y otros objetos luminosos (abajo) así como bocadillos transparentes

tación de polígonos 3D, muchos otros utilizan OpenGL para ejecutar efectos y gráficos mas suaves.

Valyria Tear

Valyria Tear es un juego de rol muy bueno, con excelentes gráficos y música. Todavía está en desarrollo, pero la versión 1.0 que está en mi repositorio permite jugar el primer episodio al completo. La portada y la breve introducción muestran los magníficos efectos que se pueden esperar del OpenGL. Las nubes flotantes, los rayos y los torbellinos girando son sólo algunos de los sorprendentes efectos de Valyria Tear. Aunque muchas funcionalidades todavía se están escribiendo, Valyria Tear ofrece una completa experiencia de juego, con mapas, menús y batallas totalmente funcionales, todo lo que puedes esperar de un clásico:

- ~50 mapas creados
- 18 32x32 Tilesets usados
- 2 personaje y medio jugables
- 19 enemigos muy diferentes
- Una docena de NPCs, habilidades, hechizos y diversos objetos

- Los rangos se pueden aplicar sobre los personajes y los enemigos
- Total compatibilidad con joystick
- Juego completamente traducido a 6 idiomas, y más que están en camino
- Tres temas de interfaz de usuarios
- Misiones principales y secundarias, tesoros y trampas ocultas
- Considerándolo todo, entre 7 y 10 horas de juego
- ~140.000 líneas de código C++ y script lua

Es un gran juego, aunque para la versión 1.1 los desarrolladores han cambiado a SDL 2 y OpenGL 2.0, así que puede que las próximas versiones del juego ya no funcionen en ODROID, puesto que GLshim aún no soporta OpenGL 2.0. Recientemente he podido conseguir algunos juegos más que funcionan,



Figura 14 - La portada de FreeOrion, un juego de estrategia muy complejo que ahora está disponible para ODROID

como FreeOrion, un juego inspirado en la serie Master of Orion.

Aquaria

Hace poco, he logrado ejecutar un juego comercial muy conocido en ODROID usando GLshim llamado Aquaria, fue lanzado en 2007 y es uno de los llamados juegos "indie". Este juego es realmente asombroso. La música es muy intensa, pudiéndola escuchar durante horas. Los gráficos son magníficos, con efectos muy buenos. Si no has oído hablar de este juego, deberías echar un vistazo a su trailer en <http://bit.ly/1QUQOpw>.

El juego es muy tranquilo y rico en diferentes entornos que se perciben



Figura 15 - Portada de Aquaria, donde ya se puede escuchar y sentir la magia y la belleza del juego

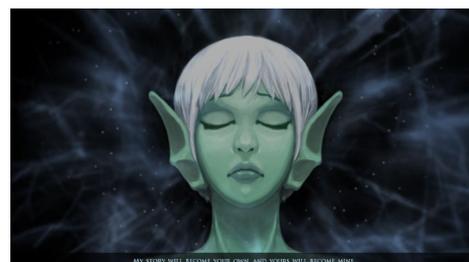


Figura 16 - Naija, el personaje principal de Aquaria, con una impresionante interpretación de voz de la hermosa Jenna Sharpe



Figuras 17 y 18 - Magnificas escenas del juego con trasfondos disipados y un montón de colores, diferentes tipos de efectos de agua, plantas y animales



Figura 19 - En momentos del juego incluso puedes llegar a la superficie y ver algo más que agua a tu alrededor. ¡Simplemente observa estos tonos tan asombrosos!

como nuevos mundos. Si te gustan los juegos de aventura y acción con un cierto estilo cómico, sin duda este juego está hecho para ti, y ahora también lo puedes ejecutar en tu ODROID.

Aquaria también ofrece diferentes mods, que se pueden descargar directamente desde el interior del juego.

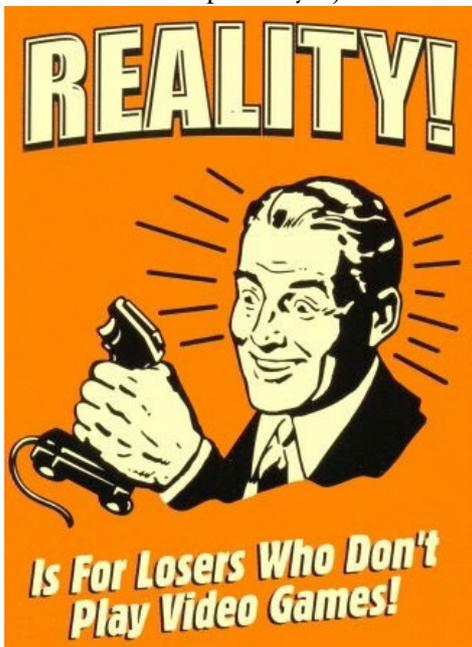
He ejecutado este juego en el U3 y XU4, y en ambos funciona muy bien incluso a 1080p, aunque en el XU4 se percibe más suave. Todas las imágenes de este artículo se han tomado directamente de un ODROID-XU4, para que puedas ver cómo se ve realmente el juego en la plataforma ODROID.

Conclusion

Espero que el proyecto GLshim continúe su desarrollo, como lo ha hecho hasta ahora. Los principales colaboradores @linuxbochs y @ptitSeb están haciendo un buen trabajo con este proyecto, y espero que seamos capaces de exportar muchos más juegos gracias a GLshim.

Preveo que pronto tendremos soporte para OpenGL ES 2.0, junto con algunas funciones OpenGL 2.x, podremos utilizar sombras y otros efectos curiosos a través GLshim en las placas ARM.

Seguiré buscando más juegos para exportarlos a ODROID utilizando GLshim, que incluiré en la sección de Juegos y emuladores de los foros ODROID en <http://bit.ly/1jUwR7u>.



WIKI DE LA COMUNIDAD

CONTRIBUYE A AMPLIAR LA BASE DE CONOCIMIENTO DE ODROID

por Rob Roy



Hardkernel ha puesto en marcha recientemente un gran recurso para los ODROIDians que deseen aportar sus conocimientos a una wiki de la comunidad, disponible en <http://wiki.odroid.in>.

Está hecha con la intención de complementar la wiki oficial de Hardkernel de <http://bit.ly/1R6D0gZ>, útil para que publiques tus consejos, enlaces a imágenes de la comunidad, proyectos y cualquier otra cosa que pueda ser beneficiosa para la comunidad de Hardkernel.

Si desea participar, haz clic en el botón “Request Account” en la parte superior derecha, e incluye tu nombre de usuario del foro ODROID en la sección “Personal Biography”. Para comentarios, preguntas y sugerencias relacionadas con la nueva wiki, por favor visite el hilo del foro original en <http://bit.ly/1QDMNoT>.

Welcome to the ODROID Support Page

This place is for the ODROID boards.

About

History

The ODROID means Open + Droid. It is a development platform for the hardware as well as the software.

Here is a brief history of ODROID.

- ODROID : The world first Android mobile game console development platform with S5PC100 (2009' Fall)
- ODROID-T : The world first Android 10.1" tablet development platform with Exynos3110 (2010' Spring)
- ODROID-S : An affordable Mobile development platform with Exynos3110 (2010' Summer)
- ODROID-7 : E-Book/CNS development platform with Exynos3110 (2010' Fall)
- ODROID-A : The world first Dual-core & 3G modem integrated tablet development platform with Exynos4210 (2011' Spring)
- ODROID-PC : Internet TV and Smart Set-top box development platform with Exynos4210 (2011' Winter)
- ODROID-A4 : Palm sized Handheld Mobile & Media player development platform with Exynos4210 (2012' Spring)
- ODROID-Q : The world first ARM Quad-Core integrated tablet development platform with Exynos4412 (2012' Summer)
- ODROID-X : The world lowest cost ARM Quad-Core development board with Exynos4412 (2012' Summer)
- ODROID-X2 : The upgrade version of ODROID-X with 1.7GHz Exynos4412 Prime and 2GB RAM (2012' Fall)
- ODROID-U2 : The upgrade version of ODROID-U with 1.7GHz Exynos4412 Prime and 2GB RAM (2012' Winter)
- ODROID-U3 : The world lowest cost ARM Octa-Core big.LITTLE board computer with Exynos5410 (2013' Summer)
- ODROID-U3 : The upgrade version of ODROID-U2 with 1.7GHz Exynos4412 Prime and 2GB RAM (2013' Winter)
- ODROID-XU3 : The world's first HMP enabled ARM Octa-Core big.LITTLE board computer with Exynos5422 (2014' Summer)

Table of Contents
• Welcome to the ODROID Support Page
• Page
• About
• History
• Getting Started
• References

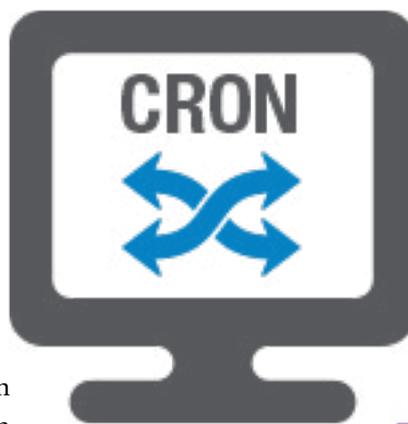
LFTP Y CRON

SINCRONIZA SERVIDORES CON FACILIDAD

por Andrew Ruggeri

Los ODROID y otras placas de desarrollo basadas en SoCs de alta gama se han utilizado desde hace un tiempo en diferentes tipos de servidores (servidores web, NAS, Bases de datos IoT, servidores multimedia, etc.) . En todos estos ejemplos, siempre es necesaria una funcionalidad que de alguna forma sea capaz de hacer copias de seguridad o una función de sincronización hacia o desde el propio servidor.

Esta guía se centrará en dos herramientas que te pueden ayudar con el proceso de sincronización.



¡Como en nuestra serie LVM, este artículo nos recuerda que la mejor estrategia es disponer siempre de una buena copia de seguridad!



LFTP: la primera herramienta que analizaremos será LFTP (<http://lftp.yar.ru/>), que es una herramienta de línea de comandos UNIX utilizada para transferir archivos a través de FTP, SFTP, HTTP, HTTPS y otros protocolos. LFTP tiene la singular función de que permite segmentar las descargas que, dependiendo del uso, puede aumentar bastante la velocidad de transferencia de los archivos.

CRON: es un programa UNIX muy práctico que puede ayudarte a programar scripts y comandos para que se ejecuten en determinados periodos de tiempo.

Como ya habrán adivinado, usaremos cron para activar una serie de comandos LFTP que nos ayudarán a sincronizar el servidor. En esta guía tendremos usaremos 'lftp' para descargar varias carpetas y archivos a/desde un servidor. Esta funcionalidad es útil en situaciones en las que respaldamos las tablas de un servidor de base de datos para guardarlas en local, nos conectamos desde algún servidor externo para vincularlas localmente a un HTPC, o simplemente cualquier escenario en el que necesitamos hacer una backup del nuevo contenido.

LFTP

LFTP es por sí misma una herramienta muy capacitada y útil. Vamos a suponer que el servidor remoto tiene un servidor SFTP ejecutándose en local. Ten en cuenta que LFTP soporta muchos protocolos a parte de SFTP, de modo que los siguientes pasos son similares para otros protocolos. Lo primero que tendremos que hacer es instalar LFTP vía apt-get:

```
sudo apt-get install lftp
```

Conceptos básicos

Para conectarte a un servidor SFTP con el nombre de usuario 'odroid' y una dirección IP como 192.168.0.101, en la ventana de terminal debes escribir lo siguiente:

```
lftp sftp://odroid@192.168.0.101
```

Bastante simple ¿no? Estamos ordenando a lftp que establezca una conexión SFTP con el nombre de usuario 'odroid', hacia un servidor ubicado en (@)192.168.0.101

Una vez conectado, te pedirá una contraseña. Si todo va bien, aparecerá algo similar a un prompt normal. No dispondrás de la totalidad de funciones bash, pero contarás con las más importantes, como cd, ls, mkdir, rm, mv, etc.

Para enviar comandos a la máquina local durante la sesión LFTP, añade al comando el signo '!', de modo que para cambiar entre directorios a nivel local el comando sería '!cd', y para imprimir el directorio de trabajo local el comando sería '!pwd'.

Descargas

Para descargar una copia de un archivo desde el servidor en el directorio de trabajo local, usaremos "pget" seguido del nombre de archivo como se muestra en el siguiente ejemplo y en la Figura 1 de la siguiente página:

```
pget myFile.dat
```

Utiliza mirror seguido del nombre del directorio para descargar un directorio. El siguiente comando descargará la carpeta 'test' y todo su contenido a el directorio de trabajo.

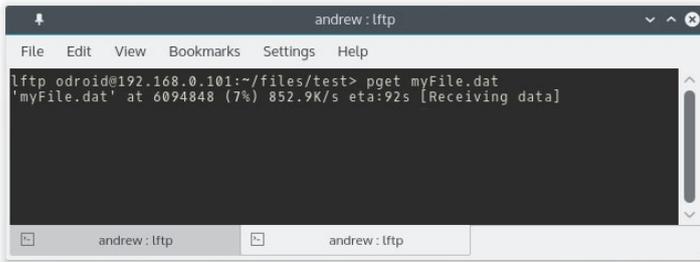


Figura 1 - Comando pget

```
mirror test
```

Una descarga o carga se puede detener con `ctrl+c`, se puede reanudar por donde se quedó con el argumento `'-c'`:

```
pget -c myFile.data
```

Subidas

Para cargar un simple archivo en el servidor, utiliza el comando `"put"` seguido del nombre de archivo como se muestra en el siguiente ejemplo y en la Figura 2:

```
put myFile.dat
```

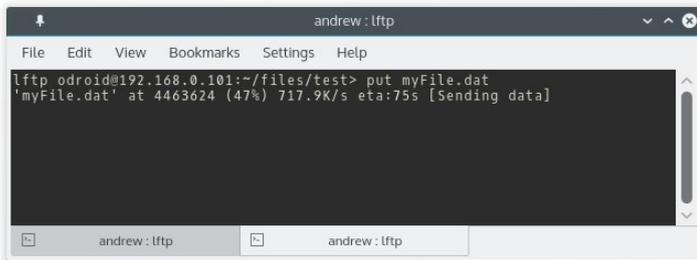


Figura 2 - Comando put

Para cargar un directorio, el comando sería `"mirror -R"` seguido del nombre del directorio local:

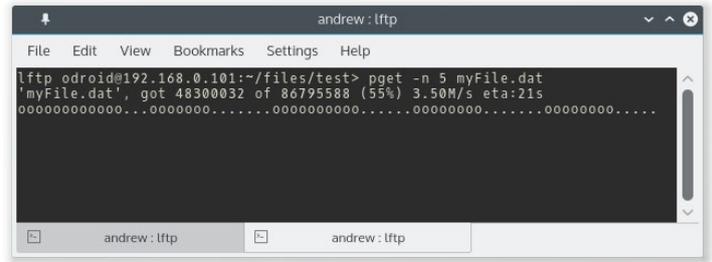
```
mirror -R test
```

Segmentos y descargas paralelas

Una característica muy importante de LFTP es la posibilidad de tener descargas paralelas y segmentadas que pueden acelerar bastante la velocidad de transferencia, sobre todo cuando estás moviendo datos a un servidor externo de tu red local.

Para un único archivo usa el comando `'wget'`, el argumento sería `'-n'` seguido del número de segmentos. Por ejemplo, para tener 5 segmentos usarías el comando que se muestra a continuación y en la Figura 3:

```
pget -n 5 myFile.dat
```



Figuras 3 - Comando wget con múltiples segmentos

Para transferir un directorio, utiliza `"mirror"` seguido de `"--usepget-n ="` y el número de segmentos para cada archivo. Por ejemplo, para transferir el directorio `"test"` en 7 segmentos deberías usar el siguiente comando:

```
mirror --use-pget-n=7 test
```

Con el argumento `'-n'` podemos descargar un único archivo en varias partes. Además, LFTP también nos permitirá descargar múltiples archivos al mismo tiempo. Cuando usamos el comando `mirror`, se descarga un único archivo dentro de ese directorio cada vez. Con el argumento `paralelo`, podemos tener el comando `mirror` descargando múltiples archivos a la vez.

Las descargas paralelas con `mirror` se activan añadiendo `"-P"` y el número de archivos a descargar. Por ejemplo, para descargar 3 archivos al mismo tiempo, deberías utilizar el siguiente comando:

```
mirror -P3 test
```

Para usar descargas segmentadas y paralelas al mismo tiempo utilizando una descarga con `mirror`, debes usar el siguiente comando:

```
mirror -P3 --use-pget-n=7 test
```

Tareas

LFLFTP te permite poner en cola tareas, así como tener tareas ejecutándose en segundo plano, lo que te permite ejecutar comandos en la sesión como `cd`, `ls`, `mkdir`, etc.

Para añadir una tarea de descarga o de subida a la cola, coloca `"queue"` justo antes del comando. Por ejemplo, para agregar el segmento `"pget"` a la cola, debes usar el siguiente comando:

```
queue pget -n 5 myFile.dat
```

Las tareas en cola se pueden activar/desactivar con los comandos `start` y `stop` de la siguiente forma.

```
queue stop
queue start
```

Para ver una lista de todas las tareas en cola utiliza el comando “jobs”, como se muestra en la Figura 4:

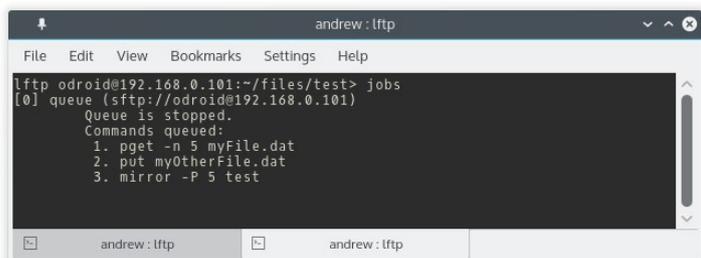


Figura 4 – Listado de tareas bajo LFTP

El comando “jobs” ofrece una gran cantidad de información útil, incluyendo el estado de la cola (parada en este caso), una lista de todas las tareas en cola (3 en este caso: pget, put, mirror) y las tareas activas actualmente (como la cola esta parada, no hay ninguna).

El comando ‘kill’ se usa para eliminar las tareas en cola, al cual le sigue el número de la tarea en concreto que se muestra en la pantalla de información de tareas. Por ejemplo, para eliminar la tarea “put myOtherFile.dat”, el comando sería:

```
kill 2
```

Marcadores

Los marcadores son sencilla forma de guardar y recuperar un nombre de usuario, contraseña, dirección de servidor remoto y el directorio del servidor remoto.

El primer paso, si así lo desea, es permitir que las contraseñas se guarden en los marcadores. Si no lo haces te pedirá que introduzcas la contraseña en cada ocasión. Los atributos tales como la dirección del servidor, nombre de usuario y directorio remoto sí que se guardarán. Para guardar las contraseñas en un marcador, utiliza el siguiente comando:

```
set bmk:save-passwords true
```

Ahora necesitarás crear un nuevo marcador de inicio de sesión para el servidor que quieres marcar. También tendrás que mover el directorio en el servidor que quieres configurar. Por ejemplo, con mi marcador quiero iniciar sesión automáticamente a mi servidor sftp en 192.168.0.101, con el nombre de usuario ‘odroid’, al directorio “~ /files/test”. Primero me conecto, luego hago “cd” a “files/test”. Después uso el comando “add” seguido del nombre del marcador, que en este ejemplo es “odtest”. Para este ejemplo, los comandos serían las siguientes:

```
lftp odroid@192.168.0.101:~> cd files/test/
lftp odroid@192.168.0.101:~/files/test/> bookmark\
add odtest
```

Para utilizar el marcador recién creado desde una sesión de terminal local, simplemente escribe ‘lftp’ seguido del nombre del marcador, como se muestra en la Figura 5:

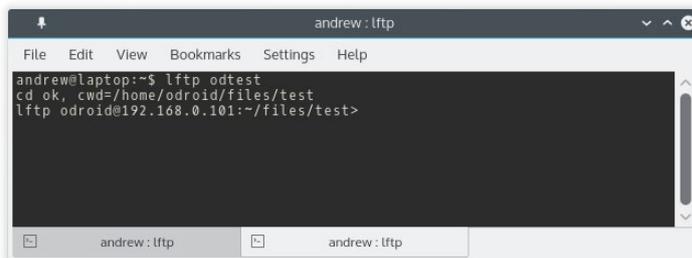


Figura 5 - Usando marcadores en LFTP

CRON

Cron es una gran herramienta bastante fácil de manejar, viene preinstalada en la mayoría de las versiones de Ubuntu para ODROID. Si Cron no está disponible, se puede instalar fácilmente con ‘apt-get’. Cron funciona editando el archivo de configuración crontab, que dictamina cuando un comando o script deben ejecutarse.

Para empezar a editar un crontab, usaremos el comando “crontab –e”. Esto abrirá el editor de texto predeterminado de tu sistema (que para Kubuntu sería ‘nano’) como se muestra en la Figura 6:

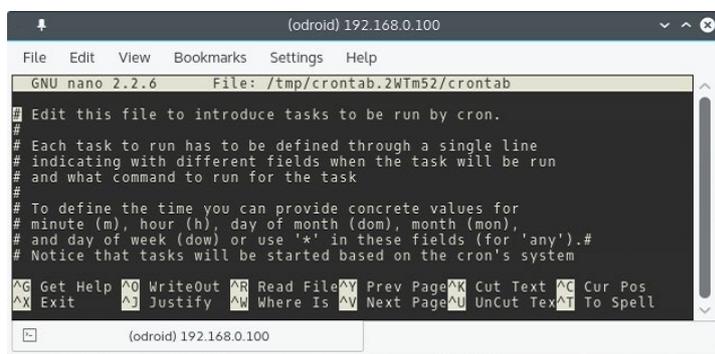


Figura 6 - El todopoderoso CRONTAB

En primer lugar desplázate hasta el final del archivo que es donde vamos a añadir nuestros comandos. Para agregar un comando que será ejecutado, tendremos que añadir una nueva fila. Cada fila tiene 6 partes, de izquierda a derecha como se muestra en la siguiente tabla:

VALUE RANGE					
0-59	0-23	1-31	1-12	0-7	36 2 * * 7 root /usr/local/sbin/backup.sh
					- COMMAND TO EXECUTE
					- EXECUTE COMMAND AS A USER ROOT
					- DAY OF WEEK: Sunday=0, Monday=1, Tuesday=2, Wednesday=3, Thursday=4, Friday=5, Saturday=6, Sunday=7
					- MONTH: January=1, February=2, March=3, April=4, May=5, June=6, July=7, August=8, September=9, October=10, November=11, December=12
					- DAY OF MONTH
					- HOUR
					- MINUTE

Cada una de las partes de la fila está separada por un espacio. Veamos un ejemplo para configurar el script `/home/odroid/foo.sh` para que se ejecute todos los días a las 10:42 am. Observa el siguiente comando:

```
42 10 * * * /home/odroid/foo.sh
```

En este comando, la primera parte ajusta el minuto a 42, la segunda parte ajusta la hora a las 10, y la última parte (parte 6) contiene la ruta del script. Las partes 3, 4 y 5 incluyen un `*`, que es uno de los símbolos especiales que se pueden utilizar. Por favor analiza la siguiente tabla y comandos de ejemplo.

Symbol	Example	Definition
*	*	All values
*/<X>	*/5	Every <X>, or skip every X
<X>-<Y>	5-7	A range from X to Y
<X>,<Y>,<Z>	2,5,6	Used list multiple values

```
# run every 10 min
*/10 * * * * /some/command.sh

# run 3 time each hour: X:00, X:15, and X:30
0,15,30 * * * * /some/command.sh

# run at the start of every hour from 6 - 8am
0 6-8 * * * * /some/command.sh
```

Otro comando muy útil que debes conocer es `crontab -l`, que muestra el archivo de configuración crontab al completo.

Script Batch

El siguiente ejemplo es un script batch que utiliza LFTP para conectarse a un servidor guardado. Este script alterna entre cada carpeta de la lista `'sync_folders'`, y descarga cualquier archivo nuevo que encuentre. El script también permitirá que se ejecute una única instancia del script batch cada vez, lo cual se selecciona creando un `'lock_file'` en `/temp`. Esto no es igual para todos los script y tendrás que modificarlo para adaptarlo a tus necesidades.

```
#!/bin/bash

# bookmark set to server with username and password
bookmarkName="odroid"

# directories to synced
remote_dir='files/data/'
local_dir="/home/andrew/backup/data/"
```

```
echo "Starting $bookmarkName Sync"
# -----
# Create a list of only the directories to be synced
# -----
declare -a sync_folders=(
"Eagle Projects"
"Images/Diagrams"
"Images/Graphs"
"MongoDB Backups"
"Thesis"
"Test Logs"
"Test Data")

# create lock file
base_name="$(basename "$0")"
lock_file="/tmp/$base_name.lock"
trap "rm -f $lock_file" SIGINT SIGTERM
if [ -e "$lock_file" ]
then
    # Only allow one instance of this script to run
    at a time
    echo "$base_name is running already."
    exit
else
    touch "$lock_file"
    for current_folder in "${sync_folders[@]}"; do
        cd "$local_dir$current_folder"
        lftp $bookmarkName << EOF
        echo "Connected to $bookmarkName"
        cd "$remote_dir"
        set mirror:use-pget-n 9
        mirror -c -P5 "$current_folder"/.
        echo done
        quit
        EOF
    done
    rm -f "$lock_file"
    trap - SIGINT SIGTERM
    exit
fi
```



CREAR UN FILTRO DE SPAM A MEDIDA

RECUPERA EL CONTROL DE TU BANDEJA DE ENTRADA

por Pascal Pucholt

El “Spam” o “correo no deseado en masa” puede superar con creces al verdadero “ham” o “correo deseado” que se envía a tu dirección de email. Si tienes tu propio dominio y también los utilizas para tus correos electrónicos, puede que tengas que soportar un servidor de correo que no hace un buen trabajo con la gestión del spam. Para solucionar esto, podrías enviar todos los correos entrantes a Google o servicio similar y dejar que ellos se encarguen del filtrado (y usar tus email para mostrarte anuncios “relevante”, además de almacenarlos en su base de datos para que la NSA y otros tengan acceso a los mismos). Sin embargo, puedes convertirte en un experto y dejar que tu ODROID se encargue del problema de spam y así tener tus correos electrónicos bajo tu propio control.

En este artículo te presento un método para filtrar el spam de tu correo electrónico personal con un servidor de correo. El servidor de correo en sí mismo no se ejecuta en tu equipo ODROID. Este método se basa en el protocolo IMAP y utiliza SpamAssassin, un software de detección de spam ampliamente utilizado. De modo que continúa siendo de aplicación universal, siempre y cuando tengas acceso a tu servidor de correo electrónico a través de IMAP. Usar SpamAssassin te permite separar el correo deseado del spam.

En mis pruebas, el filtrado redujo drásticamente la cantidad de spam recibido en mi bandeja de entrada desde varios cientos de correos al día a tan sólo unos cuantos, y espero que la detección sea aún mejor con más práctica.



No tener que lidiar con el SPAM es todo un sueño, ¿Verdad?

La Figura 1 describe el filtrado de spam para prevenir que un email no clasificado llegue a tu servidor, de modo que sólo leas los correos deseados (verde) en cualquier dispositivo conectado al servidor de correo electrónico.

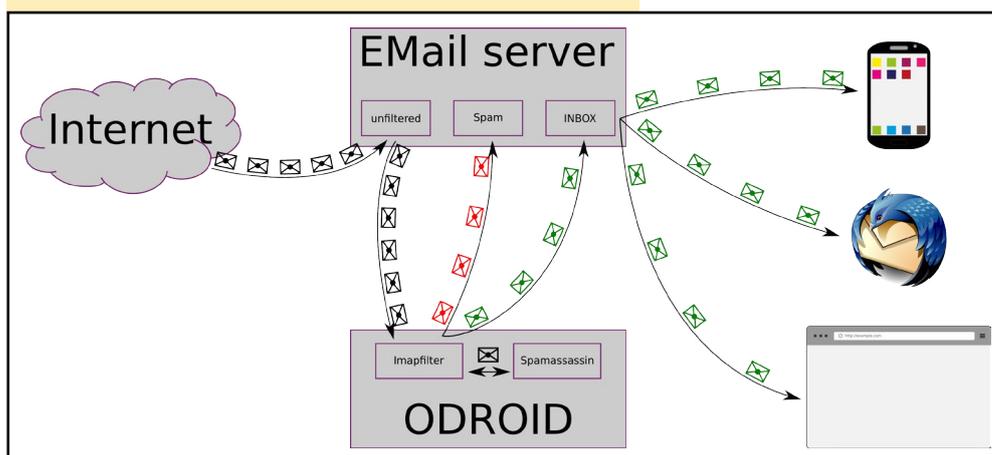
Requerimientos

- Un ODROID (yo uso un C1) que ejecute una imagen de Ubuntu
- Un servidor de correo electrónico al que puedas acceder por IMAP

Preparar tu servidor de correo

Necesitas pedir a tu servidor de correo electrónico que envíe todos los correos entrantes a una carpeta específica. Simplemente nómbrala como “No filtrados”. Primero tendrás que crear esta carpeta en el servidor, ya sea mediante un cliente de correo como Thunderbird o a través de una interfaz web. Luego, tendrá que configurar una “regla” en el servidor para reenviar todo el correo entrante a esa carpeta. El método para configurar la regla dependerá del software de servidor de correo electrónico que estás utilizando, y puede que necesites consultar la documentación de su servidor. En la mayoría de los sistemas, la opción que necesitas suele llamarse ‘filtrado’ o algo similar.

Figura 1: Representación esquemática del filtrado de spam



Instalar el software

Este filtro de spam se basa en un paquete de software llamado IMAPfilter que se comunica con tu servidor de correo electrónico, y es SpamAssassin el que hace el filtrado real. Estos paquetes se pueden instalar a través del gestor de paquetes de Ubuntu con el comando:

```
sudo apt-get install imapfilter spamassassin lua-posix
```

Los archivos de configuración que se proporcionan en este artículo han sido utilizados en un entorno con una cuenta de usuario específica y baja la cual se ejecutan los programas. Para crear esta cuenta, ejecuta el comando:

```
sudo adduser spamfilter
```

y aparecerá un script solicitando una serie de datos. Aquí es donde además configuras la contraseña para la nueva cuenta.

Después, puedes iniciar sesión en tu ordenador ODROID usando esta cuenta, y descargar los archivos de configuración desde github con los siguientes comandos:

```
sudo su spamfilter
git clone --recursive https://github.com/tux2000/imapfilter-tools.git ~/.imapfilter
```

Para incluir los detalles de tu servidor de correo electrónico, tendrás que crear una nueva plantilla de configuración de la cuenta. El último comando de los que aparece a continuación hace que el archivo solo sea legible por esa cuenta de usuario, de este modo protegeras tus credenciales del resto de usuarios:

```
cd ~/.imapfilter
cp accounts.lua.sample accounts.lua
nano accounts.lua
chmod 600 accounts.lua
```

En este archivo necesitas añadir la dirección de tu servidor de correo electrónico, tu nombre de usuario y tu contraseña.

La plantilla debería parecerse a esto:

```
account1 = IMAP {
    server = 'your.mail.server.here',
    username = 'user',
    password = 'password',
    ssl = 'ssl3',
}
```

En este punto, spamfilter ya debería estar funcionando. Puedes probarlo ejecutando “imapfilter -v” desde /home/spamfilter. Si hay problemas para conectar a tu servidor de correo elec-

trónico, recibirás mensajes de error y el programa se bloqueará. Cuando hayas solucionado todos los problemas, el programa debería entrar en modo demonio y estará listo para usarse.

Para que el spamfilter se inicie automáticamente en el arranque del sistema (y para que se reinicie si hay fallos debido a errores de comunicación con el servidor), necesitarás copiar el archivo “imapfilter.config” a la carpeta init con el siguiente comando:

```
sudo cp /home/spamfilter/.imapfilter/imapfilter.conf /etc/init/
```

Tras esto, podrás iniciar el demonio spamfilter en el sistema con este comando:

```
sudo service imapfilter start
```

El demonio registrará sus resultados en /var/log/upstart/imapfilter.log, podrás echarles un vistazo con el comando:

```
sudo tail -f /var/log/upstart/imapfilter.log
```

Preparar el filtro de spam y ajustar parámetros

SpamAssassin necesita al menos 200 mensajes “ham” y 200 “spam” para habilitar el filtrado bayesiano. Coloca más de 200 mensajes de spam en la carpeta de Spam/ False Negatives y más de 200 mensajes “ham” en la carpeta Spam/False Positives en tu servidor de correo electrónico y IMAPfilter harán el resto.

Para modificar el comportamiento de SpamAssassin, puede cambiar los parámetros en /home/spamfilter/.spamassassin/user_prefs. Añada tus propias puntuaciones para cambiar el peso con el que se decide si un correo es spam o no, o cambia el nivel de clasificación modificando el valor ‘required_score’. En mi caso reduje el valor ‘required_score’ por defecto de 5 a 4,2 de modo que el filtro Bayes puede casi por sí solo clasificar un mensaje de spam cuando está muy seguro de la clasificación. Incluso con esta configuración, aún tengo que recibir un falso positivo para clasificar correos electrónicos como spam. La Documentación sobre los parámetros que puedes utilizar para configurar el archivo user_prefs la puedes encontrar en <http://bit.ly/1iVLeIv>

Compartir Hash: No estás sólo

Los spammers generalmente quieren distribuir sus mensajes lo más ampliamente posible. Suelen seguir un enfoque muy simple, enviar el mismo mensaje a millones de direcciones. Así que lo más probable es que tú y yo tengamos exactamente el mismo correo de spam. Esta característica se puede utilizar para detectar el spam, incluso si ha logrado pasar todos los filtros basados en patrones implementados en SpamAssassin. La idea

básica es utilizar los informes de spam de muchos usuarios que han sido almacenados en una base de datos central. SpamAssassin chequea esta base de datos por cada correo electrónico. Dos de esas bases de datos que son fáciles de instalar en el sistema ODROID son 'pyzor' y 'razor'.

Cómo usuario administrador, instala los programas para comprobar la base de datos con el siguiente comando:

```
sudo apt-get install pyzor razor
```

Luego, tendrás que cambiar de nuevo a tu usuario de spamfilter:

```
sudo su spamfilter
```

Para configurar la base de datos Pyzor, necesitas ejecutar el comando:

```
pyzor discover
```

que instalará una lista de servidores para conectarse.

Para comprobar si SpamAssassin está realmente usando pyzor, puedes ejecutar el comando:

```
echo "test" | spamassassin -D pyzor 2>&1 | less
```

El resultado debería ser algo similar a esto:

```
Oct  4 12:15:33.018 [8603] dbg: pyzor: network tests
on, attempting Pyzor
Oct  4 12:15:40.866 [8603] dbg: pyzor: pyzor is
available: /usr/bin/pyzor
Oct  4 12:15:40.869 [8603] dbg: pyzor: opening pipe:
/usr/bin/pyzor check < /tmp/.spamassassin8603t9Q0e9t-
mp
Oct  4 12:15:43.099 [8603] dbg: pyzor: [8657] fin-
ished: exit 1
```

Del mismo modo, configura el programa razor con los comandos:

```
razor-admin -create
razor-admin -register
```

y pruébalo con el comando

```
echo "test" | spamassassin -D razor2 2>&1 | less
```

El resultado debería ser:

```
Oct  4 12:17:11.738 [8662] dbg: razor2: razor2 is
```

```
available, version 2.84
Razor-Log: Computed razorhome from env: /home/spamfil-
ter/.razor
Razor-Log: Found razorhome: /home/spamfilter/.razor
Razor-Log: No /home/spamfilter/.razor/razor-agent.conf
found, skipping.
Razor-Log: read_file: 2 items read from /etc/razor/
razor-agent.conf
```

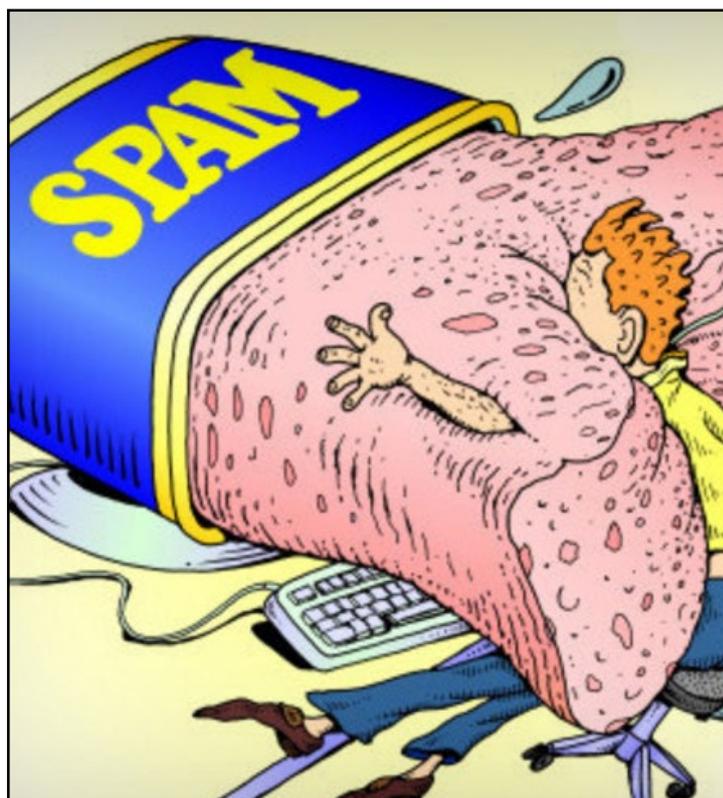
Si revisas el spam almacenado en tu carpeta de spam del servidor, verás que algunos de los mails también fueron localizados en las bases de datos pyzor y/o razor.

Conclusión

Con este sistema, puedes montar tu propio servidor de correo electrónico con todas las ventajas que ello supone. Tu ODROID no almacenará el servidor de correo en sí, dando lugar a un mejor rendimiento de tu conexión a Internet y del hardware. El filtrado de spam se hace de forma independiente a tu sistema ODROID, lo cual te permitirá ajustar los parámetros que necesites y así evitar que tus correos electrónicos sean detectados como spam, y al mismo tiempo eliminar todo lo que no quieres que aparezca en la bandeja de entrada.

Agradecimientos

La configuración del IMAPfilter que facilito en este artículo está basada en gran medida en el trabajo de Kyle Manna: (ref: <https://github.com/kylemanna>)



¡A partir de ahora, esto no te va a pasar nunca más!

HIFI-SHIELD PARA EL ODROID-C1+

HAZ QUE TU ODROID SUENE A LO GRANDE

por Justin Lee

Hemos desarrollado un nuevo producto, específico para los aficionados a la música llamado HiFi-Shield. Es un componente de sonido avanzado que incrementa la relación señal-ruido y reduce el nivel de ruido. Está disponible en la tienda de Hardkernel en <http://bit.ly/1M6UIXY> por 39\$. Lo ofrecemos como un componente adicional al ODROID-C1+, ya que hasta ahora, cuando queríamos disfrutar de música digital en alta definición en nuestro ODROID, necesitábamos conectar un DAC USB externo como este:

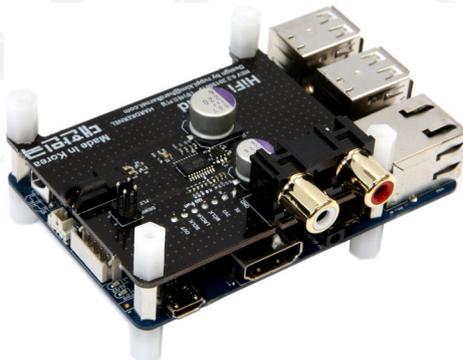
CPU -> USB Bus -> Cable USB -> Convertidor USB a I2S -> I2S -> DAC -> Amplificador -> Altavoces

Se trata de un proceso bastante largo, ¿no? Al que hay que añadir todas las cuestiones internas que afectan a las interfaces de conexión, y que pueden dar lugar a diversos problemas. El ruido de fondo, los ruidos de conexión, la inestabilidad del reloj en cascada y los complejos sistemas de alimentación pueden afectar a la calidad del sonido. Analiza este escenario en su lugar:

CPU -> I2S -> DAC -> Amplificador -> Altavoces

Esta configuración es mucho más simple y el sonido se escucha mejor. Sin embargo, podemos hacer que esta cadena sea aún más corta usando la configuración ODROID-C1+ I2S DAC. Sus ventajas son:

- Ruta de señal que evita interfaces adicionales, como USB y S/PDIF
- No usamos el problemático bus USB, que puede causar clics y chasquidos intermitentes.
- Excelente y económica configuración de sonido



HiFi-Shield para el ODROID-C1+ está diseñado para los aficionados a la música moderna.

Especificaciones técnicas

- El DAC usa un chip PCM5102 de alta gama de la compañía Burr-Brown (ahora parte de TI) que utiliza la interfaz I2S. Es compatible con los formatos de audio de 16, 24, 32 bits con mínima distorsión (-93dB) y una dinámica ideal (110dB+), además de una sorprendente frecuencia de muestreo de 384kHz.
- Los puertos de salida incluyen terminales RCA estéreo enchapados en oro y conector de audio de 3,5 mm
- Un regulador de pérdida del ruido ultra bajo se complementa con dos condensadores sólidos para el suministro de energía, lo que reduce significativamente el ruido de la fuente de alimentación y aumenta significativamente la relación señal-ruido.
- La interfaz I2S permite la decodificación directa de la entrada digital a la salida analógica mediante la sincronización de reloj maestra.
- La superficie de PCB incluye un revestimiento de oro en la parte superior de 2oz de cobre, lo que garantiza la continuidad de la señal y



reduce la reflexión y la refracción de la señal.

Calidad del audio

Algunos DACs USB de alta gama cuestan más de 1.000\$, los cuales tienen un rango dinámico inferior y más ruido en la señal que el sistema C1 + HiFi-Shield. Para analizar la calidad de audio del HiFi-Shield, hemos usado un analizador de audio y hemos invertido cerca de 10,000\$ para comprar un equipo profesional que permite analizar el audio con precisión. Con este equipo, hemos demostrado la calidad de la señal del



Analizador de audio interno de Hardkernel

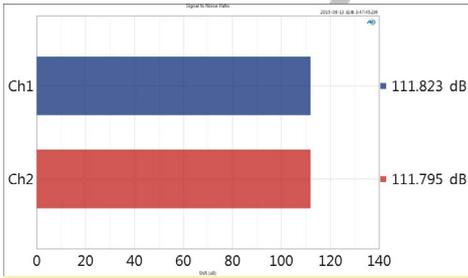
HiFi-Shield y el diseño innovador de la PCB a todos los aficionados a la música.

La relación señal-ruido (SNR) es superior a los 110 dB y el nivel de ruido es inferior a -107dB como muestra los gráficos de la siguiente página, de modo que el rango dinámico está cercano a los 110dB, un valor realmente asombroso.

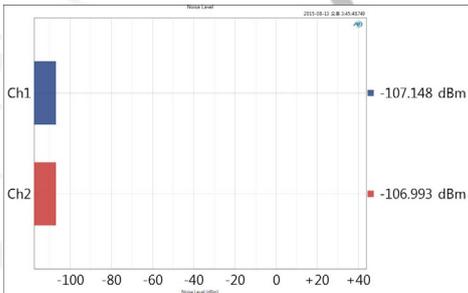
Hardware

Existen dos tecnologías responsables de la implementación del audio de alta calidad en el HiFi-Shield:

El nivel de oscilación de salida debe ser superior a 2 voltios RMS para alcan-



Relación señal-ruido del Hi-Fi Shield (cuanto mayor mejor)



Nivel de salida del ruido del Hi-Fi Shield (cuanto menor mejor)

zar 110 dB de rango dinámico. La salida a tierra de 2Voltios RMS se implementa con el multiplicador de tensión negativo. TI lo denomina tecnología "DirectPath".

Un regulador de pérdida del ruido ultra bajo se acopla con dos condensadores sólidos a la fuente de alimentación, reduciendo significativamente el ruido de ésta y aumentando bastante la relación señal-ruido. La interfaz I2S permite la decodificación directa de la entrada digital a la salida analógica usando la sincronización de reloj maestro.

Está listo para funcionar como un sistema de reproducción de sonido de alta fidelidad simplemente añadiendo un amplificador con altavoces. Creative Gigaworks T20 es un buen sistema de

Un sistema de altavoces Creative Gigaworks T20 para el HiFi-Shield



También puedes utilizar un amplificador de Clase D con altavoces de estantería

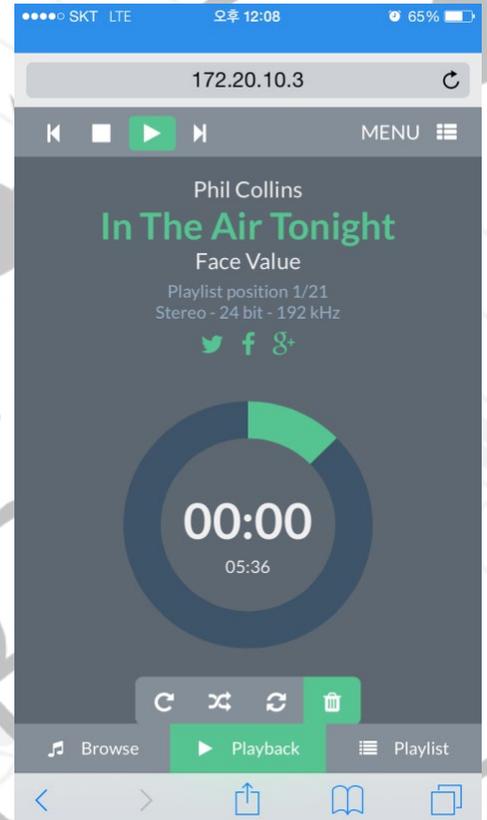
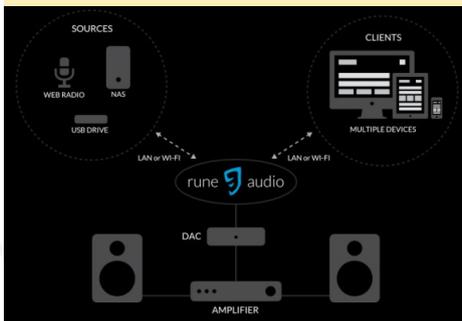
altavoces de nivel básico para disfrutar del audio de alta definición con un precio relativamente asequible. Un pequeño amplificador de Clase D con sistema de altavoces de estantería también valdría.

Software

Existen dos Sistemas Operativos muy conocidos dedicados al sonido de alta fidelidad: Volumio y RuneAudio. Puedes utilizarlos en tu sala de estar, en tu coche o en el trabajo. Pueden reproducir tu biblioteca de música digital desde unidades USB locales o montadas en red (NAS) y también permite hacer streaming vía Internet, como la radio web. Puedes controlar la reproducción de música a través de tu smartphone, tablet o PC con una interfaz web muy amigable. Ten en cuenta que HiFi-Shield sólo funciona bajo Linux, los drivers para Android todavía no están disponibles.

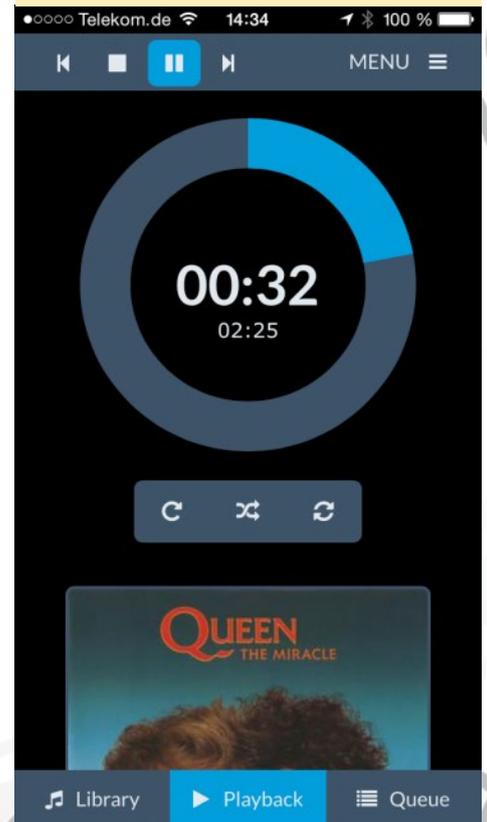
Debido a las mejoras realizadas en el ODROID-C1+ para que soporte audio de alta fidelidad, el HiFi-Shield no es compatible con el ODROID-C1 original. Sin embargo, si quieres utilizar un ODROID-C1 con un sistema de sonido envolvente 5.1, hay un adaptador USB-SPDIF disponible en <http://bit.ly/1NUsxCa>.

Típica configuración del RuneAudio



Volumio ofrece apps para iPhone y Android y recientemente ha anunciado que han agregado el ODROID-C1+ a su lista de placas soportadas (<http://bit.ly/1Oec783>)

RuneAudio se ha asociado con Hardkernel para asegurar que el ODROID-C1+ es 100% compatible con la última versión cliente de RuneAudio



LIBERAR CON CUIDADO HANDBRAKE

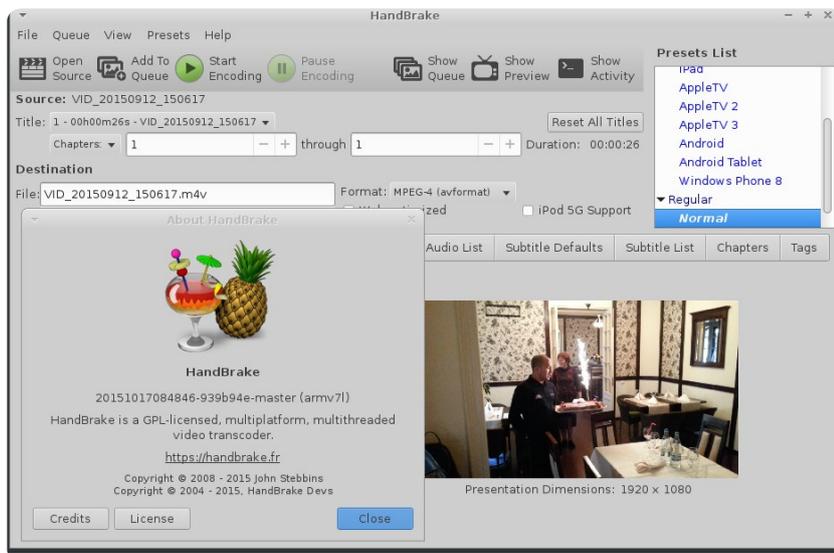
TRANSCODIFICA VIDEOS EN CUALQUIER FORMATO

por Adrian Popa

HandBrake es una herramienta para convertir videos desde casi cualquier formato a varios codecs modernos compatibles. Los formatos de salida incluyen MP4 y MKV como contenedores con H264, H265, MPEG 2 Theora y V8 como codecs de vídeo y AAC/AC3/MP3/Vorbis/FLAC como codecs de audio. Viene con un conjunto de ajustes preestablecidos para convertir a formatos compatibles con dispositivos específicos como Apple y Android, aunque puedes crear tus propias configuraciones para adaptarlos a tus necesidades. Aunque los ODROID no están diseñados para transcodificar vídeo, lo pueden hacer si te encuentras apurado. Desafortunadamente, no existe una versión de Handbrake pre-empaquetada para las plataformas ARM, por lo que es necesario compilarla desde el código fuente.

Cuando empecé a escribir este artículo, tenía la intención de que fuese una introducción a la configuración y a los problemas de compilación más comunes, y qué hacer para solucionarlos. Sin embargo, al parecer todos los problemas que encontré se debían a sólo dos causas. En primer lugar, encontré fallos en el archivo Léame y observé que faltaban dependencias. Además, con la ayuda del programador principal del HandBrake, fui capaz de encontrar una manera rápida y sencilla de resolver los problemas instalando un paquete que faltaba en mi sistema Ubuntu, lo cual hizo que todos los problemas desaparecieran.

Para compilar HandBrake desde la fuente, primero tienes que conseguir la



última versión de GitHub, lo puedes hacer usando un usuario normal no-root en Ubuntu:

```
$ git clone https://github.com/HandBrake/HandBrake.git
$ cd HandBrake
```

Llegado a este punto, es muy útil leer las instrucciones de compilación para ver lo que tienes que hacer, están disponible en doc/BUILD-Linux:

```
$ less doc/BUILD-Linux
```

Puede instalar la mayoría de los paquetes necesarios y compilar el entorno ejecutando este comando:

```
$ sudo apt-get install subversion
cmake yasm build-essential auto-
conf libtool \
zlib1g-dev libbz2-dev li-
bogg-dev libtheora-dev libvorbis-
dev \
libsamplerate-dev libxml2-
dev libfribidi-dev libfontconfig-
dev \
libfontconfig1-dev libass-
dev libbmp3lame-dev libx264-dev
libjansson-dev \
intltool libglib2.0-dev
libdbus-glib-1-dev libgtk-3-dev
```

```
libgudev-1.0-dev \
libwebkitgtk-3.0-dev libno-
tify-dev libgstreamer1.0-dev \
libgstreamer-plugins-
base1.0-dev libappindicator-dev
```

Además, necesitarás libtool-bin, que no se especifica claramente en las instrucciones pero que te puede causar muchos quebraderos de cabeza si no lo instalas, tal y como se describe en <http://bit.ly/1RUrmkW>.

```
$ sudo apt-get install libtool-
bin
```

Ahora, estás listos para compilar. En primer lugar ejecuta el script de configuración para seleccionar las opciones que deseas compilar:

```
$ ./configure --disable-gtk-up-
date-checks --enable-x265 \
--enable-fdk-aac --enable-
libav-aac --launch-jobs=6
```

Esta línea configurará HandBrake con la interfaz GTK y el codec X265, y dos implementaciones AAC. Los otros codecs ya están integrados. Si sólo deseas la versión CLI, también puede añadir `--disable-gtk` a las opciones. Ahora, es el momento de ejecutar `make`:

	Intel(R) Core(TM) i7-3612QM CPU @ 2.10GHz x8, Fujitsu-Siemens S792 laptop	ODROID-XU3 with interactive governor, maximum frequency on all cores	ODROID-XU3 with powersave governor, maximum frequency of <u>600MHz</u>
Average FPS during first pass	36	7.5	1.5
Maximum temperature	N/A	94C	65C
Total encoding time for a 26s video	1m5.236s	4m15.786s	29m1.483s
Price	1600€	158€	158€

Tabla 1 – Comparando pruebas de rendimiento de codificación

```
$ cd build
$ make
```

Cuando make haya finalizado (es de esperar sin errores), tienes dos opciones:

- sudo make install - copiará los archivos pertinentes a /usr/local/bin e instalará el software
- sudo checkinstall – te hará algunas preguntas y luego creará e instalará un paquete deb. Vamos a utilizar esta opción ya que es la propia de Debian

```
$ sudo checkinstall -D
--install=yes --pkgname=handbrake
\
--provides=handbrake --requires
=libtheora,libogg,libvorbis,libsa
mplerate,libxml2,\
libfribidi,libfreetype6,libfont
config1,libass,libmp3lame,libx264,
libjansson,\
libglib2.0,libgtk-
3,libgudev-1.0,libwebkitgtk-
3.0,libnotify,libstreamer1.0,\
libstreamer-plugins-
base1.0,libappindicator
--showinstall=yes --strip=yes
```

Ahora, el paquete handbrake debería estar instalado en tu sistema, proporcionando /usr/local/bin/HandBrakeCLI para tus necesidades de programación y /usr/local/bin/ghb para la interfaz GTK. Si todo ha ido bien, puedes eliminar el

directorio HandBrake y reutilizar el espacio usado. Si compilar desde la fuente no está entre tus preferencias, puedes intentar instalar el paquete ya compilado desde <http://bit.ly/1M5SDSm>, que debería funcionar en Ubuntu 15.04. Si te quedas atascado puedes pedir ayuda en el hilo de soporte <http://bit.ly/1MC4Atx>.

En términos de rendimiento, HandBrake solo utilizar la CPU para la codificación de vídeo (incluso usa instrucciones NEON si están disponible). Comparándolo con otros sistemas, la Tabla 1 muestra lo que podemos esperar cuando codificamos un mp4 1080p con x264, dos passes, sonido AAC, x264-preset=slow y x264-profile=high. Si codificas video con una cierta frecuencia, asegúrate que tu placa no funcione durante mucho tiempo a altas temperaturas. La instalación de cualquier sistema de refrigeración o simplemente reduciendo la frecuencia máxima hara que no se sobrecaliente.

Como era de esperar, ODROID no se puede comparar con su hermano mayor (Intel) en términos de rendimiento, pero es 10 veces más barato siendo tan sólo 4 veces más lento, de modo que es una buena solución si no tienes prisa. Yo utilizo HandBrake con ODROID para convertir vídeos con alta tasa de bits (20Mbps) tomados con el teléfono a una tasa de bits más manejable (6 Mbps), manteniendo la misma calidad, todo con un trabajo cron automatizado. ¡Diviértete con la transcodificación!



ODROID Magazine ahora esta en Reddit!



ODROID Talk Subreddit
<http://www.reddit.com/r/odroid>



CLOUDSHELL CON UN ODROID-XU4 COMO SERVIDOR DOMESTICO

UN DISPOSITIVO PERSONAL EN LA NUBE TODO EN UNO

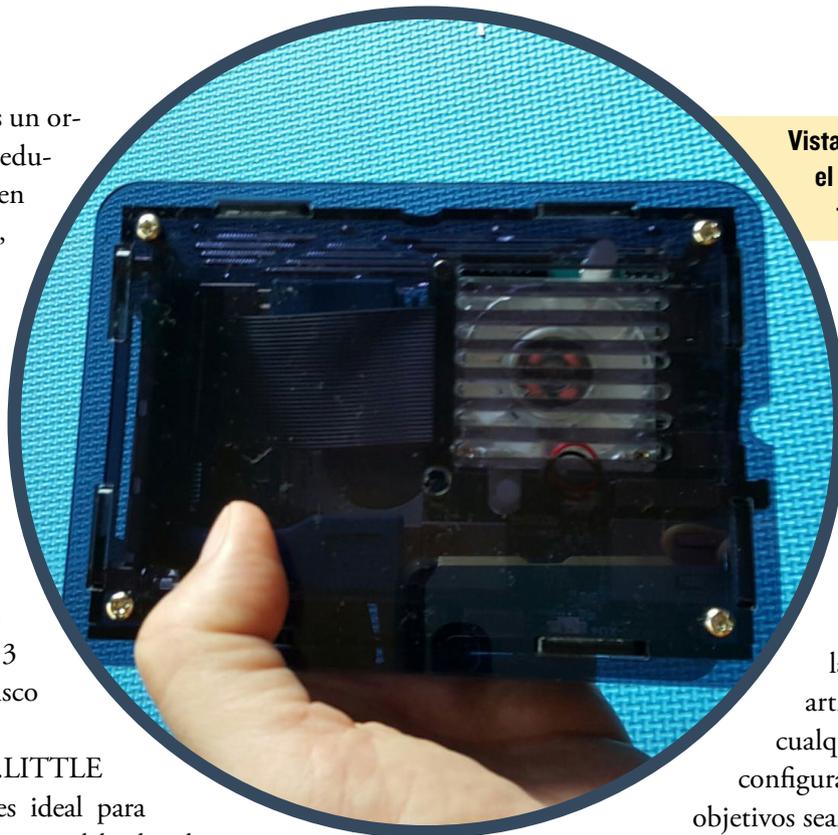
por Andy Yuen

El ODROID-XU4 es un ordenador de placa reducida ARM basado en el Samsung Exynos 5422, que cuenta con los núcleos Cortex-A7 a 1.4 Ghz y Cortex-A15 a 2 GHz, además de 2 GB de memoria. El XU4 sustituye al XU3 y XU3 Lite originales. CloudShell es una caja compacta para el XU4 con una TFT LCD de 2,2 pulgadas y un adaptador USB 3 a SATA para conectar un disco SATA de 2,5 pulgadas.

La tecnología ARM big.LITTLE usada en el Exynos5422 es ideal para montar un servidor en casa por el hecho de que cuando la carga de la CPU es baja, se utilizan los núcleos A7 más eficientes y cuando la carga es alta, pasas a utilizarse los núcleos A15 de mayor rendimiento. Dependiendo de la carga se puede usar 1 núcleo, más de 1 o todos los núcleos a la vez. Esto se traduce en un ahorro de energía a largo plazo.

Cuando la gente habla de servidores domesticos, por lo general suelen referirse a Kodi que actúa a modo de servidor multimedia. Para eso puedo utilizar un televisor inteligente o un decodificador. Mis necesidades de servidor en casa son algo más específicas, tiene que permitir acceder desde Internet a los siguientes servicios que se ejecutan en el mismo:

- Servidor WordPress para ejecutar mi blog
- Servidor SSH para acceso remoto
- Una Aplicación demo que gestione reglas para que los usuarios interactúen
- Ejecutar un servidor NFS
- Servir de nodo principal para mi clúster informático doméstico (no expuesto a Internet)



Vista superior de la CloudShell, el accesorio perfecto para tu fiel XU4

Las siguientes secciones describen cómo configurar el ODROID-XU4 como servidor doméstico, incluye enlaces con información útil sobre el procedimiento de configuración, y detalles de los puntos más sensibles de la instalación y uso. Espero que este artículo le sea de utilidad a cualquiera que esté pensando en configurar un CloudShell y que sus objetivos sean similares a los míos.

Montaje del CloudShell

Montar el CloudShell no es difícil. Sin embargo, si eres nuevo en Linux es probable que te preguntes dónde localizar tu disco duro tras arrancar la máquina puesto que la unidad no aparece tras utilizar el comando `df`. Para que aparezca la unidad, tendrás que particionar el disco, añadir una entrada en el archivo `/etc/fstab` y, o bien usar el comando `'mountall'` o bien reiniciar la máquina. Debes comprobar si hay algún error en tus nuevas entradas `/etc/fstab` utilizando `'mountall'`. Si existe un error en la entrada, la máquina no arrancará después de apagarla. Si esto ocurre, tendrá que retirar la tarjeta SD, insértela en tu máquina Linux, montarla y corregir el archivo `fstab` para que tu ODROID arranque de nuevo. Aquí tiene alguna información y enlaces útiles que te ayudarán a configurar tu CloudShell.

El montaje físico es muy simple, sólo tienes que seguir el procedimiento que se describe en <http://bit.ly/1N3xNm7>

La información sobre cómo configurar la LCD y el recep-

tor IR la puedes encontrar en la Wiki de ODROID (<http://bit.ly/1R6DOgZ>) haciendo una búsqueda por “CloudShell”.

El procedimiento para configurar el disco duro lo puedes localizar aquí (<http://bit.ly/1Gu1smv>)

Además, puede que quieras configurar una partición swap siguiendo las siguientes instrucciones (<http://bit.ly/1Xnn48x>)

Ten en cuenta que el UUID necesario para /etc/fstab no se genera hasta que no utilices el comando “mkswap”.

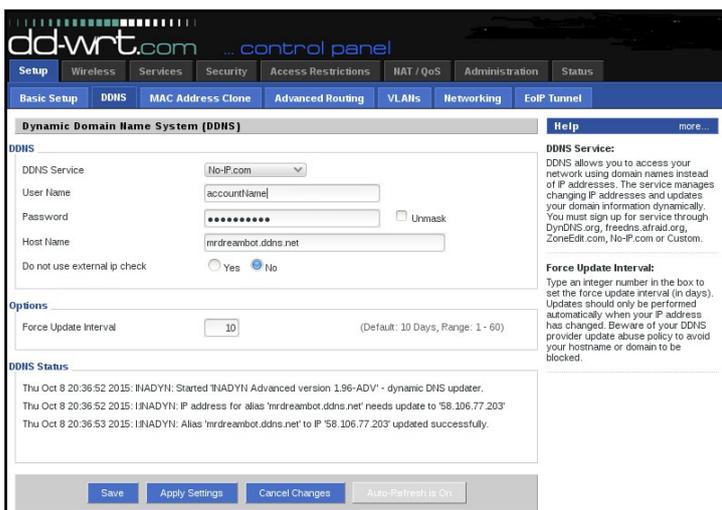
Red y DNS dinámico

Antes de exponer tu servidor a Internet, debes asegurarte de que la seguridad es la adecuada. Aquí tienes un artículo (<http://bit.ly/1kxJuWw>) de mi blog sobre cómo configuraré la seguridad de mi red doméstica usando un DMZ.

Si tu proveedor de Internet es como el mío, te proporcionará una dirección IP dinámica. En este caso, necesitarás configurar DNS dinámico para poder acceder a tu servidor desde Internet utilizando el mismo nombre de dominio aunque tu dirección IP cambie regularmente. Para ello, es necesario:

Registrarte para obtener una cuenta de Dirección IP dinámica gratis aquí (<http://www.noip.com/free>). Por supuesto, puedes utilizar otros proveedores si lo deseas.

Configura DDNS en tu router de acceso a Internet utilizando el panel de administración del router. Necesitaras tu nombre de cuenta, la contraseña y el nombre de host (Nombre de dominio que has elegido para tu servidor domestico registrado con el proveedor cuando creaste la cuenta). Si está usando DD-WRT en tu router, la pantalla debería tener este aspecto:



Configuración de DDNS en el router

Servidor WordPress para blogs

Configurar WordPress no es fácil. Tienes que configurar el sistema LAMP. LAMP es el acrónimo de los nombres de sus cuatro componentes de código abierto: Sistema operativo Linux, Apache HTTP Server, sistema de gestión de base de datos relacional MySQL (RDBMS) y lenguaje de programación

PHP. Esto significa que tendrás que crear una base de datos MySQL, configurar el servidor Web Apache y WordPress, incluidos los archivos y directorios apropiados. Hay momentos en los que me dije a mí mismo “tal vez sea más fácil usar un sitio de blogs gratuito”. Tan sólo hay que tener un poco de paciencia y podrás beneficiarte de las ventajas de tener tu propio blog en un servidor doméstico. Como dice el refrán: “Lo importante es el viaje, no el destino”.



Mi Blog (<http://MrDreamBot.ddns.net>) impulsado por Cloudshell

Aquí tienes algunos consejos que te serán muy utiles:

Instala el sistema LAMP siguiendo estas instrucciones (<http://do.co/1sLDg7T>)

Instala WordPress tal y como se describe aquí (<http://do.co/1Auy0sz>). La instalación de WordPress es muy diferente a la de otros paquetes. No puedes utilizar el paquete instalado después de hacer un “apt-get install”. Tendrás que copiar todo ese contenido a la raíz del servidor Web Apache como se describe en el enlace.

Si hacer clic en botón “Add Media” y no aparece ningún cuadro de diálogo, agrega “define (“CONCATENATE_SCRIPTS”, false);” al final del archivo wp-config.php.

Reemplaza todos los enlaces simbólicos del directorio /YourPath/www/html y subdirectorios con una copia del archivo al que apuntan.

Si recibes un mensaje al acceder a WordPress que indique que no tienes acceso a un determinado directorio, lo más probable es que sea por /youPath/www/html/.htaccess. Elimínalo o cámbiale el nombre y comprueba si te da resultado.

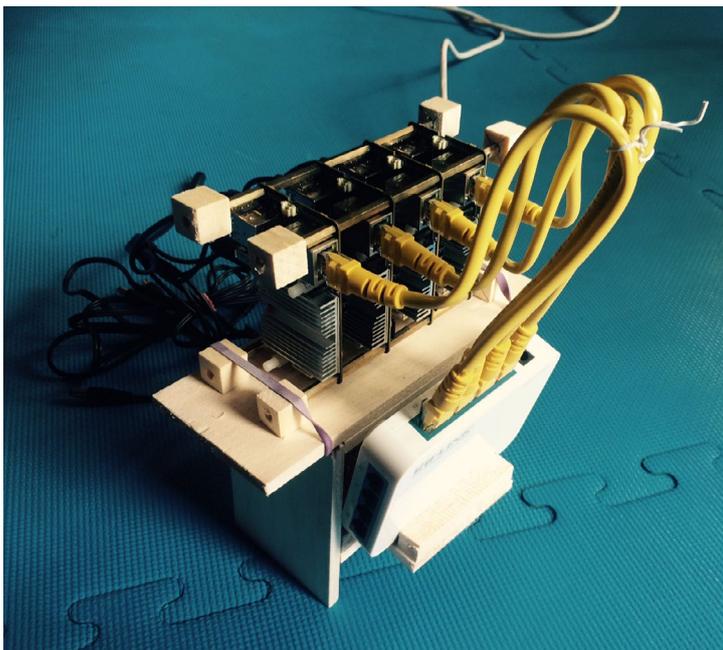
+FollowSymLinks” y “AllowOverride All” en /etc/apache2/sites-available/default.conf son importantes si personalizas los enlaces permanentes de WordPress que requieren reescribir la URL. Asegúrate de usar los comandos “sudo a2ensite default.conf” y “sudo service apache2 restart” después de hacer cambios en el archivo default.conf, como muestra el siguiente ejemplo:

```
ServerAdmin webmaster@localhost
DocumentRoot /yourpath/www/html
<Directory /yourPath/www/>
Options -Indexes +Fol-
lowSymLinks
AllowOverride All
Require all granted
</Directory>
```

Configura tu router de Internet para redireccionar el puerto 80 a la dirección IP del CloudShell IP.

NFS Server y nodo principal Clúster informático doméstico

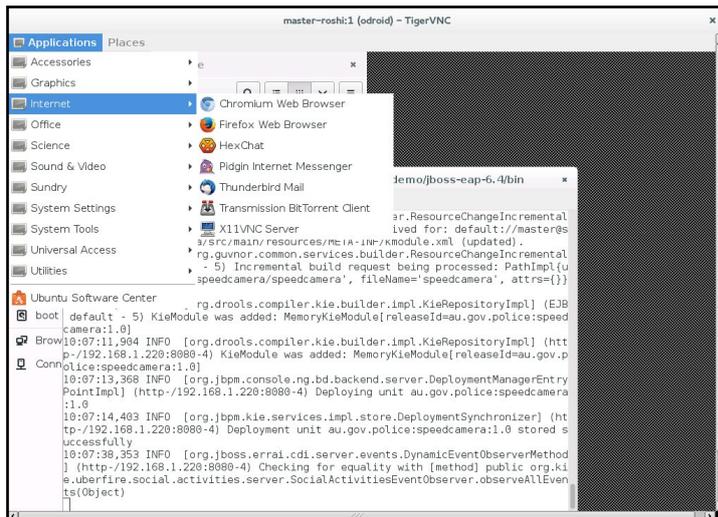
Desarrollé hace algún tiempo un clúster doméstico con 4 placas ODROID U3 como nodos informáticos con un total de 16 núcleos de potencia. Ahora es el momento de hacer una reforma. Estoy añadiendo el CloudShell como nodo principal al clúster y utilizándolo como servidor NFS, de modo que cuando quiera ejecutar un programa MPJ Express (<http://mpj-express.org/>) en el clúster no tengo que copiar ese programa en cada uno de los nodos informáticos. Al utilizar un servidor NFS, todo lo que tengo que hacer es copiar el programa en el servidor y los nodos informáticos accederán a él. No tengo que hacer copias. Poseo una gran variedad de ordenadores de placa reducida Linux como son una Raspberry Pi, BeagleBone original, BeagleBone Black, CubieBoard V1, ODROID-XU3 Lite y una ODROID-U3. Algunos sistemas operativos no tienen el módulo NFS incluido en el kernel para ser utilizados como servidor NFS. Me sorprende gratamente que el ODROID-XU4 no presente este tipo de problemas.



Mi Clúster informático doméstico utiliza 4 ODROID-U3

Servidor SSH - Acceso remoto

También puedes configurar un servidor VPN (Virtual Private Network) para acceder a tu red doméstica de forma segura desde cualquier lugar. Para mí, configurar una VPN es excesivo, aunque tengo la intención de echarle un rato por diversión cuando tenga tiempo. Todo lo que necesito es poder acceder a mi CloudShell desde Internet utilizando SSH. Una vez que sea capaz de acceder a mi CloudShell desde Internet, podre entonces redireccionar el puerto SSH para crear un túnel cifrado para acceder al escritorio de mi CloudShell usando VNC.



Conexión por VNC a mi CloudShell Via SSH

Así es cómo configure mi servidor SSH de acceso desde Internet:

Configura el servidor como se describe en esta dirección (<http://do.co/1Gu1Mlj>)

Configura el router para redireccionar el puerto que tú elijas a la dirección IP de tu CloudShell y el puerto 22. Cuando puedas conectarte por SSH al CloudShell desde Internet, utiliza el puerto que ha elegido en lugar del puerto 22 por defecto para mayor seguridad.

Configura la autenticación por clave pública (<http://bit.ly/1LfGhF6>) en tu CloudShell y en las máquinas desde las que te conectaras por SSH desde Internet. Luego fuerza la conexión SSH a través de la autenticación de clave pública, siguiendo estas instrucciones (<http://bit.ly/1hYEgBg>). Con esta configuración no te solicitará una contraseña. La sesión simplemente finalizará si la máquina que conecta no tiene la clave pública correcta.

Demo Reglas de Negocio

Soy un programador de soluciones de software intermedio de profesión. De modo que, no debería sorprenderte que experimente y desarrolle pequeñas aplicaciones con fines de demostración todo el tiempo. Una aplicación que he desarrollado es un servicio web que utiliza reglas de negocio para determinar la mejor ubicación, entre muchas, para ubicar una cámara de

RNEAUDIO MUSIC PLAYER

CREA TU PROPIO REPRODUCTOR DE MUSICA CON CALIDAD PROFESIONAL

por Justin Lee

Un reproductor de música en red es un dispositivo que te conecta a una colección prácticamente ilimitada de música. Puedes cambiar de un género a otro, aunque la mayoría te permiten reproducir música desde:

Tus colecciones almacenadas en tu unidad USB o NAS, u otros archivos de música almacenados en tu ordenador, smartphone o una tablet

Miles de estaciones de radio por Internet

Gran variedad de servicios de suscripción de música online que ofrecen formas interesantes de descubrir y mezclar canciones. ¿Crees que esto es algo que sólo le gusta a los frikis? ¡Piénsalo de nuevo! El mejor de los reproductores de música en red de hoy día está listo para estremecerte desde el momento en el que entras por la puerta. No tienes que encender tu ordenador o insertar un CD. Simplemente inicia una aplicación en su teléfono o tablet y toca la pantalla un par de veces, y ahí la tienen: música para calmarte o inspirarte. También puede añadir una pantalla táctil a color para que muestre información de la canciones y artistas, así como poder organizar tus archivos de música favoritos.

Este artículo te ayudará a montar un reproductor de audio en red, así como crear un sistema embebido independiente con un ODROID. Al encender el dispositivo, aparecerá una bonita interfaz de usuario y reproducirá música automáticamente.

Requisitos

ODROID-C1+ (<http://bit.ly/IUpx5yl>)

HiFi-Shield (<http://bit.ly/IM6UIXY>)

LCD HDMI de 7 pulgadas (<http://bit.ly/INWxgDx>)

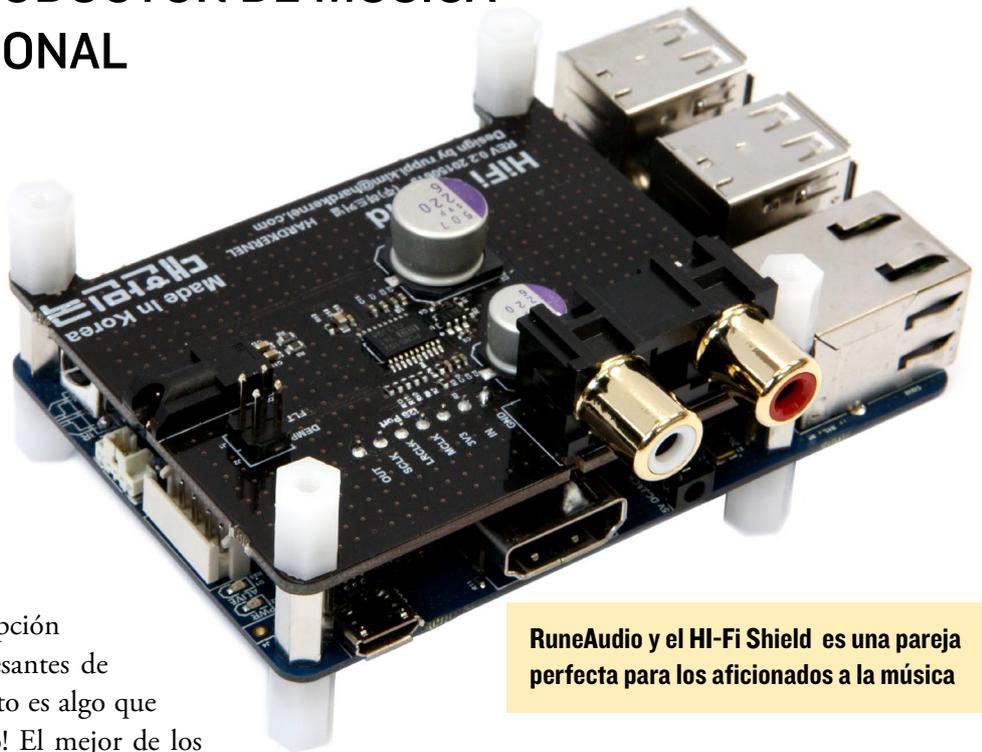
Sistema de altavoces con amplificador

Mando a distancia por infrarrojos opcional (<http://bit.ly/IM6UGiR>)

Hardware

Coloca el HiFi-Shield sobre tu ODROID-C1+ y monta los tornillos, luego conecta la pantalla LCD HDMI de 7" al C1+ con un cable HDMI y un cable micro-USB.

Una vez que finalices la instalación del software correspondi-



RuneAudio y el HI-Fi Shield es una pareja perfecta para los aficionados a la música

ente, dispondrá de un sistema reproductor de música totalmente independiente.

La instalación y configuración del software pasa por tres etapas principales y una cuarta opcional:

- Instalar la imagen de RuneAudio y configurar el entorno básico.
- Habilitar el driver de la LCD HDMI de 7 pulgadas e instalar X11 y el navegador
- Aprender a hacer auto-login, auto-inicio de X-windows y de aplicaciones como un profesional.
- Configurar los parámetros del controlador remoto IR

Sigue este largo proceso, y podrás mostrar una interfaz de usuario web para reproducir música en una pantalla LCD utilizando el navegador web Chromium en modo quiosco.

Instalar SO RuneAudio

Instala la imagen RuneAudio para C1+ en tu eMMC o SD. Puedes hacer esto descargando la imagen RuneAudio_odroid-c1_0.3-dev_20150824_2GB.img desde <http://bit.ly/1GhrkCw>. Utiliza la consola serie (con el kit USB-UART ODROID), SSH a la dirección IP asignada a la placa por el route o utiliza una pantalla HDMI con un teclado. El nombre de usuario es "root" y la contraseña es "rune".



```
# Disable VPU (Video decoding engine, Saves RAM!!!)
# 0 = disabled
# 1 = enabled
setenv vpu "1"
```

Cambia el valor de "1" a "0" :

```
# Disable VPU (Video decoding engine, Saves RAM!!!)
# 0 = disabled
# 1 = enabled
setenv vpu "0"
```

Guarda los cambios con "Ctrl-x + y + INTRO" y reiniciar el sistema. La memoria accesible total cambiará desde los 836MB a 948MB. ¡Hemos ganado más de 100 MB!

Optimizar el arranque

Para acelerar el proceso de arranque si utilizas una conexión Ethernet directa (sin Wi-Fi), introduce el siguiente comando:

```
$ systemctl disable netctl-auto@wlan0
```

Maximizar el almacenamiento

La imagen oficial del SO está hecha para tarjetas de 2 GB. En la mayoría de los casos tu tarjeta tendrá más capacidad, de modo que puedes utilizarla como almacenamiento local o para instalar otros paquetes. Escribe lo siguiente para entrar en fdisk:

```
$ fdisk /dev/mmcblk0
```

- Obtener el tamaño actual ("p + INTRO" en tu teclado)
- Eliminar la primera partición ("d + INTRO")
- Agregar una nueva partición con los parámetros por defecto ("n + INTRO + INTRO + INTRO + INTRO + INTRO")
- Escribir la tabla de particiones en tu tarjeta, ignorando los mensaje de error ("w + INTRO")
- Reiniciar el sistema ("reboot + INTRO")
- Iniciar sesión de nuevo
- Redireccionar el sistema de archivos raíz:

```
$ resize2fs /dev/mmcblk0p1
```

Verifica el nuevo tamaño::

```
$ df -h
```

Maximizar la RAM

Elimina la memoria reservada para el driver de vídeo editando el archivo de configuración de arranque:

```
$ nano /boot/boot.ini
```

Desplázate hacia abajo hasta llegar a esta parte:

Habilitar el driver I2S DAC

Activa los requisitos para el HiFi-Shield de ODRROID editando de nuevo el archivo de configuración de arranque:

```
$ nano /boot/boot.ini
```

Desplázate hacia abajo hasta llegar a esta parte:

```
# PCM5102 audio DAC Enable/Disable
# Uncomment the line below to __ENABLE__ Audio-DAC (PCM5102)
#setenv enabledac "enabledac"
```

Elimina una almoadilla (#) de siguiente forma:

```
# PCM5102 audio DAC Enable/Disable
# Uncomment the line below to __ENABLE__ Audio-DAC (PCM5102)
setenv enabledac "enabledac"
```

Guarda los cambios con "Ctrl-x + y + ENTER" y reinicia.

Ajustar la resolución HDMI

La resolución HDMI por defecto es 720p (1280x720). Si tienes problemas con el arranque desde la SD UHC-1, prueba lo siguiente. Encuentra esta sección dentro de /boot/boot.ini y añade una almohadilla (#) en la última línea:

```
# UHS Card Configuration
# Uncomment the line below to __DISABLE__ UHS-1 MicroSD support
# This might break boot for some brand models of
# setenv disableuhs "disableuhs"
```

Instalar escritorio

Vamos a instalar un gestor de ventana y un navegador web

con los paquetes correspondientes. Primero, debe sincronizar la base de datos con el servidor de paquetes Arch-ARM:

```
$ pacman -Syy
```

Instala los archivos del driver de vídeo (uno para x11 y otro para framebuffer):

```
$ pacman -S xf86-video-odroid-c1 xf86-video-fbdev
```

Instala el servidor xorg y xinit (startx):

```
$ pacman -S xorg-server xorg-xinit
```

Instala la interfaz gráfica de usuario de escritorio LXDE, asegurándote de seleccionar todos los paquetes opcionales:

```
$ pacman -S lxde
```

Instala una fuente adicional si lo deseas. Yo instalé la fuente coreana para mostrar la información de la canción en coreano.

```
$ pacman -S ttf-baekmuk
```

Lo siguiente es instalar el navegador web Chromium. Sin embargo, el paquete chromium no funciona en el servidor Arch-ARM. La solución es descargar un paquete de Chromium funcional desde una fuente alternativa e instalarlo manualmente. He usado el paquete disponible en <http://bit.ly/1W4AFo5>.

```
$ wget http://odroidxu.leeharris.me.uk/chromium-45.0.2454.101-1-armv7h.pkg.tar.xz
$ sudo pacman -U chromium-45.0.2454.101-1-armv7h.pkg.tar.xz
```

A continuación, ejecute el gestor de ventanas LXDE:

```
$ lxdm
```

Prueba la interfaz de usuario de RuneAudio en la pantalla HDMI. Debería funcionar con el siguiente comando:

```
$ chromium --user-data-dir --kiosk 127.0.0.1
```

Configurar modo Quiosco

Vamos a hacer que el sistema funcione de forma totalmente automática, de modo que en el arranque se inicie todo. Primero, activa el inicio de sesión automático en la consola de HDMI:

```
$ systemctl edit getty@tty1
```

Copia y pega estas tres líneas, guardar los cambios con “Ctrl-x + y + INTRO” y reinicia el sistema:

```
[Service]
ExecStart=
ExecStart=-/usr/bin/agetty --autologin root --noclear
%I 38400 linux
```

Puedes pasar por alto el proceso de inicio de sesión en pantalla, pero aún tienes que iniciar sesión en las consolas Serie y SSH.

Auto-inicio de LXDE

Ejecute el siguiente comando para crear un `.bash_profile`:

```
$ nano /root/.bash_profile
```

Copia y pega las siguientes líneas y guarda los cambios (“Ctrl-x + y + INTRO”):

```
#
# ~/.bash_profile
#
echo Run me first
if [[ -z $DISPLAY ]] && [[ $(tty) = /dev/tty1 ]];
then
exec lxdm
fi
```

Activar inicio de sesión automático en LXDE

Edita el archivo `/etc/lxdm/lxdm.conf` en la tercera línea, guardarlo y luego reinicia para comprobar que el sistema se inicia directamente con el escritorio LXDE:

```
autologin=root
```

Auto-inicio de Chromium

Edita el archivo `/etc/xdg/lxsession/LXDE/autostart`, y añade esta línea al final:

```
@chromium --user-data-dir --kiosk 127.0.0.1
```

Puedes ejecutar la interfaz de usuario web de RuneAudio 20 segundos después del inicio si arrancas desde el módulo eMMC.

Control remoto

Opcionalmente, puede utilizar un mando a distancia IR para controlar RuneAudio, disponible para su compra por 4\$ desde la tienda de Hardkernel en <http://bit.ly/1M6UGiR>.

Las funciones con las que cuenta son:

- Subir volumen = “+”
- Bajar volumen = “-”
- Siguiente canción = “flecha hacia abajo” (coincide con la cola)
- Canción anterior = “flecha hacia arriba”
- Buscar hacia adelante = “flecha a la derecha”



La mejor forma de interactuar con tus canciones mientras te relajas: Un clásico mando a distancia por Infrarrojos

- **Buscar hacia atrás = “flecha izquierda”**
- **Alternar entre reproducir/pausar = “mute”**
- **Apagar = “Botón encendido” (presionar y mantener durante 3 segundos) + “OK” (presionar y mantener durante 1 segundo)**
- **Reiniciar = “Botón encendido” (presionar y mantener 3 segundos) + “flecha hacia arriba” (presionar y mantener 1 segundo)**
- **Si el reproductor está parado en la siguiente o anterior canción (flecha hacia abajo o hacia arriba) inicia la reproducción**

Primero, edita el archivo “/etc/lirc/lircd.conf.d/lircd.conf”:

```
# this config file was automatically generated
# using lirc-0.9.0(default) on Mon May 25 19:47:48
2015
#
# contributed by
#
# brand: ODROID                lircd.conf.conf
# model no. of remote control:
# devices being controlled by this remote:
#
begin remote

    name    lircd.conf
    bits    16
    flags   SPACE_ENC|CONST_LENGTH
    eps     30
    aeps    100

    header  8964 4507
    one     544 1692
    zero    544 561
    ptrail  544
    repeat  8965 2285
    pre_data_bits 16
```

```
pre_data    0x4DB2
gap         107872
toggle_bit_mask 0x0

begin codes
    KEY_PREVIOUS    0x53AC
    KEY_NEXT        0x4BB4
    KEY_FORWARD     0x837C
    KEY_BACK        0x9966
    KEY_OK          0x738C
    KEY_HOME        0x41BE
    KEY_MUTE        0x11EE
    KEY_MENU        0xA35C
    KEY_VOLUMEDOWN 0x817E
    KEY_VOLUMEUP    0x01FE
    KEY_POWER       0x3BC4
end codes
```

end remote
También edita /etc/lirc/lircrc:

```
## irexec
begin irexec
begin
    prog = irexec
    button = KEY_POWER
    ignore_first_events = 20
    config = php /srv/http/command/ui_notify.php
    php Power "Power down = press OK\nReboot = press UP"
    simplemessage
    mode = powerdown
    repeat = 0
end

begin
    prog = irexec
    button = KEY_PREVIOUS
    config = mpc prev && mpc play
    repeat = 0
end

begin
    prog = irexec
    button = KEY_NEXT
    config = mpc next && mpc play
    repeat = 0
end

begin
    prog = irexec
    button = KEY_VOLUMEUP
    config = mpc volume +2
    repeat = 1
end

begin
    prog = irexec
    button = KEY_VOLUMEDOWN
    config = mpc volume -2
    repeat = 1
end

begin
    prog = irexec
    button = KEY_FORWARD
    config = mpc seek +00:00:02
    repeat = 1
```

```

end

begin
  prog = irexec
  button = KEY_BACK
  config = mpc seek -00:00:02
  repeat = 1
end

begin
  prog = irexec
  button = KEY_MUTE
  config = mpc toggle
  repeat = 0
end
end irexec

begin powerdown
  begin
    prog = irexec
    button = KEY_OK
    ignore_first_events = 5
    config = /srv/http/command/rune_shutdown;
shutdown now -h
    repeat = 0
  end

  begin
    prog = irexec
    button = KEY_PREVIOUS
    ignore_first_events = 5
    config = /srv/http/command/rune_shutdown;
reboot
    repeat = 0
  end

  begin
    prog = irexec
    button = KEY_VOLUMEUP
    mode = irexec
    repeat = 0
  end

  begin
    prog = irexec
    button = KEY_VOLUMEDOWN
    mode = irexec
    repeat = 0
  end

  begin
    prog = irexec
    button = KEY_FORWARD
    mode = irexec
    repeat = 0
  end

  begin
    prog = irexec
    button = KEY_BACK
    mode = irexec
    repeat = 0
  end

  begin
    prog = irexec
    button = KEY_RIGHT
    mode = irexec
    repeat = 0
  end

  begin
    prog = irexec
    button = KEY_LEFT
    mode = irexec
    repeat = 0
  end
end

```

```

begin
  prog = irexec
  button = KEY_MUTE
  mode = irexec
  repeat = 0
end
end powerdown

```

Ejecuta los siguientes comandos para activar los demonios de infrarrojos

```

$ systemctl enable lircd
$ systemctl start lircd
$ systemctl enable irexec
$ systemctl start irexec

```

Para utilizar el monitor con pantalla táctil de 7 pulgadas, necesitas seleccionar la resolución de 800x480 y el modo de salida “dvi” en lugar de “hdmi”. Si el color de los iconos o del cursor es extraño, puedes solucionarlo cambiando la profundidad de color del sistema a 24 bits.

```

# setenv m "vga" # 640x480
# setenv m "480p" # 720x480
# setenv m "576p" # 720x576
setenv m "800x480p60hz" # 800x480
# setenv m "800x600p60hz" # 800x600
# setenv m "1024x600p60hz" # 1024x600
# setenv m "1024x768p60hz" # 1024x768
# setenv m "1360x768p60hz" # 1360x768
# setenv m "1366x768p60hz" # 1366x768
# setenv m "1440x900p60hz" # 1440x900
# setenv m "1600x900p60hz" # 1600x900
# setenv m "1680x1050p60hz" # 1680x1050
# setenv m "720p" # 720p 1280x720
# setenv m "800p" # 1280x800
# setenv m "sxga" # 1280x1024
# setenv m "1080i50hz" # 1080I@50Hz
# setenv m "1080p24hz" # 1080P@24Hz
# setenv m "1080p50hz" # 1080P@50Hz
# setenv m "1080p" # 1080P@60Hz
# setenv m "1920x1200" # 1920x1200

# HDMI DVI Mode Configuration
# setenv vout_mode "hdmi"
setenv vout_mode "dvi"

# HDMI BPP Mode
# setenv m_bpp "32"
setenv m_bpp "24"
# setenv m_bpp "16"

```

No olvides instalar la última versión del kernel para activar el driver multi-táctil capacitivo. Puede que tardes un par de horas en instalar y configurar los paquetes para hacer sistema autónomo. También existe una imagen precompilada con todos los pasos, que puede ser descargada desde <http://bit.ly/1RXCLAs>.

Creditos

Gracias a Frank Friedmann (@ hondagx35) que desarrolla y mantiene la imagen RuneAudio para la plataforma ODROID. Puede encontrar más información en <http://bit.ly/1MSSStwB> y <http://bit.ly/1jSB2AL>.

ODROID-VU7: PANTALLA MULTI-TÁCTIL HDMI DE 7 PULGADAS PARA EL ODROID-C1+

UN ECONOMICO MONITOR PORTATIL INTERACTIVO

por Justin Lee

La pantalla multi-táctil de 7 pulgadas para ODROID-C1+, conocido como el ODROID-VU7, ofrece a los usuarios la posibilidad de crear un Todo-en-uno, proyectos integrados tales como tablets, consolas de videojuegos, sistemas de entretenimiento y sistemas embebidos. La pantalla 800x480 se conecta a través de una placa de conexión HDMI y una placa de conexión micro-USB que se encargan de la alimentación y de la señal. Sólo tiene que conectar la fuente de alimentación a la toma DC del C1+ y estás listo para divertirte, una vez que instales la última actualización del sistema operativo. Su precio es de sólo 55\$, y puedes comprarla en <http://bit.ly/1NWxgDx>.

Esta pantalla táctil de alta calidad ha sido diseñada específicamente para funcionar con Android y Linux en el ODROIDC1+. Puede conectar fácilmente el C1+ a la parte trasera de la pantalla LCD y así crear un sistema totalmente integrado. También funciona con la placa original C1 si utilizas un cable micro-HDMI y un cable micro-USB. La pantalla LCD HDMI de 7 pulgadas viene con una placa de conexión HDMI, una placa de conexión micro-USB, tres soportes para la PCB de 8mm y 6 tornillos.

Los drivers de la pantalla táctil tienen soporte para 5 puntos de contacto, y están integrados en las últimas imágenes de Ubuntu y Android. La resolución de la pantalla debe estar configurada para WVGA (800x400) y la opción DVI debe estar activada en el parámetro "vout" editando el archivo boot.ini.

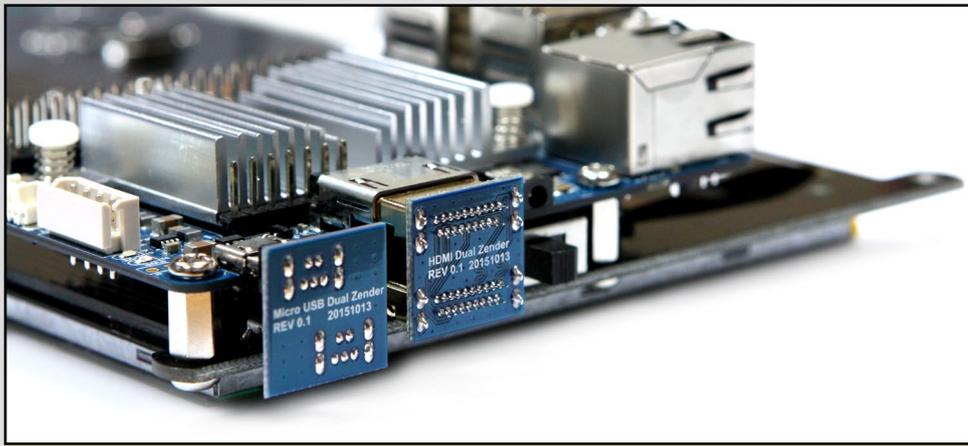


La pantalla táctil HDMI de 7 pulgadas es el compañero perfecto para tu ODROID-C1+

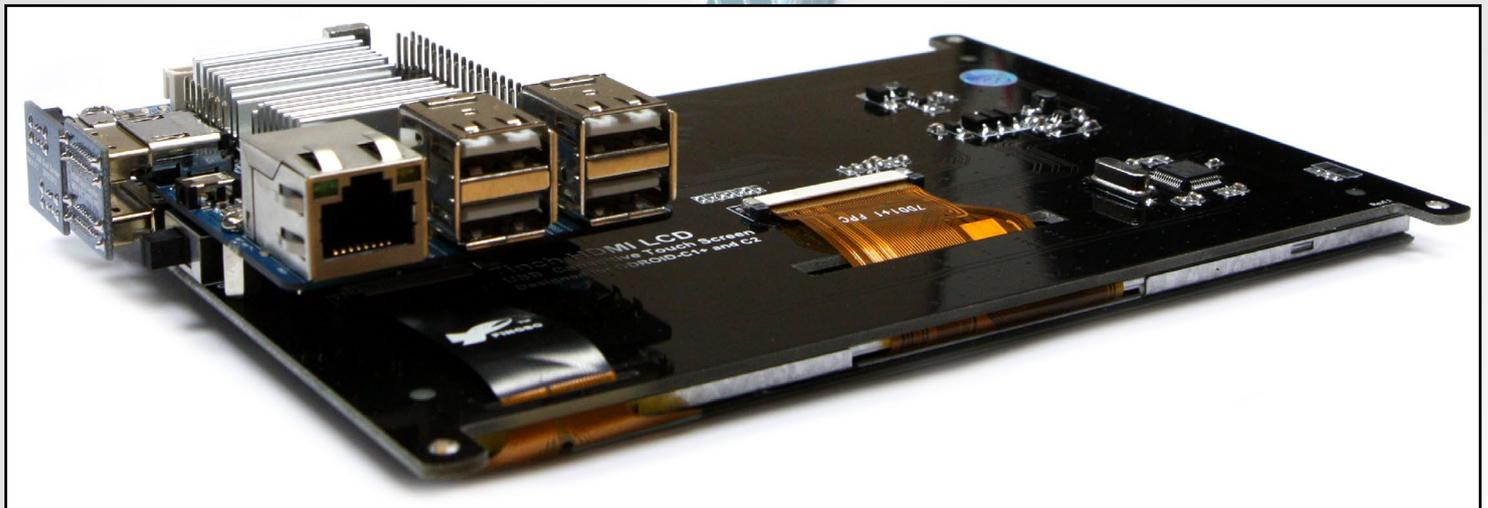
Guía de montaje

Opción 1: Montar la placa C1+ sobre la unidad LCD con los 3 soportes PCB de 8mm. La placa de conexión HDMI y la placa de conexión micro-USB son utilizadas para la interfaz y la alimentación.

Opción 2: También se puede usar un cable USB micro-micro y un cable HDMI. El kit de cables de 35 cm se vende por separado.



Primer plano de los adaptadores USB que unen la pantalla multi-touch al CI+



La pantalla táctil montada directamente con la placa CI



Especificaciones

LCD-TFT de 7 pulgadas

Resolución de pantalla: 800x480 píxeles

Entrada táctil capacitiva de 5 puntos

Consumo de energía: 600 mA/5Voltios

Interruptor on/off de Luz de fondo

Ángulo de visión (en grados): 70 Izquierda, 70 Derecha, 50 Arriba, 70 Abajo

Dimensiones de la pantalla: 172.9 mm x 124.3 mm x 15 mm (incluyendo interruptor y conectores)

Tamaño visible la pantalla: 153.6 mm x 86,64 mm (área activa)

Usar cables también es otra opción para montar la pantalla multi-táctil

Configuración

Para utilizar la pantalla táctil en el sistema Linux, la versión del kernel debe ser 3.10.80-128 o superior. Además, no olvides configurar en el archivo boot.ini la resolución a 800x480 y el modo de salida DVI:

```
# Possible screen resolutions
# Uncomment only a single line! The line with
# setenv written.
# At least one mode must be selected.

# setenv m "vga" # 640x480
# setenv m "480p" # 720x480
# setenv m "576p" # 720x576
setenv m "800x480p60hz" # 800x480
# setenv m "800x600p60hz" # 800x600
# setenv m "1024x600p60hz" # 1024x600
# setenv m "1024x768p60hz" # 1024x768
# setenv m "1360x768p60hz" # 1360x768
# setenv m "1440x900p60hz" # 1440x900
# setenv m "1600x900p60hz" # 1600x900
# setenv m "1680x1050p60hz" # 1680x1050
# setenv m "720p" # 720p 1280x720
# setenv m "800p" # 1280x800
# setenv m "sxga" # 1280x1024
# setenv m "1080i50hz" # 1080I@50Hz
# setenv m "1080p24hz" # 1080P@24Hz
# setenv m "1080p50hz" # 1080P@50Hz
# setenv m "1080p" # 1080P@60Hz
# setenv m "1920x1200" # 1920x1200

# HDMI DVI Mode Configuration
# setenv vout_mode "hdmi"
setenv vout_mode "dvi"
# setenv vout_mode "vga"
```

El navegador Chromium soporta zoom táctil, para que puedas acercar y alejar la imagen como un teléfono inteligente. La pantalla táctil también funciona con el sistema operativo Android para el C1+. Sin embargo, es necesario utilizar versión 2.0 o superior para poder utilizar las funciones táctiles.

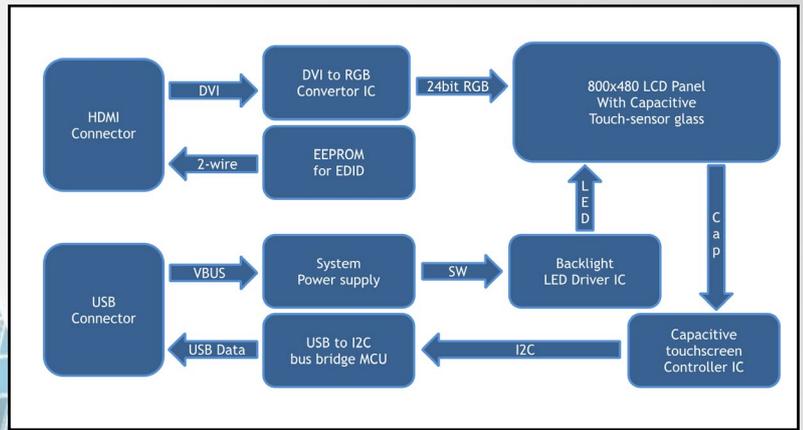
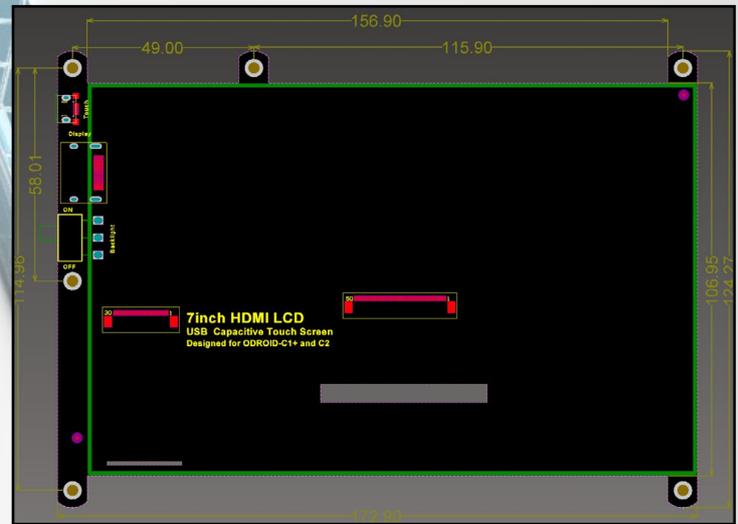


Diagrama por bloques de la pantalla Multi-táctil



Vista trasera y dimensiones físicas



Demostración de cómo es posible acercar y alejar la imagen

CONOCIENDO UN ODROIDIAN

MANUEL ADAMUZ,
EDITOR ESPAÑOL DE
ODROID MAGAZINE

editado by Rob Roy

Por favor, hablanos un poco sobre ti.

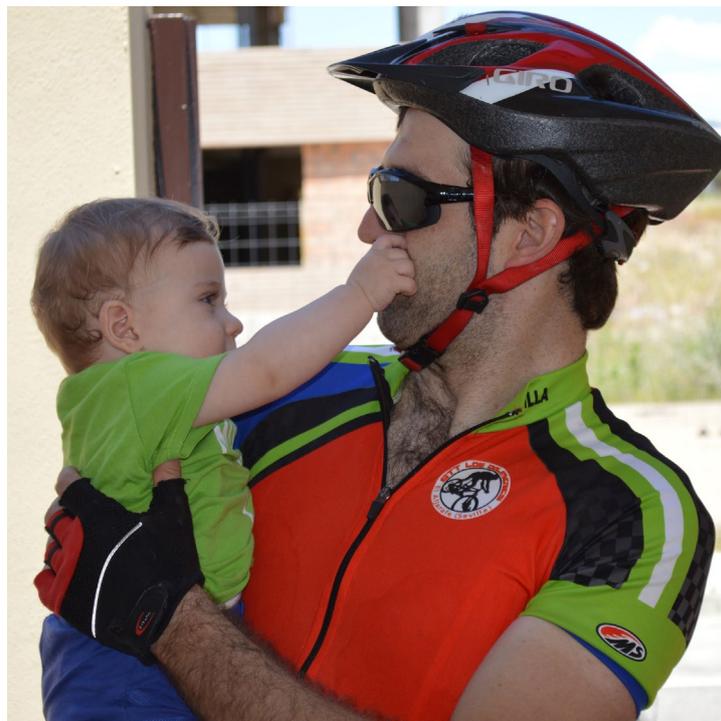
Tengo 33 años y vivo en Sevilla, España, aunque nací en Granada donde se encuentra gran parte de mi familia. Estoy casado y tengo un hijo que es muy poco tasto. Hace unos años trabajé como técnico informático y programador, pero mi trabajo actual está relacionado con la gestión de calidad y las tecnologías de la información: ISO 9001, ISO 27001, ISO 20000. Soy un apasionado de la informática, especialmente de los microordenadores como el ODROID, Raspberry Pi, etc.

¿Cómo fueron tus inicios con los ordenadores?

Al ir traduciendo Odroid magazine me he dado cuenta que muchos odroidian se han iniciado en el mundo de la informática a una edad muy temprana, incluso siendo niños. En mi caso no fue así tuve mi primer ordenador a los 17 años, estando en el instituto, y fue gracias a mi madre. Se trataba de un ordenador Petium II de 233 Hz con 64 GB de Ram, un disco duro de 3 GB, una tarjeta grafica de 4 MB y una sound blaster de 64 bits como tarjeta de sonido. Era uno de los primeros ordenadores con Windows 98 ¡El primer ordenador de tu vida se recuerda muy bien! Por desgracia no lo conservo ya que mi padre al ver que me compraba nuevos ordenadores más potentes decidió desecharlo. Tengo muy buenos recuerdos de aquella época. Recuerdo muy bien el día en el que vendedor me trajo el ordenador en casa, apenas dormí la noche anterior y lo único que hacía era mirar por la ventana del salón para ver si llegaba algún coche a casa. También recuerdo que tarde una semana en “fastidiar” el SO Windows 98, después de instalar cientos de juego y aplicaciones. Llego un momento que Windows no se iniciada ¡Quería probar todo lo que tenía a mi alcance!

¿Qué te atrajo de la plataforma ODROID?

Gracias a un amigo, que por aquel entonces vivía en Inglaterra, llegue a conocer la tan popular Raspberry Pi, que aún conservo en la actualidad y la utilizo para diversos fines. Dispongo de dos modelos: Raspberry Pi B y la Raspberry Pi 2. Realizando pruebas con estas placas me di cuenta rápidamente que les faltaba potencial. Necesitaba algo más potente y más rápido. Empecé a buscar por internet alternativas y en primer lugar compre la CubieBoard2 (CPU 1 Ghz y 1 GB de Ram), principalmente por su precio.



Manuel y su hijo David - ¡Esperamos que siga los pasos de su padre como aficionado al ciclismo!

Comprobé al poco tiempo que había cometido el error de comprarla y actualmente la tengo como un objeto de adorno en mi despacho. Fue poco después cuando descubrí por casualidad las placas Odroid leyendo un bloc que incluía un artículo donde se comparaban varios “Single Board Computer” entre ellos el odroid U3. Pensé: 1.7 Ghz + 2 GB de RAM ¡Suenan muy bien! Y a partir de aquí es donde comienza mi odisea con las placas Odroid, los foros odroid y por supuesto convertirme en el traductor de Odroid Magazine. He de reconocer que las placas odroid son una maravilla en muchos sentidos, son muy potentes y versátiles. También quisiera mencionar a la Comunidad Odroid que es impresionante y que gracias a ésta he podido resolver muchos de los problemas que he ido teniendo a la hora de instalar y configurar los diferentes SO que he ido instalado en mis placas ODROID: Ubuntu, Android, OpenElec.

¿Cómo utilizas tus ODROIDS?

Tengo un Odroid XU3 encendido permanentemente en casa, desde el cual navego por internet, consulto redes sociales, descargo software, etc... Es impresionante el poco consumo eléctrico que tiene. Mi PC Windows consume 10 veces más que el Odroid. Por otro lado dispongo de un Odroid-U3 que lo utilizo como Sistema de entretenimiento, el cual tiene instalada la distribución OpenElec. Fundamentalmente lo utiliza mi hijo para ver películas de Disney. Hace poco tiempo recibe un Odroid-C1+ que estoy probando y me está dando



Participando en una carrera de bicicletas semi-profesional

muy buenas sensaciones. Estoy pensando en utilizarlo como un sistema NAS.

¿Cuál es tu ODROID favorito?

Difícil pregunta. Me gustan todos, Cada uno tienes sus particularidades. La gente habla muy bien del odroid U3 y realmente pienso que es una placa magnífica. Si tuviera que elegir uno, simplemente lo que haría sería pensar para que lo iba a destinar y luego seleccionar el más adecuado para ese proyecto en concreto.

¿Qué tipo de mejoras de hardware te gustaría ver en futuras placas de Hardkernel?

He podido leer muchas opiniones sobre este tema al ir traduciendo Odroid Magazine. He visto que con el tiempo hardkernel han mejorado sus placas en muchos aspecto: CPU, GPU, Puertos USB, eMMC, GPIOs, etc. Pero hay una cuestión que no ha modificado con el tiempo, la RAM. Hardkernel no ha fabricado placas que superen los 2 GB de RAM ¿Por qué? ¿No es posible? ¿Piensan que no es necesario? Yo pienso que sí. Me gustaría ver en un futuro placas Odroid con 3 o 4 GB de RAM. Pudiendo así llegar a sustituir casi por completo a un PC de toda la vida

¿Qué aficiones e intereses tienes aparte de los ordenadores?

Tengo varios hobbies, pero indudablemente el que más me apasiona es la bicicleta.. Como dice una amiga “me gusta subir cuevas y en marchas largas...”. De vez en cuando participo en carrera semiprofesionales, aunque nunca gano pero tampoco me quedo el último. Es muy difícil ganar, hay gente muy obsesionada con la bicicleta. También me gusta viajar, aunque



En 2012, Manuel visitó Machu Picchu en Perú

últimamente no viajo mucho. He estado en Perú, Marruecos, Inglaterra, Escocia y en Praga.

¿Qué consejo le darías a alguien que quiere aprender más sobre la programación y/o hardware?

A pesar de que estuve un tiempo trabajando como programador, nunca me ha llegado a gustar demasiado. Creo que un buen programador debe tener ante todo paciencia, mucha paciencia, ya que serán muchas las veces que tendrá que reescribir código y depurarlo hasta conseguir los resultados deseados. Es verdad que hoy día internet ayuda mucho, existe multitud de manuales y tutoriales en línea casi para todo. Además, si encuentra comunidades como la que hay detrás de los foros odroid te puedo asegurar que ahorrar mucho tiempo y esfuerzo.

¿Cómo te convertiste en el traductor de ODROID magazine?

Cuando descubrí las placas odroid, observe que se estaba publicando una revista en formato electrónico en torno a estas placas. Empecé a echar un vistazo a los primeros números publicados en 2014 y la verdad es que los artículos y manuales que incluían eran muy interesantes. De hecho he tenido que recurrir a la revista en varias ocasiones para resolver algunos problemas de instalación y configuración de las placas. Rápidamente eche en falta que la revista estuviese en español. Así que me aventure y envié un email a Hardkernel proponiendo la traducción. Reconozco que al principio me costó un poco ya que algunos artículos eran y siguen siendo muy técnicos, pero realmente disfruto traduciendo esas magníficas aportaciones de usuario que adoran estos diminutos ordenadores.

El querido ODROID-XU3 de Manuel con un disco duro externo se mantiene en funcionando 24 horas al día, 7 días a la semana.

