

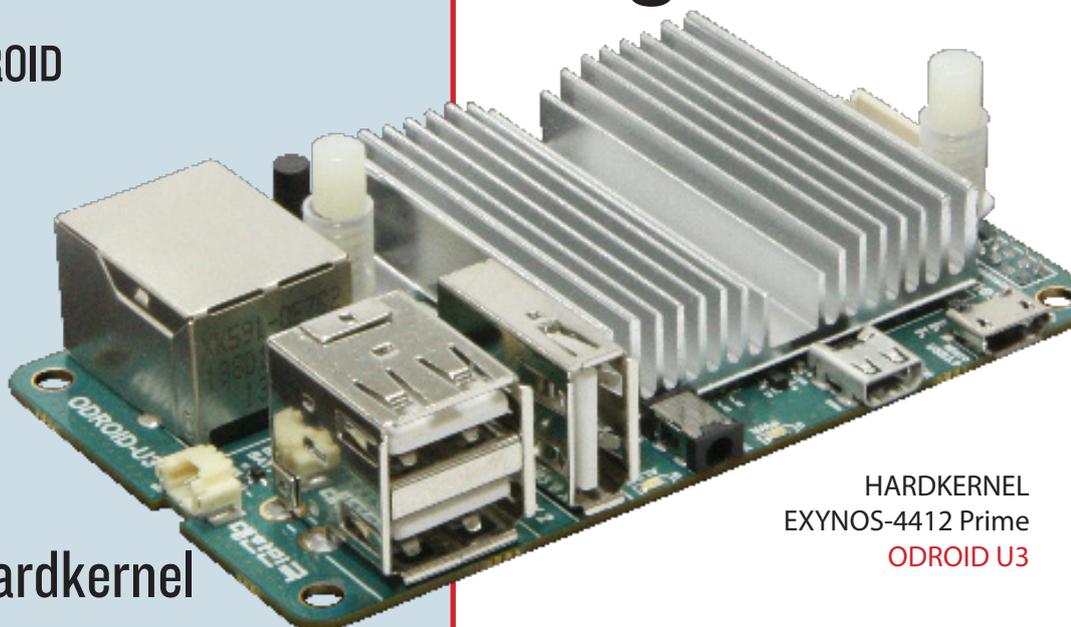
ODROID

Magazine

Si te gusta ODROID-U2,
te encantará el nuevo ODROID

U3

La nueva arma de hardkernel
para el desarrollo en ARM
¡Y es asombrosa!



HARDKERNEL
EXYNOS-4412 Prime
ODROID U3

RPi vs U3

Te mostramos
cómo se enfrentan
uno al otro

Ubuntu 13.10

Instalar y Solucionar problemas
(Una forma sencilla)

- Usa C/C++/Python/Java y IDE para:
Piraterar tu hardware y controlar LEDs
- ¡Empieza a programar de una vez!
- Informática de alto
Rendimiento con múltiples ODROIDs

• Conociendo a un Odroidian
Justin Lee, CEO de Hardkernel

• Juegos Linux

• Reconocimiento ocular en tiempo real

• Introducción al Reproductor
de Video HTML5



Al servicio de las comunidades “Open Source” y ODROID de todo el mundo, HardKernel tiene el placer de presentar su aportación más reciente a la tecnología ARM:

ODROID Magazine, Una revista gratuita mensual en formato PDF.

Esta moderna publicación electrónica ofrecerá las últimas noticias de la plataforma ODROID, así como artículos destacados de la comunidad de expertos que ha surgido en torno a esta sorprendente familia de microordenadores.

Dirigida a cualquier persona desde principiantes a expertos, ODROID Magazine presenta guías paso a paso para configurar su ODROID, instalar sistemas operativos y software, además de soluciones a los problemas más comunes. Para los usuarios más expertos contará con consejos y trucos, discusiones de seguridad, proyectos especializados y artículos técnicos que permitirán explorar todas las posibilidades que ofrece ODROID.

ODROID Magazine es una gran oportunidad para que la comunidad pueda compartir sus experiencias con todo el mundo.

Cada mes se publicarán una serie de temáticas para que cualquier persona pueda enviar sus aportaciones. Los artículos seleccionados que se incluirán en la revista serán premiados con regalos que se enviarán a los autores.

Los mejores artículos son lo que a través de una lectura sencilla, el lector pueda comprender y asimilar conceptos complejos. Es necesario enviar al menos una imagen por artículo y éste debe tener una extensión entre 500 y 2000 palabras.

¿Estás interesado en formar parte de ODROID Magazine? Envía un PM a @odroid, @mdrjr y/o @robroyhall en los foros ODROID, y nos pondremos en contacto con usted para darte más detalles. Habrá una pequeña compensación mensual para aquellos que deseen contribuir cada mes en el diseño y producción de la revista electrónica, incluyendo traducciones y correcciones. Sin duda, puede ser interesante para tu curriculum. ¡Colaborador activo en una de las comunidades más ingeniosas del mundo ARM!

ODROID

Magazine

Robert Hall, Editor Jefe

Soy un programador que vive y trabaja en Silicon Valley, CA, EE.UU. Diseño sitios web como Vevo, Hi5, Dolby Laboratories y Hyundai. Mis lenguajes principales son jQuery, angular JS y HTML5/CSS3. También desarrollo sistemas operativos pre-compilados, Kernels a medida y aplicaciones optimizadas para ODROID basadas en las versiones oficiales Hardkernel, por los cuales he ganado varios premios. Poseo una gran cantidad de ODROIDS, que uso para diversos fines: centro multimedia, servidor web, desarrollo de aplicaciones y plataforma de juegos.

Bo Lechnowsky, Editor

Soy el presidente de Respectech, Inc., consultoría tecnológica en Ukiah, CA, EE.UU. que fundé en 2001. Con mi experiencia en electrónica y programación dirijo a un equipo de expertos, además de desarrollar soluciones personalizadas a empresas, desde pequeños negocios a compañías internacionales. Los ODROIDS son una de las herramientas de las que dispongo para hacer frente a estos proyectos. Mis lenguajes favoritos son Rebol y Red, ambos se ejecutan en los sistemas ARM como el ODROID-U2. Tengo extensa experiencia con la mayoría de sistemas operativos conocidos.

Bruno Doiche, Editor Arte

¡Sólo soy un tipo al azar!

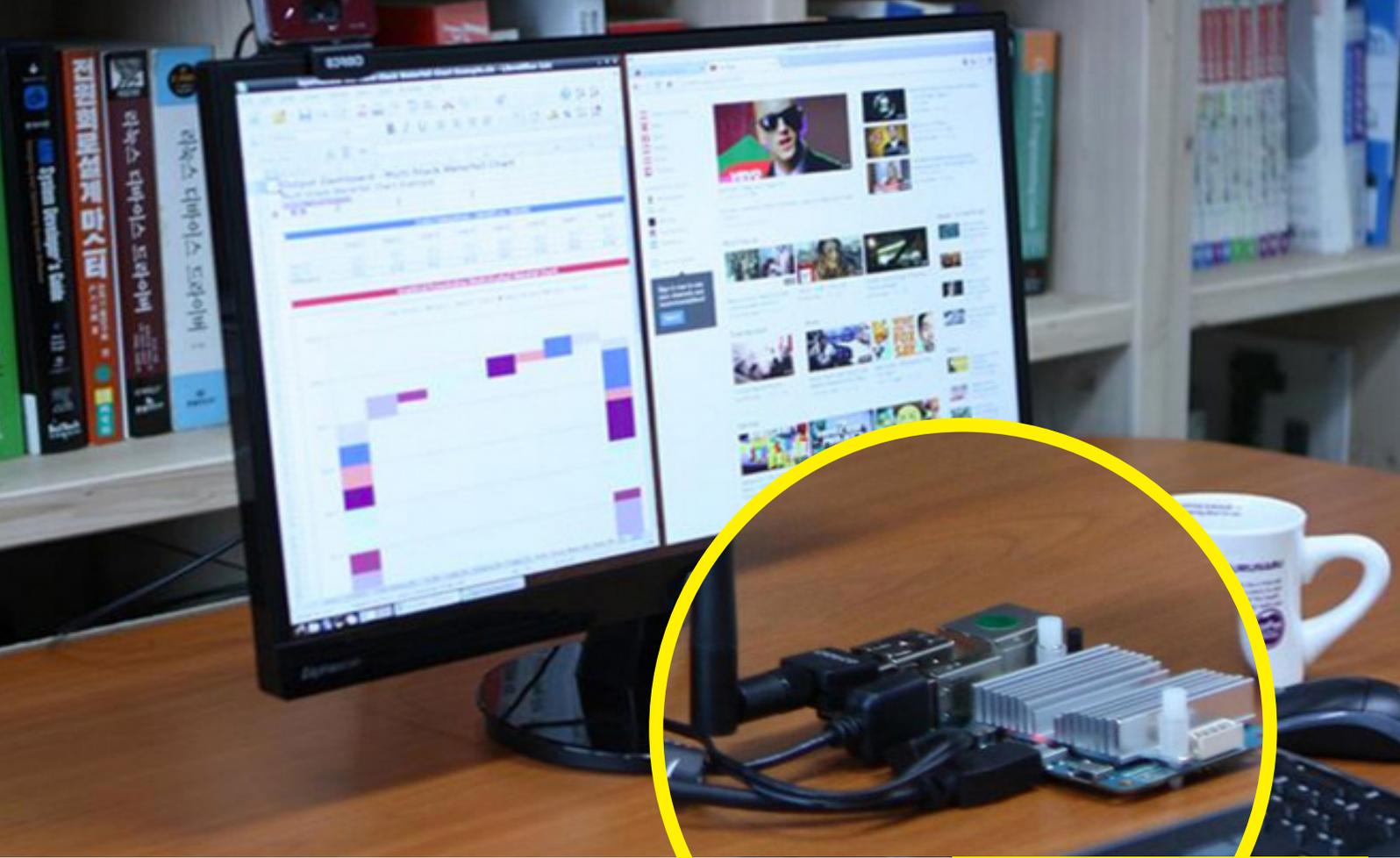
Manuel Adamuz, Traductor

¡Un espantajo más!

ODROID Magazine, publicada mensualmente en <http://magazine.odroid.com/>, es la fuente de todas las cosas ODROIDian. Únete a la comunidad ODROID en <http://forum.odroid.com/>, y explora las nuevas tecnologías que ofrece Hardkernel en [http://www.hardkernel.com/Hard Kernel, Ltd.](http://www.hardkernel.com/)

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.

Únete a la comunidad ODROID con miembros en más de 135 países en <http://forum.odroid.com/>.



COMENZAMOS CON EL ODROID-U3

Sigues esperando a actualizar tu sistema porque los precios de los actuales PC o Mac son muy elevados o buscas un segundo equipo para la familia, el trabajo o la universidad. La Familia Hardkernel de bajo coste, potentes dispositivos ARM que ahora son más asequibles con la llegada del U3 a un precio especial de 59\$.

Con un tamaño de una tarjeta de crédito y su bajo coste, ODROID-U3 puede verse como un simple equipo para aficionados. Sin embargo, su rendimiento, software y facilidad de uso lo convierten en una excelente alternativa a un PC de elevado coste. Compatible con Android y Linux permite ejecutar los miles de títulos disponibles de forma gratuita o con de bajo coste. ODROID-U3 también ofrece muchas ventajas frente al típico ordenador con Windows o OSX: un funcionamiento silencioso, consumo medio de 5W y su portabilidad, puesto que coge en el bolsillo de una camisa.

Su procesador es un Samsung Exy-

nos 4412 Quad-core de 1.7GHz con 2 GB de RAM. Presenta unas dimensiones de sólo 83 x 48 mm y un peso de 48 gr.

A diferencia de un PC, ODROID-U3 tiene la placa base, la memoria, la tarjeta de sonido e incluso el disipador de calor ensamblados. En lugar de usar un costoso disco duro como ocurren en un PC, el sistema operativo se ejecuta desde una pequeña y barata micro SD, como las utilizadas en las cámaras digitales. La tarjeta SD, así como el módulo eMMC (más rápido) pueden ser solicitados junto con el ODROID-U3, que incluirán el popular sistema operativo Ubuntu ya instalado.

Inserta la tarjeta SD en el lector, conecta un monitor, un teclado, un ratón, el cable Ethernet y el cable de alimentación, y pulse el botón de encendido. ¡Esto es todo lo que tienes que hacer para usar ODROID-U3! Navega por la Web, ejecuta juegos y programas de ofimática, edita fotos, desarrolla aplicaciones, y vea videos al instante. Si tiene un módulo eMMC, colócalo sobre

el área marcada en la placa y encájalo.

Es normal que ODROID-U3 incorpore un disipador conectado a la CPU para que disipe el calor sin la necesidad de conectar un ventilador.

¡Vamos a jugar (y trabajar) juntos!

El ODROID-U3 es adecuado para cualquier persona desde ingenieros de software profesionales que desarrollan proyectos de robótica hasta niños que aprenden a programar con el lenguaje Scratch. Capaz de ejecutar los últimos sistemas operativos Ubuntu 13.10 y Android Jellybean, es idóneo para programar, aprender, jugar, como centro multimedia, servidor web, realizar trabajos de oficina y universidad, como plataforma de hardware E/S entre otras muchas aplicaciones. Su potente procesador de 1.7Ghz de cuatro núcleos, el bajo coste, el consumo eficiente de energía y la amplia biblioteca de software disponible hacen de ODROID-U3 un ordenador moderno y perfecto para trabajar o jugar.

Una configuración típica con ODROID-U3. Un monitor de 21 pulgadas conectado al puerto HDMI. El Teclado, ratón, adaptador WiFi y cámara USB conectados a los 3 puertos USB.

Un Tour por la Placa

Vamos a empezar con rápido vistazo de lo que te encontrarás cuando abrás la caja. Es parecido a tu PC pero con algunas características adicionales.

A. El Procesador. El corazón de ODROID-U3 presenta el mismo procesador que utiliza el teléfono Samsung Galaxy S3. Se trata de un sistema de 4 núcleos a 1,7 GHz basado en la arquitectura ARM Cortex-A9. Cuenta con 2 GB de RAM unidos a la CPU.

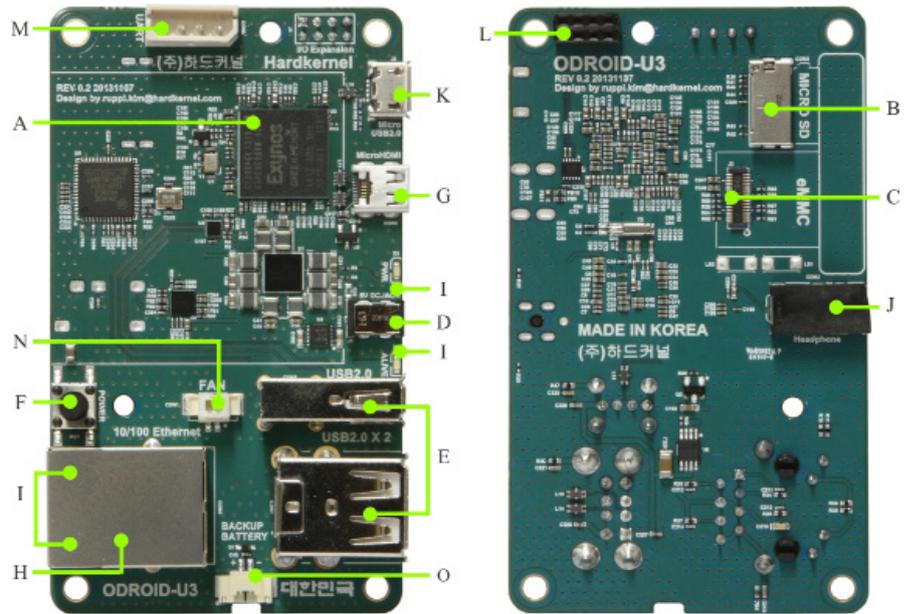
B. La Ranura para tarjetas Micro Secure Digital (MicroSD). Hay dos formas diferentes de almacenar el sistema operativo. Una es utilizar una tarjeta micro SD y la otra usar un módulo eMMC. Se utilizan normalmente como almacenamiento externo en los smartphones y cámaras digitales.

C. La Ranura del Modulo eMMC. El acceso al sistema de almacenamiento eMMC es 3-4 veces más rápido que la tarjeta SD. Hay 3 opciones de 8GB, 16GB y 64GB. El uso de eMMC aumentará la velocidad y capacidad de respuesta, similar a la que se experimenta cuando se cambia un disco duro mecánico (HDD) por una Unidad de Estado Solido (SSD) en un PC.

D. La clavija de Energía. Utilizado para la entrada de corriente de 5 voltios, con un diámetro interior de 0,8 mm y un diámetro exterior de 2,5 mm. El ODROID-U3 consume menos de 1A en la mayoría de los casos, pero puede subir a 2A si le conectamos directamente varios periféricos USB sin alimentación externa.

E. Los Puertos USB. Hay 3 puertos USB host 2.0. Puedes conectar un teclado, ratón, adaptador WiFi entre otros dispositivos. También puede cargar tu Smartphone con estos puertos. Si necesitas más de 3 puertos o tienes un periférico que requiere más potencia, puede utilizar un hub externo con alimentación.

F. El botón de Corriente. Puedes encender y apagar ODROID-U3. Una vez que se pulsa para apagar el sistema, verás un cuadro de diálogo en pantalla para confirmar la petición de apagado.



Los cuatro estado LEDs

ENERGIA	Rojo	Conectado al adaptador 5V/2A
ENCENDIDO	Azul	Luz tenue: Gestor de arranque se está ejecutando
		Luz firme: Kernel esta cargado
		Liz Parpadeante : Kernel esta funcionando
ETHERNET	Verde Amarillo	Indicador de actividad de la red Si la conexión de red es de 100Mbps

G. Conector HDMI. Para reducir al mínimo el tamaño de la placa usamos un micro HDMI Tipo D. La salida HDMI soporta resoluciones de 720p y 1080p.

H. Puerto Ethernet. El puerto Ethernet RJ45 estándar para conexiones LAN con una velocidad de 10/100 Mbps. La conexión Wi-Fi a través de un adaptador USB es otra opción que puedes utilizar.

I. Estados LEDs. ODROID-U3 tiene cuatro indicadores LED que ofrecen información visual (ver Tabla arriba).

J. Audio Analógico. Es un conector de audio analógico estándar de 3,5 mm. El conector tiene 4 polos para soportar una entrada de micrófono mono mediante los auriculares de los Smartphone Android.

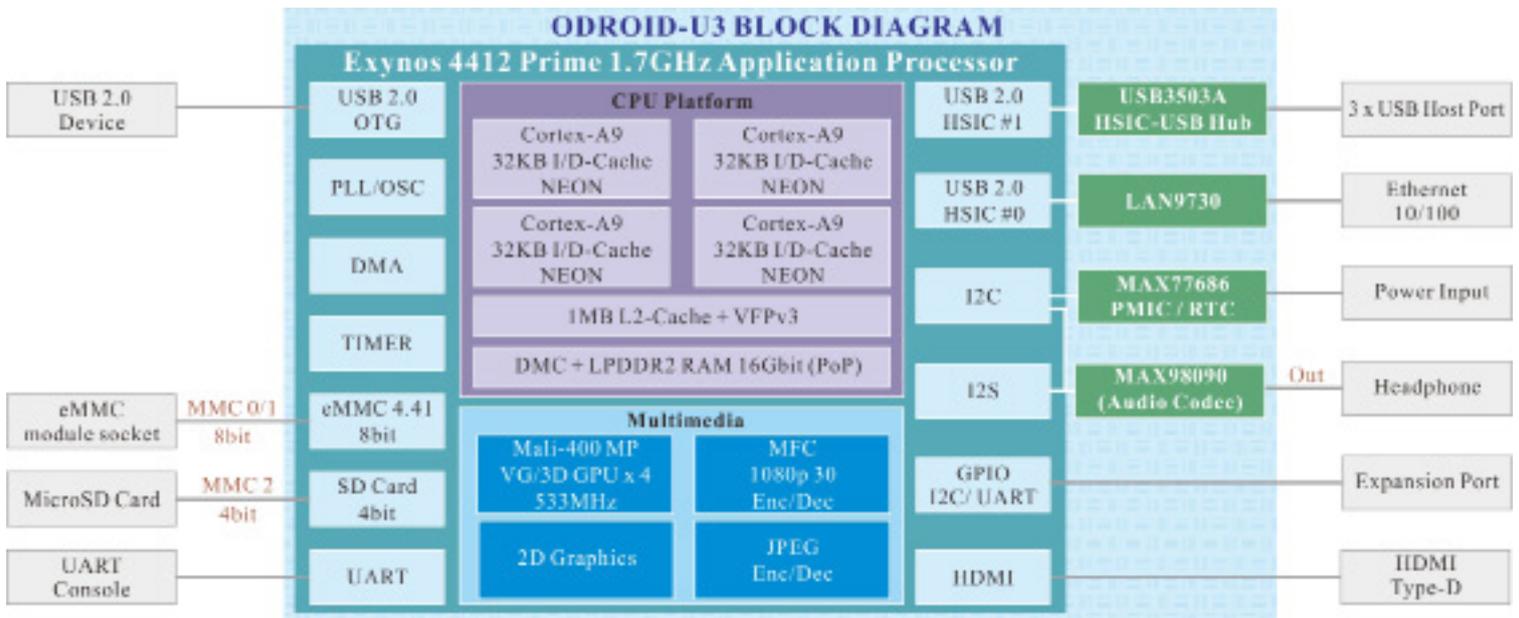
K. Conector Micro USB. Es el único conector Micro USB del dispositivo. Se puede usar con los drivers de Linux Gadget en tu PC, lo que significa que los recursos de ODROID-U3 se pueden compartir con los típicos PCs. No se puede utilizar como sistema de alimentación.

L. General Purpose Input and Output (GPIO). Se puede usar para IRQ/I2C/UART en electrónica y robótica. Estas conexiones pueden ampliarse a muchos más puertos PWM/ADC/GPIO.

M. Puerto Terminal Serial. Conectado a un PC permite el acceso a la consola de Linux. Puedes ver el registro del arranque o cambiar la configuración de vídeo o de red. Tenga en cuenta que utiliza una interfaz de 1,8 voltios. Recomendamos utilizar el módulo USB-UART de Hardkernel.

N. Salida PWM para el ventilador de refrigeración. Si conectas U3 en una habitación con una temperatura elevada, sería útil conectar un pequeño ventilador. La salida PWM aumenta o disminuye la velocidad del ventilador de forma proporcional a la temperatura de la CPU.

O. RTC (Real time clock) Conector batería de reserva. Si deseas añadir una función RTC para los archivos logs o el mantenimiento cuando se desconecta, basta con conectar una batería de reserva. RTC está implementado en ODROID-U3.



Los Periféricos

Ahora que conoces la distribución de la placa, necesitas conocer algunas cosas sobre los periféricos más apropiados (algunos se muestran más abajo) que puedes utilizar con ODROID-U3.

Además de estos periféricos, también necesitas un monitor, un teclado y un ratón. Cualquier teclado y ratón inalámbricos o conectados por cable USB son compatibles con ODROID-U3. El monitor o TV debe soportar resoluciones de 720p (1280x720) o 1080p (1920x1080), aproximadamente.

A. Una fuente de Alimentación. Este es el periférico más importante. Debes usar un adaptador de 5V/2A. Enchufe con diámetro interior de 0,8 mm y diámetro exterior de 2,5 mm, con centro positivo y exterior negativo.

B. Una Tarjeta MicroSD. Si no usas un módulo eMMC, necesitarás una tarjeta MicroSD. Se recomienda una MicroSD de 8GB clase 10. Las tarjetas de clase más baja pueden afectar al rendimiento.

La Tarjeta MicroSD oficial de Hardkernel tiene Ubuntu preinstalado y se puede arrancar inmediatamente después de abrir la caja.

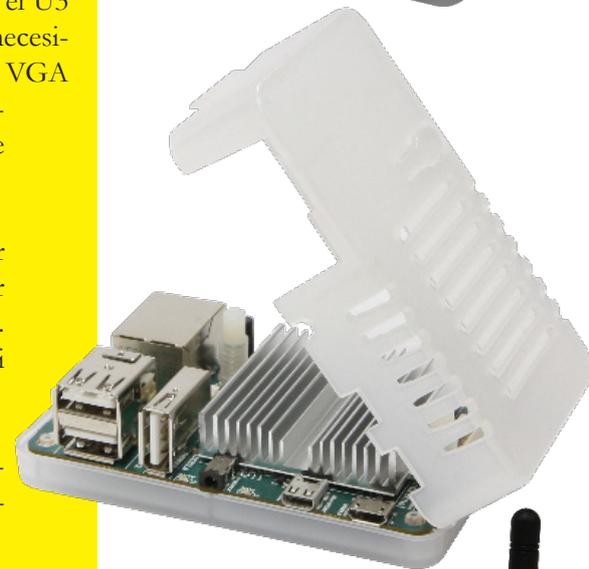
C. Un Módulo eMMC. Si quiere mejorar el rendimiento, elige el módulo eMMC. Es mucho más rápido que la MicroSD y también tiene preinstalado Ubuntu.

D. Un cable HDMI a micro HDMI. Necesitas este cable para conectar el U3 a un monitor. También podrías necesitar un adaptador para un monitor VGA ya que ODROID-U3 no tiene salida VGA. Se recomienda el cable HDMI de Hardkernel.

E. Adaptador WiFi. Puede utilizar un adaptador USB WiFi en lugar de la conexión de red por cable. Recomendamos el adaptador WiFi RTL8188CUS.

F. Caja. Para proteger tu ODROID-U3 de daños inesperados te recomendamos una carcasa protectora.

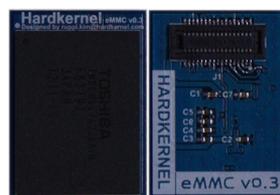
Un esquema en bloques que muestra la arquitectura del ODROID-U3.



Fuente Alimentación



Tarjeta MicroSD



Módulo eMMC



Cable HDMI



Adaptador WiFi

USAR ODRUIDS EN LA INFORMATICA DE ALTO RENDIMIENTO (HPC)

ARM: MUY POR ENCIMA DE LA COMPETENCIA.

Kurt Keville, MIT

@<http://meegs.mit.edu>

¿Por qué HPC?

Para dar servicio los modernos centros de datos usan demasiada electricidad y aire acondicionado. Un proveedor de servicios de Internet con ARM puede cargar páginas web con un menor consumo de energía que si utiliza arquitecturas convencionales (<http://tinyurl.com/ApacheOnARM>). Al mismo tiempo, la comunidad científica está adoptando con cierta rapidez la tecnología ARM por su capacidad de cálculo matemático en operaciones complejas de Coma Flotante, su mínimo consumo de energía y una planificación de desarrollo con vista a mejorar aún más su rendimiento por vatio. Desde la comunidad de desarrolladores ARM, también ha crecido el interés por HPC y su uso en áreas más amplias. En el ámbito académico existe igualmente la denominada informática de Alto Rendimiento Extrema (<http://www.ieee-hpec.org/>) y la supercomputación IEEE que fue presentada en la reciente conferencia SC13 en Denver (Colorado). Además, el Clúster IEEE ha experimentado un aumento de las publicaciones centradas en la tecnología ARM.

Los centros de supercomputación modernos tienen miles de núcleos dedicados a sus necesidades de procesamiento. Cada vez que un programador logra ejecutar una aplicación que permite mejorar el rendimiento (por vatio, dólar o metro cuadrado) representa un triunfo tanto para el propio centro como para los clientes a los que se les dan soporte. Cada vez más se dan circunstancias en las que las aplicaciones se deben ejecutar varias veces y donde comparten recursos en HPC, parecido a la nube. Se vuelve a la vieja técnica del uso compartido de recursos.

¿Por qué ARM?

La arquitectura ARMv7 ha resultado ser un auténtico desafío para HPC en numerosas ocasiones en las que anteriores arquitecturas ARM no lo han sido. Al margen de su eficiencia energética, ARM se presenta como la tecnología en vía de desarrollo de los denominados "sistemas embebidos". Cuando un centro de datos se compone de más de 10 mil núcleos, las mejoras graduales se traducen en importantes ventajas. Estos pequeños cambios pueden representar un significativo ahorro de espacio y energía. Cuando la memoria está compartida entre la CPU y la GPU los SoCs ARM (Sistema en un Chip) duplican las extensiones SIMD (Una Instrucción, Múltiples Datos) en Cortex-A15 NEÓN GPU mejorando considerablemente el acceso a la memoria. Los beneficios aparecen en el momento adecuado, cuando la solicitud (aplicación) y el conjunto de datos se encuentran en la cache on-die. Con la creciente acogida del sistema GP-GPU (General Purpose GPU) y la expansión de aplicaciones de tipo HPC basadas en aplicaciones complejas, los rápidos modelos de cálculo ARM son ahora más importantes que nunca, una tecnología con una participación cada vez mayor de HPC.

¿Por qué ODRUID?

Actualmente, ODRUID tiene un procesador Exynos con 4 núcleos ARM. La próxima serie Exynos5 tiene 8 núcleos, 4 de los cuales son ARM Cortex-A15. Hardkernel así como el consorcio de investigación de Soluciones Informáticas han experimentado mejoras significativas de potencia

y rendimiento en la serie XU, en comparación con otras arquitecturas de hoy en día. Con su ambicioso calendario de lanzamientos en nuevas tecnologías, Hardkernel ha adoptado esta plataforma siguiendo la ley de Moore y la tecnología ARM, lo que les ha permitido unirse a la ola de los más novedosos, mejores y de bajo coste sistemas de alto rendimiento, a la vez que su compatibilidad ha ido creciendo significativamente.

Es más, Cortex-A15 gana en la mayoría de las pruebas de rendimiento ARM-HPC (<http://tinyurl.com/ODROID-HPC>). El equipo de Soluciones Informáticas de Entornos de ejecución ha evidenciado las principales ventajas del A15 sobre A9 en el desafío HPCC, el estándar de evaluación preferido para HPC (<http://hpcchallenge.org/>). Esta prueba utiliza solamente los núcleos A15 de Exynos 5410 y los mantiene a su máxima capacidad durante el ensayo (<http://tinyurl.com/ODROID-LINPACK>), la cual no es una comparación completamente equitativa. De cualquier modo, se puede extraer la información más importante; XU Jessie es dos veces mejor que U2 Whisper en G-HPLINPACK.

Conclusión

Hoy en día podemos ejecutar muchas aplicaciones HPC en ODRUID, y la creciente demanda de eficiencia energética lleva a los profesionales de la industria a replantearse el diseño de los centros de datos. El futuro es muy prometedor a medida que la denominada "Informática Exascale" marca el inicio de un nuevo y emocionante contexto en las tecnologías SoC embebidas.

INSTALAR UN S.O. EN UN ODROID

CRONICAS DE UN CIENTIFICO LOCO

Bohdan Lechnowsky
@bo@respectech.com



Estás sentado en tu laboratorio semi-oscuro. Mientras trabajas en tu próximo proyecto para dominar el mundo que a la carrera garabateaste en la servilleta de la cena de anoche, te sorprende el sonido de un claxon. Aunque en el momento de instalar la bocina pensaste que sería un modo idoneo para que un científico loco fuese alertado de un intruso o visitante, ahora lo estás reconsiderando mientras te frotas la mano quemada. No habías tenido en cuenta la posibilidad de ser sorprendido mientras usas el soldador de rayos láser.

Llegas a la puerta principal, pero el autor del incidente que te ha provocado la quemadura ya no está. A medida que revisas tu material de vigilancia y activar tu sistema perimetral láser, observas una pequeña caja marrón junto a la puerta con la etiqueta "Hardkernel". Tu corazón palpita con excitación al recoger el diminuto paquete. Una vez que conoces como aprovechar la energía, el mundo tan sólo ha comenzado a darse cuenta.

Mientras bajas en el ascensor a tu cámara subterránea, se te pasan por tu mente retorcidas ideas. A toda prisa, abres el paquete rompiéndolo mientras te diriges a la zona de ordenadores del laboratorio. Comienzas la tarea aburrida de enchufar el cable micro-HDMI al monitor LED, el adaptador inalámbrico USB para teclado/ratón y la fuente de alimentación desde una toma de corriente a la unidad

ODROID. Piensas en, "podría usar una salida normal, pero prefiero la conmutada." Algún día, cuando logres dominar el mundo, te permitirás el lujo de tener un ayudante a quien podrás gritar: "¡Activa el interruptor!" hasta entonces, pulsa el interruptor por ti mismo. Ahora es el momento de dar una carcajada malvada mientras se enciende la unidad.

Al instante te das cuenta que hay un problema. ¡No aparece nada en la pantalla! Miras de un lado a otro pensando en decenas de posibilidades mientras consideras que algún enemigo sería lo suficiente inteligente como para impedir tu retorcidos planes. Al buscar alguna pista evidente, te percatas de que alguien te ha robado la tarjeta eMMC. Solo entonces, te acuerdas de que nunca compraste una tarjeta eMMC (o una tarjeta microSD, para el caso)

Como científico loco, instintivamente sabes que las tarjetas eMMC son mucho más rápidas, pero requiere un adaptador para conectarlas al ordenador y son también más difíciles de conseguir que las tarjetas microSD, que son muy comunes, de bajo coste y disponibles en la droguería en la calle de abajo. (Tus instintos se apoyan en datos concretos y reales recogidos en algún otro sitio de este número de ODROID Magazine.) "No importa", te dices, "el método es el mismo en ambos casos."

A continuación, te preguntas ¿De-

bería usar Windows o Linux para grabar la imagen? Mientras analizas los pros y contras, inicias la descarga del sistema operativo elegido desde <http://dn.odroid.com>. Ahora que tienes tu imagen del sistema operativo descargada, puedes garabatear en la parte trasera de uno de tus planes de dominación del mundo ya abandonado:

Sabes que los mejores planes de dominación requieren ensuciarse un poco las manos, por lo que te agrada hacerlo. Por supuesto, eliges el método de línea de comandos de Linux ya que es el que mejor se ajusta a tu perfil de científico loco.

Coge la tarjeta de tu estación de trabajo, apaga tu ODROID (pulsa el interruptor), inserta la tarjeta en ODROID, y grita "¡PULSA EL INTERRUPTOR!" a nadie en particular, pero esta vez eres consciente de que funcionará tu siguiente plan para dominar el mundo.

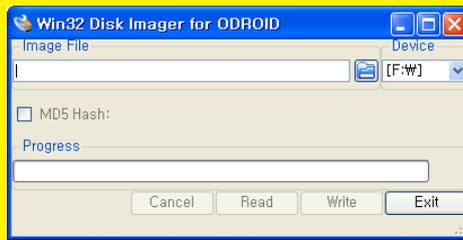
Por unos instantes, tu rostro refleja un cierto desconcierto al ver la pantalla en negro, pero después te das cuenta que es normal ya que ODROID no tiene salida en modo de consola. ODROID se muestra gustosamente una vez que se inicia el entorno gráfico por completo. En este punto se introduce los datos supersecretos de acceso (introduce "odroid" como usuario y "odroid" como contraseña), y tu gigantesca pantalla LED se ilumina.

¡PARA DOMINAR EL MUNDO!

Windows

Descarga Win32DiskImager.exe para ODRROID v1.1 disponible en <http://www.hardkernel.com/?b144>. Windows debería abrir automáticamente el archivo zip después de descargarlo permitiéndote copiar el archivo .exe en su escritorio o en otra ubicación. La ventaja de esta versión sobre la versión disponible en Sourceforge.com es que permite limpiar la memoria flash antes de escribir y verifica los datos durante el proceso de escritura. Esto hace que el proceso necesite el doble de tiempo (tres veces más si realizamos primero una limpieza).

El archivo de descarga puede estar comprimido en Formato .xz Para descomprimir, es necesario descargar 7-zip o una utilidad similar desde Internet. 7-zip se encuentra disponible en <http://www.7-zip.org/download.html>. Si usa 7-zip, es necesario abrir el explorador de archivos y navegar hasta la ubicación de la descarga.



A continuación, abra el archivo y guarde el contenido.

El campo "Device" está vacío si inicias Win32DiskImager antes de insertar la tarjeta. ¡Recuerda que no debes hacer esto!. Una vez guardado el archivo .img, asegúrate que la tarjeta microSD/eMMC esta insertada en el puerto USB.

Después, abre Win32DiskImager.

Si desea borrar los datos anteriores, recuerda limpiarla primero. Después, haga clic en el botón "Browse", busque el archivo .img y pinche en "Write". Ahora puedes trabajar en otros proyectos durante un tiempo, hasta que finalice el proceso.

Línea de Comandos Linux

dd ya está incluido en muchas distribuciones, por lo que no es necesario descargar nada.

xz ya está incluido en muchas distribuciones, pero si necesitas instalarlo, usa el comando `sudo apt-get install xz-utils` (el método varía dependiendo de la versión de Linux).

Introduzca `unxz my-odroid-image.img.xz` para extraer el archivo .img (Reemplaza "my-odroid-image.img.xz" por el nombre del archivo de imagen descargado).

Introduzca `df -h` y toma nota de los dispositivos montados.

Inserte la tarjeta microSD/eMMC y ejecuta `df -h` de nuevo. Debe aparecer al menos un nuevo dispositivo en la lista. Verá algo así como `"/dev/mmcblk0p1"` o `"/dev/sdd1"`. La última parte ("p1" o "1") es el número de partición. Sin embargo, para escribir toda la tarjeta, necesitas eliminar esa parte del nombre (en el ejemplo anterior `"/dev/mmcblk0"` o `"/dev/sdd"`) como dispositivo para

toda la tarjeta. Si hay más de una partición en la tarjeta, ésta puede aparecer más de una vez en el resultado de "df". Desmonte la tarjeta mediante:

```
sudo umount /dev/mycard
```

Donde `"/dev/mycard"` es reemplazado por el nombre completo de la partición que localizas en tu sistema (como `"/dev/mmcblk0p1"` en el ejemplo anterior). Necesitas ejecutar este comando para cada partición que se monta en la tarjeta.

Recuerda que, "dd" es sinónimo de "borrado de disco". Si no usar el comando adecuadamente, puede eliminar todo el contenido del disco duro. (Esto me da una idea: ¡Ejecutar este comando con una determinada configuración en el caso de que mi equipo caiga en manos de un enemigo!)

```
sudo dd bs=1M if=my-odroid-image.img of=/dev/mycard
```

Reemplaza "my-ODROID-image.img" por la ruta donde se encuentra el

archivo de imagen que ha descargado.

Reemplaza `"/dev/mycard"` por el nombre completo de la tarjeta anotado en los pasos anteriores.

Esto tardará varios minutos, puedes pasar a otros proyectos hasta que finalice el proceso. No es posible mostrar el progreso. Si deseas ser notificado con una alerta de audio cuando "dd" haya terminado, introduce el siguiente comando en lugar del anterior:

```
sudo dd bs=1M if=my-odroid-image.img of=/dev/mycard && echo -e '\a' || (echo -e '\a'; sleep 1; echo -e '\a')
```

Escucharás un pitido si "dd" finaliza el proceso con éxito y dos pitidos si falla por cualquier razón.

Cuando finalice "dd", ejecuta `sudo sync` para vaciar la caché de escritura (por si acaso).

Ejecuta `df -h`, usa el mismo método anterior para desmontar las particiones de la tarjeta y luego retire la tarjeta del ordenador.

Horas más tarde, bien entrada la noche...

Oyes aproximarse tormentas. Las luces brillantes de los equipos de tu laboratorio se apagan momentáneamente y ves como la red eléctrica intenta hacer frente a las fluctuaciones causadas por los rayos. Ha realizado varios cambios en la tarjeta

desde que encendiste tu ODRROID. No queriendo que tus proyectos fracasas por un fallo técnico elemental, decides realizar una copia de seguridad.

Una forma sencilla de hacer esto es usando Win32DiskImager en Windows. Hay maneras de hacerlo igualmente en Linux (`dd if=/dev/mycard of=/home/`

`user/My-odroid-image.img bs=8M`). Inserte la tarjeta en tu ordenador con Windows, inicie Win32DiskImager, seleccione la letra de unidad de tu tarjeta, introduce un nombre del archivo para la imagen con la copia de seguridad y haga clic en "Read". El contenido de la tarjeta se lee y se guarda con total seguridad.

JUEGOS LINUX EN ODROID

LA SIGUIENTE EVOLUCIÓN EN LOS JUEGOS

Tobias Schaaf

@schlurf82@googlemail.com

Los videos juegos siempre han jugado un papel importante en mi vida y es la razón principal de mi participación activa en ODROID. En este artículo me centraré en los juegos de Linux, me parece que Linux es un sistema operativo mucho más interesante que Android. Además, la mayoría de los juegos en Android están hechos para pantallas táctiles y apenas soportan mandos. Hay algunos juegos para Android que admiten teclado o mandos, que igualmente funcionan muy bien en ODROID. Se ven estupendamente y son muy divertidos, pero si ya tienes un dispositivo Android en casa, entonces sabrás que en un ODROID puede jugar a cualquier juego de Android con una TV HD.

Me parece más interesantes los juegos en Linux, especialmente por el desarrollo de juegos y emuladores que pueden ser ejecutados en un sistema como ODROID, con un entorno de escritorio que ofrece muchas más posibilidades que Android. Por ejemplo, prefiero la combinación de diferentes elementos en un pack completo, como mi imagen ODROID GameStation Turbo. Esta agrupa en un único pack un sistema multimedia y una plataforma de juegos, pero en realidad se compone de muchas partes que interactúan entre sí. Este nivel de personalización es mucho más difícil de lograr en Android.

He de decir que Android está basado en Linux pero lo considero

como un sistema operativo independiente, ya que actúa de forma distinta al entorno normal de Linux.

Primer acercamiento

Cuando ODROID U2 y X2 eran recientes y los drivers Mali 3D acababan de salir, me pregunté qué podía hacer con ellos. Busqué en la red juegos OpenGL ES (GLES), pero era muy difícil localizar versiones GLES. La mayoría de los juegos y programas que encontré para GLES sólo eran para dispositivos Android, y resultaba bastante difícil de encontrar versiones GLES nativas. Aunque @mdrjr (uno de los principales desarrolladores Hardkernel) encontró una versión de Quake 3 para Mali, es difícil encontrar algo más que se pueda ejecutar en ODROID usando GLES.

Finalmente, localicé mi primer juego para compilar en GLES: Descent 1 y 2 Rebirth. El juego funciona muy bien en ODROID a una velocidad constante de 100 FPS (fotogramas por segundo) lo cual era impresionante, y GLES hacía que el juego se viese realmente bien.

Aparte de Descent, resulto muy difícil encontrar algo para Linux que usara GLES, por lo que probé otras opciones de juegos en Linux para ODROID.

Una forma fácil de ver lo que hay disponible es echar un vistazo al Centro de Software de Ubuntu en la imagen de Linaro 12.11 (basado en Ubuntu 12.04) proporcionada por Hardkernel. Hay cientos de juegos en el Centro de Software, muchos de ellos son juegos en 2D. Rápidamente encontré viejos clásicos en Internet, como Dune Legacy, un clon de Dune 2 con imágenes pre-compiladas para armhf que funcionan sin realizar modificaciones.

Viendo estos juegos y observando las técnicas que hay detrás, llegué a la conclusión de que aunque estos juegos se hicieron para sistemas x86 y usan principalmente la CPU (y no la GPU), estos juegos también se ejecutan en ODROID. De hecho, esto demuestra que ODROID tiene un gran potencial si lo comparamos con otras placas ARM, como Open Pandora o Raspberry Pi.

Mientras que para Open Pandora o Raspberry Pi se tenían que desarrollar versiones específicas de juegos y emuladores (e.j. Emulador Amiga), que requieren de una gran cantidad de ajustes y optimizaciones. Nosotros hemos sido capaces de usar los emuladores x86 ya existentes y

Descent - Comparación entre el rendimiento de software y hardware.



Referencias de juegos y versiones SDL:

http://en.wikipedia.org/wiki/List_of_games_using_SDL
http://libregamewiki.org/SDL_games

únicamente los hemos recompilado para ODROID. La CPU es lo suficientemente potente como para ejecutar este tipo de programas y juegos en modo software.

Esto es posible gracias a SDL, siglas de Simple DirectMedia Layer. Muchos de los programas y juegos se ejecutan sobre SDL y son fáciles de compilar para ODROID.

Los emuladores son una buena forma de conseguir juegos de antiguas consola como NES, SNES, SEGA, GBX, Amiga, NeoGeo, o incluso juegos de PlayStation 1 que se ejecutan en PCs de hoy en día. Una vez más, mientras que otras placas tienen problemas con

permitiéndome ejecutar muchos emuladores con un solo programa y representa uno de mis primeros grandes avances en los emuladores, dado que soporta escaladores de software como hq2x entre otros.

Jugar con emuladores es una muy buena experiencia para probar antiguos juegos haciendo uso del hardware moderno. Juegos que en su día se ejecutaban en pequeños televisores CRT ahora son capaces de funcionar con altas resoluciones en grandes televisores TFT.

El potencial de ODROID combinado con SDL hace que todo esto sea posible.

Desarrollo Tecnológico: OpenGL ES (GLES)

Ha pasado algún tiempo desde que desarrolle la primera versión y mucha gente ha trabajado duro para mejorar ODROID. Un número cada vez mayor de juegos y programas GLES han sido trasladados a Linux, junto con la versión de Android. En la actualidad hay bastantes juegos que se ejecutan bajo GLES y presentan grandes gráficos. Juegos como Hurrican (un clon gratuito de Turricon) y Extreme Tux Racer funcionan bajo GLES y se ejecutan perfectamente en ODROID.

Pero no éramos el único grupo de trabajo que exportaba juegos GLES a placas ARM. Así pues y con la ayuda de @pitSeb de la comunidad Open Pandora, que ha hecho muy buen trabajo exportando juegos que utilizan el motor de Quake 3, ha sido posible trasladar incluso juegos más espectaculares a ODROID.

Ahora contamos con RetroArch, un emulador multi-sistema modular, que se ejecuta sobre GLES y utiliza diferentes núcleos para emular diversos sistemas.



Izquierda: freedroid RPG
 Arriba: Corsix-TH
 Más arriba derecha: Legend of Edgar
 Arriba derecha: Open Jazz
 Abajo derecha: OpenTyrian
 Más abajo derecha: Jagged Alliance 2 - Stracciatella
 Abajo - PainTown
 Abajo izquierda - OpenXCom

La Magia de SDL

La Búsqueda de SDL y 2D me ayudó a encontrar un montón de juegos y emuladores. Así versiones como Freedroid RPG, Corsix-TH (un clon de Theme Hospital), Flare, KoboDeluxe, Legend of Edgar, Open Jazz, OpenXCom, OpenTyrian, Zod Engine, PainTown, Jagged Alliance 2 - Stracciatella y ScummVM podían perfectamente ejecutarse en ODROID.

Mientras que probar un solo juego es interesante y puede ser muy divertido, aún más interesante es poder jugar a cientos de juegos usando una única plataforma: un emulador.

algunos emuladores debido a la limitada potencia de su CPU, ODROID es capaz de ejecutar incluso emuladores de SNES y Amiga en SDL con relativa facilidad. Podría rellenar este artículo con miles de fotos de juegos de todos los emuladores que soporta ODROID.

Uno de los primeros emuladores multi-sistema que usé fue Mednafen,



Arriba: Jedi Knight 3
 Jedi Academy
 Arriba izquierda: Conflict:
 FreeSpace -
 The Greater War
 Derecha: FreeSpace 2
 Abajo izquierda: Open Arena
 Más arriba derecha: UFO -
 Alien Invasion

Esto permite que muchas consolas puedan ser emuladas sin problemas en ODROID. Por ahora, las opciones están limitadas a los blobs binarios Malí proporcionados por Samsung.

@AreaScout, colaborador de los foros ODROID, nos comenta que en

más que con los drivers estándar de Mali.

Las opciones de juego en ODROID son muy extensas, con las nuevas versiones de emuladores PSP y NDS. Un grupo incluso ha exportado Dolphin (Emulador Nintendo Wii y GameCube) para ejecutarse en las placas ARM. No todo funciona correctamente de momento, pero con el gran número de personas dedicadas al desarrollo ARM y con placas ARM cada vez más rápidas, los avances serán visibles muy pronto.

Hardkernel ofrece con cada generación de placas el hardware más reciente hasta la fecha. El último ODROID-XU es mucho más potente que el X2 o U2, y los emuladores que se ejecutan en esas arquitecturas puede hacerlo mejor en el XU, y aún mejor en la próxima generación de ordenadores Hardkernel..

Un Futuro Prometedor

Los Juegos en Linux han sido, en mucho tiempo, algo que la gente no ha tomado en serio. Pero ahora con Steam disponible oficialmente para Linux, pack de juegos como Humble Bundle así como otros están siendo exportados a Linux. Tan sólo es una cuestión de tiempo hasta que estos juegos estén también disponibles para dispositivos ARM

Canonical está centrandose en el mercado móvil con un diseño unificado de Linux. ODROID nos acerca un poco más al hecho de poder utilizar un smartphone como una estación con múltiples conexiones convirtiéndolo en un potente PC de escritorio.

USAR ODROIDS EN APLICACIONES COMERCIALES DE INFORMÁTICA DE ALTO RENDIMIENTO (HPC)

Anthony Skjellum, Cooper Filby, RunTime Computing Solutions, LLC
 @ <http://www.runtimecomputing.com>

Kurt Keville, MIT
 @ <http://meegs.mit.edu>

La tecnología ODROID, basada en ARM multinúcleo y una Unidad de procesamiento gráfico de aplicación general (GP-GPU), ofrece una tecnología microservidor que está a punto de transformar el mundo de la informática en la nube, la informática paralela y los sistemas embebidos. Este artículo muestra algunas tecnologías interesantes de los sistemas actuales, con un precio inferior a 200 dólares (sin incluir los dispositivos de memoria masiva).

Tecnología big.LITTLE

El ODROID XU + E presenta arquitectura ARM big.LITTLE, con procesadores de bajo consumo y alta potencia: Cortex- A7 y Cortex- A15. Con

ocho núcleos, ¿Cómo se distribuyen los procesos? El objetivo de la arquitectura big.LITTLE es reducir el consumo de energía usando los núcleos A15 más potentes cuando sea necesario. Los algoritmos de programación de procesos del kernel reflejan precisamente esto. Basándonos en los datos que hemos recopilado usando sensores en el XU + E, hemos concluido que el algoritmo de programación de los procesos de migración del clúster funciona. En este algoritmo de programación los procesadores A7 son usados hasta que la carga del procesador alcanza un cierto nivel, momento en el que los núcleos A15 se activan y se hacen cargo de cálculo hasta que la carga disminuye de nuevo. Otros algoritmos

de programación big.LITTLE incluyen In-Kernel Switcher (IKS) que empareja cada núcleo A7 con un núcleo A15 para la migración del proceso y Global Task Schedule (GTS), que distribuye los procesos entre los núcleos en base a sus necesidades de procesamiento. Esto hace que el bajo consumo y el alto rendimiento sean posibles en una única arquitectura. El procesador A15 se complementa con una GP-GPU que permite en una única instrucción multiplicar la aceleración de datos, además del procesamiento vectorial de precisión individual con las extensiones de coma flotante A15 NEON. Para muchas aplicaciones en la nube, la precisión individual es más que suficiente para los cálculos de coma flotante.



Gestión de Puppet y Clúster

Con ODROID XU + E y Ubuntu, hemos podido instalar y configurar correctamente un clúster HPC usando software gratuito. Para una mejor administración del nodo usamos Puppet, que es un sistema de gestión de contenidos para sistemas Linux. Con Puppet, hemos sido capaces de crear una configuración básica para todos los ODROIDs de nuestro clúster, lo que garantiza una configuración homogénea para nuestros nodos. Además, si alguna vez necesitamos añadir más programas a nuestra configuración básica, sólo tenemos que actualizar la configuración en nuestro servidor Puppet y los cambios se aplicarán al resto del clúster. La flexibilidad de Puppet nos permite hacer incluso tareas como la configuración de los usuarios locales, autofs y LDAP, todo al mismo tiempo que añadimos con facilidad nuevos nodos.

Aplicaciones en la nube

La informática en la nube basada en varios ODROID XU + E de baja potencia ofrece una interesante alternativa a procesadores x86-64 sumamente virtualizados. Al ofrecer muchos más sistemas pequeños, junto con una red gigabyte habilitada por el USB 3.0 del XU + E, las arquitecturas en la nube posibilitan los niveles de red al igual que las arquitecturas “más tradicionales” que virtualizan mediante la capa del sistema operativo host. Debido a que los sistemas son relativamente

baratos, es posible un mayor control del hardware físico dentro de la infraestructura en la nube. Esto es comparable a algunos de los proyectos actualmente, como HP Moonshot, hecho realidad hoy en día, sin marca registrada. Cualquiera puede desarrollar una nube XU + E. Además, la capacidad E/S de los módulos eMMC ofrecen un rendimiento/velocidad superior a 100Mbyte/s y su capacidad llega actualmente hasta a los 64 gigabytes. Esto permite una mayor carga de trabajo.

Soporte para librerías matemáticas de alto rendimiento

Las librerías matemáticas tanto gratuitas como comerciales tienen o pronto tendrán soporte para ODROID. ATLAS, por ejemplo, (<http://sourceforge.net/projects/math-atlas>) se encuentra en pleno desarrollo para la familia ARM y ya ha demostrado beneficios para muchas aplicaciones que utilizan procesos BLAS.

Además, las librerías comerciales para el procesamiento de imágenes y señales están en desarrollo, en particular las basadas en el estándar VSIPL (<http://www.omg.org/hot-topics/vsipl.htm>). Por ejemplo, “RunTime Computing Solutions” (www.runtime-computing.com) pronto lanzará VSIPL para varios núcleos ARM. Esto facilita la transición de plataformas más antiguas, como la familia de procesadores PowerPC Altivec directamente a ARM

con el correspondiente traslado de rendimiento, logrando que éste sea más alto en las bibliotecas de código abierto como FFTW.

Problemas ODROID

Desafortunadamente, no existe el sistema perfecto y ODROID XU + E no iba a ser una excepción. Hemos tenido que solventar dos problemas cuando desarrollamos nuestro clúster con NFS y LDAP por primera vez. Nos encontramos que el kernel usado en la imagen de Xubuntu 13.10 en realidad no soportaba NFS. Este se solventó fácilmente con UNFS3 disponible gratuitamente y ejecutándose en el nodo principal. En segundo lugar, nos encontramos con algunos problemas al configurar los nodos para la autenticación en el servidor LDAP en el nodo principal usando `nscd` y `ldap-auth-client`. La solución a este problema pasaba por vincular simbólicamente el archivo `/lib/arm-linux-gnueabi/nss_ldap.so` a `/lib/libnss_ldap.so.2`. Las buenas noticias son que el soporte de Linux está evolucionando rápidamente, y hemos visto grandes mejoras en el rendimiento y la funcionalidad con cada versión.

Conclusiones

Las opciones de HPC disponibles en ODROID son innumerables. Ahora podemos ejecutar actividades tradicionales de HPC como el arranque por red, soporte NFS, enrutamiento múltiple, control `cpufreq` de carga (que puede estar conectado a un sistema de colas como `slurm`), compilar y vincular aplicaciones modernas de supercomputación con librerías empresariales como MPI y OpenMP.

Y ahora, al pasar nuestro clúster y HPC a ARM, podemos rastrear con facilidad el actual desarrollo de microservidores ODROID, pendientes de la liberación de los cada vez mejores procesadores ARM.

RECONOCIMIENTO OCULAR MOVIL EN TIEMPO REAL CON LA PLATAFORMA DE DESARROLLO ODROID

Christopher D. McMurrugh
@murrman@sbcglobal.net

En este artículo se recoge una introducción al reconocimiento ocular usando un sistema de bajo coste y la plataforma de desarrollo ODROID U2. Nuestro objetivo es probar el potencial del procesador como plataforma de vídeo oculografía en tiempo real. Duplicar esta configuración experimental resulta muy barato por la disponibilidad de componentes de hardware de bajo coste y software de código abierto.

INTRODUCCION

El reconocimiento ocular por vídeo o vídeo oculografía, es una tecnología emergente que tiene la capacidad de cambiar el modo en que las personas interactúan con sistemas inteligentes. La utilidad del seguimiento ocular ya ha sido probada en usuarios con graves discapacidades físicas, como parálisis, ELA y enfermedades neurológicas, proporcionando una funcionalidad del tipo manos libres. En estos sistemas, la interacción persona-ordenador se establece mediante el seguimiento de la mirada del usuario en un punto concreto (PoG) sobre un monitor, en lugar de depender de un teclado o ratón estándar o del reconocimiento de voz.

Existen diferentes tipos de dispositivos de vídeo oculografía tanto a nivel comercial como de investigación, pero el principio general de funcionamiento es el mismo. En primer lugar, una cámara (por infrarrojo) observa el ojo del usuario y calcula la ubicación de la pupila. A continuación, un proceso de calibración conecta las posiciones de la pupila a un sistema de coordenadas, mientras que el

usuario mira los puntos en la pantalla. Por último, un PoG interpola los puntos de calibración.

Mientras que el reconocimiento ocular es una nueva y apasionante tecnología, los dispositivos comerciales tienden a ser muy caros. Además, hay muchas cuestiones abiertas que actualmente impiden que la tecnología sea usada más ampliamente. Estas limitaciones, sin embargo, no impiden que un simple aficionado o un investigador realicen sus propios experimentos. Este artículo ofrece ideas sobre cómo iniciarse en el seguimiento ocular móvil con ODROID utilizando hardware y software de código abierto.

HARDWARE

Como se ha mencionado, los reconocedores oculares disponibles en el mercado tienden a tener un coste excesivo para aficionados y experimentados. Por suerte, hay algunos diseños que se pueden hacer pirateando aparatos electrónicos de consumo comunes. En este artículo, vamos a usar un auricular EyeWriter [1]. Este se puede crear conectando una cámara PlayStation Eye modificada a unas gafas de sol baratas utilizando un brazo de aluminio flexible. Sorprendentemente, la calidad de



Usuario utiliza el auricular EyeWriter

vídeo y el nivel de fotogramas por segundo proporcionados por la cámara correctamente modificada son similares o exceden la de muchos reconocedores oculares disponibles en el mercado. Gracias a la facilidad de fabricación y al buen rendimiento de EyeWriter, vamos a utilizar este dispositivo para nuestra configuración experimental. La imagen de arriba muestra una persona usando el auricular.

Una vez que hemos montado nuestro auricular de oculografía, necesitamos procesar la imagen del ojo para localizar la pupila. Realizaremos este proceso en un ODROID-U2. U2 fue elegido por su tamaño relativamente pequeño, bajo coste, y su potencial para el procesamiento de imágenes. También nos aprovecharemos

de varios proyectos de software de código abierto que se ejecutan en Ubuntu y soportados por ODROID.

SOFTWARE

Nuestro sistema se basa en técnicas de procesamiento de imágenes para realizar un seguimiento de la pupila a partir de fotogramas de vídeo que proporciona la cámara USB. En concreto, utilizaremos el software libre Robust Pupil Tracker [2], una aplicación OpenCV escrito en C++. Esta aplicación intenta ajustar una elipse alrededor de la pupila en un fotograma ocular. El centro de esta elipse proporciona una estimación precisa del centro de la pupila en coordenadas de píxeles, que podemos usar en

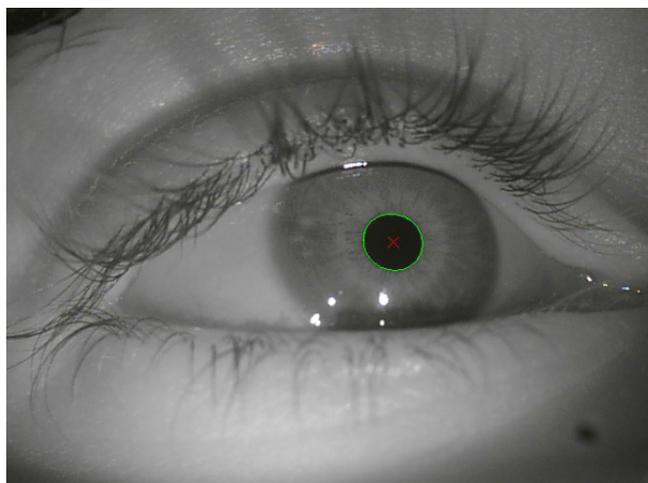


Imagen ocular con seguimiento de pupila

nuestras propias aplicaciones. Puede ver un ejemplo de un fotograma de vídeo desde el auricular en la imagen de arriba.

Para rastrear la pupila, tenemos que configurar el sistema operativo Ubuntu en nuestro ODROID-U2. También necesitamos instalar OpenCV, Boost, y las librerías Threaded Building Blocks en nuestro sistema, que son dependencias de la aplicación de seguimiento de la pupila. Este proceso requiere de alguna compilación desde código fuente y puede llegar a ser un poco tedioso. Por suerte, podemos saltarnos estos pasos

si descargamos la imagen de Ubuntu 12.11 Robotics Edition desde los foros ODROID. Esta imagen contiene las librerías necesarias para este proyecto, así como otras herramientas útiles de robótica y comprensión, como Robotic Operating System (ROS), OpenNI, Point Cloud Library (PCL), etc. Para nuestra configuración, la imagen del sistema operativo se ha descargado e instalado en una tarjeta SD de 8 GB. Tras arrancar ODROID-U2, podemos loguearnos con las credenciales por defecto y trabajar en nuestro experimento.

RESULTADOS

Por lo general, el seguimiento de la pupila por video imagen ocular suele ser un proceso costoso desde el punto de vista informático. Se deben procesar varias imágenes de forma secuencial con el fin de obtener resultados precisos. Con el fin de evaluar el rendimiento de ODROID, ejecutamos Robust Pupil Tracker con fotogramas oculares en tiempo real proporcionados por el auricular EyeWriter en 3 plataformas diferentes: Un PC de escritorio con cuatro núcleos de alta capacidad, un Netbook con un núcleo Intel Atom y un ODROID U2.

Hemos captado fotogramas originales con la cámara EyeWriter llegando a una tasa de 30 fps (fotogramas por segundo) y con una resolución de 640x480 píxeles (la cámara puede tener un mayor rendimiento, pero no es necesario). Los fotogramas de vídeo fueron captados y

procesados en tiempo real durante 30 segundos por cada plataforma. La tasa real de procesamiento de fotogramas de cada plataforma se muestra en la imagen de abajo.

Como reflejan los resultados, ODROID-U2 ofrece un mejor rendimiento que el netbook estándar. Esto es bastante impresionante, dado su tamaño, menores requisitos de energía y menor coste. Aunque una velocidad de 11,8 fps puede no bastar para seguir el irregular movimiento de la pupila (movimiento de los ojos durante la exploración visual), representa una buena tasa para la mayoría de aplicaciones en tiempo real, especialmente la detección de la mirada para la comunicación y el control.

CONCLUSIONES

El reconocimiento ocular es una emocionante y emergente tecnología que, en gran medida, está siendo impulsada por la comunidad de código abierto. El tema en sí tiene muchos frentes abiertos, pero experimentadores y aficionados pueden empezar a beneficiarse de varios diseños de hardware y software de uso público. La video oculografía ha marcado un antes y un después en las vidas de muchas personas con graves discapacidades físicas y con un mayor desarrollo puede llegar a tener un impacto aún más positivo en estos usuarios.

REFERENCIAS

- [1] Z. Lieberman, C. Sugrue, T. Watson, J. Powderly, E. Roth, and T. Quan, "The EyeWriter," EyeWriter Initiative, 2009. [Online]. Available: <http://www.eyewriter.org/>.
- [2] L. wirski, A. Bulling, and N. Dodgson, "Robust real-time pupil tracking in highly off-axis images," in Proceedings of the Symposium on Eye Tracking Research and Applications - ETRA '12, 2012, p. 173.
- [3] C. McMurrugh, "Ubuntu 12.11 Robotics Edition v2 (ROS+OpenCV+OpenNI+PCL) U2," ODROID Forum, 2013. [Online]. Available: <http://forum.odroid.com/viewtopic.php?f=8&t=2096>

Procesamiento de la tasa de fotogramas - Comparación

Plataforma	Tasa de Fotogramas (FPS)
PC Escritorio	30.0 FPS
Netbook	9.8 FPS
ODROID-U2	11.8 FPS

EJECUTAR PROGRAMAS LINUX BAJO ANDROID

UNA GUIA PARA CHROOT LINUX

Marian Mihailescu

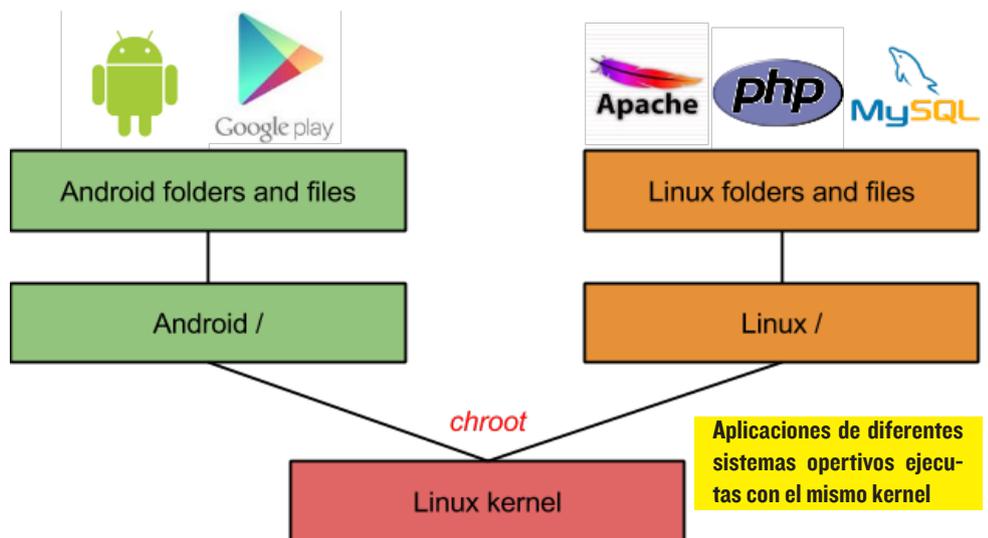
@mihailescu2m@gmail.com

Android es el sistema operativo amigable de Google disponible para muchos móviles y dispositivos compactos en todo el mundo, incluyendo ODROID. Al igual que Debian y Ubuntu, Android está basado en el kernel de Linux. Puedes comprobarlo entrando en la aplicación de Terminal desde tu dispositivo Android y ejecutando:

```
root@android:/ # uname -a
Linux localhost 3.0.51-g4732a09
#3 SMP PREEMPT Sun Aug 4 07:26:36
CST 2013 armv7l GNU/Linux
```

A diferencia de Linux, Android tiene su propio sistema de ventanas que reemplaza X.org, librerías independientes y un sistema de archivos raíz diferente. El sistema de ficheros raíz o directorio root es con el que el sistema arranca, comúnmente conocido como / (barra). Lugar en el que se montan los otros sistemas de archivos y donde residen todos los ficheros. Cada aplicación sabe dónde está el directorio raíz preguntando al kernel (llamadas al sistema). chroot es un método para pedir al núcleo (usando la llamada al sistema chroot) que cambie el directorio raíz aparente por la aplicación actual y sus descendientes.

Ya que Linux y Android usan el mismo kernel, es posible entonces tener dos sistemas de archivos raíz diferentes (uno para Linux y otro para Android) y usando chroot, se fija uno u otro dependiendo de la aplicación que estás ejecutando – una aplicación Android utilizaría el sistema de archivos raíz Android, mientras que una aplicación de Linux usaría el sistema de archivos raíz de Linux. Puesto que sólo hay un núcleo en ejecución, ambos sistemas compartirían los recursos. De ahí que, no hay penalización en el rendimiento o sobrecarga al usar chroot, cada aplicación accedería directamente al mismo kernel.



Cuando trabajas con chroot, hay algunas cuestiones que tienen que salvar:

Primero, el sistema que ejecutes en chroot (cliente) se iniciará desde un estado ya “arrancado”, por lo que todas las inicializaciones y scripts de inicio no están ejecutados. En realidad esto es bueno, ya que el sistema de ficheros raíz original que arrancó primero (host) probablemente ya hizo algo similar. Sin embargo, cualquier programa que tenía que ser iniciado al arrancar el cliente, tendrá que ser iniciado manualmente cuando usamos chroot.

Segundo, el kernel de Linux usa varios sistemas de archivos “virtuales” para administrar los recursos, que van desde componentes de hardware a la simple ejecución de aplicaciones. En el sistema de archivos de host, estos sistemas de virtuales se crean y montan en el arranque, de modo que en los clientes necesitas montarlos tú mismo. (como los recursos son compartidos, se deben montar los mismos sistemas de archivos creados en el host). Por último, es necesario tener los sistemas de archivos raíz del cliente ya montados en el sistema de archivos ejecutado en el host. Esta es la opción en la

que los sistemas de archivos raíz del cliente se encuentran físicamente en una partición en la misma unidad que el sistema de archivos host, en una unidad diferente, o incluso en un archivo de imagen que actúa como otra unidad.

A efectos de este artículo, el sistema de archivos host es Android y el sistema de archivos cliente es Linux (Ubuntu) ejecutado desde una imagen. Usar una imagen tiene varias ventajas, como la portabilidad (sólo tiene que copiar un archivo para mover todo el sistema), la facilidad de uso (la mayoría de las distribuciones ofrecen imágenes para sus sistemas de archivos raíz), y una mejor compatibilidad con Android (Android requiere un determinado formato de partición y no es recomendable cambiarlo). Como requisito, se necesita una aplicación Terminal en Android para acceder a la línea de comandos y Busybox, que es una aplicación para Android que ofrece varias herramientas Unix reducidas al mínimo en un ejecutable.

Asumiendo que tienes una imagen con el sistema de archivos raíz cliente (Ubuntu), primero tienes que montarla bajo Android, en este caso en la carpeta /data/local /ubuntu:

```
root@android:/sdcard/ubuntu # busybox mknod /dev/block/loop127 b 7 127
root@android:/sdcard/ubuntu # busybox losetup /dev/block/loop127 ubuntu.
img
root@android:/sdcard/ubuntu # mkdir /data/local/ubuntu
root@android:/sdcard/ubuntu # busybox mount -t ext2 /dev/block/loop127 /
data/local/ubuntu/
```

Navegando hasta la carpeta donde se monta la imagen, la raíz del sistema de archivos de Linux ahora es visible:

```
root@android:/sdcard/ubuntu # ls -la /data/local/ubuntu
drwxr-xr-x root    root    2012-03-29 09:13 bin
drwxr-xr-x root    root    2008-08-05 16:38 boot
drwxr-xr-x root    root    2012-03-29 09:19 dev
drwxr-xr-x root    root    2012-06-07 14:19 etc
drwxr-xr-x root    root    2009-01-10 18:58 home
drwxr-xr-x root    root    2012-03-29 09:13 lib
drwx----- root    root    2008-11-08 03:56 lost+found
drwxr-xr-x root    root    2008-11-08 04:50 media
drwxr-xr-x root    root    2008-08-05 16:38 mnt
drwxr-xr-x root    root    2008-11-08 04:50 opt
drwxr-xr-x root    root    2008-08-05 16:38 proc
drwxr-xr-x root    root    2012-06-30 14:45 root
drwxr-xr-x root    root    2012-03-29 09:13 sbin
drwxr-xr-x root    root    2008-11-08 04:50 srv
drwxr-xr-x root    root    2008-08-12 14:26 sys
drwxrwxrwt root    root    2012-06-07 14:19 tmp
drwxr-xr-x root    root    2009-01-08 21:35 usr
drwxr-xr-x root    root    2008-11-08 04:50 var
```

Después, monta los sistemas de ficheros virtuales creados por el Kernel de Linux en Android, en las ubicaciones correspondientes en el sistema de Linux:

```
root@android:/sdcard/ubuntu # busybox mount -t devpts
devpts /data/local/ubuntu/
dev/pts
```

```
root@android:/sdcard/ubuntu # busybox mount -t proc proc /
data/local/ubuntu/proc
```

```
root@android:/sdcard/ubuntu # busybox mount -t sysfs sysfs
/data/local/ubuntu/sys
```

También puede montar otros sistemas de archivos desde Android si quieres tenerlos disponibles en Linux, por ejemplo, la tarjeta SD:

```
root@android:/sdcard/ubuntu # mkdir /data/local/ubuntu/
media/sdcard
```

```
root@android:/sdcard/ubuntu # busybox mount -o bind /sdcard
/data/local/ubuntu/media/sd-
card
```

Asegúrate que la red está funcionando para el sistema de ficheros de Linux:

```
root@android:/sdcard/ubuntu # busybox sysctl -w net.ipv4.
ip_forward=1
```

Ahora, sólo tiene que iniciar un *prompt* de Linux para que crea que el sistema de ficheros raíz es Linux:

```
root@android:/sdcard/ubuntu # busybox chroot /data/local/
ubuntu/ /bin/bash

root@localhost:/#

root@localhost:/root# cat /
etc/issue

Ubuntu 12.04 \n \l
```

Empieza a ejecutar programas de Linux, instalar nuevos programas, etc. Es una buena idea instalar ssh, si no está ya instalado. Con ssh iniciado, puedes acceder a tu sistema Linux sin tener que volver a hacer todos estos pasos.

Dado que este sistema de ficheros raíz no ha sido realmente arrancado, la inicialización no se ha realizado. Hay algunos demonios y varias variables de entorno que desearías activar. (como ssh, por ejemplo).

```
root@localhost:/# ex-
port PATH=/bin:/sbin:/
usr/bin:/usr/sbin:/
usr/local/sbin
```

```
root@localhost:/# ex-
port TERM=linux
```

```
root@localhost:/# ex-
port HOME=/root
```

```
root@localhost:/#
service ssh start
```

Configura el sistema host para tener Linux disponible después de cada arranque de Android, agrupando los comandos en un script y ejecutandolo cuando Android arranque (hay apps en Android para ejecutar scripts en el arranque). Crea un script en la imagen de Linux para que todas las inicializaciones sean activadas en el arranque, y asegúrate de ejecutar el script bajo chroot (por ejemplo, /root/init.sh):

```
root@android:/sdcard/ubuntu # busybox chroot /data/local/
ubuntu/ /bin/bash /root/init.
sh
root@localhost:/#
```

Añade cosas útiles, las posibilidades son infinitas, ¡disfruta!

```
# environment variables
export PATH=/bin:/sbin:/usr/
bin:/usr/sbin:/usr/local/
sbin
export TERM=linux
export HOME=/root
```

```
# fix some issues with up-
start
dpkg-divert --local --rename
--add /sbin/initctl > /dev/
null 2>&1
ln -s /bin/true /sbin/initctl
> /dev/null 2>&1
```

```
# create /etc/mtab file with
the mounted filesystems
gep -Ev "rootfs|tmpfs|cgroup|
mmcblk|usbfs|asec|storage" /
proc/mounts | sort -r | uniq
> /etc/mtab
```

```
# start ssh daemon
service ssh start
# start other daemons...
```

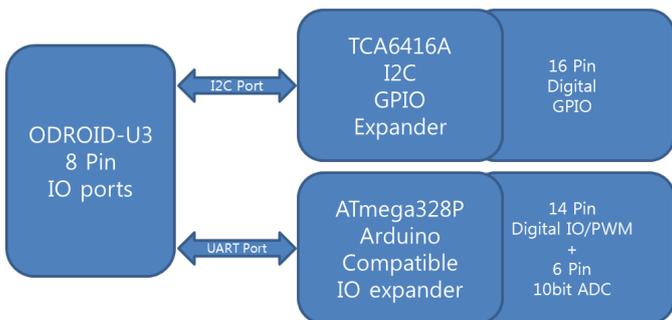
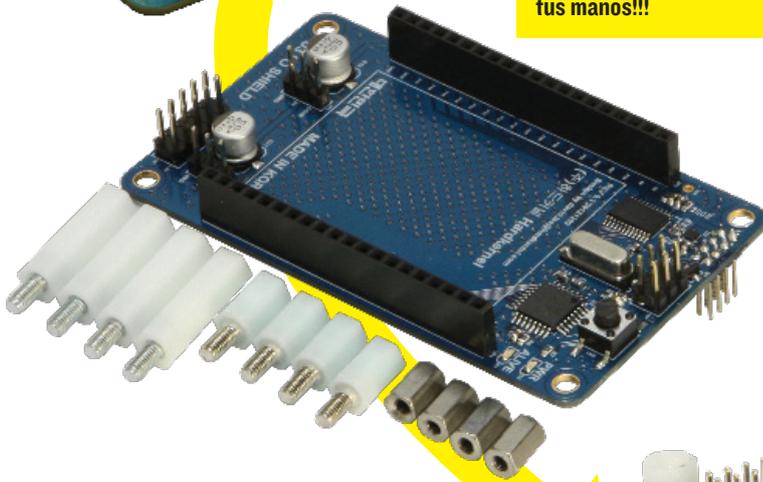
PUERTOS E/S EN ODROID-U3

EMPIEZA CON PEQUEÑOS AJUSTES

Justin Lee
@justin.lee@hardkernel.com



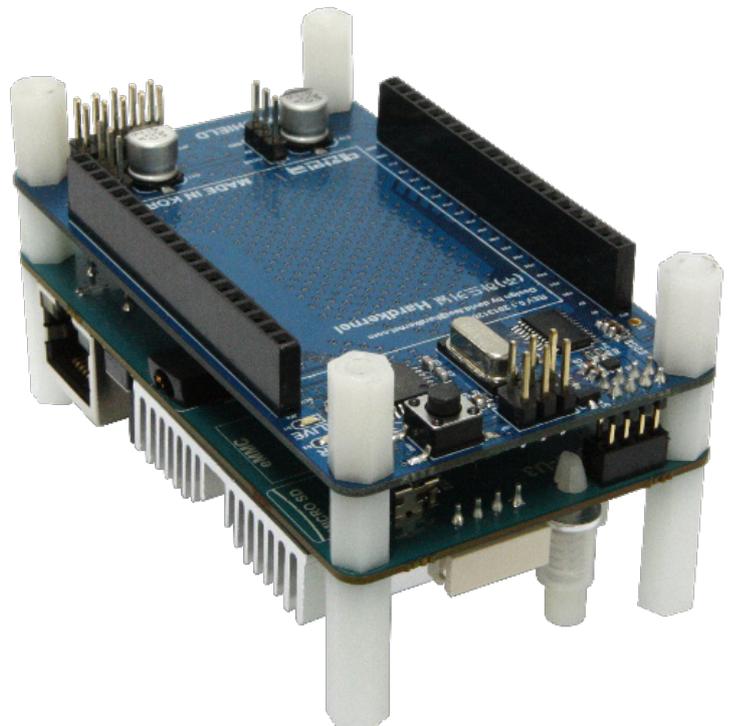
Lo sabemos, lo sabemos. Apenas puedes esperar a tener uno de estos en tus manos!!!



Puede que quieras conectar algunos servomotores, LEDs y conmutadores a ODROID-U3. Pero sólo hay un conector de 8 pines, así que ¿cómo conseguir más? Las corrientes eléctricas son de 5V, 1.8V y tierra, dejando sólo 5 puertos para usarse. Aún más confuso es utilizar una interfaz de 1,8V aún cuando la mayoría de aplicaciones requieren una interfaz de 5V. ¿Cómo un aspirante a técnico de hardware puede resolver estos problemas?

No te preocupes, existe una solución: la placa protectora E/S. Esta placa se puede conectar a ODROID-U3 mediante un diminuto conector E/S de 8 pines. Tiene exactamente la misma forma y dimensiones que ODROID-U3, y viene con 12 separadores PCB para que su montaje sea sencillo y sólido.

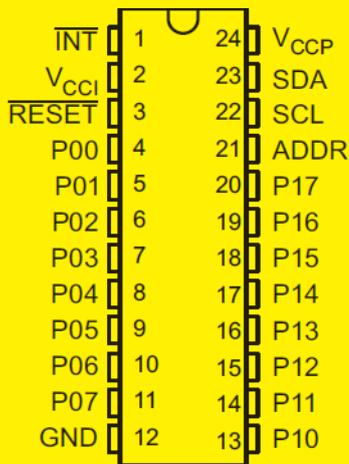
La placa E/S tiene 2 bloques. “Bloque de expansión I2C E/S” y “bloque compatible con Arduino”.



Expansión I2C E/S: 16 x GPIO

Este bloque usa la extensor TCA6416A I2C a puerto paralelo.

El principal beneficio de este dispositivo es su amplio rango de VCC. Puede funcionar desde 1,65V a 5,5V en el lateral puerto-P y en el lateral SCL /SDA por separado. Permite Expansión bidireccional con interpretación a nivel de tensión GPIO entre 1,8 V SCL/SDA (VCCI) y 5V Puerto (VCCP).



Los 16 puertos son asignados entre GPIO #289 ~ #304, que se puede leer desde la línea de comandos. Aquí hay un ejemplo para acceder a GPIO. Tenga en cuenta que necesitas para cargar el controlador primero el comando modprobe.

```
# modprobe gpio-pca953x
# echo tca6416 0x20 > /sys/
devices/platform/i2c-gpio.4/
i2c-4/new_device
# echo 289 > /sys/class/gpio/
export
# cd /sys/class/gpio/gpio289
/sys/class/gpio/gpio289#
echo "in" > /sys/class/gpio/
gpio289/direction
/sys/class/gpio/gpio289# cat
direction
in
/sys/class/gpio/gpio289# cat
value
1
```

Puede utilizar las librerías Python o C/C++ genéricas para acceder a los 16 GPIO en tu programa.

Arduino Uno compatible E/S

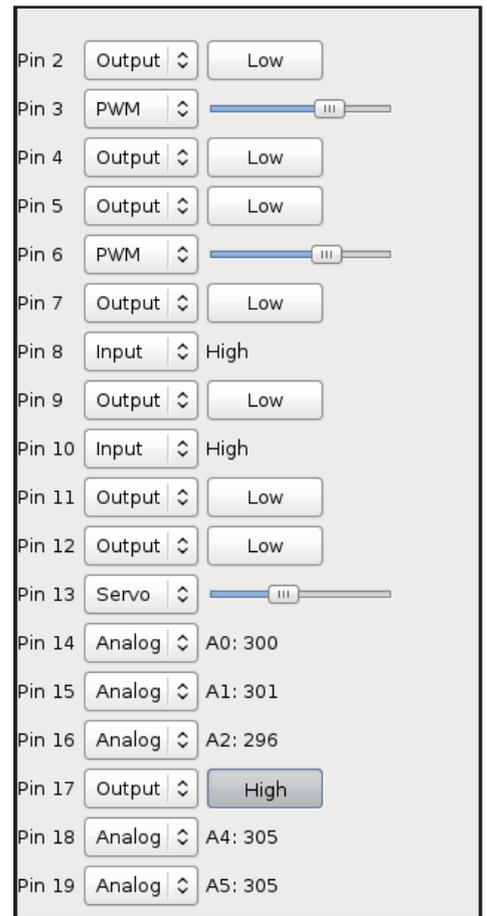
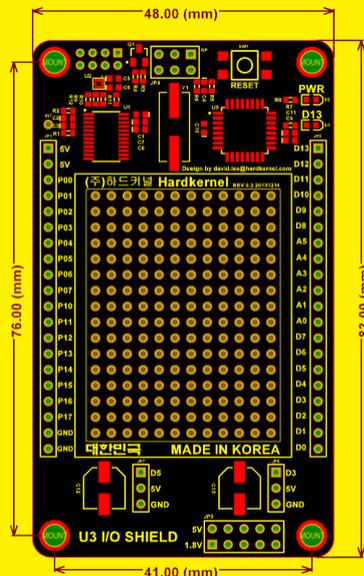
ATMEGA328P de Atmel se utiliza como una CPU esclava para expandir el puerto E/S. Puede utilizar Arduino IDE en ODROID-U3 para desarrollar y subir tus bocetos a este bloque compatible con Arduino, sin conectar cables adicionales/externos. Sólo tienes que elegir “/ dev/ttyACM99” en el Menú >> Herramientas >> Configuración del puerto serie. ATMEGA328P es un dispositivo esclavo en el bus UART y ODROID es el maestro.

Al igual que Arduino Uno, tiene 14 entradas/salidas digitales (de las cuales 6 se puede utilizar como salidas PWM), 6 entradas analógicas, que proporciona 10 bits de resolución. Funcionan a 5 voltios.

ATmega328P en la placa protectora E/S se envía con Arduino IDE compatible con el gestor de arranque optiboot-v5.0a y StandardFirmata Ver2.3 firmware.

Si necesitas probar rápidamente tu hardware, este programa independiente puede acceder a todos los pines. Código fuente de esta suite de pruebas: http://www.pjrc.com/teensy/firmata_test/firmata_test_OSL.tgz

Este es un dibujo CAD PCB de referencia. Los archivos PCB Gerber serán liberados en breve para tu propio desarrollo de protección DIY



En Hardkernel, hemos probado Firmata (<http://firmata.org/>) para hacer una interfaz interactiva entre ODROID-U3 y ATmega328P. Firmata es un protocolo genérico para la comunicación con los microcontroladores desde el software en un ordenador host.

La placa E/S es muy flexible. El módulo de control y adquisición de datos de aplicación general proporciona una conexión directa al conector E/S de ODROID-U3. Hemos añadido un montón de cosas básicas pero esenciales. En primer lugar, hay una gran área para el desarrollo de prototipos, de modo que puedes cablear los chips DIP, sensores, etc. En los bordes de la zona de proto, los pines GPIO / ADC / PWM y corriente están divididos en 0.1 stips para facilitar las conexiones. Hay dos cabezales de 3 pines para conectar pequeños motor servidores. Por último, hemos añadido un conector de 10 pines de I2C/GPIO para una mayor expansión.

Es hora de hacer ajustes y crear un prototipo con tu potente placa ARM 1.7Ghz quad-core.

SOBRE EL RENDIMIENTO DE LAS TARJETAS SD

SABER MÁS SOBRE LAS OPCIONES DE ALMACENAMIENTO

Jussi Opas
@jussil0@welho.com

Actualmente las mejores tarjetas micro SD disponibles son las de clase 10 UHS-I. Según sus especificaciones la velocidad de lectura mínima de una tarjeta UHS-I es de 10 MB/s. La velocidad máxima de lectura no está definida, de modo que cada tarjeta supera considerablemente su límite mínimo. La velocidad máxima suele anunciarse con valores de 10, 20, 40 o incluso 70 MB/s. En artículos publicados, las pruebas de lectura/escritura se hacen con lectores de tarjetas que a menudo no están disponibles para los usuarios comunes. ¿Cómo este rendimiento anunciado afecta en sí mismo al uso de una placa ODROID o un ordenador de un usuario ocasional? Este artículo recoge los resultados de las pruebas con tarjetas SD realizadas por nosotros.

Tarjetas probadas:

SanDisk Ultra 8GB:

Se envía junto a la placa ODROID

Samsung Pro 8GB:

Destinada a almacenar video de las cámaras digitales y se anuncian con una velocidad de escritura de 20 MB/s y de lectura de 70 MB/s.

Kingston 8GB:

Otra tarjeta SD de nivel UHS-I

eMMC card 64 GB:

Almacenamiento integrado en ODROID. Se usa de manera similar a las tarjetas micro SD con un adaptador que se envía con la placa.

Para las pruebas de lectura usamos el comando `hdparm` de Linux. Se puede invocar, como `sudo hdparm-t /dev/mmcblk0p1`. La prueba se hicieron con un portátil de 4 años (Celeron x86), un portátil de 2 años, (x86 i5) y una placa ODROID XU (Exynos ARM). Los resultados son estos:

LECTURA MB/S	PORTATIL 2 AÑOS	PORTATIL 4 AÑOS	XU
SanDisk Ultra	13.8	19.4	17.6
Samsung Pro	16.0	16.6 - 18.0	16.9 - 18.1
Kingston	11.8 - 14.7	15.7 - 16.8	15.4 - 16.5



las tarjetas. Por lo tanto, para alcanzar la máxima velocidad de la tarjeta SD más rápida, compramos un lector más rápido o usamos otro método de prueba.

Para comparar la velocidad de lectura con otras unidades y tarjetas eMMC, se llevaron a cabo una serie de pruebas de lectura `hdparm` en un par

de unidades de disco duro y una unidad SSD a través de una conexión eSATA. Los resultados se muestran en la imagen de abajo en orden decreciente.

La lectura más rápida se da un SSD instalado en un portátil (500 MB/s) y 268 MB/s con conexión eSATA. El rendimiento de lectura en eMMC varía en las distintas pruebas realizadas y entre los diferentes sistemas de archivos utilizados,

LECTURA MB/S	
SSD en portátil de 2 años	500
SSD vía eSATA en portátil de 2 años	268
eMMC en XU	90 - 117
HD en portátil de 4 años	63
HD en portátil de 7 años	30
SD en XU	17.6

ya sea en FAT o EXT4. El resultado más rápido alcanzado fue de 117 MB/s. El disco duro del portátil de 4 años solo puede alcanzar la mitad de la velocidad de lectura de la eMMC, 63 MB/s. Aún menor es el rendimiento de un disco duro en un ordenador de 7 años. Una tarjeta SD logra la misma de velocidad de lectura en una placa XU, similar a un viejo disco duro.

La escritura fue probada con el comando `dd`. El comando arroja unos resultados excelentes, ya que recoge el tiempo transcurrido y la velocidad de escritura real. En la práctica, `dd` se utiliza para borrar una tarjeta y luego montar una imagen del sistema operativo en ella.

Los resultados de escritura en un portátil de 4 años, se muestran más abajo.

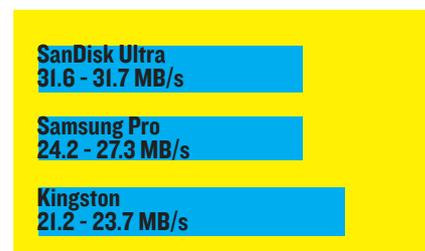
Tanto los portátiles de 2 y 4 años, escriben aproximadamente a la misma velocidad, lo que indica que la velocidad de escritura no depende del hardware. La SanDisk Ultra es la tarjeta más rápida en limpiar y montar, con ambas operaciones realizadas a la misma velocidad. Es sorprendente que la velocidad de escritura de la tarjeta de Samsung Pro sea de sólo 6 MB/s, muy lejos de la capacidad prometida de 20 MB/s. Kingston trabaja de manera constante a 8,4 MB/s. Limpiar una tarjeta eMMC se realiza a 8,6 MB/s mientras que el montaje se hace

a 6,6 MB/s. Para limpiar y montar más rápido una tarjeta eMMC, se debe ejecutar Linux en una tarjeta SD y tener la tar-

jeta eMMC instalada en la placa XU al mismo tiempo, pero esta configuración no se intentó en esta prueba.

Cuando se compara con otras tarjetas similares, la tarjeta SD enviada con ODROID rinde igual o mejor que las otras tarjetas analizadas en esta prueba. La tarjeta eMMC se comporta mejor que los discos mecánicos ya que la velocidad de lectura alcanza aproximadamente la mitad de la velocidad de un SDD que ha sido montado a través de eSATA.

A diciembre de 2013, las actualizaciones para el kernel ODROID-XU han mejorado la velocidad de lectura. El ODROID-XU con kernel 3.4.70 o más reciente presenta los siguientes resultados en la prueba de rendimiento:



REPRODUCIR CONTENIDO DE YOUTUBE EN LINUX

Jussi Opas

@insert email here

Una cuestión que desconcierta a usuarios (y especialmente en Linux) es por qué algunos vídeos no se pueden reproducir. Hay ocasiones en las que no se reproduce nada, otras en las que se ve el vídeo sin sonido y otras en las que el vídeo no se muestra pero se oye el audio. La reproducción de vídeo es en realidad un proceso bastante complejo. Para transferir contenido de forma eficaz, hace falta un método de empaquetado y desempaquetado eficiente y códecs de vídeo y audio apropiados. Luego, el hardware hace su trabajo a través de la conexión HDMI o el puerto de audio Jack para auriculares, ambos disponibles en ODROID-XU. Todas estas partes tienen que funcionar conjuntamente para poder alcanzar una experiencia de vídeo agradable.

Los Codecs pueden comercializarse bajo licencia o pueden ser de código abierto. Especialmente en el mundo ARM, un codec o reproductor libre puede no estar disponibles para Linux. El ejemplo más evidente es Adobe Flash Player, el cual fue cancelado en 2012. Vivir sin Flash, también es una gran ventaja – Los anuncios Flash no se visualizan cuando navegas. Pero ¿Qué pasa con los contenidos de vídeo y audio de YouTube u otras fuentes de contenido? Si un usuario no logra solucionar el problema de la reproducción, puede pensar que el problema está relacionado con el hardware. Hay una forma simple y fácil que permite solucionar los problemas de reproducción, tal y como se indica a continuación.

Si hay problemas para reproducir de vídeos de YouTube, podemos intentarlo en Firefox en lugar de Chromium, o viceversa. Si ninguno de los dos funciona, podemos pasar al reproductor HTML5 integrado en cualquier navegador. Mientras intentamos visualizar el vídeo, haz clic con el botón derecho del ratón y selecciona la opción de menú “Acerca del reproductor HTML5”, y una página con la dirección de <http://www.youtube.com/html5> se abrirá

Las versiones actuales del navegador soportan vídeo WebM con el codec de audio VP8. Pulse el botón “Solicitar el reproductor HTML5”, vuelve a cargar el vídeo y éste se reproducirá

Los usuarios pueden ver el formato de vídeo y codec de audio del contenido haciendo clic derecho sobre el vídeo y seleccionando la opción “Estadísticas para nerds” del menú. Si aparece `Mime Type: video/WebM; codec = “vp8.0, vorbis”`, el vídeo se reproducirá sin problemas con Firefox en un ODROID_XU con Xubuntu 13.10.

ODROID-U3 VS ODROID-U2

LAS DIFERENCIAS EN EL HARDWARE DESDE EL AÑO PASADO

Justin Lee

@justin.lee@hardkernel.com

Lanzamos ODROID-U2 en diciembre de 2012. Ya ha pasado un año, y el U2 sigue siendo uno de los mejores productos dentro del mundo de los PC ARM. El último Ubuntu 13.10 y Android 4.2.2 funcionan muy bien con el nuevo Kernel 3.8 de Linux y muchos miembros del foro están desarrollando activamente programas para ODROID. Por ello, decidimos mejorar el hardware de U2, y así nació la ODROID-U3.

Tenga en cuenta que ODROID-U3 es 100% compatible con el software de ODROID-U2.

Hay 7 diferencias clave entre la placa U3 y U2.

Bajada en el precio: De 89\$ a 59\$

Sí, ¡Es la diferencia más importante! Esta bajada de un 34% se hizo posible con la ayuda de muchos proveedores (incluido Samsung) así como de nuestros ingenieros de hardware. El precio normal es de 65\$, pero se puede obtener a un precio especial de 59\$ al registrarse en los foros: <http://forum.odroid.com/>.

Puerto Host USB

ODROID-U2 sólo tiene 2 puertos host USB, insuficiente para un uso normal de escritorio. Por ejemplo, si usas un teclado y un ratón, no hay lugar para un adaptador WiFi. Por lo tanto, hemos decidido añadir otro puerto USB. Esta nueva característica reduce significativamente la necesidad de un hub USB con alimentación externa.

Otra mejora fue añadir un conmutador inteligente que anula la fuga de corriente a los dispositivos USB cuando el sistema está apagado, reduciendo posibles daños en los dispositivos.



Encendido y apagado

Hemos añadido un botón para encender y apagar ODROID-U3, aún cuando ya cuenta con el encendido automático. Ya no es necesario conectar y desconectar la clavija DC. Para apagar el sistema, basta con pulsar el botón y aparecerá un cuadro de diálogo de apagado en el monitor. Para reiniciar, basta con pulsar el botón de nuevo y el ODROID-U3 se reiniciará.

Puertos E/S

ODROID-U2 no tenía puertos E/S, sino que usaba la interfaz USB para agregar sensores, motores, LEDs, conmutadores, etc. Un diminuto conector de 8 pines en ODROID-U3 permite UART, IRQ, I2C, GPIO y energía.

Para los que desean usar ODROID-U3 para una aplicación de robótica, nuestra placa de expansión E/S tiene 36 puertos incluyendo 24 GPIO, 6 PWMs y 6 ADCs.

Dimensiones

A mucha gente le gustaba el diminuto tamaño de U2. Pero, algunas personas le preocupaba su altura y estábamos de acuerdo en que el U2 podría ser más pequeño. Así que aumentamos ligeramente la superficie del PCB y redujimos la altura ganando alrededor de 50% de

espacio, lo que significa que dos U3 pueden encajar en el espacio de una sólo U2.

El volumen de la ODROID-U2 es 198.360 mm³ (58 x 60 x 57 mm con disipador de calor) mientras que el volumen de la ODROID-U3 es 87.648 mm³ (83 x 48 x 22 mm con disipador de calor).

Reloj en tiempo real (RTC)

Simplemente añadiendo una batería de repuesto del tamaño de una moneda, Es posible mantener el reloj y el calendario exacto cuando se va la luz, sobre todo para proyectos autónomos sin conexión a la red.

Protección en la entrada de energía

Protección en Tensión inversa, sobretensión, Bajada de tensión y ESD se han tenido en cuenta en el sistema de entrada de energía con un circuito especial (IC). Proporciona protección contra sobretensiones positivas y negativas, de hasta 28 V y justo hasta -28 V.

En condiciones de sobretensión, la salida del CI permanece desactivada si la tensión de entrada es superior a 6 V. Esto da a ODROID-U3 una mejor resistencia frente a un mal uso (abusivo) de fuentes de alimentación.

ODROID-U3 VS RASPBERRY PI

¿NECESITAS MAS POR TU DINERO? ¡U3 TE LLEVARA HASTA ALLI!

Mauro Ribeiro
@mdrjr@gmail.com

Justin Lee
@justin.lee@hardkernel.com

Ambos son ordenadores compatibles con Linux, económicos y de una única placa ARM para diversos fines y propósitos.

Aunque ODROID-U3 es un económico ordenador de una sola placa ARM, incorpora un quad-core Samsung ARM. Sus especificaciones incluyen un SoC Samsung Exynos 4412 quad-core ARM Cortex-A9 a 1.7GHz, Mali-400 MP con 4 núcleos de GPU, tres puertos 2.0 host USB, un puerto Ethernet 10/100MB, salida de vídeo 1080p a través de micro-HDMI, 2 GB de memoria LP-DDR2, eMMC y micro-SD para el almacenamiento. Soporta los sistemas operativos Debian Wheezy, Ubuntu 13.10 Linux y Android Jellybean.

Hardware

Comparándolo con el Raspberry Pi, el número de núcleos de CPU/GPU es 4 veces más alto y la frecuencia de reloj de funcionamiento de la CPU es alrededor de 2,4 veces más rápida. Además, el tamaño de la RAM es también 4 veces mayor y la frecuencia acceso a la ésta es 1,6 veces más rápida. En superficie, el RPI es 1,2 veces más grande y el peso es muy similar si se considera el disipador de calor. También se puede utilizar en ODROID-U3 hardware RTC (reloj en tiempo real) con una pila tipo botón.

El RPI utiliza un SoC (System on a Chip) Broadcom que es un ARM v6 con una velocidad de reloj de 700Mhz (se puede incrementar hasta un 1GHz), sin embargo el RPI probado no superó los 800Mhz). En cambio, el U3 utiliza un procesador ARM v7 más reciente, el Samsung Exynos 4412. Es el mismo chip que encontrarás en los smartphones como el Galaxy Note 2 y el Galaxy S3. Dispone de



Si, si, La RPi no esta ni siquiera bien fijada. Esto es lo mucho que soporta.

ODROID-U3

Raspberry Pi

(Modelo B / 512MB)

CPU

SAMSUNG EXYNOS-4412 Prime
4 x ARM Cortex-A9 @1.7Ghz
Arquitectura ARMv7

BROADCOM BCM2835
1 x ARMII @700Mhz
Arquitectura ARMv6

GPU

4 x ARM Mali400 @400Mhz

1 x VideoCore IV @ 250 MHz

RAM

2GB LP-DDR2 @440Mhz

512MB SDRAM @400Mhz

USB 2.0 Host + device

3 Puertos • 1 Puerto Linux USB Gadget driver

2 Puertos • NO

ETHERNET

10/100 Mbit/s

10/100 Mbit/s

SALIDA VIDEO

HDMI (480p/720p/1080p)

HDMI / Composite RCA

SALIDA AUDIO

3.5mm Jack / HDMI

3.5mm Jack / HDMI

RELOJ EN TIEMPO REAL

SI (Conexión batería reserva)

NO (A menos que uses un módulo GPIO)

GPIO

5 (1.8Volt)

17 (3.3Volt)

TAMAÑO

83 x 48mm (3.27" x 1.89")

85.6 x 56mm (3.37" x 2.2")

PESO

30g (1.06 oz)

48g (1.69 oz) with a heat sink

45g (1.6 oz)

PRECIO

59\$

35\$

4 núcleos Cortex-A9 a 1.7Ghz (se puede incrementar hasta un 2.0Ghz). Recuerde que el “overclocking” (aumento de velocidad) es una cuestión de suerte. No todas las placas tendrá el mismo resultado “overclocking”.

Al igual que el RPi, el U3 tien un puerto Ethernet 10/100. Aun así, el U3 muestra velocidades de transmisión más altas, ya que cuenta con un nodo separado. El ODROID-U3 incorpora 3 puertos USB permitiendo conectar más periféricos. Los puertos USB de U3 pueden proporcionar hasta 500 mA. El sistema no se reinicia al conectar dispositivos USB.

El U3, a diferencia de RPi, utiliza tarjetas microSD en lugar de una de tamaño normal. Las tarjetas microSD son cada vez más popular debido a que la mayoría de los smartphones del mercado que permiten expandir su almacenamiento usan este tipo de tarjetas. Además, el U3 va aun más lejos y añade la opción de utilizar memoria eMMC. eMMC es una memoria flash MLC de alta velocidad como la usada como memoria interna en los teléfonos modernos y tabletas (eMMC se vende por separado como un accesorio). La velocidad de acceso a eMMC es 3-4 veces mayor que en las tarjetas SD.

El número de pines GPIO de E/S es menor que en RPi. Sin embargo, puedes ampliar E/S con la placa de expansión I2C E/S o el kit USB E/S, además de Arduino o interfaces similares.

Rendimiento

Las pruebas se realizaron con las imágenes ofrecidas por el fabricante y con una instalación limpia, además de apt-get update && apt-get upgrade para que ambas placas estuviesen actualizadas.

RPI fue elevado a 800MHz y usamos una tarjeta SD Sandisk Extreme UHS-1 de 8GB con Debian Wheezy.

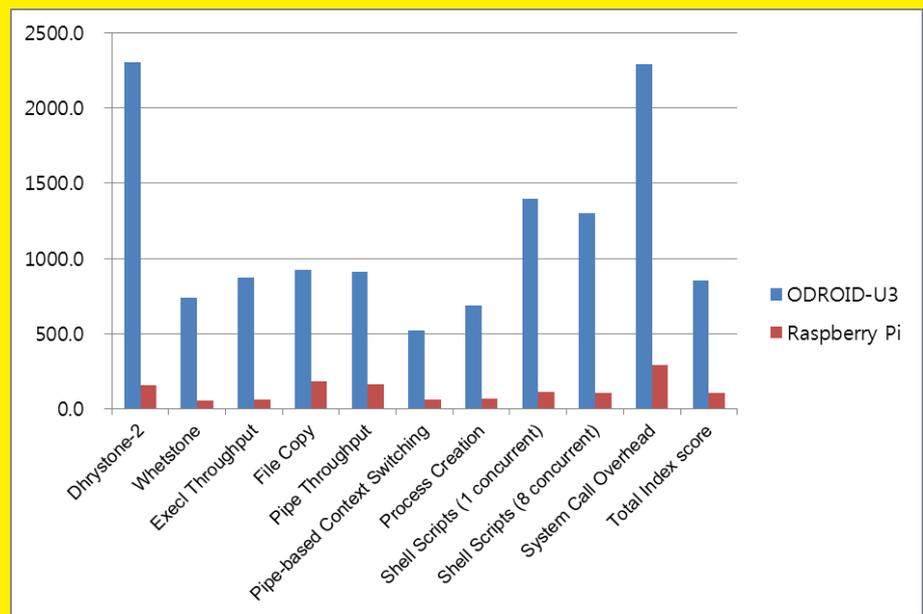
U3 fue ejecutado a su velocidad estándar de 1.7Ghz usando una eMMC de 16 GB con Ubuntu 13.10.

Ambas unidades fueron alimentadas por una fuente de 5V/2A y conectadas a la salida de HDMI 1920x1080.

Ejecutamos la famoso y simple banco de prueba: Unix-Bench version 5.1.3

Activamos los cuatro núcleos en ODROID-U3 con el comando./Run-c 4.

Comparando el Rendimiento

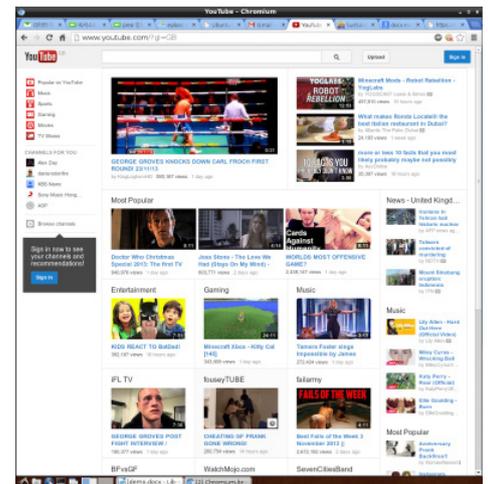


La evaluación Dhrystone-2 es aproximadamente 14 veces más rápida en ODROID-U3. La evaluación E/S en torno a 5 veces más rápida en ODROID-U3, debido a que el sistema eMMC es más rápido. En términos globales, los resultados de las pruebas muestran que ODROID-U3 es alrededor de 8 veces más rápido que RPI. Sin embargo, el precio de la U3 es sólo 1,7 veces mayor que el del RPI.

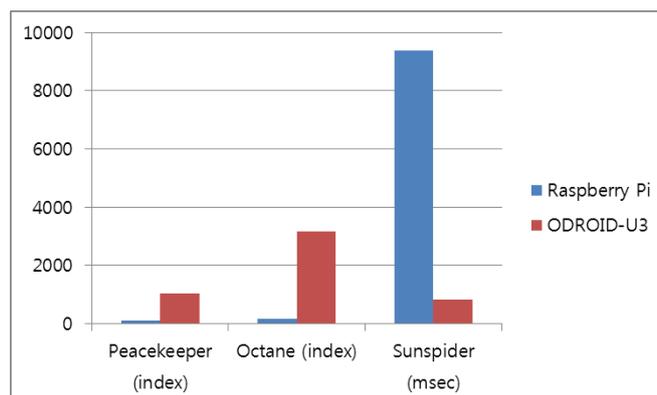
Uso de Software

Navegador Web:

ODROID-U3 proporciona una experiencia de navegación completa con un desplazamiento de ventana fluido y sin efectos borrosos. HTML5 y Flash player están disponibles para Google Chrome, por lo que puedes disfrutar de los vídeos de YouTube, juegos HTML5 y mucho más en tu U3.



U3 te ofrece todos los recursos necesarios para navegar. Raspberry, no tanto



Evaluación Peacekeeper (Más alto mejor)

RPI: 99

U3: 1036

Evaluación Octane (Más alto mejor)

RPI : 154

U3: 3156

Evaluación Sunspider (Más bajo mejor)

RPI: 9372ms

U3: 834ms

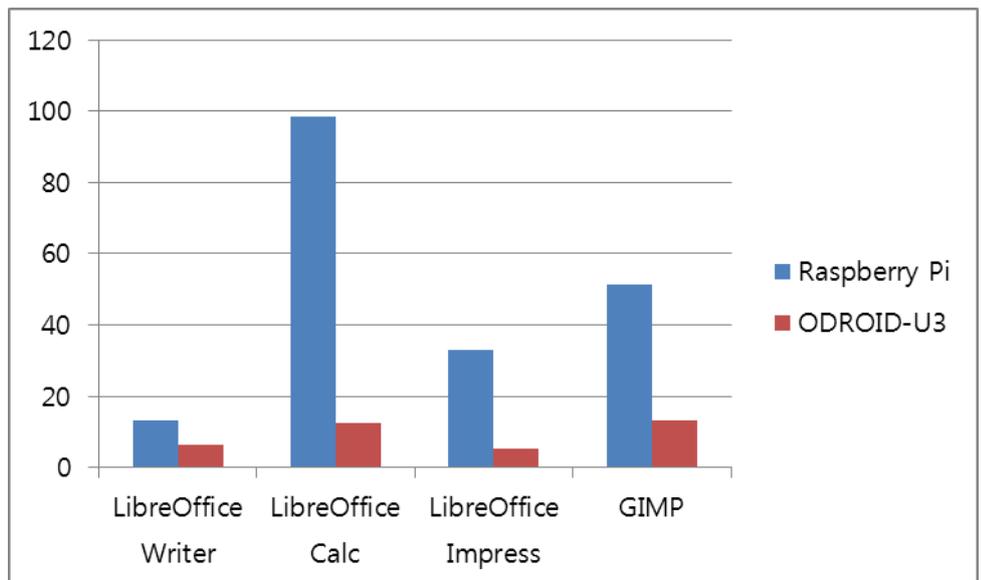
U3 muestra 10-20 veces mejor rendimiento comparado con Raspberry Pi.

Suite de ofimática y editor gráfico

Muchas personas usan software de oficina con frecuencia. Se realizó una comparación del tiempo de apertura y carga de archivos usando el famoso LibreOffice y GIMP.

Hemos seleccionado un fichero en particular en el explorador de archivos para medir el tiempo que tarda en abrirse el programa más abrir el archivo en sí mismo.

La mayoría de los programas necesitan mucho recursos, Calc es aproximadamente 8 veces más rápido en el U3.



Programación y Desarrollo

También comparamos varios IDEs y compiladores para desarrollo de software. Probamos Scratch, Arduino IDE, Python, compilación del Kernel Linux y Eclipse.

El U3 es, obviamente mucho más rápido al cargar el IDE y compilar el código, mientras que RPi no podía ejecutar Eclipse debido a la RAM insuficiente.

La compilación completa del código del kernel de Linux es 17 veces más rápida que en RPi.

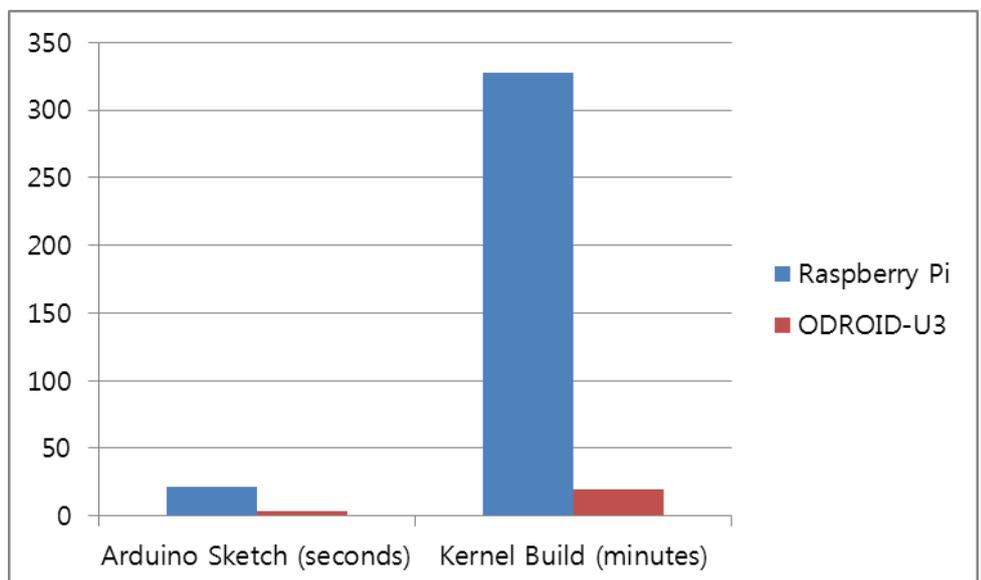
Hemos utilizado la opción `-j5` en el comando `make` para usar los 4 núcleos.

Software de Entretenimiento y otros XBMC

ODROID soporta XBMC en Linux y Android. Al igual que el RPi, el ODROID-U3 permite la decodificación del vídeo a 1080p. Sin embargo a 720p, obtendrás una experiencia más fluida. No obstante el desarrollo de Linux XBMC para U3 está todavía en curso. Tenga en cuenta que ODROID-U3 no tiene el mismo receso al decodificar el vídeo a 1080p en Android.

Sistema Operativo Android

El Soporte para Android es completo con todas las funciones. U3 se convierte en una completa tablet HD cuando se conecta a un TV. Si instalas Google Play



Esquema Arduino: Tiempo de desarrollo de un simple esquema LCD

RPi: 21.7 segundos

U3: 3.2 segundos

Compilar Kernel: Tiempo de desarrollo del código fuente completo del kernel 3.4 de Linux

RPi: 327 minutos (5 horas y 27 minutos)

U3: 19 minutos

en ODROID, puedes disfrutar de más de un millón de aplicaciones y contenidos de Android. RPi también tiene una versión para android, pero carece de muchas características para un uso cotidiano.

Conclusion

Como muestra los resultados anteriores, ODROID-U3 quad-core de 1,7 GHz supera claramente a Raspberry Pi con su único núcleo de 700MHz (incluso cuando se incrementa su velocidad). Muchos de los resultados de las

pruebas muestran de seis a doce veces mejor rendimiento de la plataforma U3, siendo el aumento del coste de tan sólo 1,7 veces. Aunque ambas plataformas son dispositivos compatibles con linux, la relación rendimiento/coste es completamente diferente.

Si estás considerando adquirir un diminuto ordenador para un uso general, desarrollar software o como plataforma de proyectos, ODROID-U3 te dará mucha más satisfacción y diversión con un increíble rendimiento a un precio muy bajo.

CONOCIENDO A UN ODROIDIAN

UN CARA A CARA CON JUSTIN LEE, CEO DE HARDKERNEL

Robert Hall

Como CEO de Hardkernel, mantienes contacto con la comunidad a través de los foros. ¿Qué es lo que más te gusta de la comunidad ODROID?

Recopilo muchas ideas, problemas, quejas a través de nuestros foros. Mejoramos el hardware y el software de las placas ODROID con esas valiosas aportaciones de los usuarios. Puesto que todos los temas de nuestro foro marcan el futuro de ODROID, me las arreglo para mantener el contacto con el foro de la comunidad tanto como me es posible.

Mi foro favorito es el de proyectos, que muestra muchas aplicaciones brillantes. No sabía muy bien hasta donde podían llegar los usuarios con ODROID. He aprendido muchas cosas de este foro.

Este es un robot impulsado por la placa ODROID-U2 ejecutando ROS con un sensor complejo de imagen 3D. ¿No es emocionante?



¿Tus inicios con los ordenadores?

Cuando tenía 13 años, en el año 1983 había muchos PCBs clon de Apple II en Corea. Tuve que soldar sobre 300-400 componentes en un gran PCB Apple II para aprender el lenguaje Basic. Tarde más de 4 semanas en montar mi primer PC y finalmente lo logré. Pero perdía mucho tiempo con los juegos.

Más tarde aprendí el lenguaje 6502 ASM, así como lenguaje Z80 ASM con una tarjeta CP/M. Fue en 1984-1985.

¿Qué te motivó a crear Hardkernel?

Hardkernel se creó a finales de 2008. Tiene 5 años. Sus fundadores habían estado trabajando en productos de consumo [sector con productos] como reproductores multimedia portátiles, PC portátiles, sistema de navegación para coches. Cuando los Smartphone se expandieron y eliminaros a los



otros dispositivos de TI, nosotros [tuvimos que] buscar otro negocio.



Pensamos que los desarrolladores necesitan un hardware de alto rendimiento basado en linux. Creamos una nueva empresa Hardkernel que es simplemente Hardware + Linux Kernel.

Empezamos a desarrollar un producto para desarrolladores por desarrolladores. El primero fue una placa ARM11 que ejecutaba Android 1.5 en marzo de 2009. La primera plataforma ODROID original, nació en octubre de 2009

con ARM Cortex-A8 y Android 2.0. ODROID significa Open + Android.

¿Quién está trabajando contigo en el equipo Hardkernel?

Tenemos 2 equipos internamente
Equipo de Marketing y Ventas (Lisa, Anna y Emily), dedicado al Marketing, Finanzas, Ventas, gestión web y otras actividades de operación.

Equipo de Investigación y Desarrollo (Charles, Ruppi, Kevin, Mauro, Chris, David, Juan y Brandon) dedicado al diseño de Arquitectura, esquemas de hardware y planos PCB, desarrollo de software, producción, soporte a clientes y envío.



sentar un par de periféricos cada mes.

Escribiremos un par de artículos para la revista ODROID cada 2 meses. Este es el plan a seguir para el 2014.

¿Qué otros intereses y aficiones tienes?

Desde primavera a otoño, suelo jugar en mi pequeña granja cada fin de semana. Planto maíz y algunas verduras como tomates, pimientos y coles. Parece que es realmente bueno para mi salud, así como



para mi mente. Sorprendentemente reduce la presión mental.



¿Tienes algún proyecto personal en el que estás trabajando usando ODROIDS?

Estoy usando ODROID como una



El equipo Hardkernel, para saber más sobre esta foto, jecha un vistazo a nuestro blog!

consola de juegos para mi familia. Algún día quisiera hacer una maquina arcade real con un detector de moneda para jugar con mis colegas en la oficina.

Que será lo próximo de Hardkernel y ODROID en 2014?

Como hemos venido haciendo en los últimos 5 años, seguiremos desarrollando la plataforma de software y hardware.

Acabamos de lanzar ODROID-U3 con un precio muy asequible. Habrá muchos periféricos emocionantes y software para ODROID-U3 en 2014. Todas las ideas son bienvenidas y estamos listos para hacerlas realidad. Esperamos pre-

Gracias por tomarte un tiempo en hablar con nosotros, Justin. ¡Te deseamos buena suerte con Hardkernel!



Primer prototipo de Hardkernel, Ha pasado mucho tiempo desde esto. ¡Únete a nosotros para conocer lo que se avecina!