

# ODROID

Año Seis  
Num. #68  
Ago 2019

Magazine

## *Kubernetes*

CREANDO UN  
SISTEMA DE  
ORGANIZACION DE  
CONTENEDORES  
EFICIENTE



CLUSTER  
ODROID-N2:  
INFORMATICA DE  
ALTO RENDIMIENTO  
ASEQUIBLE

ARRANQUE MULTIPLE:  
UN ODROID, MULTIPLES SISTEMAS

MOOSEFS:  
UN SISTEMA DE ARCHIVOS POSIX EN RED



## Monta una Tablet ODROID Dual-Boot, Root, Toot: Utilizando el ODROID-C0 para Crear una Tablet de Calidad Profesional por Menos de 100\$

© August 1, 2019

Oculto a simple vista en forma del potente ordenador de placa reducida de Hardkernel (SBC), el modesto ODROID-C0. Ligeramente más grande que una batería de 3.000 mAh pero impulsado por un Amlogic S805 System-on-a-Chip (SoC) que consume 1 Gb de RAM, dos puertos USB (2.0), una interfaz eMMC y un ▶



## El Punto G: Tu Destino para Todas las Cuestiones Relacionadas con Juegos Android

© August 1, 2019

Cuando el número de julio estaba a punto de salir, varios bombazos de juegos que, sin solicitarlo fueron a parar a mi escritorio. De modo que, empezaré este artículo para el número de agosto con una actualización sobre cada uno de estos eventos importantes. Estos eventos importantes incluyen: el Electronic ▶



## Implementación del Manipulador GPIO IRQ: Usando Python 3 para Controlar RPi.GPIO

© August 1, 2019

Este código y esta guía tienen la intención de probar la gestión de GPIO IRQ en el ODROID-C1+/C2/XU4/N2. La guía ha sido adaptada partiendo de la página wiki de ODROID [https://wiki.odroid.com/odroid-xu4/application\\_note/gpio/rpi\\_gpio\\_irq](https://wiki.odroid.com/odroid-xu4/application_note/gpio/rpi_gpio_irq). Simplemente tenemos que implementar el manipulador GPIO IRQ con Python 2/3. En esta guía, usaremos Python 3 para programar ▶

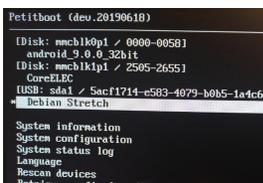


kubernetes

## Ejecutando Kubernetes en el ODROID-N2: Crea un Sistema de Organización de Contenedores de Eficiencia Energética

© August 6, 2019

El despliegue de aplicaciones en contenedores es la nueva moda. Con ello, surge la necesidad de automatizar las implementaciones, ampliarlas para satisfacer el aumento de las cargas de trabajo y administrar su ciclo de vida. Kubernetes (K8s) es una de esas utilidades que cubre estas necesidades.



## Arranque Múltiple en ODROID-N2: Añadiendo la Posibilidad de Seleccionar Múltiples Sistemas Operativos desde un Único Disco

© August 6, 2019

Hay habido mucha demanda para administrar múltiples sistemas operativos y muchos usuarios de SBC han intentado ejecutar múltiples sistemas operativos instalados en el almacenamiento USB, uSD y eMMC y, como tal, se proporcionaron instrucciones con un script o comandos de Linux. Muchos usuarios lo lograron, pero las instrucciones no son ▶



## Juegos Linux: PC-Engine TurboGrafx - Parte 5

© August 1, 2019

En la anterior entrega de esta serie, logre jugar a una gran variedad de diferentes géneros. Descubrí algunos juegos de disparos que me gustaron, uno de mis juegos de aventuras favoritos de todos los tiempos, e incluso el único juego de deportes que me ha llegado a gustar en el [▶](#)



## Usando el Kernel 5.2 con Armbian: Ejecuta WiringPi, HomeAssistant y Mucho Más con el Ultimo Soporte del Kernel

© August 6, 2019

Este artículo se centra en la creación de una imagen arm utilizando el kernel principal Linux linux 5.2.0-rc3, y en la configuración del popular software de domótica, Home Assistant. En este ejercicio trataremos específicamente el popular ordenador de placa reducida (SBC) ODROID-C1.



## Cómo Crear una Consola de Juegos Retro Monku - Parte 2: Configuración del Software

© August 1, 2019

Esta es la continuación del artículo de la consola de juegos retro del mes pasado, donde aprendimos cómo fabricar una carcasa para una consola de juegos retro. Esta entrega te ayudará a configurar el software para el proyecto. En este artículo, te mostraré con detalle cómo configurar el sistema operativo [▶](#)



## Un Excelente Servidor Web de Bajo Coste: Utilizando ODROID-N2 para Hosting de Internet

© August 6, 2019

Como desarrollador de NEMS Linux, siempre estoy ansioso por probar nuevas placas que prometen alto rendimiento y estabilidad en el campo de los servidores.



## Montando un Clúster ODROID-N2: Informática de Alto Rendimiento Asequible

© August 6, 2019

En mi intento por encontrar un verdadero SBC de quad-core de 64 bits en el que cada núcleo estuviera cercano a los 2 Ghz y que contase con al menos 4 GB de RAM, me encontré con esta belleza recién lanzada llamada ODROID-N2. Vamos a montar un clúster con ella.

# Monta una Tablet ODROID Dual-Boot, Root, Toot: Utilizando el ODROID-C0 para Crear una Tablet de Calidad Profesional por Menos de 100\$

© August 1, 2019 By Dave Prochnow Android, ODROID-C0, Mecanico



Oculto a simple vista en forma del potente ordenador de placa reducida de Hardkernel (SBC), el modesto ODROID-C0. Ligeramente más grande que una batería de 3.000 mAh pero impulsado por un Amlogic S805 System-on-a-Chip (SoC) que consume 1 Gb de RAM, dos puertos USB (2.0), una interfaz eMMC y un zócalo para tarjeta de almacenamiento microSD, Este SBC es el secreto mejor guardado del mundo ODROID.



**La tablet ODROID que ejecuta Ubuntu 18.04 Mate con un teclado y ratón opcionales.**

La mayoría de las cualidades más comunes del ODROID-C0 son conocidas por todos y están bien documentadas: puede ser alimentado por una batería de 3.7V, hay un cargador integrado para una batería,

la mayoría de los principales puertos de interfaz se pueden soldar al SBC (a través del paquete de conectores vendido por separado) "según sea necesario", y se puede conectar un monitor externo a través de un puerto HDMI disponible. Sin embargo, lo que quizás no sepas del ODROID-C0, es que puedes conocerlo simplemente estudiando el diagrama esquemático de este SBC.

La lectura de este diagrama esquemático revela que el jumper J1 se puede usar para seleccionar las interfaces de la tarjeta eMMC o microSD para iniciar el ODROID-C0. Además, los pines de entrada/salida de propósito general (GPIO) cuentan con una fuente de alimentación de 5V. Estas son dos joyas increíbles de hardware que pueden explotarse para crear una tablet ODROID de arranque dual por menos de 100\$

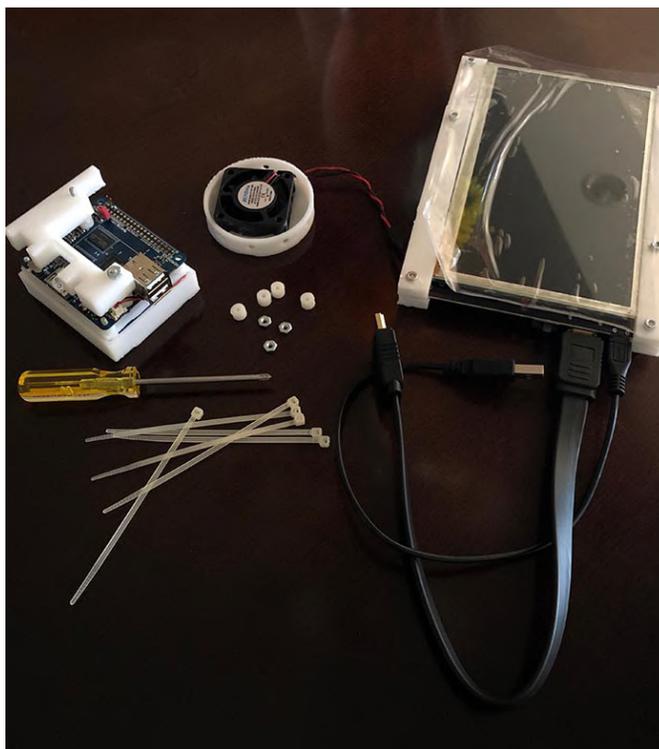


**Figura 2: Cambiar el jumper J1 convierte la tablet ODROID en un dispositivo portátil Android 4.4.4.**

### Componentes

Nota: las piezas para este proyecto se pueden comprar en Hardkernel o ameridroid.com

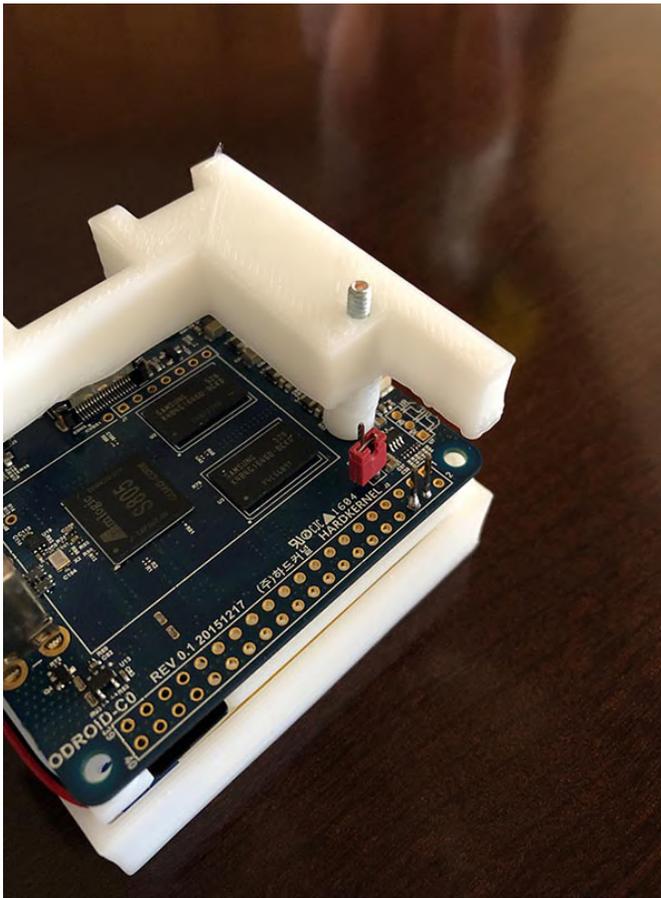
ODROID-C0 ODROID-VU7 Batería de respaldo RTC Módulo Android eMMC C1+/C0 de 16 Gb Tarjeta Linux microSD C0/C1 de 16 Gb Batería de 3.000 mAh Pack de conectores para ODROID-C0 Ventilador de refrigeración USB de 40x40x10 mm (opcional) (1) Cabezal de 2 pines con jumper (se puede comprar en BGMicro.com) (1) Cabezal de 2 pines (2) Cables Jumper hembra



**Figura 3: Todos los componentes que necesitarás para montar tu propia tablet ODROID.**

### Paso a paso

1. Suelda el puerto USB dual, desde el pack de conectores, al ODROID-C0.
2. Suelda el cabezal de 2 pines con el jumper al puerto J1 en la placa de circuitos impresos (PCB) de la SBC.
3. Suelda el otro cabezal de 2 pines a los pines 4 y 6 en la interfaz GPIO de PCB. Ten en cuenta: es posible que tengas que aumentar la temperatura de tu soldador a 365 grados Celsius (690 Fahrenheit) para realizar una conexión sólida a estos pines.



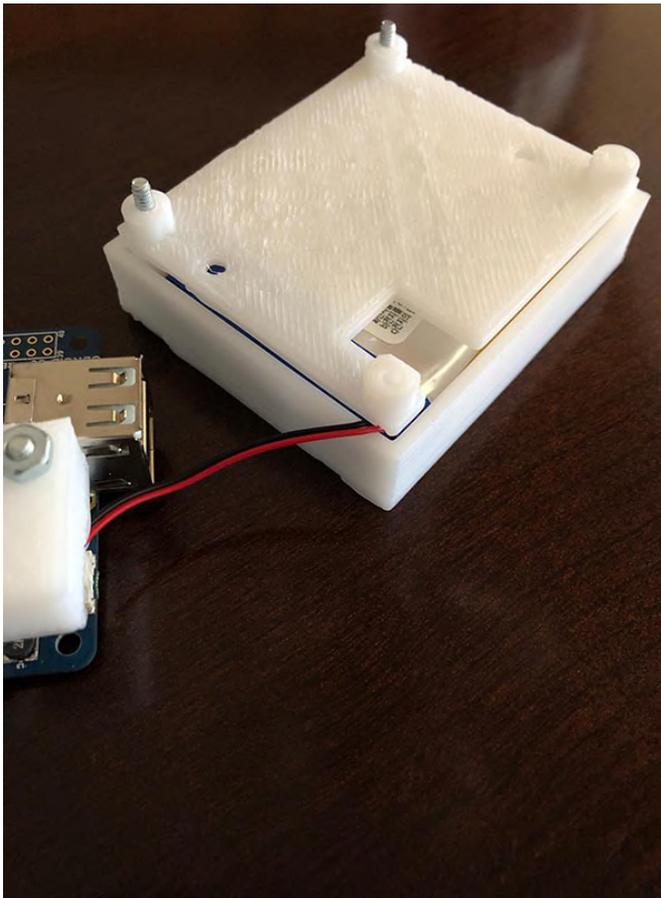
**Figura 4 - Se ha soldado un cabezal de 2 pines al jumper J1 y se ha conectado otro cabezal a los pines GPIO del ODROID-C0.**

4. Recorta la clavija USB del ventilador dejando al aire los cables rojo y negro y suelda los dos cables del jumper hembra a estos cables rojo y negro.



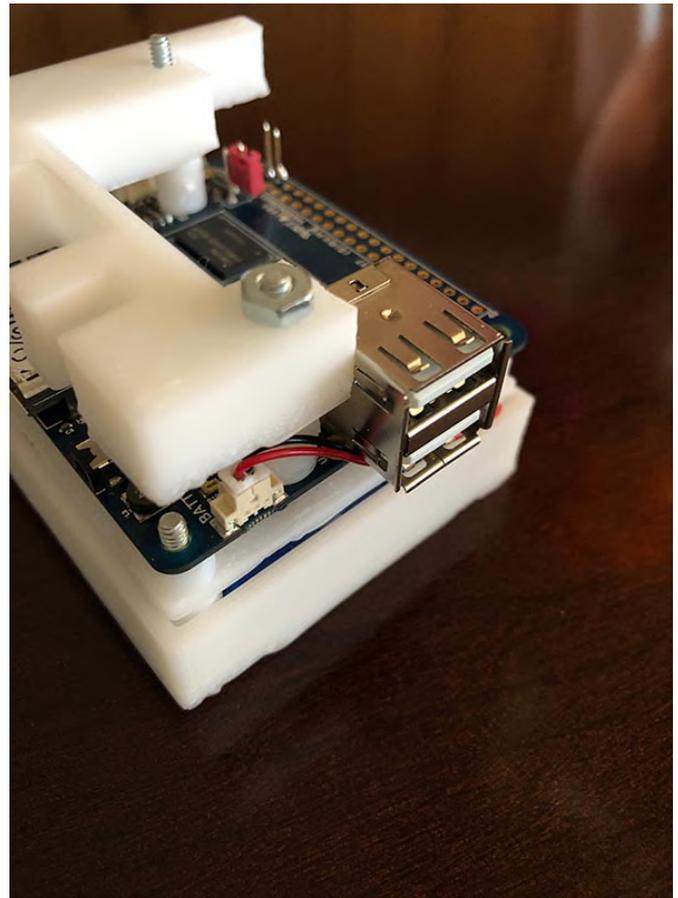
**Figura 5: el ventilador USB opcional ahora se puede conectar a los pines de alimentación GPIO 5V.**

5. Opcionalmente, puede construir una siempre estructura con tu material de fabricación favorito (por ejemplo, madera, plástico, impresiones en 3D, LEGO, etc.) para sostener el ODROID-VU7, ODROID-C0, la batería RTC, la batería de alimentación principal y el ventilador.



**Figura 6:** He imprimido en 3D una estructura y una carcasa para la batería de mi tablet ODROID.

6. Ensambla los componentes de la tablet: conecta la batería RTC, la batería de alimentación principal, el ventilador y el jumper de arranque J1. Colocando el jumper sobre AMBOS pines J1 arrancará desde la tarjeta microSD. Al retirar el jumper, el ODROID-C0 se iniciará desde la interfaz eMMC. Además, conecta el cable rojo (+) del ventilador en el pin #4 en el GPIO y el cable negro (GND; -) del ventilador en el pin #6.



**Figura 7 - Listo para la conexión al ODROID-VU7.**

7. Conecta el ODROID-VU7 al ODROID-C0. Usa el cable HDMI y el cable USB micro Tipo A.

8. Inserta una tarjeta microSD de arranque preconfigurada con el sistema operativo Linux en la toma de la tarjeta y conecta un módulo eMMC de arranque preconfigurado con el sistema operativo Android en su correspondiente interfaz.

9. Ajusta el jumper J1 para arrancar con el sistema operativo elegido. Asegúrate de que los cables USB y HDMI del ODROID-VU7 estén conectados al ODROID-C0. Comprueba que el cable rojo del ventilador esté firmemente conectado al pin GPIO #4 y que el cable negro esté firmemente conectado al pin GPIO #6.

10. Enciende la pantalla del ODROID-VU7 y pon el interruptor de alimentación del ODROID-C0 en la posición ON.

Disfruta de tu tablet ODROID multiarranque controlada por hardware.

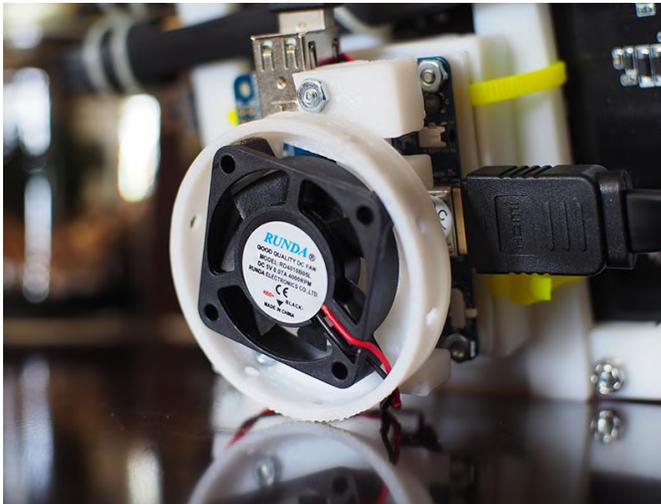
### **Datos divertidos sobre la tablet ODROID**

Si ejecutas la "prueba de esfuerzo" de números primos de sysbench con 10.000 números primos, la

tablet ODROID-C0 terminará en aproximadamente 45.9 segundos.

Durante esta "prueba de esfuerzo", si se activa el ventilador de enfriamiento, la temperatura del SoC llega a los 54 grados Celsius (129 Fahrenheit) y luego bajará rápidamente a 44C (110F).

La temperatura de reposo, sin el ventilador de enfriamiento, del Amlogic S805 es de aproximadamente 47 grados Celsius (116 Fahrenheit). La activación del ventilador de enfriamiento reduce esta temperatura a 38 grados Celsius (100 Fahrenheit).



**Figura 8 - El ventilador de enfriamiento opcional conectado al ODROID-C0**

Arrancando desde la tarjeta microSD, Linux tarda unos 53,65 segundos.

Una batería de 3.000 mAh recién cargada ejecutará el sistema operativo Linux basado en microSD durante aproximadamente 94 minutos.

Las baterías de repuesto de 3.000 mAh se pueden intercambiar en la tablet. Sin embargo, la tablet DEBE apagarse antes de cambiar las baterías.

El uso del ventilador opcional podría causar un problema de consumo de energía con un dongle WiFi conectado. Si notas una pérdida de energía cuando se conecte a tu router, simplemente desconecte el ventilador de los pines de alimentación GPIO.

# El Punto G: Tu Destino para Todas las Cuestiones Relacionadas con Juegos Android

August 1, 2019 By Dave Prochnow Android, Juegos



Cuando el número de julio estaba a punto de salir, varios bombazos de juegos que, sin solicitarlo fueron a parar a mi escritorio. De modo qué, empezaré este artículo para el número de agosto con una actualización sobre cada uno de estos eventos importantes. Estos eventos importantes incluyen: el Electronic Entertainment Expo (E3) que aterriza en Los Ángeles; Google lanza una actualización de Stadia; y el primer avistamiento público "real" del próximo PC portátil para juegos equipado con AMD Ryzen™ V1000.

Con mucho espectáculo, ¿verdad? Excepto que la convención de este año no fue como otras convenciones de E3. ¿Por qué? Porque Sony no se presentó. Sí, el fabricante de Playstation no asistió a ésta, la 25ª Exposición de Entretenimiento Electrónico en el Centro de Convenciones de Los Ángeles en California. Teniendo en cuenta que E3 es el autoproclamado "evento mundial más importante"

para ordenadores y videojuegos y productos relacionados, la ausencia de Sony fue una ausencia importante para un evento que prefiere centrarse en la promoción de los nuevos y futuros lanzamientos de videojuegos.



Figura 1 - Sony no asistió al E3 2019

Afortunadamente, Nintendo, Microsoft, Square Enix y otras compañías hicieron suficientes anuncios importantes para compensar la notable ausencia de Sony: ya sabes, el elefante en la habitación del que nadie quería hablar.

Desplazándome por la enorme lista de los próximos lanzamientos de juegos anunciados en le E3 2019, localice un título relacionado con Android que debería

interesar a los usuarios de ODROID: Commander Keen de Bethesda llegará a las máquinas a finales de este año.

### Google Stadia is Almost 'Readia'

También está previsto que llegue a finales de este año el servicio de streaming de juegos de Google conocido como Stadia. En un anuncio dos días antes del E3 2019, el jefe de Stadia de Google, Phil Harrison, informó a los jugadores que Stadia costaría 9.99\$ al mes para el servicio ilimitado de juegos. Podrás jugar a estos juegos a una resolución de 4K a velocidades de hasta 60 frames por segundo (fps), y molestar a los vecinos con un sonido envolvente 5.1. Además, estos juegos Stadia estarán disponibles tanto en los televisores Chromecast Ultra como en los navegadores Chrome para ordenadores portátiles y de escritorio. Los únicos dispositivos inteligentes compatibles serían los teléfonos Google Pixel.

Hay un paquete especial "Founder's Edition" que ofrece acceso total a Stadia, un mando de juego exclusivo, un dongle Google Chromecast Ultra y más productos al precio de compra online de 129\$. Se trata de un paquete pre-pedido que se enviará en noviembre de 2019.

[https://store.google.com/us/product/stadia\\_founders\\_edition?hl=en-US](https://store.google.com/us/product/stadia_founders_edition?hl=en-US)

### Gamer SMACH

Los jugadores empedernidos agonizarán por el próximo lanzamiento de SMACH Z, un PC portátil para juegos que se asemeja a la PSP de Sony. Financiado por una campaña de Kickstarter que recaudó más de 500K \$, el SMACH Z fue visto y jugado por algunos jugadores afortunados en el E3 2019.



Figura 2: unos pocos afortunados en el E3 2019 pudieron jugar el próximo título del juego DOOM en el PC de juegos portátil SMACH Z que se lanzará próximamente

<https://youtu.be/g68bH2E8B3g>

### And Finally

Diseñado en colaboración con WB Games (Portkey Games) y Niantic, el lanzamiento del nuevo título de realidad aumentada (AR) Harry Potter: Wizards Unite ha visto algunas descargas serias de Google Play Store. Con un concepto similar a Pokemon: Go (también desarrollado por Niantic), Harry Potter: Wizards Unite es un juego mágico que es totalmente compatible con el creador de Harry, J.K. Rowling. Lo mejor de todo, Wizards Unite es gratuito.



Figura 3: Prepárate para la multitud de aspirantes a magos que correrán por los parques con teléfonos inteligentes en la mano este verano

### Selección de Juegos Android para los Dog Days del Verano

Jurassic World Alive - GRATUITO Farm Punks - GRATUITO Harry Potter: Wizards Unite - GRATUITO Talion - GRATUITO Toy Story Drop - GRATUITO

# Implementación del Manipulador GPIO IRQ: Usando Python 3 para Controlar RPi.GPIO

© August 1, 2019 By Justin Lee ODR0ID-C0, ODR0ID-C1+, ODR0ID-C2, ODR0ID-H2, ODR0ID-N2, ODR0ID-XU4, Tutoriales



Este código y esta guía tienen la intención de probar la gestión de GPIO IRQ en el ODR0ID-C1+/C2/XU4/N2. La guía ha sido adaptada partiendo de la página wiki de ODR0ID [https://wiki.odroid.com/odroid-xu4/application\\_note/gpio/rpi.gpio\\_irq](https://wiki.odroid.com/odroid-xu4/application_note/gpio/rpi.gpio_irq).

Simplemente tenemos que implementar el manipulador GPIO IRQ con Python 2/3. En esta guía, usaremos Python 3 para programar el manipulador. Sin embargo, antes de empezar, tenemos que instalar RPi.GPIO para ODR0ID. Consulta la página Wiki [https://wiki.odroid.com/odroid-xu4/application\\_note/gpio/rpi.gpio](https://wiki.odroid.com/odroid-xu4/application_note/gpio/rpi.gpio) para las instrucciones de instalación.

## Código de muestra

```
#!/usr/bin/env python3
```

```
import sys
import time
```

```
import RPi.GPIO as GPIO

# https://wiki.odroid.com/odroid-
xu4/application_note/gpio/rpi.gpio#about_bcm_numbe
ring
IRQ_GPIO_PIN = 25
IRQ_EDGE = GPIO.FALLING
count = 0

def handler(channel):
    global count

    count += 1

def print_status():
    global count

    print(count)
    count = 0

if __name__ == '__main__':
    GPIO.setmode(GPIO.BCM)
```

```

GPIO.setup(IRQ_GPIO_PIN, GPIO.IN,
pull_up_down=GPIO.PUD_UP)
GPIO.add_event_detect(IRQ_GPIO_PIN, IRQ_EDGE,
callback=handler)

print('Press Ctrl-C to exit')
try:
while True:
time.sleep(1)
print_status()
except KeyboardInterrupt:
GPIO.cleanup()
sys.exit(0)

```

Este es un script modelo muy básico para usar el manipulador GPIO IRQ. Cuenta cuántas interrupciones se solicitan en 1 segundo y muestra el recuento total. Si un usuario presiona Ctrl + C, el script se cerrará.

Hay una función handler () que recibe un parámetro. Este parámetro es necesario ya que la librería lo necesita para registrar internamente la función del manipulador. Esta función aumenta el número de conteo en 1 cuando se recibe una interrupción. La función print\_status () muestra el número del conteo e pone la variable de conteo a 0. Si el archivo de script es el principal archivo ejecutado, que significa que es el primer archivo del proyecto Python, RPi.GPIO inicialmente configurado usa GPIO.setmode (). Esta función autoriza al usuario y usa la numeración BCM para seleccionar un pin GPIO.

En GPIO.setup (), el pin GPIO seleccionado está fijado como una fuente de interrupción en este tiempo para usarse de la forma que el usuario pretendía. Deberíamos introducir 3 parámetros, que son el número de pin GPIO en la numeración BCM, la dirección de la señal y el modo pull. Deberíamos fijar la dirección de la señal y el modo pull up para recibir la interrupción GPIO.

Tenemos que añadir una función de manipulador de eventos usando GPIO.add\_event\_detect (). Esta tiene 3 parámetros, que son el número de pin GPIO en la numeración BCM, el modo edge IRQ y el puntero de la función del manipulador como su nombre. La librería RPi.GPIO registrará el manipulador para el pin GPIO en un bucle interno. Especificamos el modo edge de

interrupción para reducir edge utilizando esta función.

Finalmente, hay códigos de gestión de excepciones que detectan interrupciones del teclado (SIGINT). Si se recibe la interrupción, el RPi.GPIO se limpia solo con la función GPIO.cleanup () y el programa se cerrará. Si no se recibe la interrupción, el bucle infinito se ejecuta y llama a print\_status () cada 1 segundo, así que podemos limpiar estas largas descripciones.

## Funciones

- def handler(): un manipulador de interrupciones. Aumenta el número de conteo en 1 cuando ocurre la interrupción. Requiere al menos 1 parámetro.
- def print\_status (): muestra el número de conteo actual e inicializa a 0.
- GPIO.setmode (): Inicia RPi.GPIO con una guía de números pin que debe usarse.
- GPIO.setup (): fija el pin GPIO como fuente de interrupción. También configura la dirección de la señal y el modo de extracción.
- GPIO.add\_event\_detect (): ajusta el mismo pin GPIO que se configuró antes, con el modo edge de interrupción y la función del controlador.
- Ejecuta la función print\_status cada 1 segundo y detecta la interrupción del teclado para cerrar el programa adecuadamente.

## Entornos

Para probar su rendimiento de gestión IRQ, utilicé el siguiente conjunto de dispositivos de prueba

- ODROID C1/C2/XU4/N2
- Generador de funciones
- Osciloscopio

Configuré el generador de funciones para generar una onda cuadrada de 1 KHz, y verifiqué esa onda usando un osciloscopio. Cambié la amplitud de cada objetivo a 1.8V o 3.0V cada vez que cambiaba la placa. Luego elegí los pines #22 para la fuente de interrupción y #20 para la puesta a tierra. El pin físico #22 es equivalente al #25 en la numeración BCM. Los tres modelos tienen el mismo factor de forma en el cabezal GPIO. Finalmente, conecté los cables tal y como se muestra en la Figura 1.

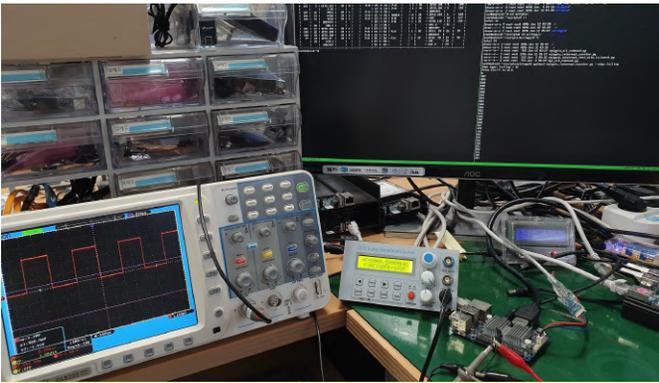


Figura 1 - Configuración de prueba C1 + IRQ

## Ejecución

Probemos primero con el C1+. Simplemente ejecute el script sin ningún cambio. Tal como está el script, debería mostrar número en torno a 1000 cada segundo, ya que está configurado para manejar las interrupciones edge descendentes y el generador de funciones está configurado para generar una onda cuadrada de 1 KHz. Para asegurarme, también comprobé la versión actual del kernel:

```
root@odroid:~# uname -a
Linux odroid 3.10.107-13 #2 SMP PREEMPT Wed Jun 19
02:31:43 -03 2019 armv7l armv7l armv7l GNU/Linux
root@odroid:~# python3 test.py
Press Ctrl-C to exit
1000
1003
1000
1000
1000
1001
1001
987
1001
1001
```

Éste muestra los números tal y como esperábamos. No podían ser exactamente 1000 en cada momento, porque hay muchos factores [no] previstos que afectan la detección de las interrupciones. Este es el resultado en el ODR0ID-C2:

```
root@odroid:~# uname -a
Linux odroid 3.16.68-41 #1 SMP PREEMPT Tue Jun 18
15:06:16 -03 2019 aarch64 aarch64 aarch64
GNU/Linux
root@odroid:~# python3 test.py
Press Ctrl-C to exit
981
```

```
993
993
984
985
996
977
995
1001
981
```

El siguiente resultado es del ODR0ID-XU4, que usa el pin #26 para la fuente de interrupción y el #28 para la puesta a tierra:

```
root@odroid:~# uname -a
Linux odroid 4.14.120-160 #1 SMP PREEMPT Fri May
17 01:18:14 -03 2019 armv7l armv7l armv7l
GNU/Linux
root@odroid:~# python3 test.py
Press Ctrl-C to exit
1005
1007
1014
1005
1008
1003
1005
1001
1005
1004
```

Finalmente, así es como se ve en el ODR0ID-N2

```
root@odroid:~# uname -a
Linux odroid 4.9.182-31 #1 SMP PREEMPT Tue Jun 18
14:45:56 -03 2019 aarch64 aarch64 aarch64
GNU/Linux
root@odroid:~# python3 test.py
Press Ctrl-C to exit
1000
1002
1002
1003
1000
1000
1000
999
1000
1000
```

Todos los modelos funcionan bien en la gestión de GPIO IRQ con RPi.GPIO. Ten en cuenta que el número

que se muestra en el rango de error aceptable no tiene importancia, ya que cambia continuamente.

Configurar el tipo de Edge GPIO IRQ

También podemos cambiar el tipo de detección edge del pin. Hay tres tipos de edge:

- Falling
- Rising
- Both

Esto se puede cambiar cuando configuras el manipulador GPIO IRQ en el código utilizando la función `GPIO.add_event_detect()`. Estos se definen respectivamente como:

```
GPIO.FALLING
GPIO.RISING
GPIO.BOTH
```

Si deseas detectar utilizando el modo both-edge, reemplaza el valor `IRQ_EDGE` existente en la línea #9 por el siguiente:

```
# From
IRQ_EDGE = GPIO.FALLING
```

```
# To
IRQ_EDGE = GPIO.BOTH
```

Aquí tienes una prueba para verificar si funciona. Con C1+, el script modificado produce una onda cuadrada de 1 KHz.

```
root@odroid:~# python3 test.py
Press Ctrl-C to exit
1994
1997
1994
1998
1999
1980
2002
2001
2001
2002
```

Muestra aproximadamente 2000 porque el controlador reacciona para ambos momentos de edge, lo que significa que funciona sin problemas.

Referencias

<https://sourceforge.net/projects/raspberry-gpio-python/>

# Ejecutando Kubernetes en el ODROID-N2: Crea un Sistema de Organización de Contenedores de Eficiencia Energética

© August 6, 2019 👤 By Thomas Kruse 📄 Linux, Tutoriales



# kubernetes

El despliegue de aplicaciones en contenedores es la nueva moda. Con ello, surge la necesidad de automatizar las implementaciones, ampliarlas para satisfacer el aumento de las cargas de trabajo y administrar su ciclo de vida. Kubernetes (K8s) es una de esas utilidades que cubre estas necesidades.

Los contenedores se pueden implementar en dispositivos de hardware de muy diversas características y de una amplia gama de precios. Un ordenador de placa reducida (SBC) totalmente autónoma es uno de esos dispositivos. El ODROID-N2 es un SBC altamente capacitado, y lo convierte en un dispositivo K8 muy rentable. Está disponible con 4 GB de RAM, una tarjeta eMMC como almacenamiento de alta velocidad y una gran variedad de soporte E/S. Junto con la fuente de alimentación, cuesta menos de 100 euros (~ 115\$). Con 4 + 2 núcleos de CPU ARM64, el ODROID-N2 es una plataforma interesante para poner en marcha un pequeño clúster K8 con un gasto

energético relativamente bajo. Incluso se puede utilizar para experimentar con una implementación K8 de bajo coste, antes de implementar soluciones más costosas.

Este artículo explica cómo configurar K8 en el ordenador de placa reducida ODROID-N2. Dado que hay varias opciones para los sistemas operativos, así como los métodos de distribución y configuración de K8, este artículo parte de las siguientes premisas:

- Utiliza Arch Linux ARM64 como sistema operativo base (éste es bastante sencillo y está muy actualizado)
- Vanilla K8s se usará, compilará y empaquetará como paquetes Arch ARM64 en el ODROID-N2
- Se usará Kubeadm plano para configurar el clúster K8
- CRI-O como tiempo de ejecución del contenedor (en lugar de Docker)
- Nodo maestro único y 4 nodos de trabajo

Desafortunadamente, no hay soporte del kernel Linux estándar para ODROID-N2. Sin embargo, Hardkernel ha prometido trabajar en ello. Las siguientes características no funcionan actualmente como cabría esperar:

- zram para memoria comprimida como dispositivo de intercambio
- Deshabilita la asignación de memoria de la GPU para usar los 2GB/4GB completos del ODROID-N2

Las experiencias anteriores con Arch Linux ARM 64bit y K8s en Raspberry Pi y ODROID (ODROID-C2 para ser precisos) las puedes encontrar aquí:

Kubernetes auf Raspberry Pi (<https://bit.ly/30PDiyu>)  
Kubernetes auf ODROID mit zram (<https://bit.ly/2OgXjMX>)  
Kubernetes auf Arch Linux ARM (<https://bit.ly/2LEsqzT>)  
Kubernetes mit CRI-O auf Arch Linux ARM (<https://bit.ly/2JR5nj4>)  
Kubernetes mit CRI-O Worker auf Arch Linux ARM64 (<https://bit.ly/2Y6pl21>)  
Kubernetes auf ODROID Arch Linux ARM Mainline Kernel (<https://bit.ly/2JT9A6b>)  
Kubernetes Dashboard auf ARM 64 (<https://bit.ly/2LIVPZR>)

## Instalando Arch Linux en ODROID-N2

Arch Linux es bastante fácil de configurar. Las instrucciones generales de instalación las puedes encontrar aquí: <https://bit.ly/2JSlahB>. Para facilitar la configuración de múltiples nodos, se puede recurrir a la programación para semi-automatizar el tema del almacenamiento (eMMC o tarjeta SD) y extraer el sistema base. Puesto que debemos aplicar varias personalizaciones, como la copia de claves SSH, la configuración de los derechos sudo y la configuración del nombre de host, realmente vale la pena recurrir a la automatización.

Tras la instalación, también se instalan los siguientes paquetes

- sudo, htop
- socat, ethtool, ebttables (para redes K8s CNI)
- cpupower (reduce el consumo de energía permitiendo regular la CPU durante los períodos de inactividad)
- nfs-utils (si el almacenamiento NFS se va a usar con K8)

Para utilizar los 6 núcleos de la CPU al comprimir los paquetes de Arch Linux, se pueden configurar los siguientes parámetros en /etc/makepkg.conf:

```
COMPRESSXZ=(xz -T0 -c -z -)
```

Esto configurará la compresión multiproceso para la creación de paquetes de Arch Linux

## Compilando paquetes K8s Arch Linux ARM 64

Al principio, los paquetes actualizados para K8 y los servicios de soporte serán compilados como paquetes de Arch Linux. Se recomienda crear un directorio para cada paquete que se compile y colocar el archivo PKGBUILD en cada uno.

Puedes encontrar los archivos PKGBUILD usados aquí:

- runc
- CNI-Plugins
- CRI-O
- CRI Tools
- Kubernetes, Install File

Por lo general, la creación de un paquete se lleva cabo introduciendo el siguiente comando en cada directorio:

```
$ makepkg -s
```

De momento, se pueden compilar todos los paquetes, excepto el paquete Kubernetes Arch.

Para los K8, se deben realizar algunos pasos específicos, ya que una compilación de Kubernetes requiere muchos recursos: en un ODROID-N2 de 4GB es posible llevar a cabo una compilación sin memoria de intercambio adicional, pero como mínimo se necesitan unos 3.5 GB. Si vas a utilizar un modelo de 2 GB, puedes añadir un archivo de intercambio swap:

```
$ sudo fallocate -l 1000M /swapfile  
$ sudo mkswap /swapfile  
$ sudo swapon /swapfile
```

Además del archivo de intercambio, debemos realizar dos configuraciones. El kernel debe permitir exceder la memoria disponible en lugar de estar expectante a la hora de asignar memoria:

```
$ sudo sysctl -w vm.overcommit_memory=1
```

y se debe evitar que la cadena de compilación realice compilaciones paralelas con el número de núcleos disponibles, lo cual aumentaría el consumo de memoria:

```
export GOFLAGS="-p=1"
```

Aunque cada compilación en sí no se ejecutará en paralelo, cada parte del paquete de Kubernetes puede aprovechar todos los núcleos durante su compilación individual, evitando así reducciones importantes del rendimiento. Dado que Arch usa un sistema de archivos tmpfs para /tmp, primero debemos desmontarlo, de lo contrario, la memoria se asignará a compilaciones temporales y posiblemente conduzca a un estado de "sin memoria":

```
$ sudo umount /tmp
```

Una vez finalizadas estas configuraciones, se puede compilar K8:

```
$ makepkg -s
```

Cuando se complete la compilación, los siguientes paquetes deberían estar presentes:

```
cni-plugins-0.7.5-1-aarch64.pkg.tar.xz
cri-o-1.14.0-1-aarch64.pkg.tar.xz
crictl-bin-1.14.0-1-aarch64.pkg.tar.xz
runc-1.0.0rc8-1-aarch64.pkg.tar.xz
kubernetes-1.14.1-1-aarch64.pkg.tar.xz
```

Estos paquetes ahora se pueden distribuir a todos los nodos ODROID-N2 que participan en el clúster. Por supuesto, también se pueden usar otras máquinas, siempre que todas ellas sean plataformas de hardware ARM64.

## Configuración general del nodo ODROID-N2 K8s

Antes de instalar los paquetes, debemos configurar ciertas cosas para que el funcionamiento de la red de contenedores sea el correcto

Las siguientes características del kernel deben estar presentes, de lo contrario, la red K8s no funcionará y podría conducir a errores realmente difíciles de diagnosticar como los siguientes:

```
iptables: No chain/target/match by that name
```

```
Unexpected command output Device 'eth0' does not exist:
```

- CGROUP\_PIDS
- NETFILTER\_XTABLES, XT\_SET

Si al kernel le falta algún servicio o características, como se muestra en el siguiente resultado, la solución más rápida es compilar un nuevo paquete de kernel que incluya las características necesarias.

```
$ zgrep XT_SET /proc/config.gz
# CONFIG_NETFILTER_XT_SET is not set
$ zgrep CONFIG_NETFILTER_XTABLES /proc/config.gz
CONFIG_NETFILTER_XTABLES=m
```

## Verificación de las características del kernel para K8s CNI

La compilación es bastante fácil, ya que el paquete de kernel Arch Linux se puede compilar utilizando las herramientas habituales. Para acelerar el proceso de compilación, se recomienda editar /etc/makepkg.conf y habilitar la compilación multiproceso usando MAKEFLAGS = "-j6", que contempla los 6 núcleos disponibles en el ODROID-N2.

```
$ git clone
https://github.com/everflux/PKGBUILDs.git
$ cd PKGBUILDs/core/linux-odroid-n2
$ git checkout patch-1
$ makepkg -s
```

La instalación del paquete del kernel se realiza con pacman. Luego configuraremos la red.

```
$ sudo sh -c 'echo "net.ipv4.ip_forward=1" >>
/etc/sysctl.d/30-ipforward.conf'
$ sudo sysctl -w net.ipv4.ip_forward=1
$ sudo sh -c 'echo "br_netfilter" > /etc/modules-
load.d/br_netfilter.conf'
$ sudo sh -c 'echo "xt_set" > /etc/modules-
load.d/xt_set.conf'
$ sudo modprobe br_netfilter xt_set
```

En cada nodo, deben instalarse los paquetes de herramientas de contenedor y Arch8 K8 compilado previamente. Si se compila un paquete de kernel personalizado, también se instalará igualmente.

```
$ sudo pacman -U *pkg.tar.xz
loading packages...
resolving dependencies...
looking for conflicting packages...
Packages (5) cni-plugins-0.7.5-1 cri-o-1.14.0-1
crictl-bin-1.14.0-1 kubernetes-1.14.1-1 runc-
1.0.0rc8-1
Total Installed Size: 1065.89 MiB
:: Proceed with installation? [Y/n]
...
```

## Instalación de todos los paquetes

Después de la instalación, el tiempo de ejecución del contenedor CRI-O requiere configuración. CRI-O respeta la configuración de todo el sistema de registros de contenedores fiables que hay dentro de /etc/containers/policy.json. Para poder extraer imágenes de docker.io (y otros registros) se puede instalar la política por defecto: policy.json

Aquí tienes una configuración mínima para CRI-O: cri.conf. Debe ubicarse en /etc/crio/crio.conf. Para evitar que CRI-O desactive la red de contenedores debido a la ausencia de una configuración de red CNI por defecto, montamos una configuración CNI con un simple circuito cerrado.

```
$ sudo sh -c 'cat >/etc/cni/net.d/99-loopback.conf
<<-EOF
{
  "cniVersion": "0.2.0",
  "type": "loopback"
}
EOF'
```

Posteriormente, el servicio CRI-O se puede habilitar e iniciar.

```
$ sudo systemctl daemon-reload
$ sudo systemctl enable crio
$ sudo mkdir -p /etc/cni/net.d
$ sudo systemctl start crio
$ sudo systemctl enable kubelet.service
```

## Configuración maestra Kubernetes ODROID-N2

En el nodo maestro, la configuración del clúster se realizará utilizando kubeadm. Puesto que incluso el último ODROID-N2 con 4 GB de RAM está bastante limitado con memoria, me viene a la mente el servicio

adicional zram-swap o usar un archivo de intercambio. Para ejecutar K8 con el intercambio habilitado, se debe facilitar la configuración "--ignore-preflight-errors Swap" a kubeadm:

```
$ sudo kubeadm init --ignore-preflight-errors Swap
--cri-socket=/var/run/crio/crio.sock
```

Luego puedes conectar cuantos nodos de trabajo quieras ejecutando lo siguiente en cada uno de ellos como root

```
kubeadm join 10.23.200.120:6443 --token c11wrg... --
discovery-token-ca-cert-hash sha256:3f5dc1..
```

## Configuración maestra de Kubernetes

Una vez finalizada la configuración de kubeadm y aparezca el token de unión, se pueden configurar los nodos de trabajo. Pero primero debemos hacer una copia de la configuración del clúster en el directorio de inicio del usuario, para que luego se pueda recuperar para poder configurar kubectl.

```
$ mkdir -p $HOME/.kube
$ sudo cp -i /etc/kubernetes/admin.conf
$HOME/.kube/config
$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

## Configuración del nodo trabajador de Kubernetes

Puesto que la configuración es la misma para los nodos de trabajo y maestros, queda muy poco por hacer: el comando kubeadm se usará para unirse al clúster, después de ello la configuración de trabajo de K8s en el nodo habrá finalizado.

Si los nodos de trabajo tienen habilitado el intercambio, también se debe facilitar el parámetro "--ignore-preflight-errors Swap":

```
$ sudo kubeadm join 10.23.202.120:6443 --ignore-
preflight-errors Swap --token c11wrg.... --
discovery-token-ca-cert-hash sha256:3f5dc1...
```

## Acceso y red del Cluster

Para acceder al clúster K8s, el archivo de configuración generado para kubectl se obtiene del maestro.

```
$ mkdir ~/.kube/config
$ scp master:~/admin.conf ~/.kube/config
```

Luego, se debe poder acceder al clúster desde kubectl.

```
$ kubectl get nodes
NAME           STATUS    ROLES    AGE    VERSION
n2-master0     NotReady  master   11m    v1.14.1
n2-worker0     NotReady             5s     v1.14.1
n2-worker1     NotReady             10s    v1.14.1
n2-worker2     NotReady             9s     v1.14.1
n2-worker3     NotReady             8s     v1.14.1
```

## Acceder al clúster K8s recién configurado

Todos los nodos están en estado NotReady ya que no se ha configurado ninguna red para el clúster. Esto se puede solucionar rápidamente usando weave como proveedor de CNI:

```
$ kubectl apply -f
"https://cloud.weave.works/k8s/net?k8s-
version=$(kubectl version | base64 | tr -d '
')"
```

Una vez consolidada la red Weave, los nodos cambian al estado Ready.

```
$ kubectl get nodes
NAME           STATUS    ROLES    AGE    VERSION
n2-master0     Ready     master   77m    v1.14.1
n2-worker0     Ready             65m    v1.14.1
n2-worker1     Ready             65m    v1.14.1
n2-worker2     Ready             65m    v1.14.1
n2-worker3     Ready             65m    v1.14.1
```

Para hacernos con una interfaz basada en web para el clúster, instalamos el panel de K8s. Aunque es proporcionado como una imagen ARM64, la implementación por defecto usa amd64 como plataforma, por lo que es necesario llevar a cabo una pequeña sustitución con sed:

```
$ curl -sSL https://bit.ly/2G4e9Hu | sed 's/-
amd64:/-arm64:/' | kubectl apply -f -
```

## Referencias

<https://kubernetes.io/>

<https://www.trion.de/news/2019/05/06/kubernetes-odroid-n2.html>

# Arranque Múltiple en ODROID-N2: Añadiendo la Posibilidad de Seleccionar Múltiples Sistemas Operativos desde un Único Disco

© August 6, 2019 By Dongjin Kim ↗ ODROID-N2, Tutoriales

```
Petitboot (dev.20190618)

[Disk: mmcblk0p1 / 0000-0058]
  android_9.0.0_32bit
[Disk: mmcblk1p1 / 2505-2655]
  CoreELEC
[USB: sda1 / 5acf1714-e583-4079-b0b5-1a4c69]
* Debian Stretch

System information
System configuration
System status log
Language
Rescan devices
Retrieve config from UBI
```

El ODROID-N2 es un ordenador de placa reducida (SBC) fabricado por Hardkernel, que fue lanzado en febrero de 2019 y que funciona con un S922X de Amlogic (4xA73@1.8GHz y 2xA53@1.9GHz). Una de las nuevas características del ODROID-N2 es una memoria flash SPI de 8 MB. Obviamente, la memoria flash SPI integrada no es una tecnología nueva y algunos SBC ya la tienen para usarse en ciertos propósitos. Normalmente, la memoria flash SPI tiene un gestor de arranque que permite cargar la imagen del sistema operativo desde un almacenamiento más grande como es un almacenamiento USB o uSD

La memoria flash SPI en el ODROID-N2 puede ser de arranque y 8 MB es un espacio bastante grande para un gestor de arranque que generalmente ocupa menos de 1 MB. Por otro lado, es muy pequeña para la imagen del kernel de Linux, que suele ser de 8 ~ 9 MB para una imagen comprimida de ODROID-N2. Nuestro objetivo es mejorar el uso de este pequeño

almacenamiento en lugar de simplemente instalar un gestor de arranque.

Ha habido mucha demanda para administrar múltiples sistemas operativos y muchos usuarios de SBC han intentado ejecutar múltiples sistemas operativos instalados en el almacenamiento USB, uSD y eMMC y, como tal, se proporcionaron instrucciones con un script o comandos de Linux. Muchos usuarios lo lograron, pero las instrucciones no son fáciles para los usuarios que no están familiarizados con un sistema Linux. Queremos ayudar a esos nuevos usuarios que desean utilizar dos sistemas operativos pero que tienen problemas para configurar múltiples entornos de sistemas operativos.

Por estas dos razones, mi compañero de trabajo Joy y yo empezamos a estudiar Petitboot, que es un gestor de arranque de sistema operativo basado en Linux kexec. Tuvimos que esforzarnos bastante para compilar una imagen completa que se ajustara a los

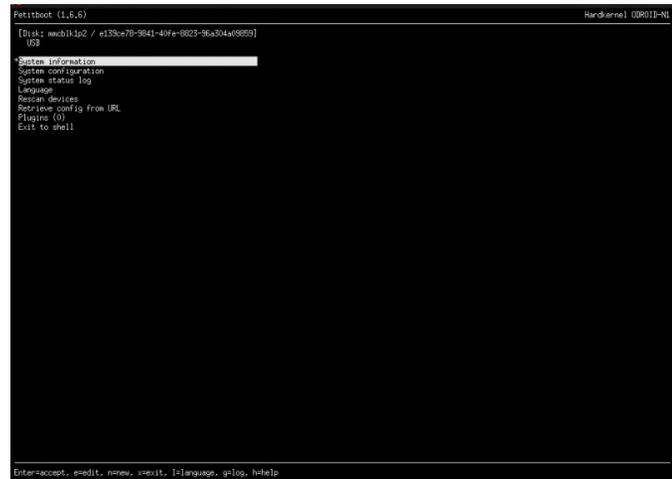
8MB, en concreto, tuvimos que personalizar el kernel de Linux para reducirlo a un tamaño más pequeño de unos 3.8MB, que es más de la mitad del tamaño del kernel de Linux genérico. Al mismo tiempo, puesto que Petitboot es una aplicación que se ejecuta en el espacio de usuario de Linux, tuvimos que compilar un pequeño sistema de archivos raíz que se ajustara a los 3.2MB. De lo contrario, no podríamos instalar el pequeño sistema Linux en 8MB. Aquí tienes el mapa de memoria de la imagen para arrancar Petitboot en 8MB.

Offset	Partition	Size (bytes)	
		Dec	Hex
00000000	U-boot	983040	F0000
000EFFFF			
000F0000	ENV	65536	10000
000FFFFFFF			
00100000	DTB	131072	20000
0011FFFF			
00120000	Kernel (uImage)	3932160	3C0000
004DFFFF			
004E0000	Initrd (rootfs.cpio.gz)	3276800	320000
007FFFFFFF			
00800000			
<b>Total size</b>		<b>8388608 bytes</b>	<b>8 MBytes</b>

**Figura 01: Mapa de memoria de la imagen para iniciar Petitboot en 8 MB.**

Después de salvar varios obstáculos, hemos logrado compilar con éxito el pequeño sistema Linux de arranque que incluye U-boot en 8 MB y de esta forma ODROID-N2 pueda ejecutar Petitboot desde la memoria flash SPI. Aunque la primera versión únicamente mostraba los menús de Petitboot y la entrada de inicio configurados manualmente para un sistema operativo ya que había formatos de inicio

como Grub o PXE que no eran compatibles con el ODROID-N2.



**Figura 03 - menu petitboot**

Como mi intención era ayudar a los usuarios a iniciar un sistema operativo sin tener que configurar prácticamente nada, tuve que añadir un componente que pudiera reconocer la imagen del sistema operativo aportada por la comunidad de código abierto. Todas las imágenes del sistema operativo para SBCs ODROID tienen un script de arranque especial boot.ini que contiene los comandos del gestor de arranque para configurar los parámetros del kernel, qué kernel debe cargarse, etc. De lo contrario, el usuario tendría que aprender el nuevo formato de configuración de arranque para Petitboot y asignar los archivos de arranque manualmente.

He escrito un código que analiza el archivo boot.ini y pude hacer que Petitboot lea la imagen del sistema operativo del almacenamiento conectado al ODROID-N2 y se muestre en pantalla para que el usuario pueda seleccionar la correspondiente entrada de arranque usando un Teclado USB conectado o un control remoto IR. Creo que éste es un gran logro y permite al usuario seleccionar fácilmente un sistema operativo instalado y conectado al ODROID-N2.

He administrado la versión de Petitboot para ODROID-N2 durante el mes de junio de 2019 y, finalmente, pude corregir los errores más importantes y mejoré algunas funciones que normalmente lleva a cabo U-boot para arrancar un sistema operativo. Aun así, no es una solución perfecta. Como no puedo ocuparme completamente de esta tarea de Petitboot, el Petitboot actual no

funciona tan bien como esperaba y soñaba, pero de vez en cuando lo mejoraré y lo pondré a disposición de cualquiera.

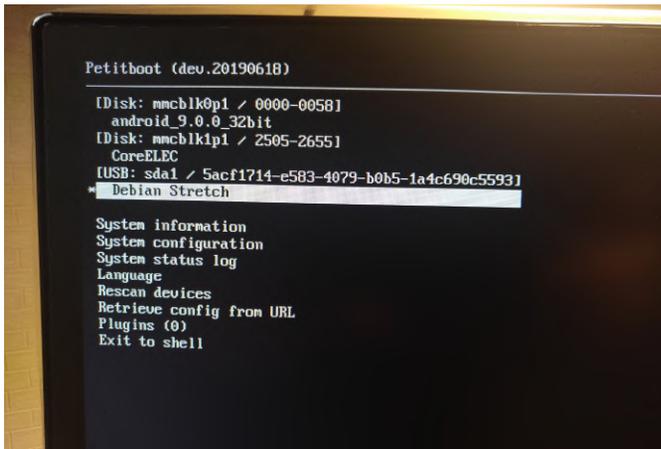


Figura 03 - Menu Petitboot

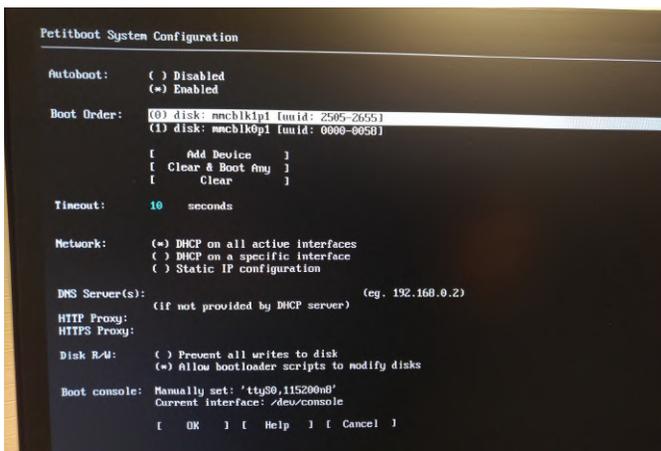


Figura 04 - Menu Petitboot

## La versión actual puede hacer lo siguiente:

- Reconocer las diferentes imágenes del sistema operativo de la comunidad de código abierto tal como son, lo que significa que perfectamente puede iniciar la imagen del sistema operativo tal como se inicia desde la ranura eMMC o uSD sin ningún cambio. Descubrí que Armbian ha cambiado la estructura de su sistema de archivos, por lo que es necesario realizar algunos cambios.
- Se puede arrancar fácilmente desde un almacenamiento USB. Muchos SoC para SBC no permiten arrancar desde un almacenamiento USB que

no sea uSD o eMMC. ODROID-N2 es capaz de arrancar desde USB seleccionando fácilmente la entrada de arranque del almacenamiento USB en lugar de utilizar una línea de comando.

- El orden del dispositivo de arranque se puede configurar desde el menú, puedes elegir uSD como medio de arranque prioritario o el USB.
- Arranque automáticamente para los soportes de arranque de alta prioridad en la segunda unidad.

## La versión actual tiene algunas limitaciones:

- La configuración de arranque debe almacenarse en uSD, lo cual es incómodo. Esto se debe a la limitación de la función SoC, aunque esto estudiando una solución alternativa.
- Todavía es un proyecto WIP y tiene algunos problemas para trabajar con el arranque heredado desde uSD o eMMC. Debido a que parte del trabajo que se realizó en U-boot se está moviendo a Petitboot y no todas las funciones se mueven o no son compatibles.
- El arranque desde soportes de red como PXE no es compatible, aunque lo he incluido en mi lista que cosas pendientes por implementar
- Al no ofrecer UX sofisticado debido a la pequeña capacidad de la memoria flash SPI, 8MB es el margen para almacenar los BLOB obligatorios, únicamente.

Me alegró ver que muchos usuarios están intentando usar Petitboot en su ODROID-N2 y muchos de ellos están contentos porque pueden iniciar un sistema operativo con un propósito diferente. Mantengo un hilo en el foro ODROID en <https://forum.odroid.com/viewtopic.php?f=182&t=33873>, y seguiré actualizando la versión cada vez que haga correcciones y/o mejoras. Para comentarios, preguntas y sugerencias, visite el artículo original en <https://medium.com/@tobetter/multiple-os-on-odroid-n2-3a5f3a14a726>.

# Juegos Linux: PC-Engine TurboGrafx - Parte 5

© August 1, 2019 By Tobias Schaaf Juegos, Linux



En la anterior entrega de esta serie, logre jugar a una gran variedad de diferentes géneros. Descubrí algunos juegos de disparos que me gustaron, uno de mis juegos de aventuras favoritos de todos los tiempos, e incluso el único juego de deportes que me ha llegado a gustar en el Sistema de Entretenimiento de Nintendo (NES) con mejores gráficos que TurboGrafx-16 (PCE). En general, la experiencia ha sido muy variada, con muchos y buenos juegos, aunque también encontré algunos que no me agradaron demasiado. Veamos cómo son los siguientes juegos a medida que nos acercamos al final de esta serie.

## Juegos que me han gustado

### Pop'n Magic



Figura 1: Pop'n Magic tiene unos gráficos muy coloridos y está diseñado especialmente para niños



**Figura 2: Los jefes pueden llegar a ser bastante difíciles de eliminar en este juego**

Este juego es muy similar a la serie Bubble Bobble con un sistema de juego muy parecido. Atrapas a los enemigos disparándoles, quedando atrapados en una burbuja de color/bola de energía. Puedes coger la burbuja y arrojarla contra otras burbujas para destruirlas, lo cual hará que suelten muchos elementos de bonificación, como son los potenciadores, armas más potentes y la posibilidad de desplazarte más rápido. Para esto, debes lanzar la burbuja contra otra de color diferente. Si la lanzas contra una burbuja del mismo color liberarás al enemigo nuevamente. En lugar de lanzar las burbujas unas contra otras, también puedes disparar y destruirlas de esta forma, pero te darás menos puntos o menos potenciadores. El juego es bastante aleatorio, con una buena banda sonora CD y escenas de anime entre los mundos (aunque todos japoneses). Al final de cada mundo, lucharás contra un jefe que es bastante difícil de vencer, pero te recompensará con varias vidas a partir de todos los puntos que vas acumulando.

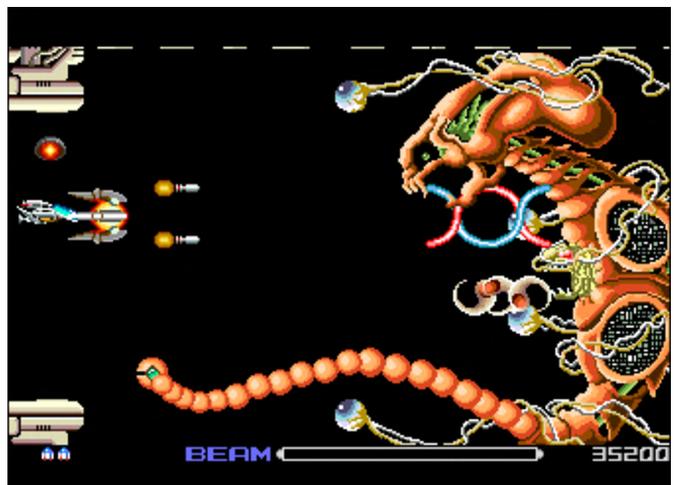
### R-Type Complete CD

Hace poco jugué a este juego de la C64 en el ODR0ID Go, que es algo más difícil que esta versión, pero ésta tampoco es nada fácil. Esta versión tiene gráficos mejorados, viene con una introducción y secuencias de video cada par de misiones. La banda sonora se ha mejorado y ahora es audio CD, por lo que es de mayor calidad, aunque aún es posible reconocer las melodías originales. En general, disfruto bastante con

este juego, aunque también existe una versión de HuCard de este juego. En realidad, hay varias, en Japón existe R-Type I y R-Type II, que es el mismo juego, pero dividido en dos fichas de juego, no una secuela. Norteamérica contaba con una versión HuCard (TurboChip) que tenía ambos juegos combinados en una tarjeta, y cuando salió el CD, en Japón finalmente se publicó también el juego completo, incluido un combate de jefe adicional en el nivel 6. En conjunto, el juego es muy bueno y vale la pena jugando.



**Figura 3: R-Type de hecho se juega en pantalla panorámica en la PC-Engine**



**Figura 4: Los combates de los jefes siempre siguen un determinado patrón**

Otra particularidad de este juego son los gráficos a pantalla panorámica. La mayoría de los juegos se ejecutan a 256x243, pero a R-Type se juega a 352x243. Este formato solo fue incluido en unos cuantos juegos.

### Rainbow Island



Figura 5 - Bubble Bobble 2 - Rainbow Island



Figura 6: Rara vez he visto jefes tan "graciosos", ¡aunque realmente detesto las arañas!

Rainbow Island es otro de los clásicos de Amiga que jugué de niño. En este gracioso juego, literalmente escalas el arcoiris. De hecho, puedes disparar arcoíris y caminar sobre ellos para subir a plataformas más elevadas o matar enemigos. Al final de cada isla, tienes que luchar contra un jefe. Todo el juego es muy amigable y está diseñado para niños, tiene gráficos al estilo comic muy coloridos. La música ahora se reproduce con calidad CD, pero sigue siendo la original.

### Renny Blaster

El hecho de que finalizar el juego de un tirón ya debería ser motivo suficiente como para ponerlo en esta lista, y de hecho esa es la principal razón por la que está aquí, ya que el juego no es nada "espectacular". Los gráficos son bastante mediocres en el mejor de los casos, sin desplazamiento de paralaje, y las secuencias de vídeo son solo imágenes

estáticas con excesivas interpretaciones de voz, las cuales terminé omitiendo por completo. Este es un juego de lucha similar a Double Dragon o Streets of Rage. Puedes elegir entre dos jugadores, ambos tienen estilos de lucha completamente diferentes. Uno es campeón de artes marciales y lucha con puños y pies, el otro es una especie de místico que lucha con poderes mágicos. Personalmente, descubrí que usar los pies y las manos para luchar es mucho más efectivo que disparar pequeños rayos de energía, especialmente porque estos últimos no se pueden lanzar en el aire, lo cual hace que los ataques en salto sean imposibles de realizar.



Figura 7: EL combate contra los enemigos normales pueden ser muy fácil o similares a los combates de los jefes



Figura 8: Cada nivel tiene varios jefes con lo que tienes que luchar.

También tienes un ataque que puede cargarse hasta en tres pasos, y causa tres ataques diferentes, todos ellos bastante débiles, pero a veces útiles en

determinadas situaciones. A lo largo de los niveles, encontrarás pergaminos de texto, los cuales activan nuevas habilidades que luego puedes elegir antes de empezar un nuevo nivel. Aunque los gráficos y el sonido/música no son nada del otro mundo, el juego es bastante amigable y puedes finalizarlo en tan sólo media hora.

### Riot Zone



Figura 9: Golpear a los malos nunca fue tan divertido



Figura 10 - Al final de cada nivel, te espera un jefe con el que tienes que luchar

Se trata de otro juego como Double Dragon o Streets of Rage y es muy bueno. Los gráficos son geniales, con algo de desplazamiento de paralaje en los trasfondos. Los controles son bastante buenos, y realizar ataques de salto es fácil. Puedes seleccionar entre dos luchadores diferentes y luego simplemente darles a tus enemigos la paliza que se merecen. Me

divertí mucho jugando a este juego y lo recomiendo encarecidamente

### Seirei Senshi Spriggan

Este es uno de los juegos de los que me costó mucho alejarme, ya que me mantuvo ocupado y despertó mi interés durante bastante tiempo. Es uno de los muchos shooters que existen para el sistema y una vez que lo dominas, es bastante bueno.



Figura 11: Este juego puede arrojar toneladas de enemigos al mismo tiempo



Figura 12: Normalmente luchas contra dos jefes por nivel

Los gráficos son bastante buenos, aunque no vi demasiado desplazamiento de paralaje, sueles cambiar constantemente de escenario. También tienes lava vibrante o cascadas de fondo, lo que compensa la falta de desplazamiento de paralaje. No puedes mejorar tu ataque, pero puedes recolectar

burbujas de energía de diferentes colores, como verde, azul, amarillo y rojo, que puede combinar hasta llegar a tres para obtener diferentes resultados. A veces simplemente tienes tres tipos de armas disparando al mismo tiempo, y a veces creas ataques completamente nuevos. Algunos ataques son dirigidos y localizan sus objetivos por sí solos, mientras que otros se expanden por toda la pantalla. Es muy divertido probar las diferentes combinaciones.

### Shape Shifter



Figura 13: El calabozo suele estar repleto de enemigos



Figura 14: Los jefes son grandes y requieren algunas tácticas para deshacerse de ellos

Este juego de plataformas de acción está completamente en inglés. Incluso las interpretaciones de las voces están en inglés, y juegas con pistas sonoras, que este juego tiene casi al 100. Incluso los

dueños de las tiendas te hablan, que se reproduce desde el CD. El juego en sí es bastante divertido, aunque algunos monstruos pueden ser muy molestos, especialmente los monstruos que vuelan sobre ti y no puedes golpearlos, pero si puedes pasar por alto esto, el juego es bastante divertido.

Puedes comprar un par de armas diferentes, e incluso mejorarlas con un hechizo mágico. Más adelante, cuando luches contra algunos jefes, liberarás a 5 magos, que te concederán el poder de transformarte en otras cosas, como una pantera, por ejemplo, lo cual te proporciona mayor velocidad y fuerza por un tiempo limitado. En la parte superior central de la pantalla, puedes ver un sol y un planeta que lo rodea. Cada vez que el planeta se ponga delante del sol, un rayo aparecerá del cielo y te golpeará, que como contrapartida repondrá toda tu salud. De modo que, a veces es mejor sentarse y esperar a que suceda. En general, el juego es muy divertido, aunque necesitas tiempo para acostumbrarte a la lucha, ya que algunos enemigos siguen apareciendo a menos que destruyas ciertos objetos, y algunos pueden llegar a ser difíciles de eliminar o alcanzar.

### Spriggan Mark II - Re-Terraform

Aunque está completamente japonés no pudiendo entender ni una sola palabra del juego, cuenta con una introducción de película y tiene varias secuencias de video entre los niveles.



Figura 15: Muchas configuraciones diferentes y un montón de desplazamiento de paralaje en segundo plano



Figura 16: Como de costumbre, al final del nivel tienes que luchar contra un jefe

Los gráficos en general son bastante buenos, y todos los niveles tienen algo de desplazamiento de paralaje en segundo plano. Se te lanzan muchos objetos y enemigos diferentes todo el tiempo. Con el primer botón atacas, y con el segundo giras tu mecha para que puedas disparar en la otra dirección. Con SELECT, puedes recorrer diferentes armas. No existen potenciadores o cosas por el estilo, pero a medida que vas avanzando en el juego, obtienes nuevas mechas, y entonces puedes elegir diferente armamento. Hablando precisamente de las armas, una es, por ejemplo, una espada que puede ser realmente muy útil en el combate cuerpo a cuerpo, y si mantienes presionado el botón, puedes usarla para volar hacia los enemigos y provocarles un daño constante. Realmente me gusta este juego como uno de los muchos y buenos shooters que existen para la consola.

### Star Parodia

Este es uno de esos adorables juegos que enganchan, realmente es muy bueno, con un montón de gráficos coloridos, buena música y efectos de sonido. Hay tres personajes diferentes para elegir, uno es Bomberman y otro es PC Engine. Este juego lo tiene todo. Potencias tu arma recogiendo los elementos que representan el arma que tienes en ese momento. Recoge un arma diferente para cambiar a otro estilo de ataque. Tiene bombas para los grandes ataques y potenciadores como escudos y bombas adicionales. El juego viene con un buen desplazamiento de

paralaje en algunos niveles, muchos monstruos y combates contra jefes.

Se desplaza rápido y nunca se ralentiza. Es muy bien para las personas que no son tan buenas con lo shooters, ya que el nivel de dificultad no es muy elevado. Recibes un golpe y no mueres instantáneamente, sino que pierdes un nivel de arma. Solo cuando estás en el mínimo y eres golpeado de nuevo pierdes una vida. Consigues un montón de armas muchas luchas desde los puntos de recolección o recogiendo potenciadores. En general, este juego es muy divertido y lo recomiendo.



Figura 17 - Buenos gráficos y un sistema de juego fluido en este agradable juego



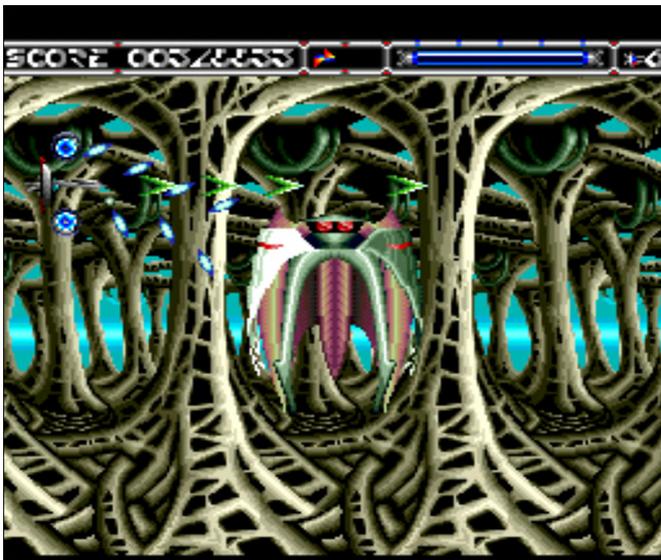
Figura 18 - Jefes extraños como montañas rusas y papeles origami

### Syd Mead's Terra Forming

Este shoot 'em up es bastante bueno, con un gran desplazamiento de paralaje tanto en primer plano como en segundo plano. Es muy agradable a la vista. La acción es rápida pero manejable. Los jefes requieren algunas tácticas, ya que a menudo solo pueden ser atacados desde ciertos ángulos.



**Figura 19: Hay muchos desplazamientos de paralaje y diseño dinámico en este juego**



**Figura 20: Los jefes a menudo tienen ciertos puntos difíciles que necesitas encontrar para destruirlos**

Este es uno de esos juegos que eliges simplemente para probarlo y terminas jugando durante horas. Tienes un arma principal que se puede mejorar recogiendo rectángulos azules con una línea en el medio. También puede tener una de las tres esferas de apoyo que tienen diferentes ataques. Éstas se activan al recoger burbujas amarillas, rojas o azules. El amarillo es un ataque que se expande y se disparará en línea recta en todas direcciones y que te

permite atacar a los enemigos de delante y atrás al mismo tiempo. El rojo es una potente arma que solo va directa al blanco. El azul es mi favorita y es una serie de balas teledirigidas. Cada bala va automáticamente a cada enemigo de la pantalla. El primer nivel se puede superar con tan solo situarse en el medio de la pantalla y mantener presionado el botón de disparo mientras las balas apuntan y destruyen todo lo que te arrojan. Todas las armas se pueden actualizar varias veces para hacerse más fuertes. Si te golpean, pierdes un nivel de potencia de tu arma y así sucesivamente has que mueres.

## **Juegos que están bien**

### **Prince of Persia**

Esta versión de Prince of Persia es una versión aceptable con gráficos mejorados, una intro ligeramente actualizada y una banda sonora de calidad CD. Los controles están bien en su mayor parte, y puede usar un botón de salto específico o el botón arriba de tu mando. Este último puede causar algunos problemas si simplemente intentas correr en línea recta y de repente tu personaje salta, así que asegúrate de tener un gamepad con un buen d-pad. En general, los controles podrían ser mejores, especialmente en las peleas, donde los controles a veces son un poco lentos, deberían reaccionar más rápido. Aun así, el juego es genial con una buena historia y el sistema de juego original, definitivamente vale la pena probarlo.

### **Psychic Storm**

Este shoot 'em up es algo extraño. Es divertido para un rato, con 4 personajes y naves diferentes para elegir, y diferentes armas y transformaciones, las cuales son muy diferentes entre sí. Aun así, el juego es bastante mediocre en el mejor de los sonidos, gráficos y jugabilidad. Luchas contra naves extraterrestres y monstruos de gran variedad, y al final de cada etapa hay un jefe que necesitas eliminar.

Tu nave puede transformarse en una nave espacial gigante similar a un insecto con bastante potencia de fuego durante un tiempo, y puedes recolectar energías para la nave que es lanzada por los diferentes enemigos, pero no se dejan caer lo suficiente para mantener esta forma por mucho

tiempo. Si tienes la forma normal y te golpean con demasiada frecuencia (salud casi vacía), te transformas automáticamente a la forma más potente, para evitar la destrucción. Lo cual está bien, si vuelves a cero de salud en el momento en que finaliza esta transformación, el siguiente golpe te transforma de nuevo. Después de tres transformaciones (ya sea voluntariamente o por estar cerca de la muerte), se acabó y ya no puedes usarlas más, lo cual también significa la muerte instantánea tras finalizar la transformación si eres golpeado una vez más. Puedes mejorar tu arma principal cuando recoges potenciadores azules y aumentar algo tu salud recogiendo píldoras rojas/azules. Ambas son raras y permiten aumentar la potencia de tu arma. Tienes que recolectar alrededor de 5 armas en total, lo que hace que tus armas sean bastante pobres.

### **Puyo Puyo CD / Puyo Puyo CD Tsuu**

Ambos no son malos juegos, son agradables y coloridos, divertidos y en general tienen una buena presentación. Simplemente no veo por qué están en CD: uno tiene casi 40 pistas de audio, y el otro tiene más de 80. Aparentemente, estas pistas no son para música, sino que solo contienen los diálogos que se escuchan cada cierto tiempo. Honestamente, no veo que esto merezca la pena tener formato CD. Más bien, estos juegos deberían haberse almacenado mejor en una HuCard. La música ni siquiera parece ser de CD, la calidad no es tan buena. Ambos juegos son buenos y si eres fanático de la serie Puyo Puyo, deberías probarlos.

### **Pomping World**

Básicamente se trata de una versión del clásico arcade Pang, donde disparas burbujas que se dividen en dos burbujas más pequeñas y así sucesivamente hasta que las burbujas más pequeñas finalmente desaparecen. Eliminas todas las burbujas y ganarás; si eres golpeado y mueres, tienes que empezar de nuevo. No es un mal juego y es casi perfecto como arcade.

### **Populous - The Promised Lands**

Primero jugué a este juego de estrategia en el Amiga, y sorprendentemente los gráficos para PC Engine son bastante buenos, y tal vez incluso superiores a los

gráficos originales de Amiga, pero probablemente se vea diferente debido a los gráficos de estilo cómic. Aún así, el juego es el mismo que en el Amiga, aunque los controles son mucho más difíciles debido al hecho de que el juego se juega mejor con un ratón, mientras que con un mando se vuelve lento y engorroso. Funciona bien, incluso si es un poco lento. El juego tiene una mezcla de inglés y japonés.

### **Ranma Ni Bun no Ichi**

El primero de un trío de juegos de Ranma que probé. Sigue la historia de los primeros episodios del juego, un juego de plataformas de acción con peleas y saltos. El juego tiene muy buenos gráficos y las secuencias de video son realmente impresionantes, con muchos detalles y animaciones, así como las interpretaciones de las voces (creo que utiliza el elenco original). Aunque el juego está completamente en japonés, realmente no es un problema. Lo que sí es un problema son los malos controles que están tristemente presentes en todo el juego. En algunas escenas, tus golpes no se completan y la acción simplemente se detiene a mitad del golpe o la patada. Lo mismo ocurre con el salto, lo cual hace que el juego sea innecesariamente más difícil de lo que debería ser.

### **Ranma Ni Bun no Ichi - Datou Ganso Musabetsu Kakutou Ryou!**

Este es el segundo juego de la serie de juegos de Ranma que probé. Los gráficos, las escenas musicales y todo sigue siendo tan bueno como en el primer juego, pero la jugabilidad ha cambiado bastante. Ahora, es más bien un combate cuerpo a cuerpo, y luchas contra multitud de enemigos. Esto realmente funcionaba mucho mejor que el primer juego y resultaba bastante más divertido vencer a los primeros enemigos. Pero poco después, descubrí que los enemigos se vuelven mucho más fuertes y, de hecho, son bastante injustos.

Cuando te golpean, parpadeas y te vuelves invencible durante un breve periodo de tiempo. Eso es todo, tal y como sucede en muchos otros juegos, aunque en este juego en concreto el tiempo que eres invencible es tan corto que te golpean al instante con otro ataque nuevamente. El enemigo también te ataca la

mayoría de las veces a distancia, que a menudo son ataques que se despliegan en abanico, por lo que es muy difícil de evitarlos, al mismo tiempo en la mayoría de las ocasiones tienes que acercarte mucho para golpear al enemigo. En general, se vuelve injusto y difícil muy rápido, lo cual hace que no lo disfrutes demasiado, pero aparte de esto, el juego es fantástico.

### **Ranma Ni Bun no Ichi - Toraware no Hanayome**

Este es el último juego de esta serie de juegos de Ranma que probé. Este cambia bastante. La música es mala en comparación con los otros dos. No es que sea mala en sí, es que no encaja o suena terrible, a pesar de que los dos primeros venían con audio CD, este parece contener solo chiptunes. El juego apenas tienes "acción" y, de hecho, más bien es una Novela Visual en la que simplemente tienes que elegir lo que quieres hacer a continuación. Incluso si no entiendes el idioma (como yo), igualmente puedes jugar. Simplemente eliges todas las opciones disponibles varias veces, y con el tiempo continuarás avanzando. Esto también hace que el juego sea muy largo, especialmente si no entiendes lo que se dice. Hay muchas interpretaciones de voces durante todo el juego, con una buena cantidad de animación y diseño de personajes, pero con casi ningún fondo. Este fue el juego que menos me gustó de los tres.

### **Rayxanber II + III**

Estos dos shooters son muy genéricos. Los gráficos están bien, pero no son muy buenos, sin, por ejemplo, desplazamiento de paralaje. Sin el interruptor turbo, este juego es demasiado difícil de jugar, ya que necesitas el disparo rápido y lo tienes que manejar durante mucho tiempo aplastando básicamente el botón del mando.

No cuenta con ningún sistema de potenciadores, y solo puedes recoger un par de armas diferentes. Dependiendo de la dirección del icono del arma cuando la recojas, dispararás en una u otra dirección, la de frente resultó ser la mejor, ya que necesitas potencia de fuego adicional puesto que tu nave es bastante débil. Como he dicho antes, no puedes potenciar el arma, de modo que recoger varias veces

el mismo icono no hace nada, excepto el hecho de permitirte cambiar la dirección del disparo.

### **Road Spirits**

Road Spirits es un juego de carreras de la vieja escuela similar a la serie Lotus en el Amiga o OutRun de las salas recreativas. Siempre conduces en línea recta y solo puede girar a izquierda y derecha para seguir la carretera y evitar los otros automóviles. En este juego, tienes dos movimientos: Bajo (botón arriba) y Alto (botón abajo); el primero va a 170 km/h, y en el otro va a 290 km/h. El juego tiene un poco de desplazamiento de paralaje en segundo plano, pero en general los gráficos no son muy buenos. El juego también es bastante fácil, a diferencia de OutRun, por ejemplo. Tampoco tienes muchas opciones a dónde ir y qué hacer. Solo se puede elegir la música antes de empezar. Hay 10 bandas sonoras en total para elegir y eso es todo. No está mal como juego, pero tampoco es nada del otro mundo.

### **Ryuuko no Ken**

Este juego de lucha con muy buena pinta tiene muchas características que ofrecer. No solo es uno de los pocos que admite el gamepad de 6 botones existente para el sistema, sino que también ofrece características como el zoom en los combates, lo que significa que cuando estás cerca del enemigo, la vista se acerca, y si estás más lejos se aleja. Estas son características geniales que hacen que el juego sea impresionante. Este juego de estilo Street Fighter es probablemente uno de los mejores que la consola puede ofrecer. Me gustó, pero lo encontré un poco difícil, aunque ello podría deberse simplemente a mi falta de habilidades con este tipo de juegos.

### **Shadow of the Beast**

Este es otro juego clásico de Amiga. Ganó algunos premios por sus gráficos y música en su tiempo, escuchar la música en calidad CD en PC Engine es ciertamente algo asombroso. La música se remezclaba y en algunos casos es bastante diferente a la original, pero probablemente sigue siendo la mejor de todas las versiones de consola. Los gráficos son bastante buenos siempre que te encuentres fuera de una mazmorra, pero dentro de una mazmorra se echan de menos los fondos y no hay

desplazamiento de paralaje, lo que lo convierte en el peor de todas las versiones en términos de gráficos (al menos dentro de una mazmorra), pero la impresionante música contrarestró este hecho, y el juego no tiene tan mal aspecto. No es como la versión SNES donde los gráficos se ven muy bien en general, aunque la música es simplemente terrible e incluso peor que la versión Genesis. Tienes disponible un video comparativo del juego original de Amiga comparándolo con las versiones de SNES, Genesis y PC Engine en <https://www.youtube.com/watch?v=QUT91K4mPlw>.

### **Shanghai II**

Por lo general no soy fanático de este tipo de juegos de rompecabezas, pero este en concreto tiene una música muy agradable que lo hace muy relajante, disfruté jugándolo durante un tiempo. No es nada del otro mundo, sin videos ni escenas, solo un simple juego, que es agradable y que lo disfruté durante el tiempo que lo jugué.

### **Slime World**

Este juego de plataformas realmente es bastante interesante. Juegas con tipo que lleva un traje que camina y salta a través de niveles de fango verde, con monstruos arrojándote fango y estanques de agua donde puedes limpiarte. Si tienes demasiado fango cubriéndote, mueres. Puede recopilar un montón de elementos diferentes que no logré averiguar cómo usarlos o si podía realmente usarlos. Los controles están un poco fuera de sitio. Saltas muy lento y te pegas a las paredes, sin importar si quieres o no, lo que a veces dificulta el acceso a determinados lugares. Gráficamente, el juego es agradable, aunque es un poco repelente, no por el fango, sino porque la pantalla sigue cada movimiento, lo que significa que si giras, subes o bajas, la pantalla sigue desplazándose. Giras a la izquierda o derecha y la pantalla se desplazará instantáneamente en la otra dirección.

### **Space Fantasy Zone**

Este juego es un Space Harrier que se encuentra con el juego Fantasy Zone (de ahí el nombre), y es exactamente lo que esto sugiere. Imagina Space Harrier con gráficos y enemigos de Fantasy Zone. En

realidad, es bastante divertido y tras finalizar un nivel, puedes ir a una tienda para curarte, comprar mejores armas y algunos extras. Desde el punto de vista gráfico no es un juego que impresione demasiado, pero no está mal y tiene algunas melodías agradables.

### **Space Invaders - Original Game**

Deberías tomar el "Juego original" muy en serio, ya que éste es un remake de las diferentes versiones de arcade de 1978, y esto es también lo que puedes esperar en cuanto a gráficos. El juego viene con un par de versiones diferentes e incluso tiene algunas versiones competitivas para varios jugadores, pero en general es el "buenazo" de Space Invaders, aunque no demasiado elaborado.

### **Splash Lake**

Este juego de rompecabezas es bastante singular. Juegas con un pájaro y tu objetivo es matar a todos los enemigos del mapa, no atacándolos, sino picando el suelo bajo sus pies, rompiendo cornisas y haciéndoles caer al agua. Para esto, debes conocer los diferentes tipos de baldosas que puedes romper y cuáles no, y cuáles y cuántas se desprenderán al elegirlos. El juego tiene diferentes fases, cada una con 10 niveles, después de la cual tienes una especie de combate contra un jefe, que es un enemigo que puede caer al agua más de una vez y algunos otros enemigos que aparecen más adelante. El juego es bastante divertido, aunque no impresiona demasiado. Los controles son un poco toscos, ya que una vuelta o una caminata marca la diferencia en una fracción de segundo.

### **Star Mobile**

Este juego de rompecabezas es bastante simple: tienes una balanza con tres lugares para colocar las "estrellas". Cada estrella representa un "peso" diferente. Las estrellas caen al azar, y tu objetivo es evitar que la balanza se incline hacia un lado u otro, y con ello acumulas las estrellas. Cada nivel aumenta el número de estrellas que necesitas apilar para ganar el nivel. Puede combinar estrellas del mismo color para obtener puntos adicionales o puede ganar estrellas adicionales si recolectas estrellas anteriores, esto se puede hacer cuando tiene estrellas de un

color diferente entre dos estrellas del mismo color, la estrella del medio desaparece y obtienes una estrella extra (que necesitas en caso de que cometas errores y sueltas estrellas). Es divertido por un tiempo, pero no tiene gráficos que impresionen ni elementos que destaquen dentro del juego.

### **Steam Heart's**

Este shooter es un poco extraño. Los gráficos son agradables y la música está bien. Puede recolectar potenciadores de armas y aumentar su potencial casi infinitamente (lo cual es muy interesante), pero da la impresión de que falta algo. El juego viene con dos personajes jugables, pero sus estilos de lucha son completamente idénticos. Parece ser que el juego solo tiene dos armas: un cañón vulkan que se expande y un rayo láser, siendo el láser el arma más fuerte con diferencia. Puedes conseguir un par de objetos adicionales, como misiles que atacan directamente o esferas que buscan enemigos y escudos que permiten protegerte durante un tiempo. Raramente consigues algo para reponer tu salud. En general es agradable, pero siento que le falta algo.

### **Super Air Zonk**

Este juego a menudo es conocido como una de las joyas ocultas del sistema, los gráficos y la música están bien, el estilo cómico es divertido y agradable, y aunque en general el juego es divertido, me decepciona un poco. Es un buen juego de disparos, pero para mí hay demasiadas cosas en la pantalla, y nunca estoy seguro de que si lo que me encuentro es algo que se supone que debo recoger, o si es un enemigo o una "bala". Aparecen demasiadas cosas y, a menudo, las bonificaciones que se supone que debes recoger están fuera de tu alcance, y me refiero a las bonificaciones de puntos normales, no a los poderes de las armas o lo que sea. Es molesto matar a 10 enemigos y no poder recoger ninguno de los puntos porque se alejan al instante de tu alcance. Aún así, el juego es agradable y divertido por un tiempo.

### **Super Darius I + II**

Ambos títulos son shooter conocidos de las salas de juegos, y su presentación en general es buena. El juego mejoró en el tema del sonido sobre los originales arcade, la banda sonora en calidad CD es

simplemente excelente. Los gráficos son agradables, con una buena cantidad de efectos y desplazamiento de paralaje en segundo plano. El juego no parece ralentizarse sin importar lo que esté sucediendo en la pantalla.

Sin embargo, No me gustan mucho estos juegos. El primero es extremadamente difícil, y me quedé sin vida en pocos segundos, hasta que me acostumbré. El segundo juego es más fácil, donde pude ajustar la dificultad y las vidas que tienes, lo que ayuda mucho, ya que cada vez que mueres, pierdes todos tus poderes, lo que hace que sean extremadamente difíciles los niveles posteriores ya que los enemigos necesitan más golpes y a los que les disparas es a guisantes.

### **Super Raiden**

Realmente me gustan los juegos de Raiden, y recientemente empecé a jugarlos en ODROID Go para Atari Lynx nuevamente. Por supuesto, esta versión es mejor en todo en comparación con la versión de Atari Lynx con gráficos, sonidos y música mejorados. El juego es agradable, y cuando obtienes un cañón vulkan completamente equipado con misiles guiados, el juego se vuelve simplemente un paseo relajante. Es divertido, pero le falta ese último efecto "increíble" para ser un juego perfecto. Aún así, lo recomiendo ya que es realmente divertido y puedes pasar un buen rato jugando.

### **Sylphia**

En este juego de disparos, juegas con un duende y tienes que disparar a muchos enemigos diferentes con muchas armas diferentes. Puedes recolectar diferentes armas de color, rojo, verde, azul y amarillo, que representan un tipo de arma diferente. Recoge el mismo tipo para aumentar el poder, o recoge un tipo diferente para cambiar de arma. Los combates de jefes pueden ser bastante difíciles y largos, y normalmente también hay un jefe intermedio. En general, el juego es divertido y agradable, tiene algunos efectos buenos, como un nivel bajo el agua, pero los gráficos no son tan impresionantes, por lo que no es uno de los mejores juegos de disparos disponibles para el sistema.

### **Juegos que no me han gustado**

### **Rom Rom Stadium**

Este juego de béisbol es uno de los pocos que existen para el sistema. De hecho, fue el único en el que logré golpear la pelota o anotar un punto. Por lo general, no soy fanático de los deportes, y los juegos de béisbol en las consolas nunca me han atraído demasiado. Éste tiene buenos gráficos de estilo comic, aunque también está completamente en japonés, por lo que hay muchas cosas con las que tengo dificultades. Me imagino que este juego puede ser divertido para las personas a las que les gusta el género, pero para mí no es muy bueno.

### **Shanghai III - Dragon's Eye**

A diferencia de Shanghai II, no me gustó el tercer título de la serie. La música y el sonido están demasiado fuertes, aunque no llegue a cambiar la configuración. Aunque el juego salió bastante tarde en el ciclo de vida de la consola (1992), los gráficos no impresionan demasiado. Tras finalizar el panel, aparece un gráfico de un dragón que dispara fuego y lo seguía haciendo sin importar qué botón presionaras o cuánto tiempo esperaras. Supongo que aquí es donde tenías que reiniciar la consola para volver a jugar.

### **Sherlock Holmes Consulting Detective Volume I + II**

Este juego tiene un inicio que impresiona bastante. Viene con videos, lo que significa que en realidad hay secuencias de video de Sherlock Holmes en el juego. El juego omite todas las bandas sonoras y tiene voces y música digitalizadas como datos en el disco. Hay un par de cosas que puedes ver antes de empezar el juego, tiene una introducción donde ves videos y escuchas voces, lo cual es impresionante, pero luego comienza el juego y la calidad cambia por completo.

Aun así, siguen apareciendo videos y cosas así cuando visitas lugares e intentas resolver el juego, pero no llegué muy lejos. Lo primero que te dicen que hagas es que revises el periódico, pero para ser honesto, no

he tenido la paciencia de leer 5 periódicos diferentes con 40 ~ 50 entradas de noticias, cada uno con un menú que siempre te devuelve al periódico que deseas leer antes de poder seleccionar el siguiente artículo. Este es un juego que lleva mucho tiempo, y no tuve el tiempo ni la paciencia para centrarme en él. Sin embargo, si te gustan los juegos largos, puedes echar un vistazo al análisis de <https://www.youtube.com/watch?v=6ShdmWQoLDY>.

### **Sim Earth - The Living Planet**

Este es también uno de esos juegos para jugadores pacientes. Gráficamente, he visto mejores versiones, además tiene retardos, aunque si tiene algunos buenos efectos de sonidos. En general, lleva demasiado tiempo llegar a algún lugar, yo no tengo tiempo para algo como esto. Los retardos y a veces, que el mando no responde como es debido hacen que este juego no vaya conmigo.

### **Strider Hiryou**

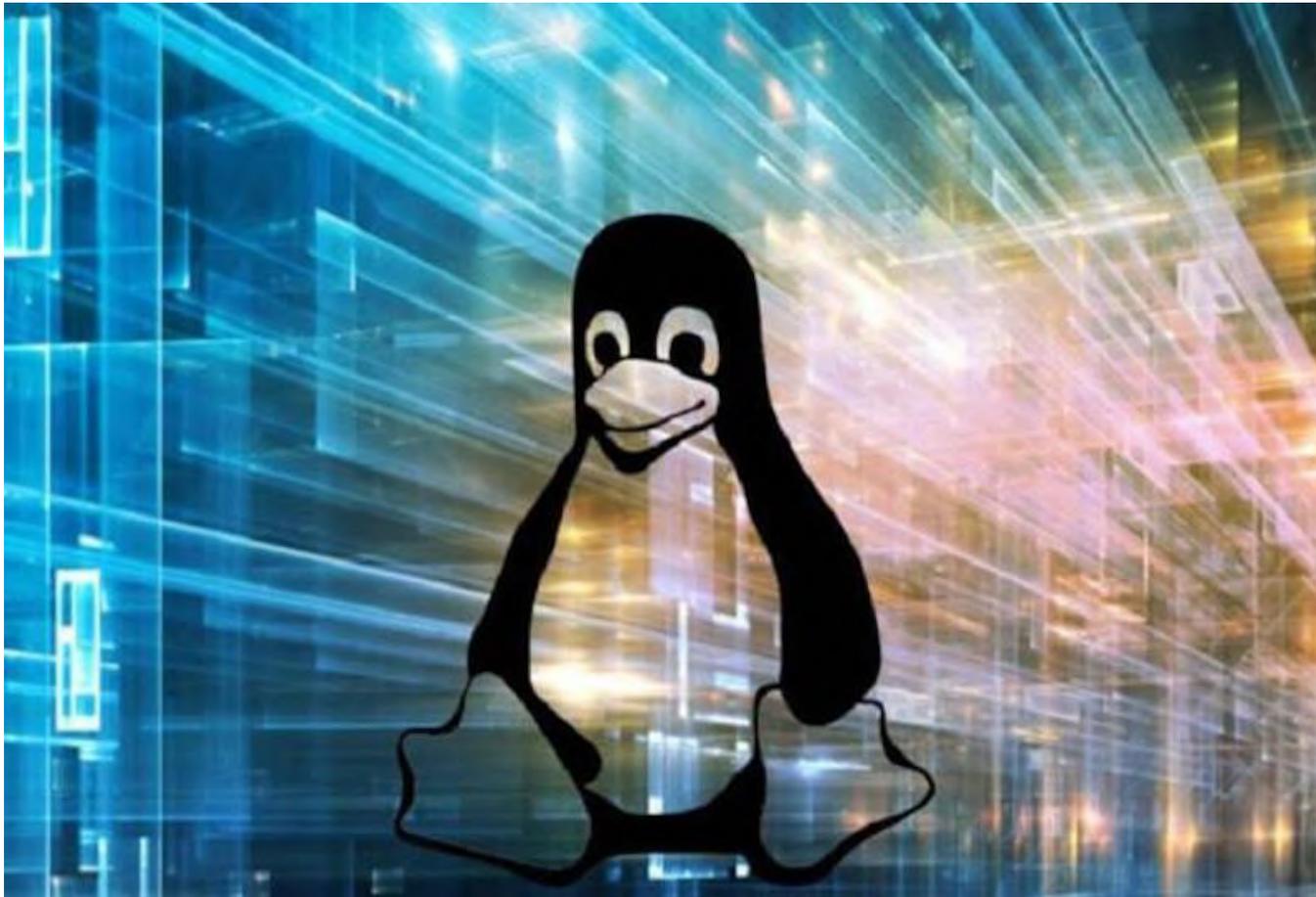
Esta es la versión para PC Engine de Strider, pero en mi opinión no es una versión muy buena. Tiene muchos gráficos que parpadean o no se ven, y los controles tampoco son los mejores, especialmente cuando intentas golpear a los enemigos que están en el suelo. No es un juego horrible, pero estoy seguro que podría haberse diseñado mejor

### **Summer Carnival '92 - Alzadick / Summer Carnival '93 - Nexzr Special**

Ambos juegos son shooter también, pero en el mejor de los casos mediocres. El segundo título tiene mejores gráficos en general, pero el sistema de juego no es muy divertido. Mueres demasiado rápido, o terminas el nivel y el juego finaliza y simplemente buscas conseguir la mejor puntuación. No disfruté mucho con estos dos juegos, aunque Summer Carnival '93 es mucho mejor en general.

# Usando el Kernel 5.2 con Armbian: Ejecuta WiringPi, HomeAssistant y Mucho Más con el Ultimo Soporte del Kernel

© August 6, 2019 By @joerg Linux, ODROID-C1+, Tutoriales



Este artículo se centra en la creación de una imagen arm utilizando el kernel principal Linux linux 5.2.0-rc3, y en la configuración del popular software de domótica, Home Assistant. En este ejercicio trataremos específicamente el popular ordenador de placa reducida (SBC) ODROID-C1.

## Pasos de instalación

Primero, instala una imagen armbian que funciona con un kernel 5.x. Estos son los pasos para compilar e instalar el kernel estandar:

```
$ export ARCH=arm
$ export CROSS_COMPILE=arm-linux-gnueabihf-
$ export
  PATH=/path/to/jour/toolchain/gcc-linaro-7.3.1-
2018.05-x86_64_arm-linux-gnueabihf/bin:$PATH
$ make odroidc1_defconfig
$ make -j 4 LOADADDR=0x00208000 uImage dtbs
modules
$ kver=`make kernelrelease`
```

```
$ sudo echo ${kver} > ../kernel.release

#put the sdcard to your host computer
#and copy the needed files to it

$ sudo mkdir /path/to/sdcard/boot/mainline
$ sudo cp arch/arm/boot/uImage
arch/arm/boot/dts/meson8b-odroidc1.dtb
  /path/to/sdcard/boot/mainline
$ sudo make modules_install ARCH=arm
INSTALL_MOD_PATH=/path/to/sdcard/boot/mainline
$ sudo cp .config
/path/to/sdcard/boot/mainline/config-${kver}
$ sudo cp ../kernel.release
/path/to/sdcard/boot/mainline

#this copies all relevant files
#then put the sdcard to the C1 and boot
#on the C1:
$ cd /boot/mainline
$ VERSION=$(cat kernel.release)
$ sudo update-initramfs -c -k ${VERSION}
```

```
$ sudo mkimage -A arm -O linux -T ramdisk -a 0x0 -
e 0x0 -n ../initrd.img-${VERSION}
-d ../initrd.img-${VERSION}
../uInitrd-${VERSION}
```

Tenga en cuenta que odroidc1\_defconfig no existe en el kernel estándar. En su lugar, tienes el multi\_v7\_defconfig. Sin embargo, cuando yo utilizo éste, el ODROID-C1 no arranca. Creo que se debe al tamaño del kernel de aproximadamente unos 10Mb. Odroidc1\_defconfig está basado en la configuración de @aplu, pero hay algunos valores de configuración que no se han ajustados. Puedes descargar odroidc1\_defconfig desde <https://bit.ly/2Y9JtQS>. Además, tenemos que editar el boot.ini para que el ODROID-C1 arranque el nuevo kernel:

```
# Booting
ext4load mmc 0:1 0x21000000 /boot/mainline/uImage
ext4load mmc 0:1 0x22000000 /boot/uInitrd-5.2.0-rc6
ext4load mmc 0:1 0x21800000
/boot/mainline/meson8b-odroidc1.dtb
#mainline kernel
#ext4load mmc 0:1 0x21800000 /boot/dtb/meson8b-odroidc1.dtb
```

Tras reiniciar debería verse algo similar a la siguiente imagen:

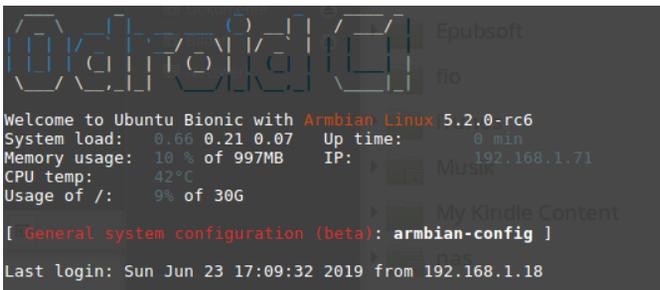


Figura 01

Tuve que modificar el kernel para lograr ejecutar wiringPi. En el kernel estándar, faltan algunos valores en /proc/cpuinfo:

```
...
Hardware : Amlogic Meson platform
Revision : 0000
Serial : 0000000000000000
```

Tras aplicar un parche, lo que creé se ve así:

```
...
Hardware : ODROIDC
Revision : 000a
Serial : 1b00000000000001
```

El parche cpuinfo se ve así:

```
diff --git a/arch/arm/boot/dts/meson8b-odroidc1.dts b/arch/arm/boot/dts/meson8b-odroidc1.dts
index f3ad939..8892151 100644
--- a/arch/arm/boot/dts/meson8b-odroidc1.dts
+++ b/arch/arm/boot/dts/meson8b-odroidc1.dts
@@ -52,6 +52,10 @@
     model = "Hardkernel ODROID-C1";
     compatible = "hardkernel,odroid-c1",
 "amlogic,meson8b";

+ hardware = "ODROIDC";
+ serial-number = "1b00000000000001";
+ revision = <0x000a>;
+
     aliases {
         serial0 = &uart_A0;
         mmc0 = &sd_card_slot;
diff --git a/arch/arm/kernel/setup.c b/arch/arm/kernel/setup.c
index d0a464e..b54a855 100644
--- a/arch/arm/kernel/setup.c
+++ b/arch/arm/kernel/setup.c
@@ -99,6 +99,9 @@ EXPORT_SYMBOL(system_serial);
 unsigned int system_serial_low;
 EXPORT_SYMBOL(system_serial_low);

+const char *system_hardware;
+EXPORT_SYMBOL(system_hardware);
+
 unsigned int system_serial_high;
 EXPORT_SYMBOL(system_serial_high);

@@ -959,6 +962,23 @@ static int __init
init_machine_late(void)
{
    system_serial_high,
    system_serial_low);

+ if (root) {
+     ret = of_property_read_u32(root,
"revision",
+         &system_rev);
+     if (ret)
+         system_rev = 0x0000;
+ }
+
+ }
```

```

+   if (root) {
+       ret = of_property_read_string(root,
"hardware",
+           &system_hardware);
+       if (ret)
+           system_hardware = NULL;
+   }
+
+   if (!system_hardware)
+       system_hardware = machine_name;
+
+   return 0;
+ }
+
+ late_initcall(init_machine_late);
@@ -1295,7 +1315,7 @@ static int c_show(struct
seq_file *m, void *v)
+   seq_printf(m, "CPU revision      : %d
", cpuid & 15);
+ }
-   seq_printf(m, "Hardware      : %s
", machine_name);
+ seq_printf(m, "Hardware      : %s
", system_hardware);
+   seq_printf(m, "Revision      : %04x
", system_rev);
+   seq_printf(m, "Serial        : %s
", system_serial);

```

No sé si es una buena idea poner el número de serie en el árbol del dispositivo. No obstante, se puede hacer así, con el comando fdtpu:

```

$ sudo fdtpu -t s /boot/mainline/meson8b-
odroidc1.dtb / serial-number 1b00000000000002

```

Tengo algunos dispositivos I2C que se ejecutan con wiringPi y Home Assistant. Sin embargo, cuando lo intenté, me sorprendió que I2C aún no estuviera habilitado. Me llevó un tiempo encontrar los valores correctos para el dispositivo. Aun así, había un problema sin resolver. Después de aplicar el siguiente parche al dispositivo, noté que /dev/i2c-0 atiende a los pines 3 y 5 y que /dev/i2c-1 atiende a los pines 27 y 28. Con el kernel 3.10.107 esto es diferente, la numeración del dispositivo del I2C empieza con 1, de modo que wiringPi busca el dispositivo i2c-1. Todavía no he encontrado una solución. Una posible solución sería modificar wiringPi solo para el ODROID-C1

usando el kernel estándar, para usar el dispositivo i2c-0.

La ruta i2c es así:

```

diff --git a/arch/arm/boot/dts/meson8b-
odroidc1.dts b/arch/arm/boot/dts/meson8b-
odroidc1.dts
index 8892151..c1d6e40 100644
--- a/arch/arm/boot/dts/meson8b-odroidc1.dts
+++ b/arch/arm/boot/dts/meson8b-odroidc1.dts
@@ -313,6 +313,25 @@
+   };
+   };
+
+&i2c_A {
+   status = "okay";
+   clock-frequency = <100000>;
+   pinctrl-0 = <&i2c_a_pins>;
+   pinctrl-names = "default";
+};
+
+&i2c_B {
+   status = "okay";
+   clock-frequency = <100000>;
+   pinctrl-0 = <&i2c_b0_pins>;
+   pinctrl-names = "default";
+   ds3231@68 {
+   compatible = "dallas,ds1307";
+   reg = <0x68>;
+   status = "okay";
+   };
+};
+
+&ir_receiver {
+   status = "okay";
+   pinctrl-0 = <&ir_recv_pins>;
diff --git a/arch/arm/boot/dts/meson8b.dtsi
b/arch/arm/boot/dts/meson8b.dtsi
index 800cd65..5831437 100644
--- a/arch/arm/boot/dts/meson8b.dtsi
+++ b/arch/arm/boot/dts/meson8b.dtsi
@@ -397,6 +397,14 @@
+   bias-disable;
+   };
+   };
+
+   i2c_b0_pins: i2c-b {
+   mux {
+   groups = "i2c_sda_b0", "i2c_sck_b0";
+   function = "i2c_b";
+   bias-disable;
+   };

```

```
+ };
};
};
```

Con todo este operativo, instalé la última versión de Home Assistant 0.94.3 en un entorno venv. También probé Docker, pero no logré hacerlo funcionar

En las próximas semanas, comprobaré si las lecturas y escrituras de I2C tienen errores. El tema es que en mi ODROID-C1 configurado con el kernel 3.10.107, el I2C se cuelga de vez en cuando y tengo que reiniciar (cada 3 ... 4 meses).

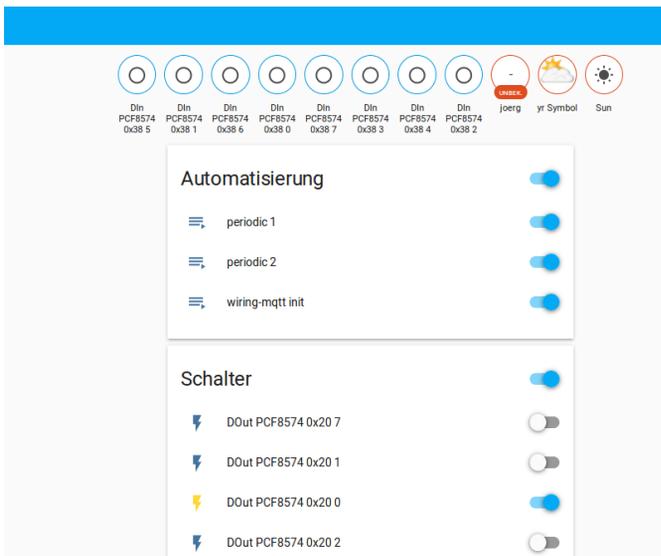


Figura 02

El miembro del foro @mad\_ady pudo instalar Hass.io a través de Docker en Armbian. Use esta información para conseguir que Home Assistant funcionase en Docker. Para tener mi propia ruta de datos para los archivos de configuración, tuve que utilizar este comando:

```
$ docker run --init -d --name="homeassistant" -v
/home/joerg/hassio:/config -v
/etc/localtime:/etc/localtime:ro --net=host
homeassistant/raspberrypi3-homeassistant
```

Luego observé lo siguiente con el comando:

```
$ docker inspect homeassistant

"Mounts": [
  {
    "Type": "bind",
    "Source":
"/home/joerg/hassio/share",
    "Destination": "/share",
```

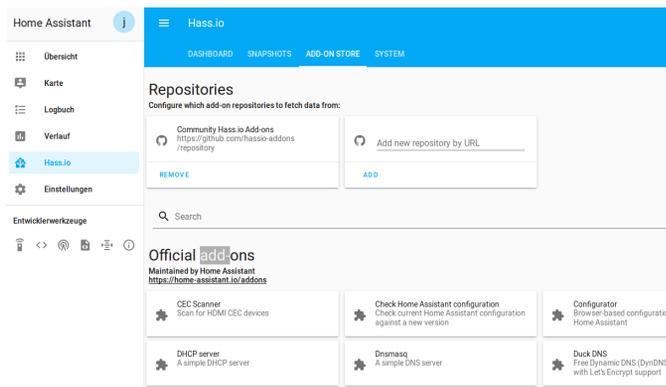
```
"Mode": "rw",
"RW": true,
"Propagation": "rprivate"
},
{
  "Type": "bind",
  "Source":
"/home/joerg/hassio/homeassistant",
  "Destination": "/config",
  "Mode": "rw",
  "RW": true,
  "Propagation": "rprivate"
},
{
  "Type": "bind",
  "Source": "/home/joerg/hassio/ssl",
  "Destination": "/ssl",
  "Mode": "ro",
  "RW": false,
  "Propagation": "rprivate"
}
],
```

Con esto, me di cuenta de por qué Home Assistant siempre daba error con el archivo de certificación mqtt. Con la versión venv, se incluye ruta completa al archivo, en Docker esto se monta en /ssl. Veo que tengo que aprender más sobre Docker.

Mi sección mqtt ahora tiene este aspecto:

```
mqtt:
# this settings for mosquitto:
broker: 192.168.1.71
port: 8883
client_id: home-assistant-test
certificate: /ssl/ca.crt
tls_insecure: true
```

Utilizo el broker mosquitto que generalmente está instalado, ya que no logré que el complemento Dockerized para mosquitto se ejecutara. Cuando uso mis archivos de certificación, aparece un error de protocolo. Así que decidí volver a la instalación normal de mosquitto. Ahora la diferencia es que hay una sección que ofrece la posibilidad de instalar complementos.



**Figura 03**

Ten en cuenta que no existe ningún apparmor, tal como se muestra a continuación. Tuve que aprender cómo habilitarlo y, en el proceso, también aprendí Docker.

```
$ sudo aa-status
apparmor module is not loaded
```

Hasta ahora utilicé el ODROID-C1 en una configuración sin monitor con el kernel estándar. Puedo decir que I2C funciona. Probé mi adaptador Bluetooth USB, pero no tuve suerte. Lo hice funcionar solo por un momento después de habilitar el soporte

de bluetooth con armbian-config, pero después de reiniciar, sin importar si el ciclo está encendido, no aparece un adaptador visible con lsusb. Mi instalación se ejecuta en una tarjeta SD. Además, aún no he probado IR y eMMC.

Docker se llegó a ejecutar con Home Assistant, pero ayer aparecieron muchos errores de la tarjeta SD. El LED parpadeaba, pero ya no podía acceder. Después de reiniciar, llevé a cabo una comprobación con fsck y me devolvió muchos errores, aunque cuando hice el chequeo con mi PC host no aparecieron errores. Una vez que arranqué con normalidad, tuve que desinstalar y reiniciar Home Assistant en Docker. De momento puedo decir que no se está ejecutando de forma estable.

Referencias

<https://forum.odroid.com/viewtopic.php?f=114&t=35474>

<https://forum.odroid.com/viewtopic.php?f=111&p=255093#p255083>

<https://forum.odroid.com/viewtopic.php?t=34570>

<https://www.home-assistant.io>

# Cómo Crear una Consola de Juegos Retro Monku - Parte 2: Configuración del Software

© August 1, 2019 By Brian Ree Juegos, ODROID-C1+, ODROID-C2



Esta es la continuación del artículo de la consola de juegos retro del mes pasado, donde aprendimos cómo fabricar una carcasa para una consola de juegos retro. Esta entrega te ayudará a configurar el software para el proyecto.

En este artículo, te mostraré con detalle cómo configurar el sistema operativo y las aplicaciones de la consola de videojuegos Monku Retro 1 (ODROID-C1+) o Monku Retro 2 (ODROID-C2). Son varios los pasos implicados, pero los describiré todos con detalle, incluidos los comandos que debes ejecutar. Si no tiene la intención de utilizar tu dispositivo como una consola de juegos retro, sino más bien como una simple máquina Linux, puede omitir las secciones de Retroarch y Antimicro e incluso la sección de optimización probablemente la podrás ignorar igualmente. De modo que, al final de este tutorial, tendrás una estupenda consola de juegos retro de tu propia creación. Bueno, estos son nuestros objetivos,

así que vayamos al grano. Reitero las características de este dispositivo a continuación, que son los mismos puntos que se incluyen en la sección de hardware. Ten en cuenta que si en cualquier momento se te solicita un inicio de sesión cuando trabajes con el terminal, usa la contraseña odroid.

## Herramientas necesarias

- Un ordenador con un lector de tarjetas SD o un puerto USB si tiene un adaptador de tarjeta SD a USB.
- Una tarjeta SD, recomendada de 64 GB o más.
- Software de escritura de imágenes en tarjeta SD para Mac. Yo uso balenaEtcher, es gratis y funciona muy bien.
- Software de escritura de imágenes en tarjeta SD para Windows. Yo uso Win32 Disk Imager, es gratis y funciona bien, aunque puede ser un poco quisquilloso con unidades muy grandes.
- Una conexión a internet. Si estás leyendo esto, es que ya tienes esta herramienta 😊

## Componentes necesarios

- 2 Tarjetas Micro SD de 64 GB: 16.99\$
- 1 Lector de tarjetas SD: 11.00\$
- 1 Gamepad Linux: \$17.00

Como incluimos las tarjetas SD en el coste del hardware, no las tendremos en cuenta aquí. Esto significa que el coste total de esta parte, suponiendo que tenga acceso a algún tipo de ordenador es de 0.00\$ a 28.00\$ y probablemente puede que encuentres un lector de tarjetas SD más barato, pero a mí me gusta este porque tiene soporte para tarjetas Micro SD y SD en diferentes ranuras que funcionan al mismo tiempo. Además, puede que ya tengas un mando para los juegos, aunque te indicaré el que yo uso y un enlace donde lo puedes encontrar a un buen precio. Suena bien, pasemos a la siguiente sección.

## Características C1

- ¡Calidad ODROID!
- Botón de control de software personalizado
- Reseteo del hardware a medida
- Soporte para Atari 2600, Atari 7800, ColecoVision, MSX-1, MSX-2, NES, GameBoy, GameBoy Color, Sega SG-1000, Sega Mark III y Sega Master System configuradas y listas para usar.
- Retroarch con XBM, scripts personalizados para monitorizar el botón de software, iniciar retroarch, mantener Antimicro
- Configurado para un consumo bajo de memoria y para usarse con el mando incluido.
- Todas las ROM testeadas para ver si se cargan y se asocian correctamente con su emulador
- Completo entorno de escritorio Linux cuando no está activo el modo kiosco del juego a través de antimicro

## Funciones del botón del software C1/R1:

- 2 segundos pulsado: se reinicia el software
- 4 segundos pulsado: se apaga el software
- 6 segundos pulsado: se desactiva el modo kiosco del juego
- 8 segundos pulsado: la resolución se cambia a 1024x768x32bpp y se reinicia
- 10 segundos retenidos: la resolución se cambia a 720px32bpp y se reinicia.

## Características del C2:

- ¡Calidad ODROID!
- Botón de control de software personalizado
- Reseteo de hardware personalizado
- Soporte para Atari 2600, Atari 7800, Atari Lynx, ColecoVision, MSX-1, MSX-2, NES, GameBoy, GameBoy Color, Virtual Boy, SNES, GameBoy Advance, WonderSwan Pocket/Color, NEO GEO Pocket/Color, Sega SG-1000, Sega Mark 3, Sega Master System, Sega Genesis, Sega GameGear, NEC Turbo Graphics 16 y NEC Super Graphics configurados y listos para usar.
- Retroarch con XBM, scripts personalizados para monitorizar el botón de software, iniciar retroarch, mantener Antimicro
- Configurado para un consumo bajo de memoria y para usarse con el mando incluido
- Control total del entorno de escritorio de Linux por gamepad cuando no está activado el modo kiosco de juego mediante Antimicro

## Funciones del botón de software C2/R2:

- 2 Segundos pulsado: se reinicia el software
- 4 segundos pulsado: se apaga el software
- 6 segundo pulsado: desactiva el modo kiosco del juego
- 8 segundos pulsado: configura el video como automático para el modo VGA, posiblemente modifique retroarch.cfg para el audio USB si está presente.
- 10 segundos pulsado: configura el modo video a 720p, cambia retroarch.cfg a audio HDMI

## Imagen Base 16.04 MATE y escritura en una tarjeta SD

Pasé una excesiva cantidad de tiempo probando las imágenes de los sistemas operativos disponibles para C1 + y C2 y descubrí que Ubuntu 16.04 LTS es la mejor imagen en términos de eficiencia, disponibilidad de paquetes, soporte, etc. Laka era genial, pero tuve muchos problemas con el audio. Ubuntu 18.04 LTS también era muy estable, pero usaba más recursos y no encontré tantos emuladores disponibles para esta versión, así que decidí usar la versión 16.04 LTS. Arranca tu Mac o Windows. Los usuarios de Linux pueden seguir las instrucciones de Mac. No voy a proporcionar ejemplos específicos para Linux. Abre tu

navegador y dirígete a la Wiki de ODROID <https://wiki.odroid.com/start>. Si tienes un C1+ o C2 ve a la sección correspondiente. Las Figuras 1 y 2 muestran los enlaces para cada dispositivo.



## ODROID Wiki

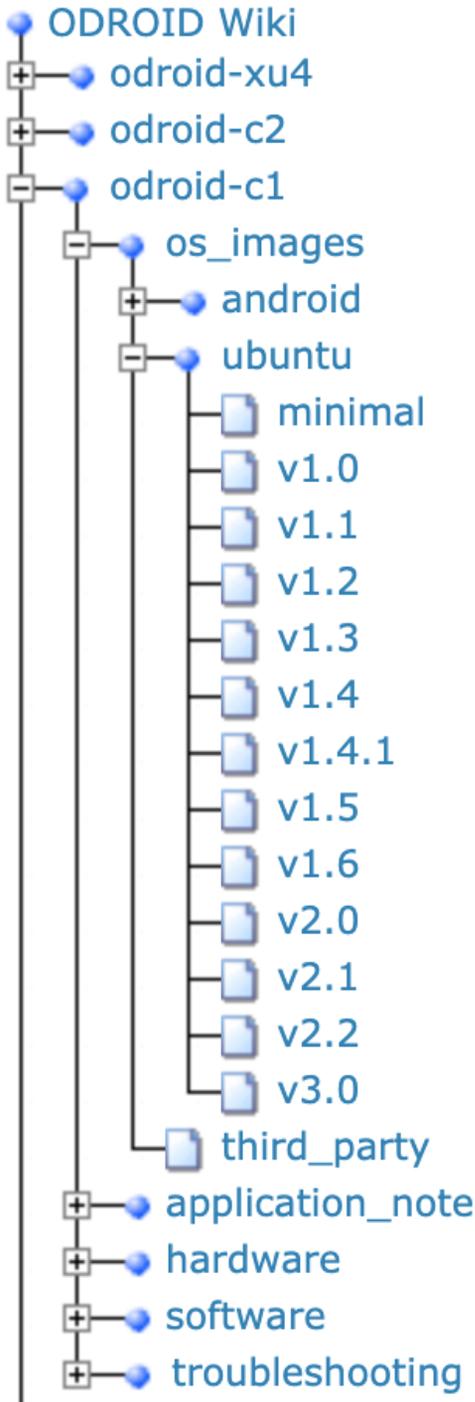


Figura 1 - Enlaces Wiki de ODROID para ODROID-C1

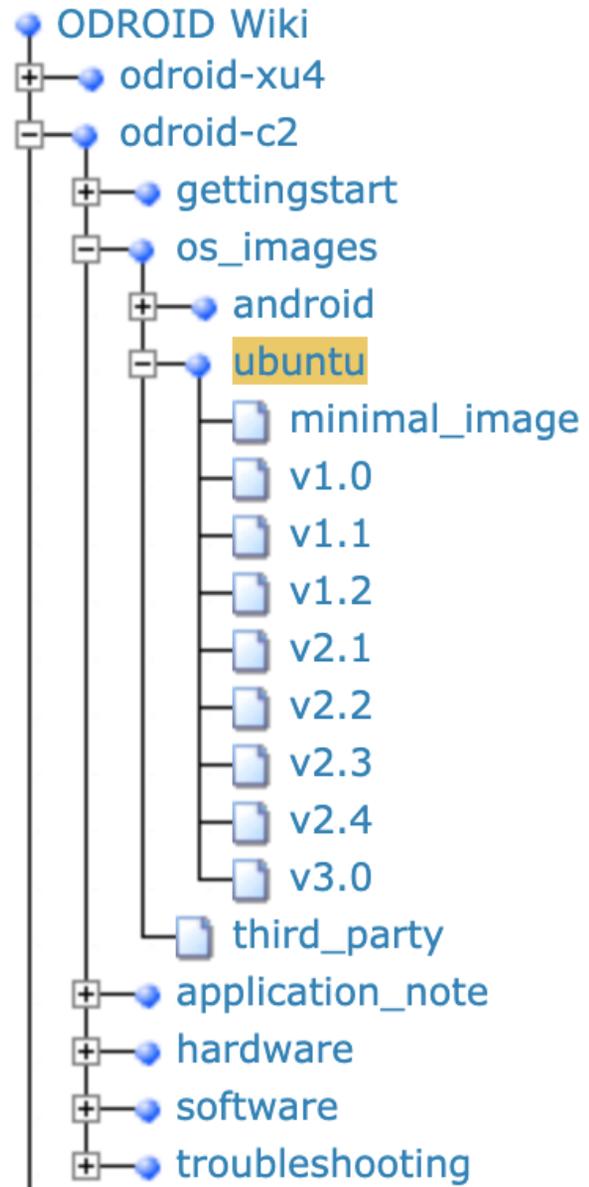


Figura 2 - Enlaces Wiki de ODROID para ODROID-C2

Selecciona tu dispositivo, luego selecciona os\_images, y finalmente selecciona ubuntu. Se cargará una página en el panel derecho. Desplázate hacia abajo hasta que veas la lista de servidores espejo. Me gusta usar el servidor de Corea del Sur, parece que funciona mejor. Aquí tienes los enlaces directos a las imágenes del sistema operativo para cada dispositivo.

Para ODROID-C1+: <https://dn.odroid.com/S805/Ubuntu/> Para ODROID-

C2: <https://dn.odroid.com/S905/Ubuntu/>

Proporcione algo más de información de la necesaria. Podría haber publicado el enlace de descarga directa, pero quería que vieras qué opciones hay. Para el C1+ utilicé <https://dn.odroid.com/S805/Ubuntu/ubuntu-16.04.2-mate-odroid-c1-20170220.img.xz> como imagen base. Si no puedes descomprimir un archivo .xz, intente usar 7-Zip. Es gratis y funciona bien en Windows. Yo no tuve ningún problema con mi Mac el cual maneja archivos .xz muy bien.

Para el C2, usé <https://dn.odroid.com/S905/Ubuntu/ubuntu64-16.04.2lts-mate-odroid-c2-20170301.img.xz> como imagen base. Recomiendo estar conectado a internet por cable si puedes, el WiFi será un poco más lento. Deja que las imágenes se descarguen. Puede verificarlas con la suma de verificación si quieres, el archivo que hay justo debajo de la imagen del sistema operativo, aunque esta cuestión no la voy a tratar aquí. Ahora en Windows, querrás descomprimir el archivo de imagen para tener una copia completa sin comprimir. Ambos archivos usarán aproximadamente 6 GB de espacio en el disco duro. Puede eliminar el archivo sin comprimir una vez que haya terminado con él para ahorrar espacio. En un Mac, recomiendo hacer lo mismo, si tienes problemas para descomprimir la imagen, te recomiendo usar The Unarchiver. Es gratis y funciona muy bien. Es la mejor opción para descomprimir cientos de ROM a la vez, créeme. También tendrás que navegar hasta la imagen sin comprimir en Mac.

¡Asegúrate de verificar doble y triplemente las letras de tus unidades! No querrás arruinar accidentalmente una de tus unidades o pen drive USB o lo que sea que hayas conectado a tu ordenador.

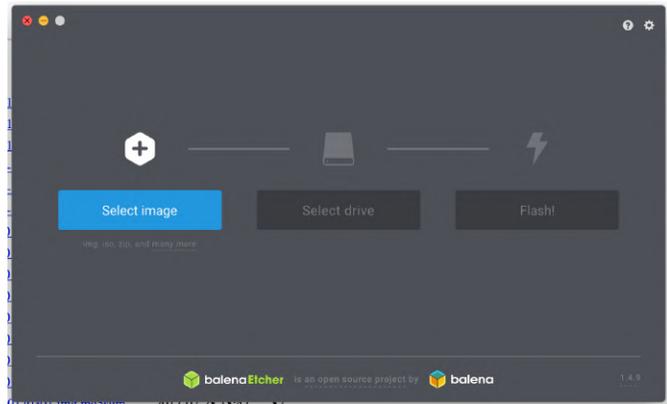


Figura 3

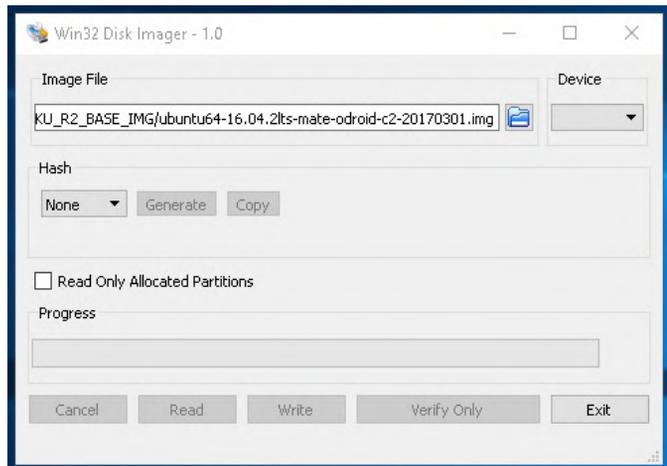


Figura 4

Deja que la aplicación para escribir imágenes haga su trabajo. Esto puede demorar unos minutos, vete a tomarte un café o una cerveza si son las 10AM. Ten en cuenta que las imágenes del sistema operativo no son intercambiables, debe usar la imagen base C1+ para tu dispositivo C1 + y la imagen base C2 para su dispositivo C2. Ignora las indicaciones para formatear la tarjeta SD desde OSX o Windows. Cuando hayas terminado, debería ver una unidad llamada "boot" donde estaba la tarjeta SD en blanco. Expulsala correctamente y colócala en el C1+ o C2, con el que estás trabajando.

Si estás trabajando con un ODROID-C1+ y planeas trabajar con una pantalla de ordenador, es posible que no puedas gestionar la salida HDMI por defecto. Es buena idea cambiar la salida de video antes de continuar. En tal caso, vuelva a colocar la tarjeta SD en tu ordenador y abre la partición de arranque. Busca el archivo boot.ini y haz una copia del mismo llamado boot.ini.orig.old. Edita el archivo boot.ini original y asegúrate de que el único modo de salida no comentado sea setenv m "1024x768p60hz", luego

desplázate hacia abajo un poco más hasta que vea la sección de señal de video y asegúrate de que únicamente setenv vout "vga" no esté comentado. Asegúrese de utilizar un editor de texto que sea adecuado para el trabajo, Pluma, nano, vi en Linux y OSX, también puede probar el editor de texto OSX nativo. En Windows Word Pad debería valer. También descubrí que Atom funciona bien en mi Mac. Si tu hardware está bien, deberías ver una luz azul cuando la tarjeta SD esté conectada y el dispositivo esté encendido. Debería ver un mensaje de inicio de sesión tras unos segundos, los C1+ son un poco más lentos. Si no fuera así, puede que tengas algún problema con la tarjeta SD o el hardware. He trabajado con al menos 6 de estas placas y las he soldado ... malamente ... y todas han arrancado sin problemas. El nombre de usuario y contraseña para tu dispositivo es odroid. Escríbelo en alguna parte para que no lo olvides.

Si no ves nada más que una pantalla en blanco, o si ves un mensaje de inicio de sesión, pero tras iniciar sesión aparece una pantalla en blanco, restablece el boot.ini original. Si está trabajando con un C1+, prepárese para trabajar frente a la TV. Comprueba que tu boot.ini ha sido restablecido al original y prueba si puede acceder al escritorio MATE. Ten en cuenta que este dispositivo no se está configurando como una maquina Linux sólida y segura. Aunque puedes hacerlo si quieres, nosotros no trataremos este tema en este artículo.

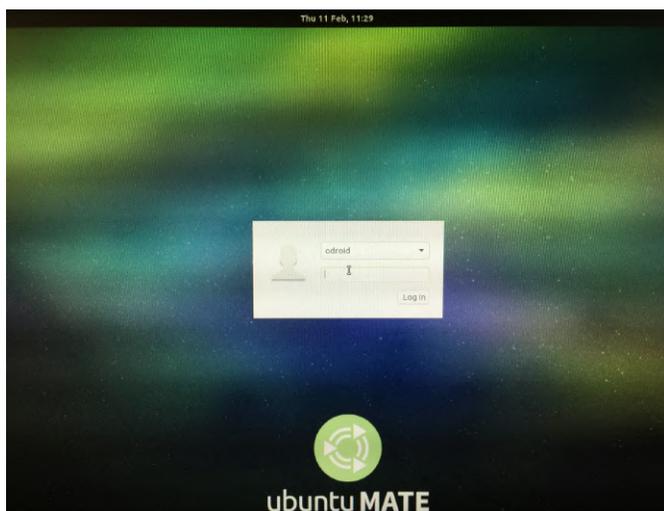


Figura 5

## Actualizaciones de software, tamaños de partición y mucho más

Primero, eliminaremos ese molesto mensaje de inicio de sesión. Si tu dispositivo tiene problemas con el USB al arrancar, simplemente reinícialo. Sin embargo, te diría que use los botones de control del software una vez que todo esté configurado y nunca uses el botón de encendido del hardware a menos que lo vuelva a encender. Para el resto del tutorial, espero que tenga un teclado y un ratón conectados al dispositivo. Si vas a la siguiente ubicación del menú Applications -> System Tools -> MATE Terminal, se abrirá una ventana de terminal con las que podrás trabajar.

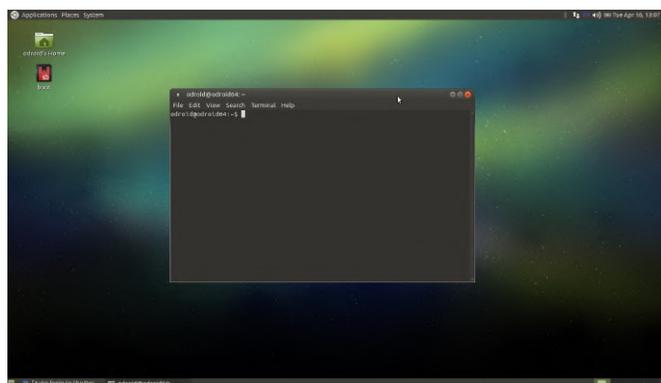


Figura 6

Ahora vamos a ejecutar una serie de comandos en el terminal. Algunos tardan un tiempo en ejecutarse, es posible que tengas que estar cerca de la pantalla por si alguna instalación te solicita algo. ¡Resulta pésimo tener que volver a un proceso que pensabas que terminaría solo y ves que aparece un cuadro de diálogo pidiéndote tu opinión! Lo que estamos haciendo exactamente es configurar el inicio de sesión automático, actualizar Ubuntu e instalar algunos paquetes.

Escribe este comando o cópialo y pégalo en la ventana del terminal.

```
$ sudo nano /usr/share/lightdm/lightdm.conf.d/60-lightdm-gtk-greeter.conf
```

Se te solicitará la contraseña, usa "odroid". Verás un texto como este en el archivo:

```
[Seat:*]  
greeter-session=lightdm-gtk-greeter
```

Añade una línea en la parte inferior, autologin-user=odroid, y luego guarda y cierra el archivo. Presiona Ctrl + O y luego presiona Intro para guardar

el archivo. Presiona Ctrl + X y luego Intro para salir del editor. En el próximo reinicio no se te solicitará que inicies sesión.

A continuación, ejecutaremos las actualizaciones de los paquetes del sistema operativo e instalaremos algunas cosas. Esta parte lleva un poco de tiempo, pero en su mayor parte se ejecuta sola. Introduce y ejecuta cada uno de estos comandos en el orden que se muestra a continuación desde la ventana del terminal.

```
$ sudo apt-get update -y
```

Si recibes un mensaje boot.ini, el comando simplemente es presionar Enter. Este comando lleva un poco de tiempo.

```
$ sudo apt-get upgrade -y
$ sudo apt-get install git -y
$ sudo apt-get install gparted -y
$ sudo apt-get install make -y
$ sudo apt-get install cmake -y
$ sudo apt-get autoremove -y
```

Ahora que todo esto ya se está ejecutando, el sistema empieza a mejorar un poco. Lo siguiente que vamos a hacer es desactivar cualquier espacio de intercambio que esté usando el sistema operativo. Se cree que las particiones de intercambio tienden a degradar la tarjeta SD con el tiempo, yo creo que están diseñadas para soportar un cierto número de operaciones de lectura/escritura antes de que empiecen a fallar. Hasta ahora las he estado deshabilitando sin observar un impacto notable en el rendimiento. Ejecuta el siguiente comando en la terminal.

```
$ swapon -s
```

Si ves un resultado similar al siguiente, es que tiene habilitado zram. Sigue los siguientes pasos para apagarlo y eliminarlo.

```
odroid@odroid64:~$ swapon -s
Filename Type Size Used Priority
/dev/zram0 partition 219824 2080 5
/dev/zram1 partition 219824 2076 5
/dev/zram2 partition 219824 2076 5
/dev/zram3 partition 219824 2076 5
```

Queremos desactivarlas para asegurarnos de que nuestra tarjeta SD dure tanto como sea posible. No he notado ningún aumento o disminución del rendimiento derivadas de esta acción, no quiero operaciones adicionales de lectura/escritura. Así que, ejecutemos el siguiente comando. Creo que el C1+ no tiene una partición de intercambio por defecto.

```
$ sudo apt-get remove --purge zram-config -y
```

Si observas una partición de intercambio estándar en la lista, introduce el siguiente comando para eliminar las particiones de intercambio estándar:

```
$ swapoff -a
```

Esto deshabilitará inmediatamente el intercambio en el sistema. Luego, elimina las entradas de intercambio de /etc/fstab editando el archivo como root y comentando las entradas de la partición de intercambio. Reinicia el sistema. Si, por algún motivo, la partición de intercambio todavía está ahí, abre gparted, System -> Administration -> GParted. Localiza la partición en la lista de particiones activas y desmóntala, después elimina la partición

A continuación, ejecutaremos una actualización del software MATE navegando hasta System -> Administration -> Software Update en el menú del sistema. Ten en cuenta que, si se te solicita que realices una actualización parcial, omite este paso y la ejecutaremos con los paquetes que hemos instalado. Me he dado cuenta de que en algunos casos los paquetes pueden volverse algo inestables. Yo solo procedería con una actualización si no apareciera el cuadro de diálogo Partial Update.

Si no se trata de una actualización parcial, haz clic en el botón Actualizar, espera un poco mientras se convierte lentamente en un cuadro de diálogo en forma de barra de progreso y realiza las actualizaciones. Tras finalizar el proceso, se te pedirá que reinicie el sistema. Pulsa en Restart.



Figura 7

Observa como ahora no aparece la ventana de inicio de sesión cuando inicias el sistema. Se parece cada vez más a una consola de juegos retro. Ok, lo último que vamos a hacer es cambiar el tamaño de la partición principal para usar todo el espacio disponible. Si vas a System -> Administration -> GParted haciendo uso de los menús del sistema, debería ver algo como lo que se muestra a continuación. Puedes ver que hay algo así como unos 50 GB de espacio sin utilizar. Si tiene poco o ningún espacio no utilizado, aunque todavía tengas espacio en la tarjeta SD, deberías cambiar el tamaño de tu partición root.

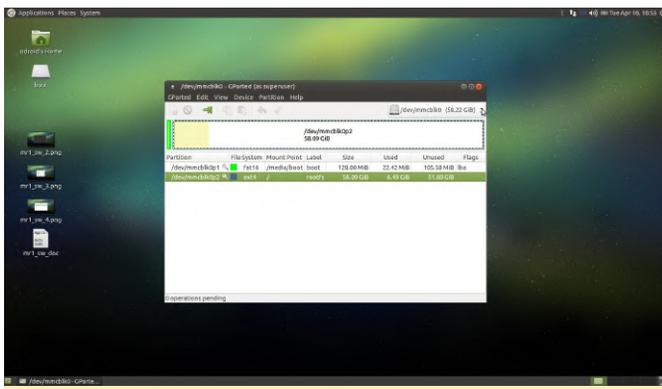


Figura 8

Te aré un rápido resumen de todo proceso. Ten en cuenta que es posible que no necesites hacer esto, pero deberías comprobarlo de todos modos. La mejor manera de trabajar en nuestro C1+ o C2 es con un sistema de archivos ext4. Usa la segunda tarjeta SD que viene con la compra recomendada que figura en el listado de hardware. Escribe la imagen del sistema operativo base en la SD como lo hemos hecho anteriormente e instala gparted. No tienes que ejecutar todas las actualizaciones, solo asegúrate de que gparted esté instalado. Usa el adaptador SD a USB que hemos mencionado anteriormente para

montar la tarjeta SD en la que queremos redimensionar las particiones del sistema operativo. Debería ver aparecer un pequeño icono de unidad, lo llamaremos SD-USB.

Arranca la tarjeta SD con Ubuntu básico si aún no lo ha hecho. Inicia gparted desde el menú System -> Administration -> GParted y selecciona la tarjeta SD-USB conectada, no el sistema de archivos root que ejecuta Ubuntu.

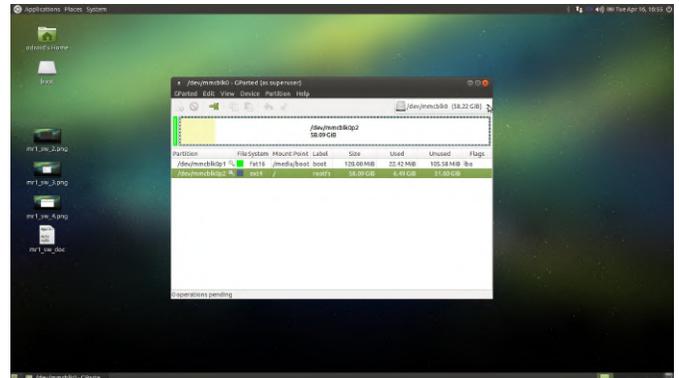


Figura 9

Selecciona el sistema de archivos root en la tarjeta SD-USB. Haz clic derecho sobre él y dirígete a la opción de cambio de tamaño. Si recibe un error, es posible que necesite desmontar esta partición y luego intentar cambiar el tamaño. Ahora puedes cambiar el tamaño de la partición arrastrando la flecha hacia la derecha o configurando los campos que hay en el formulario para que tenga 0 espacio libre. ¡Hecho! Hemos terminado con esta sección.

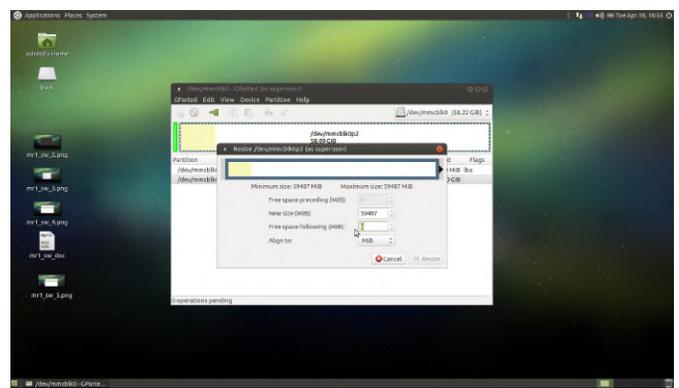


Figura 10

## Instalación de Retroarch y Antimicro

Ahora, instalaremos Retroarch y Antimicro para que podamos comenzar el proceso de configuración. Para instalar Retroarch necesitamos abrir una terminal, Applications -> System Tools -> MATE Terminal

```
$ sudo add-apt-repository ppa:libretro/stable &&
sudo apt-get update -y && sudo apt-get install
retroarch* libretro-* -y
```

Prueba el comando anterior en el C1+ y C2, presiona enter si se te solicita, y la tecla "Y" si se te solicita. Si falla no te preocupes. Noté que falla en el C2, pero no estoy seguro si también en el C1+, no te preocupes, los \* en las listas de paquetes son los culpables. Está tirando de algunos paquetes que tienen problemas de dependencia y eso está deteniendo todo el comando. Si falla intenta ejecutar este comando en su lugar.

```
$ sudo apt-get install retroarch retroarch-assets
retroarch-dbg libretro-beetle-lynx libretro-
genesisplusgx libretro-handly libretro-4do
libretro-bsnes-mercury-performance libretro-bsnes-
mercury-accuracy libretro-bsnes-performance
libretro-beetle-wswan libretro-dinothawr libretro-
beetle-ngp libretro-bsnes-balanced libretro-
gambatte libretro-fbalpha2012 libretro-fba
libretro-beetle-psx libretro-vba-next libretro-gw
libretro-mupen64plus libretro-beetle-sgx libretro-
2048 libretro-tyrquake libretro-beetle-pcfx
libretro-prosystem libretro-bsnes-accuracy
libretro-parallel-n64 libretro-picodrive libretro-
mame libretro-nestopia libretro-mednafen-psx
libretro-core-info libretro-gpsp libretro-mess
libretro-beetle-pce-fast libretro-mgba libretro-
fbalpha2012-neogeo libretro-fba-neogeo libretro-
beetle-vb libretro-tgbdual libretro-fba-cps1
libretro-fba-cps2 libretro-fmsx libretro-stella
libretro-yabause libretro-mess2014 libretro-
mess2016 libretro-desmume libretro-beetle-bsnes
libretro-glupen64 libretro-catsfc libretro-
quicknes libretro-bsnes-mercury-balanced libretro-
vbam libretro-blueumx libretro-fceumm libretro-
nxengine libretro-snes9x-next libretro-mame2014
libretro-mame2016 libretro-fbalpha2012-cps1
libretro-fbalpha2012-cps2 libretro-fbalpha2012-
cps3 libretro-fbalpha libretro-snes9x libretro-
prboom libretro-beetle-gba -y
```

Si uno de estos paquetes falla, elimínalo de la lista e intentalo de nuevo. El comando anterior es el comando exacto que yo uso en mis C2. Una vez hecho esto, actualicemos y limpiemos un poco el sistema.

```
$ sudo apt-get update -y
$ sudo apt-get upgrade -y
$ sudo apt-get autoremove -y
```

Ahoras debería tener esta opción de menú disponible tras haber ejecutado los comandos Applications -> Games -> Retroarch. Haz clic en él y debería ver algo similar a lo que se muestra a continuación.

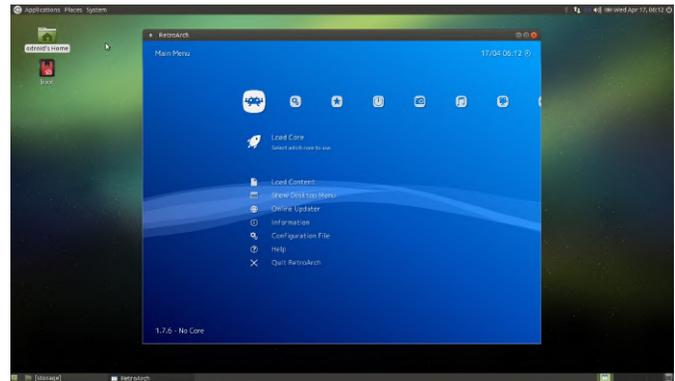


Figura 11

¡Casi estamos! No te metas a configurar Retroarch todavía; volveremos más tarde a él. A continuación, tenemos que instalar Antimicro para que podamos controlar todo con un gamepad cuando Retroarch no se esté ejecutando. Dirígete a <https://github.com/AntiMicro/antimicro/releases> y descárgate la última versión en forma de archivo zip. Deberías verlo debajo de las entradas EXE de Windows. Una vez haya finalizado la descarga, abre la carpeta de inicio de odroid. Tienes un acceso directo en el escritorio. Crea una nueva carpeta llamada install\_zips como se muestra a continuación.

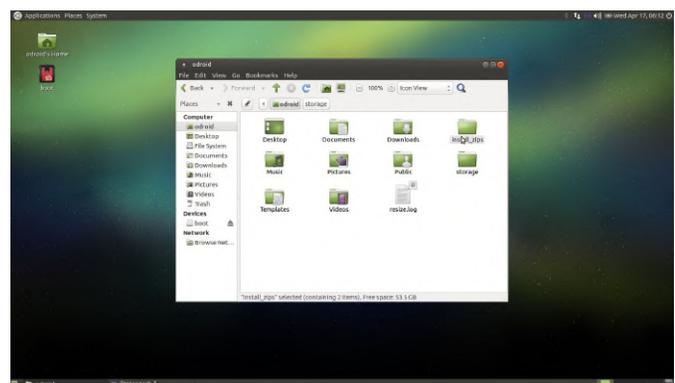


Figura 12

Ahora dirígete a la carpeta de descargas y copia el archivo zip de Antimicro y pégalo en la carpeta install\_zips. Haz clic derecho sobre él y selecciona Extract Here. Espera a que se descomprima.

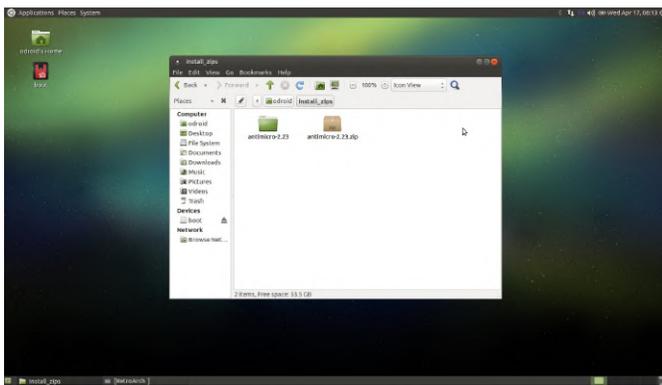


Figura 13

Ahora abre un terminal, Applications -> System Tools -> MATE Terminal, y ejecuta los siguientes comandos. Si tu carpeta Antimicro, después de descomprimir el archivo zip original, tiene un nombre diferente, usa ese nombre en el comando que aparece a continuación para cambiar de directorio.

```
$ cd install_zips/antimicro-2.23/
$ sudo apt-get install libsdl2-dev -y
$ sudo apt-get install qttools5-dev -y
$ sudo apt-get install qttools5-dev-tools -y
$ sudo apt-get install libxtst-dev -y
```

Una vez instalados todos estos paquetes, podemos compilar Antimicro sin ningún error:

```
$ cmake .
$ sudo make
$ sudo make install
```

Deberías ver algo como la Figura 14 durante este proceso. Una vez hecho esto, probemos Antimicro, ejecuta Antimicro en el terminal. Si todo está bien, debería ver algo como lo que se muestra en la Figura 15.

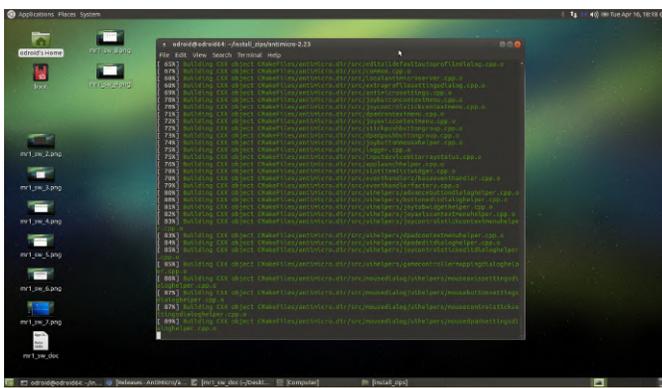


Figura 14

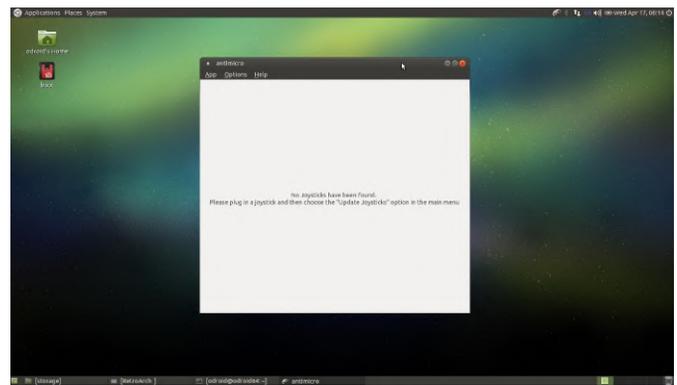


Figura 15

Conecta un mando de juegos compatible con Linux. Hardkernel tiene el mejor precio que he visto, así que si te haces con algunos ODRROID-GO u otro hardware, compra también uno de estos mandos con cable GameSir. Inicia Retroarch (Applications -> Games -> RetroArch), deberías ver un gran texto amarillo parpadeando en la parte inferior de la pantalla, míralo de cerca. He tenido algunas versiones del mando que funcionan un poco raro, aunque he estado trabajando con unas 5 y, en su mayor parte, no he tenido problemas. Cierre Retroarch y escribe en la ventana de terminal el comando antimicro. Debería ver algo similar a lo que se muestra a continuación si el mando ha sido detectado correctamente.

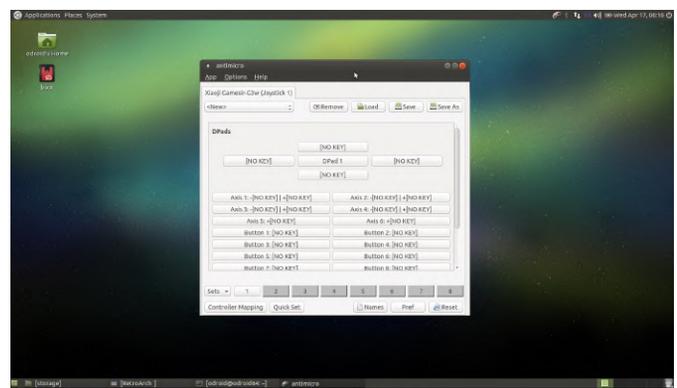


Figura 16

## Configuración de Retroarch y Antimicro

Vamos a configurar Antimicro para que podamos empezar a controlar el entorno de escritorio con el gamepad. Abre un terminal, no incluiré la ruta del menú desde aquí en adelante. Escribe Antimicro en el terminal y espera a que se inicie la aplicación. Conecta tu mando compatible con Linux y asegúrate de que Antimicro lo reconozca. Si no es así, tendrás que probar con otro mando. Haz clic en el botón Controller Mapping en la esquina inferior izquierda de la IU. Aquí es donde le indicas a Antimicro la

funcionalidad básica de tu mando. Si no tienes un botón para una posición específica de la lista, por ejemplo, Linux parece ignorar el botón central azul de los mandos GameSir, usa el ratón para hacer clic en la siguiente opción viable. Empareja los botones del gamepad con el indicador del botón verde del gráfico del mando. Ten en cuenta que algunos botones, como los disparadores, se disparan varias veces, y tendrá que usar el ratón para retroceder a la posición de la asignación y corregir la doble entrada. Haz clic en guardar cuando hayas terminado y regresa a la interfaz de usuario principal de Antimicro.

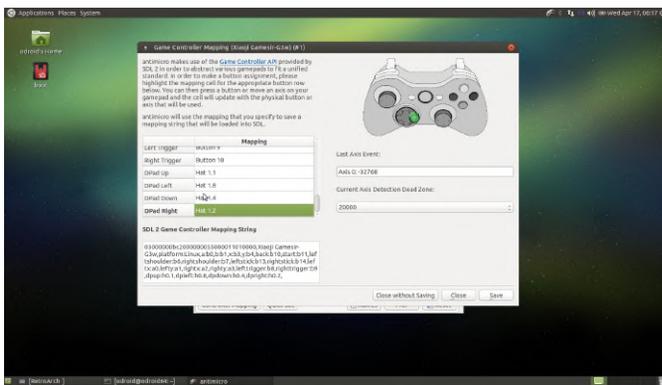


Figura 17

Verás una nueva asignación en la interfaz de usuario principal de Antimicro que contiene botones para todas las nuevas asignaciones que acaba de hacer. Lo que vamos a hacer ahora es configurar el soporte de ratón para que puedas controlar el entorno de escritorio desde el gamepad cuando Retroarch no se esté ejecutando. Usaremos el joystick izquierdo para un control más preciso y lento del ratón y el joystick derecho para un control más rápido del ratón. Los botones A y B servirán como los botones izquierdo y derecho del ratón. Haz clic derecho en el área de la barra de control izquierda y selecciona el ratón normal de la lista de opciones.

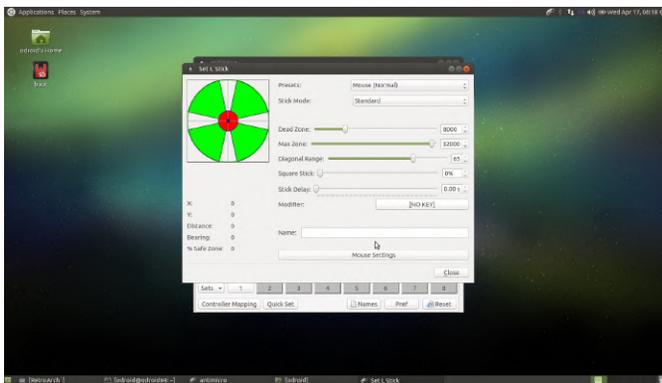


Figura 18

Vuelve a hacer clic en los botones del joystick izquierdo y busca el botón Mouse Settings en la parte inferior de la ventana. La imagen de arriba muestra el botón que estamos buscando. En la ventana de configuración del ratón, configura la Velocidad horizontal y la Velocidad vertical en 10 para el joystick izquierdo tal y como se muestra en la Figura 19.



Figura 19

Haz lo mismo con el joystick derecho, excepto que ahora configura la Velocidad horizontal y la Velocidad vertical en 30 tal y como se muestra en la Figura 20.



Figura 20

Ahora vamos a asignar los botones del ratón. Cierre todos los cuadros de diálogo y regresa a la interfaz de usuario principal de Antimicro. Busca el botón A en la lista de botones debajo de la lista de miniaturas y dpad. Haz clic en él y luego en la pestaña Mouse. Selecciona el botón izquierdo del ratón. Haz lo mismo para el botón B, excepto que hayas elegido el botón derecho del ratón para esa asignación. A continuación, se muestra una captura de pantalla que muestra la asignación del botón izquierdo del ratón en acción.

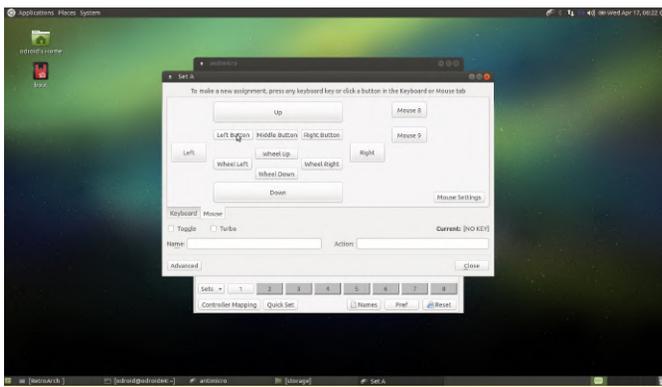


Figura 21

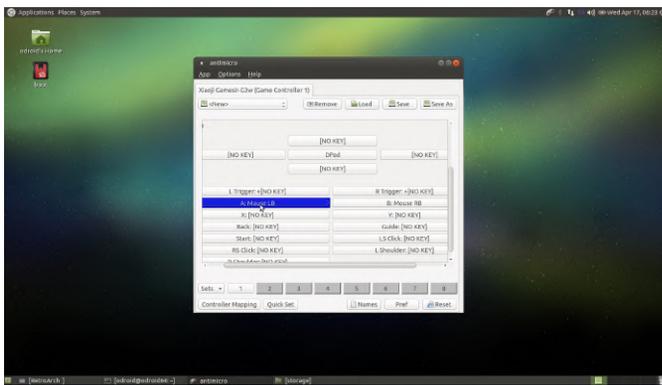


Figura 22

Pruébalo mientras la interfaz de usuario principal de Antimicro esté abierta. Debería ver que el ratón se mueve por la pantalla cuando las listas de botones en la interfaz de usuario de Antimicro se vuelven azules para indicar que están activos. Observa cómo se siente, ajusta las velocidades de los controles del ratón como mejor te parezca.

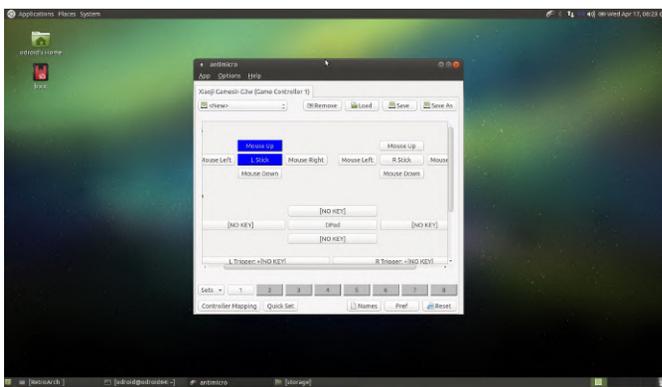


Figura 23

Cuando todo esté listo, regresa a la interfaz de usuario principal de Antimicro y haz clic en el botón Save As en la parte superior derecha de la pantalla. Guarda la configuración del mando como game\_sir\_wired.xml o como quieras nombrar a tu mando dentro del directorio de inicio de odroid, tal y como se muestra a continuación. Te proporcionaré una copia de mi archivo XML, si estás utilizando un

mando GameSir, así que puedes usarlo y ahorrar algo de tiempo. Si está utilizando un mando Easy SMX, usa este archivo.

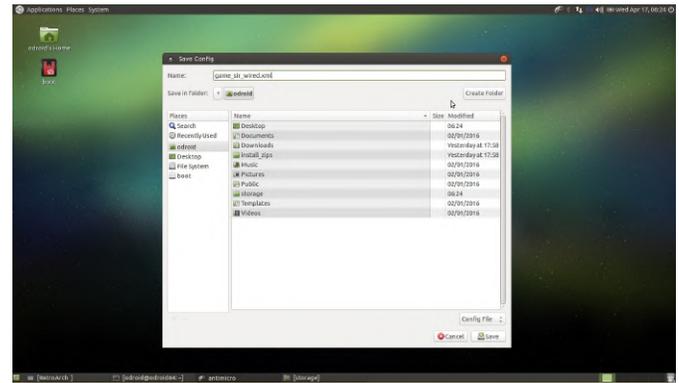


Figura 24

Haz clic en Options -> Settings en el menú Antimicro y asegúrate de que solo se haya seleccionado "Close To Tray" y "Launch In Tray". Esto permitira que Antimicro siga activo en la bandeja de aplicaciones y no desordene nuestra pantalla. ¡Tenemos una configuración más que ajustar y terminaremos con Antimicro y luego pasaremos a Retroarch!

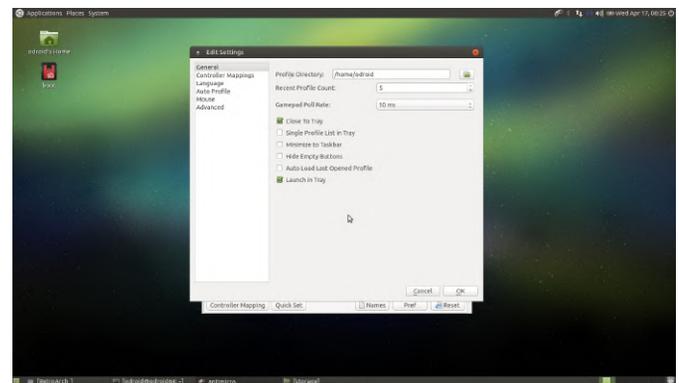


Figura 25

Estando todavía en la ventana de configuración de Antimicro, haz clic en la opción Auto Profile a la izquierda. Esto determinará qué perfil se asociará automáticamente con el gamepad conectado. Solo necesitas una asignación. Sería genial si tuviera diferentes opciones para diferentes hardware, pero hasta donde puede llegar, lo estás configurando para el mando que tienes. Haz clic en la casilla de verificación Active en la parte superior de la ventana. Luego selecciona la fila por defecto (y única) en la tabla. Haga clic en el botón Edit y busca el archivo XML de asignación del mando que guardaste antes. Haz clic en Ok y salte de Antimicro. Si aparece en el sistema, haz clic en el icono del mando en la bandeja

del sistema y salte de la aplicación. ¡Ya hemos terminado con la configuración de Antimicro!

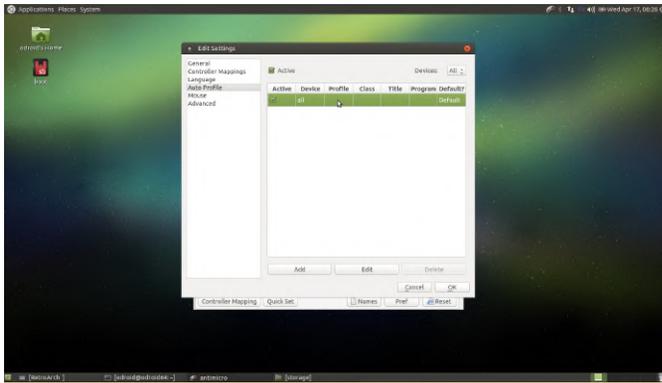


Figura 26

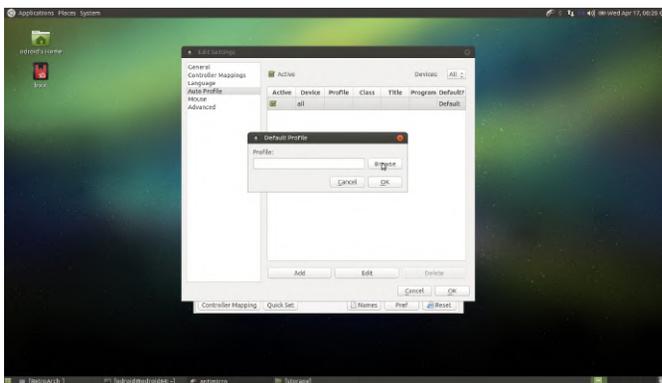


Figura 27



Figura 28

El siguiente paso es configurar correctamente Retroarch. Enciende Retroarch desde el sistema de menús. Primero, hagamos que el gamepad funcione en Retroarch. En Retroarch puede usar las teclas de flecha, intro y retroceso del teclado, para navegar por el sistema de menús sin el gamepad. Asegúrate de tener un ratón, un teclado y un mando de juego conectados a tu ODRROID. Use las flechas en el teclado para navegar directamente a la sección Settings, luego baja a la sección Input como se muestra a continuación.



Figura 29

Ajusta la configuración en esta pantalla como ves a continuación. Por lo general, yo configuro el número máximo de mandos en 4 ya que hay 4 puertos USB. Y me gusta la configuración del combo del mando para el menú "L1 + R1 + Start + Select", siendo realista si accidentalmente pulsamos esta combinación durante el juego, algo no está bien. Deja la configuración restante y desplázate hacia abajo hasta el User 1 Binds. Tendrá que configurar cada entrada del usuario de esta forma, no es tan lamentable, solo te llevará un minuto. Sugerencia: asigna los botones A y B por nombre y no por posición si está utilizando un mando GameSir, de esta manera, los colores verde y rojo se asignan al uso del botón positivo/seleccionar, negativo/atrás. ¡Es precisamente lo que me gusta hacer, tu puedes asignarlos como quieras! O puedes configurar los botones de cambio de menú y cancelar, independientemente de que funcione.



Figura 30

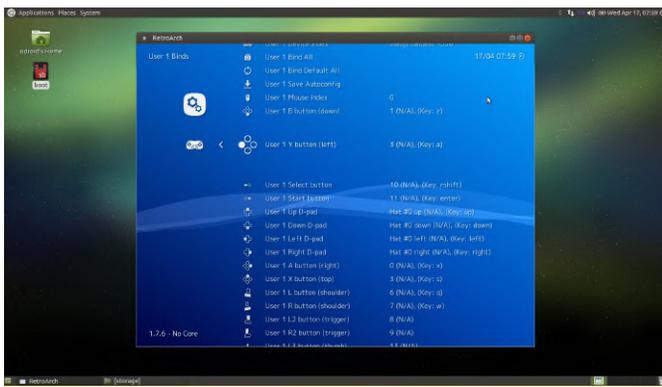


Figura 31

El siguiente paso lleva un poco de tiempo, pero requiere muy poco trabajo por tu parte. Solo tiene que hacer clic en algunas cosas y esperar a que se completen. Vuelve al menú principal, que es la primera sección en la que empieza Retroarch. Asegúrate de estar conectado a Internet. Desplázate hacia abajo hasta el Content Updater y abre esa sección.

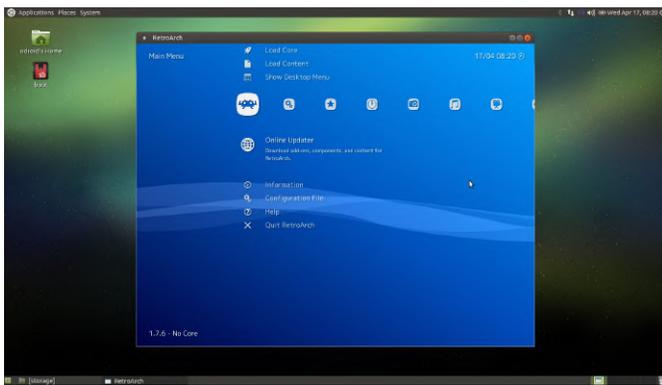


Figura 32

Normalmente ejecuto cada opción en orden. Prueba primero la opción Core Updater. Consejo: es posible que tengas que iniciar Retroarch desde el terminal de la misma manera que sudo Retroarch al actualizar los archivos principales. Si ejecuta Retroarch con el parametro -v y ves un mensaje de error durante la ejecución del comando Core Update, debe cerrar Retroarch y reiniciarlo con el comando sudo.

A continuación, dirígete al Thumbnails Updater, esta parte puede llevarte un tiempo y tienes que seleccionar de uno en uno cada sistema en el que tienes pensado cargar ROM. Resistete al impulso de hacer clic en un grupo de una fila. Lo he hecho y puede bloquear la aplicación en el peor de los casos y, como mínimo, ralentiza cada proceso individual, ya que todos compiten por los recursos. Lo mejor es dejar pasar una carga y luego seguir adelante.

Consejo: Hay algunos juegos para los que puede obtener miniaturas desde aquí, como DOOM Demo, Cave Story y Dinothawr. Podrás cargar esos juegos desde el Content Downloader.

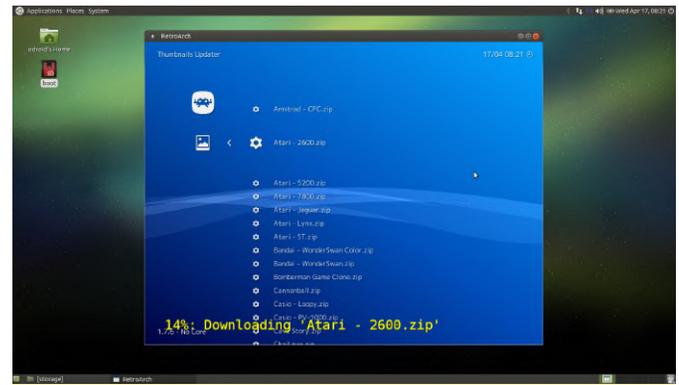


Figura 33

Ahora, querrás salir de la sección del Thumbnail Updater con la tecla de retroceso. Desplázate hacia abajo hasta el Content Downloader y descarga cualquier contenido especial que desees de la lista. Consejo: Si ves una pantalla que no tiene opciones, espera un minuto y vuelve a entrar en esa sección, simplemente no terminó de descargar el índice de opciones disponibles. Por lo general, no suelo instalar juegos únicos, aunque este es el lugar para ello. Creo que puedes ignorar la mayor parte del contenido, aunque la decisión es tuya. Ejecuta la siguiente lista de actualizaciones de una en una, básicamente todo lo que hay en la lista.

- Update Core Info Files
- Update Assets
- Update Joypad Profiles
- Update Cheats
- Update Databases
- Update Overlays
- Update GLSL Shaders
- Update Slang Shaders

Cierre Retroarch cuando se completen los pasos de actualización. A continuación, vamos a cargar algunas ROM. Poseo una copia de Contra para NES y tengo una versión en forma de ROM. Abre la carpeta de inicio de odroid usando el acceso directo del escritorio. Haz clic derecho en la carpeta y crea un nuevo directorio, asígnale el nombre ROMS. Abre la carpeta ROMS y crea una nueva carpeta y asígnale el

nombre NES. Obviamente, sustituirías la carpeta NES por cualquier sistema para el que quieras cargar ROM. No todos los sistemas son compatibles y algunos emuladores tienen pequeñas advertencias como los archivos BIOS que deben copiarse, etc. Muchos de los sistemas más antiguos son compatibles y no presentan errores. El C1+ probablemente pueda manejar SNES y Genesis, pero me gusta usarlo para juegos de 8 bits. El C2 ciertamente tiene suficiente potencia para sistemas de 16 bits y me gusta usarlo para 8 bits, 16 bits y todos los sistemas portátiles

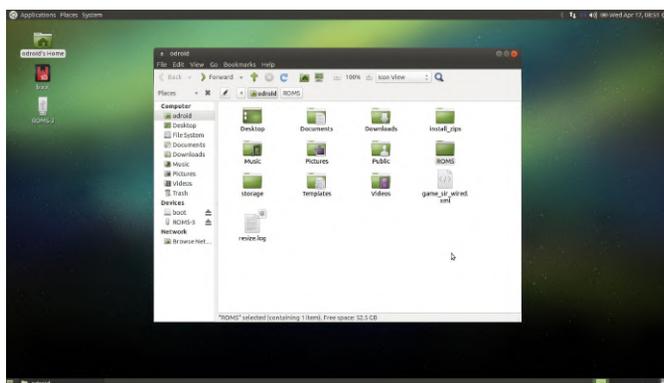


Figura 34

Una vez que tengas todas tus ROM colocadas en la carpeta ROMS dentro de sus propias carpetas separadas para cada sistema, esta es la mejor manera de hacerlo, podemos empezar a contárselo a RetroArch. Inicia RetroArch y desplázate hacia la derecha hasta que vea la opción Scan Directory



Figura 35

Navega a la carpeta ROMS, que debería estar en la lista. Luego selecciona un sistema a la vez para escanear. Escanear grandes cantidades de ROM puede llevar un tiempo. Una vez más, resiste el impulso de iniciar múltiples escaneos, las cosas serán más fáciles si dejas que se complete una operación antes de empezar una nueva. En este caso, solo

tenemos una ROM, Contra, por lo que el escaneo se completará al instante.



Figura 36

Vuelve a salir de la sección Scan Directory y ahora deberías ver un pequeño mando NES en la lista de secciones. Desplázate hasta él y ahora debería ver el Contra, o cualquier ROMs que intentaras cargar. Carga un juego seleccionándolo y luego seleccionando un emulador de destino. Tu juego debería cargarse bien en casi todos los casos.

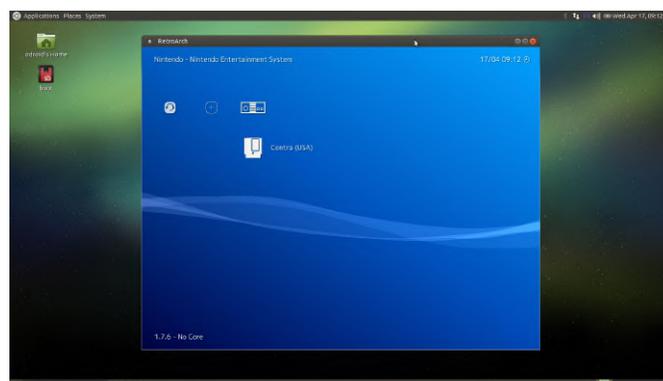


Figura 37

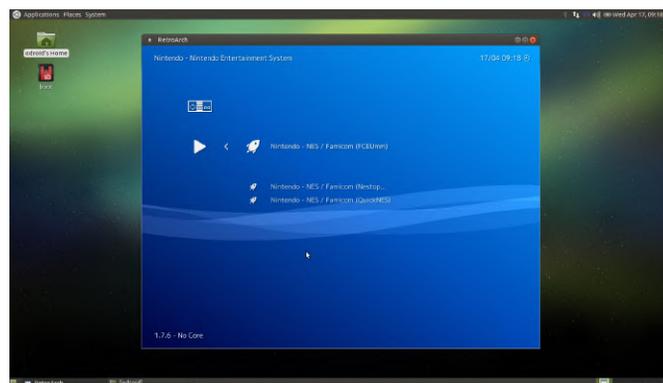


Figura 38



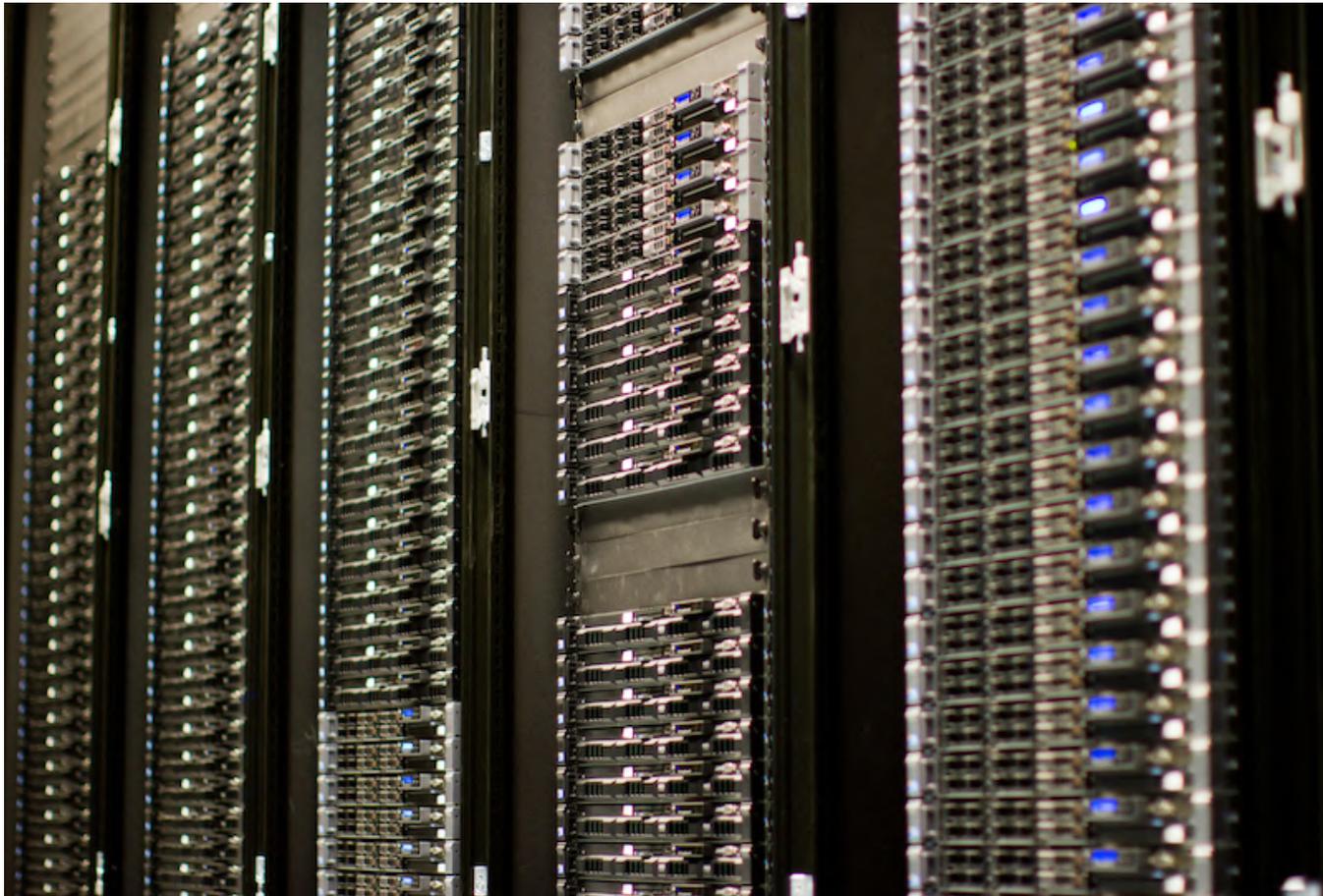
Figura 39

Retroarch no reconocerá todos los ROM de cada sistema. También es posible que desees probar

diferentes emuladores en ROM problemáticas. Eso es algo más avanzado y lo cubriré en el próximo tutorial donde lo mezclaremos todo y haremos una buena reverencia al sistema. Felicidades, montaste el hardware y ahora configuraste el software y puedes jugar algunos juegos. Tómate tu tiempo para cargar ROM. ¡Escribiré un poquito más en el tutorial 3, donde realmente hacemos que su consola de juegos destaque! Para comentarios, preguntas y sugerencias, visita el artículo original en [http://middlemind.com/tutorials/odroid\\_go/mr1\\_buid\\_sw.html](http://middlemind.com/tutorials/odroid_go/mr1_buid_sw.html).

# Un Excelente Servidor Web de Bajo Coste: Utilizando ODROID-N2 para Hosting de Internet

© August 6, 2019 By Robbie Ferguson Linux, ODROID-N2, Tutoriales



Como desarrollador de NEMS Linux, siempre estoy ansioso por probar nuevas placas que prometen alto rendimiento y estabilidad en el campo de los servidores. Dado que NEMS es, en sí mismo, una distribución de servidores sin monitor, tiene una gran ventaja ya que no necesito lidiar con los drivers de GPU o el rendimiento del escritorio. Lo que verdaderamente me importa a mí y a mis usuarios es algo muy diferente a lo que le importa a un usuario final que espera reemplazar su PC de escritorio por un ordenador de placa reducida.

Me aporta mucho, ya que me esfuerzo por proporcionar una comparativa basada en pruebas de rendimiento [-https://gigglescore.com/](https://gigglescore.com/)-con la finalidad de ayudar a los usuarios a descubrir con más facilidad qué placa es la que tiene la mejor relación calidad/precio. Tuve la oportunidad de hablar con @meveric cuando se lanzó ODROID-N2 por primera vez, y tengo que decir que no envidio para

nada cómo él y otros desarrolladores de distribuciones de escritorio tienen que lidiar con los gráficos. Trabajan muy duro para proporcionar una buena experiencia de usuario de serie, y no debe ser nada fácil. Cuando hablé con él, el desarrollo de ODROID-N2 todavía estaba en una etapa bastante temprana, y la falta de drivers X11 para la GPU Mali-G52 suponía un desafío más que interesante. En las distribuciones de servidores que compilo no tengo que lidiar con nada de esto, y el ODROID-N2 es un pequeño y excelente servidor muy estable. El uso final realmente importa; Una placa que funciona perfectamente para NEMS Linux u otras distribuciones basadas en servidor puede no ser la más idónea para montar un centro multimedia o un sistema de monitorización y control diario. Es por eso que resulta muy difícil basar las decisiones de compra solo en pruebas de rendimiento y por qué los

resultados de Giggle no siempre proporcionan una visión completa.

Es en este contexto en el que me fije en ODROID-N2 este mes. Me gustaría probarlo en un entorno en el que siento que puede encajar perfectamente; como es un servidor web SBC de alto rendimiento.

El ODROID-N2 tiene un total de 6 núcleos compuestos por un ARM Cortex-A73 quad-core y un ARM Cortex-A53 dual core. También cuenta con Ethernet Gigabit y hasta 4 GB de RAM

Usando Category5.TV SBC Benchmark v2.2 de <https://github.com/Cat5TV/cat5tv-sbctest> y con un coste para el ODROID-N2 de 79\$, mi única placa de prueba que ejecuta mi imagen base Debian Buster se muestra de esta forma:

Number of Threads:	6
SoC Contains big.LITTLE CPU:	Yes: 2 + 4 Cores
Compiler Time:	70 seconds
Multithreaded LZMA Benchmark:	8395 MIPS
Single-Threaded LZMA Benchmark:	4799 MIPS Average
sysbench CPU Score:	3,441.418 events per second. Price: ₱82.64 per unit.
sysbench RAM Score:	4,673,005.036 events per second. Price: ₱0.06 per unit.
sysbench Mutex Score:	9,441 events per second. Price: ₱3.01 per unit.
sysbench I/O Score:	1,194,456 events per second. Price: ₱238.10 per unit.
Giggle Scores:	
Multithreaded Giggle Score:	941.03 ₱v2
Single-threaded Giggle Score:	1646.17 ₱v2

Esta es ligeramente mejor que la puntuación de <https://gigglescore.com> que otorga el ODROID-N2 con un ₱v2 un resultado de 1304 (multiproceso), teniendo en cuenta que un resultado de Giggle más bajo significa mejor (medido por el rendimiento multiproceso). Por supuesto, mi prueba anterior se llevó a cabo en una instalación de Debian Buster vainilla, cuyos resultados son reales para un ODROID-N2 que de lo contrario no haría nada. Aun así, el ODROID-N2 actualmente tiene el mejor rendimiento global en GiggleScore.com y, como servidor web basado en un ordenador de placa reducida, el ODROID-N2 parece ser una excelente opción. Vamos a montar el nuestro y ponerlo a prueba.

En primer lugar, ten en cuenta que estoy empezando con la imagen base de Bald Nerd. Esta es Debian

Buster, por lo que no hay que configurar nada para que PHP 7.3 funcione. Nuestra pila LAMP incluirá Apache2, PHP 7.3 con algunos complementos útiles y ModSecurity 2.

Descarga la base de compilación de Debian Buster para ODROID-N2 y luego grábala en tu tarjeta SD o eMMC: <https://baldnerd.com/sbc-build-base/>

Puedes iniciar sesión en tu ODROID-N2 con un teclado y TV conectados, o desde otro ordenador dentro de tu red a través de SSH. El inicio de sesión y la contraseña por defecto para mi imagen base es "baldnerd".

Hay cinco cosas que debemos hacer en el primer arranque:

1) Verificar y asegurarnos de que la tarjeta SD/eMMC (/) haya cambiado de tamaño correctamente:

```
df -h
```

2) Verificar que la fecha/hora sean las correctas y corregirlas en caso contrario:

```
date
```

3) Crear una cuenta de usuario de Linux con privilegios de root. Yo crearé un usuario llamado "robbie" y le daré acceso sudo. Simplemente cambia "robbie" por su usuario.

```
$ sudo adduser robbie
$ sudo usermod -aG sudo robbie
```

4) Cierra sesión en tu ODROID-N2 y vuelve a iniciar sesión como robbie (o sea cual sea tu nombre de usuario). No inicies sesión como baldnerd nunca más  
5) Eliminar la cuenta de usuario por defecto:

```
$ sudo userdel -f -r baldnerd
```

Montemos nuestro servidor web LAMP (Linux, Apache, MySQL, PHP). Para aquellos que prefieran llevar a cabo la instalación mediante programación, he escrito un simple script:

<https://raw.githubusercontent.com/Cat5TV/linux-tools/master/install-lamp>

Para aquellos que quieran mojarse un poco y ver cada paso, pueden ejecutar lo siguiente en su ODROID-N2 con Debian Buster.

```
$ sudo apt update
$ sudo apt install apache2 php7.3 php7.3-cli
php7.3-common php7.3-curl php7.3-gd php7.3-json
php7.3-mbstring php7.3-mysql php7.3-opcache
php7.3-phdbg php7.3-readline php7.3-sqlite3
php7.3-xml libapache2-mod-php7.3 libargon2-1
libsodium23 php-curl php-rrd mariadb-server
libapache2-mod-security2 modsecurity-crs
```

## Fija una contraseña root para MySQL

Si has instalado mariadb-server anteriormente, seguramente recordarás que el procedimiento de instalación solía solicitar una contraseña root durante la instalación. Esto ha sido eliminado. En su lugar, tienes que ejecutar una aplicación posterior a la instalación con la que configurar tu usuario root y concedes acceso root a tu servidor de base de datos MySQL.

```
$ sudo mysql_secure_installation
```

Presiona Intro cuando te solicite la contraseña de root de MySQL, ya que aún no existe, y luego sigue las instrucciones para añadir una nueva contraseña de root. Recuerda cuál es esta contraseña y mantenla segura. Asegúrate de contestar "yes" a cada pregunta que se te haga para ayudar a hacer más seguro tu servidor MySQL eliminando algunas cuentas y datos de muestra.

Proporciona a tu cuenta de usuario acceso para editar archivos. Recuerda reemplazar "robbie" por tu nombre de usuario.

```
$ sudo usermod -aG www-data robbie
$ sudo chown -R robbie:www-data /var/www/html/
$ sudo find /var/www/html/ -type d -exec chmod -R
775 {} ;
$ sudo find /var/www/html/ -type f -exec chmod -R
664 {} ;
```

¡Perfecto! Tu servidor web ahora se está ejecutando, y puedes acceder a él visitando la dirección IP de tu ODROID-N2 en un navegador en cualquier ordenador de tu red. Los archivos del servidor se encuentran en /var/www/html y son propiedad de tu usuario y ya conoces las credenciales de MySQL. El acceso a MySQL está restringido a localhost, así que, para facilitar su uso, recomiendo usar la línea de comando

mysql para crear los usuarios y las bases de datos MySQL, o agregar phpMyAdmin y activarlo solo cuando sea necesario, deshabilitándolo cuando no lo estés usando. Recuerde que, puesto que SSH está habilitado en la imagen, puedes crear fácilmente archivos en /var/www/html, o usar SFTP en Filezilla Client para mover archivos de aquí para allá desde tu ordenador.

## Y ahora, las pruebas de rendimiento

Tu servidor está listo para funcionar, llevar a cabo pruebas de rendimiento puede ser muy divertido en un caso como este. Quizás sea mis recuerdos con servidores gigantes 4U de mis años más jóvenes en TI lo que me entusiasma testear ordenadores de placa reducida como centros de datos, pero sea lo que sea, me impresiona ver lo bien que funciona un ODROID-N2 en este rol. No es necesario que realices estas pruebas personalmente si no quieres, este trabajo ya está hecho, aunque este artículo no estaría completo sin mis propios resultados.

Mis pruebas de rendimiento que se detallan a continuación son muy específicas, centradas por completo en usar el ODROID-N2 como servidor web. Si te saben a poco o quieres una visión más amplia, echa un vistazo a la edición de junio de 2019 de ODROID Magazine donde Michael Larabel de Phoronix.com y Carlos Eduardo dan su opinión sobre el funcionamiento y rendimiento del ODROID-N2.

Ahora que tenemos Apache en funcionamiento, pongámoslo a prueba con Siege, una utilidad para evaluar el rendimiento HTTP. Todavía no he añadido un sitio web, así que realmente solo estamos probando la capacidad de respuesta de la página estática "It Works", aun así los resultados siguen siendo importantes. Te recomiendo que coloques tu sitio en /var/www/html y ejecutes la misma prueba de rendimiento para ver qué tal funciona tu sitio. Para este indicador de rendimiento, haremos una prueba de carga con 50 conexiones simultáneas.

```
$ apt install siege
$ siege -c50 -d10 -t3M http://localhost/
```

Here is my result:

```

robbie@debian:/home/robbie# siege -c50 -d10 -t3M
http://localhost/
New configuration template added to /root/.siege
Run siege -C to view the current settings in that
file
** SIEGE 4.0.4
** Preparing 50 concurrent users for battle.
The server is now under siege...
Lifting the server siege...
Transactions:          3540 hits
Availability:          100.00 %
Elapsed time:          179.16 secs
Data transferred:     14.85 MB
Response time:         0.01 secs
Transaction rate:     19.76 trans/sec
Throughput:           0.08 MB/sec
Concurrency:          0.12
Successful transactions: 3540
Failed transactions:   0
Longest transaction:  0.20
Shortest transaction: 0.00

```

Con cada petición se observa una respuesta con un promedio de 0.01 segundos, realmente estoy encantado con este resultado. La mayoría de las pymes no suelen ver 50 usuarios simultáneos en su sitio web. Aquí, hemos generado 3.540 visitas en 3 minutos.

## Probar el rendimiento de MariaDB

Nuestro servidor MySQL (MariaDB) está funcionando gracias a los pasos anteriores. Para probar el rendimiento de la base de datos, usaré sysbench a través del script mysql-benchmark.sh que forma parte de mi paquete cat5tv-sbctest en GitHub. No voy a hablar sobre cómo funciona, ya que para lo que estén interesados pueden ver el código fuente del simple script bash. Basta con decir que compila sysbench 1.0.17, crea un 1 millón de entradas MySQL y pone a prueba rendimiento de la base de datos durante 1 minuto.

Estos son mis resultados:

```

Performing MySQL Benchmark: sysbench 1.0.17 (using
bundled LuaJIT 2.1.0-beta2)

Creating table 'sbtest1'...
Inserting 1000000 records into 'sbtest1'
Creating a secondary index on 'sbtest1'...
sysbench 1.0.17 (using bundled LuaJIT 2.1.0-beta2)

```

```

Running the test with following options:
Number of threads: 8
Initializing random number generator from current
time

Initializing worker threads...

Threads started!

SQL statistics:
  queries performed:
    read:                35504
    write:               10144
    other:                5072
    total:               50720
  transactions:         2536
(40.83 per sec.)
  queries:               50720
(816.58 per sec.)
  ignored errors:       0
(0.00 per sec.)
  reconnects:           0
(0.00 per sec.)

General statistics:
  total time:            62.1083s
  total number of events: 2536

Latency (ms):
  min:
11.15
  avg:
194.09
  max:
4760.50
  95th percentile:
846.57
  sum:
492217.74

Threads fairness:
  events (avg/stddev):    317.0000/3.12
  execution time (avg/stddev): 61.5272/0.74

```

Todas mis consultas tuvieron éxito, se llegaron a realizar unas impresionantes 816.58 consultas por segundo. ¡Este pequeño manejará WordPress perfectamente! Recuerda que estamos ante un ordenador de placa reducida que utiliza muy poca electricidad, genera muy poco calor en la sala de servidores y cuesta 79\$. Estoy impresionado. Si está

buscando un servidor web asequible con tecnología SBC, ya sea para desarrollo, prueba o producción, el ODROID-N2 está listo para funcionar y tiene suficiente potencia para atender tus proyectos.

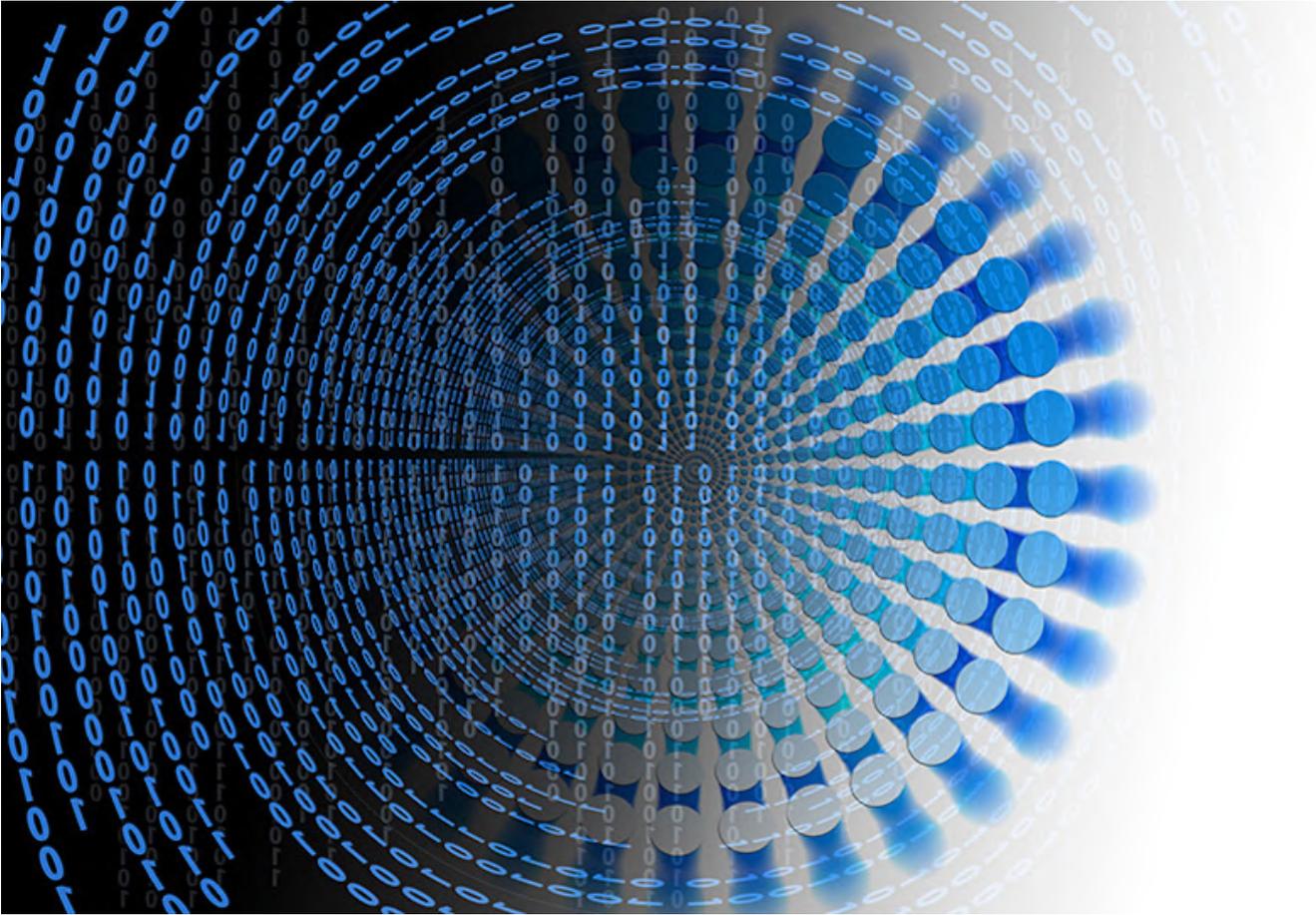
### **Sobre el Author**

Robbie Ferguson es el dueño de Category5 Technology TV y autor de NEMS Linux. Su programa de televisión se encuentra en <https://category5.tv/> y

su blog es <https://baldnerd.com/>. El proyecto de Robbie, NEMS Linux, está disponible para ODROID-N2. Puedes descargar la distribución del Servidor de Monitoreo Empresarial Nagios listo para arrancar desde <https://nemslinux.com/>

# Montando un Clúster ODROID-N2: Informática de Alto Rendimiento Asequible

© August 6, 2019 By Bhaskar S (www.polarsparc.com) Linux, ODROID-N2, Tutoriales



En el artículo “Montado un clúster ODROID-XU4”, detallamos los ingredientes necesarios para montar un clúster de laboratorio doméstico de 5 nodos utilizando el potente ODROID-XU4 de 32 bits. Tal y como se indicó en su momento, el ODROID-XU4 tiene un potente CPU Samsung Exynos5422 ARM de ocho núcleos con dos conjuntos de cuatro núcleos, conocidos como big.LITTLE, con Cortex A15 a 2 Ghz y Cortex A7 a 1.4 Ghz. Tiene la suficiente potencia como para ejecutar Linux y actuar como un mini clúster de desarrollo. Sin embargo, el ODROID-XU4 es un SBC de 32 bits con 2 GB de memoria LPDDR3 PoP.

En mi intento por encontrar un verdadero SBC de quad-core de 64 bits en el que cada núcleo estuviera cercano a los 2 Ghz y que contase con al menos 4 GB de RAM, me encontré con esta belleza recién lanzada llamada ODROID-N2 (4 GB de RAM). Tiene un potente CPU Amlogic 922X basada en ARM hex-core con dos conjuntos de núcleos, también conocida como

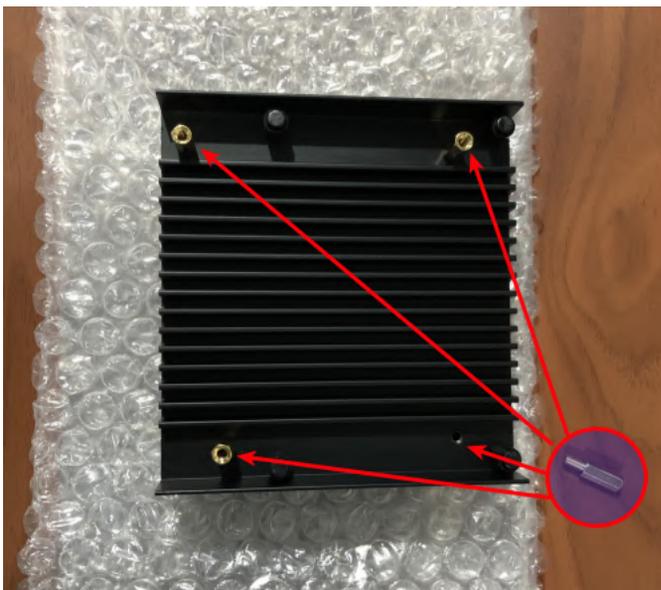
big.LITTLE, con un Cortex A73 de cuatro núcleos a 1,8 Ghz y un Cortex A53 de doble núcleo a 1,9 Ghz . Además, viene con un disipador térmico \* ENORME \* montado en la parte inferior, donde se encuentra la CPU. Está disponible en dos configuraciones, una con 2 GB de RAM DDR4 y la otra con 4 GB de RAM DDR4 (funcionando a 1320 Mhz). Tiene 4 puertos USB 3.0 y un puerto Ethernet de 1 Gb.

Los siguientes son los elementos necesarios para montar un clúster ODROID-N2 de 5 nodos:

- Cinco ordenadores de placa reducida ODROID-N2 de 4 GB
- Cinco adaptadores de corriente de 12V/2A
- Cinco tarjetas microSDXC de 64 GB de clase 10
- Cinco cables de red Ethernet
- Un pack de separadores de latón macho-hembra M3 de 60 mm
- Un pack de separadores de latón macho-hembra M3 de 12 mm

- Un pack de tornillos de acero Phillips M3 de 6 mm
- Un switch Ethernet multipuerto (si tu switch de red doméstica no tiene suficientes puertos disponibles). En mi caso, no necesitaba uno ya que tenía un switch Ethernet de 24 puertos con suficientes puertos libres para cubrir mis necesidades.
- Un lector de tarjetas USB portátil, si tu ordenador de escritorio no tiene un lector de tarjetas multimedia. En mi caso, no necesitaba uno ya que mi ordenador personal tiene un lector de tarjetas integrado.
- Una regleta con 12 tomas Belkin BE112230-08
- Necesitamos asegurar con firmeza el ODROID-N2 en una plataforma sólida y plana, que se pueda apilar una encima de otra. En mi caso, una visita al Dollar Tree local me aportó una opción interesante: una bandeja cuadrada de plástico transparente de 12,5 cm x 12,5 cm. Para mi clúster, compré 5.

El siguiente paso es ensamblar los elementos para dar forma el clúster ODROID-N2 de 5 nodos. El ODROID-N2 es un tanto único en su diseño. A diferencia del ODROID-XU4, no existen agujeros en el SBC para apilar las placas fácilmente una encima de la otra. Esto es debido al disipador térmico que ocupa todo el fondo del SBC. Sin embargo, el disipador térmico tiene una serie de orificios para atornillar los separadores de latón M3 de 12 mm en los 4 lados, tal y como se muestra en la siguiente imagen.



**Figura 1: Ubicaciones de los tornillos**

Continué con este proceso de ensamblaje para las restantes unidades. Necesitamos perforar 4 agujeros (círculos rojos) para asegurar cada SBC. Asegúrate de que los orificios están alineados con los cuatro

separadores M3 de 12 mm que atornillamos al SBC en la Figura 1. Además, necesitamos otros 4 orificios (círculos azules) en las esquinas de la bandeja de plástico para apilar las bandejas una encima de otra tal como se muestra en la siguiente imagen:



**Figura 2 - Los 8 agujeros perforados**

Sigue con este proceso en las restantes unidades. Necesitamos asegurar el ODROID-N2 en la parte superior de la bandeja de plástico transparente usando los tornillos M3 de 6 mm a través de los 4 agujeros (círculos rojos) de la Figura 2 anterior, tal y como se muestra en la siguiente imagen:



**Figura 3: ODRROID-N2 montado en una estructura de plástico, "¡hola piña!"**

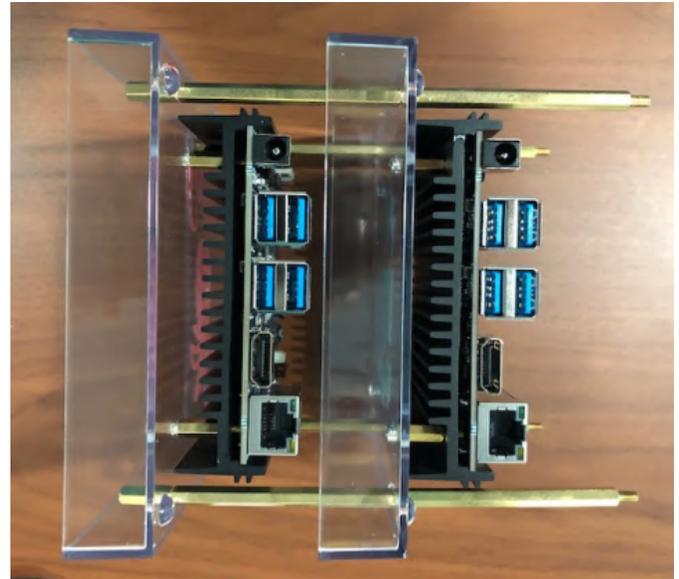
Nuevamente, continúa este proceso para las restantes unidades. Inserta los separadores de latón M3 de 12 mm a través de los 4 orificios (círculos azules) de la Figura 2 anterior y atorníllalos en la parte superior con los espaciadores de latón M3 de 60 mm tal y como se muestra en la siguiente imagen:



**Figura 4 - Ahora con los separadores incluidos**

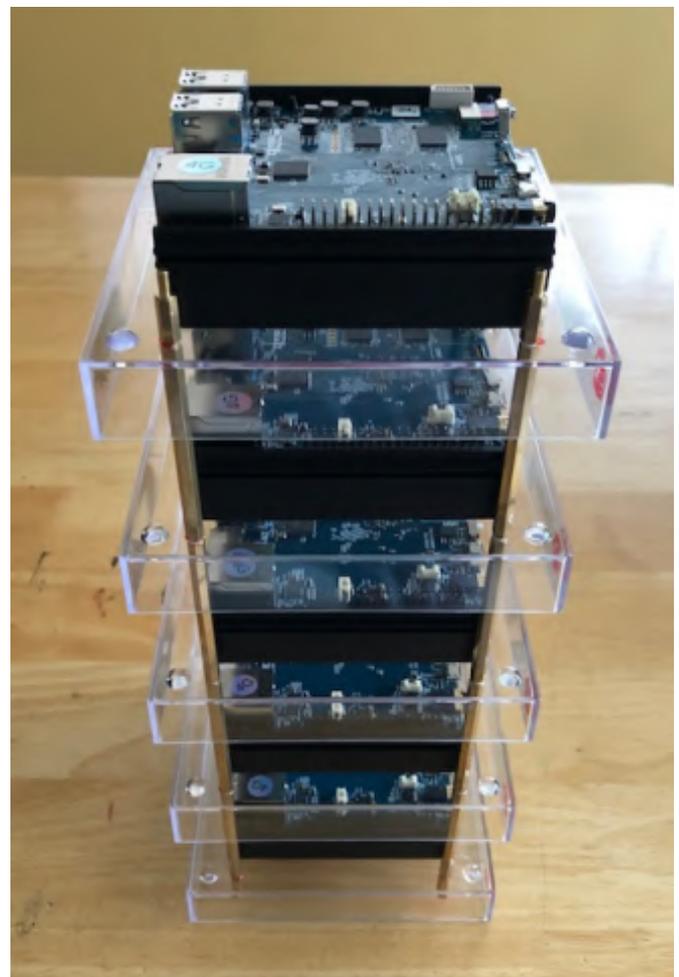
Esto es solo para una de las unidades que estará en la base de la pila. Apila la siguiente bandeja de plástico con el SBC de la Figura 3 en la parte superior de la unidad de la Figura 4 y atorníllela en la parte superior

con los espaciadores de latón M3 de 60 mm tal y como se muestra en la siguiente imagen:



**Figura 5: Dos ODRROID-N2 apilados**

Haz lo mismo para las restantes unidades. Para la bandeja superior, asegúrala con los espaciadores M3 de 12 mm.



**Figura 6: Un rack de clúster finalizado**

A continuación, descárgate la última versión estable del sistema operativo Ubuntu MATE Linux, basado en Ubuntu Bionic 18.04, tal y como se muestra en la siguiente imagen:



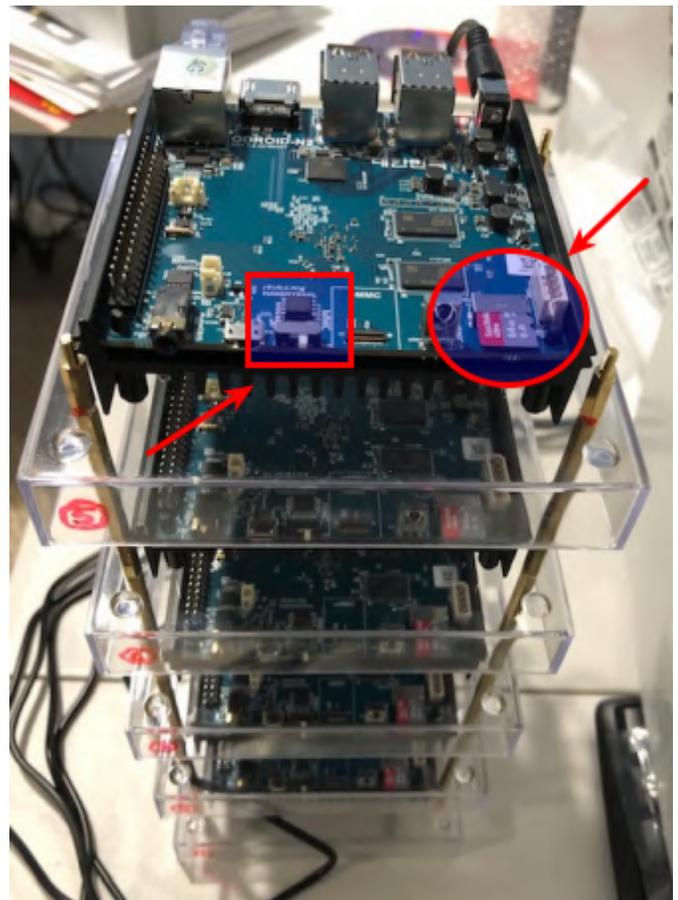
**Figura 7 - Ubicación de las descargas de Ubuntu para el ODROID-N2**

Al momento de escribir este artículo, la versión más reciente de Ubuntu MATE era la 20190325. Descarga la última versión estable de Etcher. Esta herramienta se usará para grabar la imagen del sistema operativo en cada una de las tarjetas microSDXC. Inserta una tarjeta microSDXC en el lector de tarjetas SD, inicia Etcher, selecciona la imagen Ubuntu MATE. ¡Luego presiona el botón Flash!, para grabar la imagen en la tarjeta microSDXC tal y como se muestra en la siguiente imagen:



**Figura 8: Captura de pantalla de la configuración de Etcher para grabar la SD con el sistema operativo**

Inserta cada una de las tarjetas microSDXC actualizadas en la correspondiente ranura para tarjeta SD del ODROID-N2 y asegúrate de que el interruptor de palanca del ODROID-N2 esté colocado en MMC en lugar de SPI tal como se muestra en la siguiente imagen:



**Figura 9: ubicación de la ranura SD y configuración del interruptor de arranque**

El paso final es preparar cada uno de los nodos ODROID-N2 para el trabajo. Ten en cuenta que queremos conectar, encender y configurar cada una de las placas ODROID-N2, una por una. Conecta el extremo del adaptador de corriente a la clavija de alimentación del ODROID-N2 y conecta el otro extremo a la regleta. Del mismo modo, conecta uno de los cables Ethernet a la ranura RJ45 en el ODROID-N2 y el otro extremo al switch Ethernet tal y como se muestra en la siguiente imagen:



**Figura 10: Cables Ethernet conectados al clúster**

Conecta un monitor HDMI al ODROID-N2 usando HDMI y enciende la regleta. El ODROID-N2 debería

tardar poco en iniciarse y tras unos segundos se apagaría por completo. Este es el comportamiento normal. Apaga la regleta y vuévela a encenderla. En unos segundos, aparecerá el escritorio Ubuntu MATE. El usuario por defecto es odroid y la contraseña es odroid.

Cambia la contraseña por defecto por una contraseña más haciendo clic en el menú de Ubuntu de la esquina superior izquierda y seleccionando Control Center en el menú desplegable. En la pantalla resultante, haga clic en el icono de usuarios y grupos para cambiar la contraseña de usuario por defecto.

Cambia la fecha/hora/zona horaria predeterminada al valor deseado haciendo clic en el menú de Ubuntu de la esquina superior izquierda y seleccionando Control Center en el menú desplegable. En la pantalla resultante, haz clic en el icono de Hora y fecha para cambiar la fecha/hora/zona horaria.

Abra una ventana de terminal haciendo clic en la esquina superior izquierda del menú de Ubuntu y seleccionando System Tools -> MATE Terminal en el menú desplegable. En la ventana de terminal, ejecuta el siguiente comando:

```
$ sudo pluma /etc/ssh/sshd_config
```

Cambia el valor de PermitRootLogin de yes a no. Guarda los cambios y salte del editor.

Necesitamos asignar un nombre de host único para cada una de las placas ODROID-N2 en el clúster. En la misma ventana de terminal, ejecuta el siguiente comando:

```
$ sudo pluma /etc/hostname
```

Cambia el nombre del host por, digamos, my-n2-1. Guarda los cambios y salte del editor. Para cada una de las 4 tarjetas restantes, asignaremos los nombres de host desde my-n2-2 hasta my-n2-5

También necesitamos asignar una dirección IP estática a cada una de las placas ODROID-N2 en el

clúster. Haga clic en el menú de Ubuntu en la esquina superior izquierda y elije Control Center en el menú desplegable. En la pantalla resultante, haz clic en el icono de Conexiones de red. Elije Wired connection 1 de la lista del menú y haz clic en el icono de edición en la esquina inferior izquierda (icono de rueda dentada). Selecciona la pestaña etiquetada con IPv4 Settings. Seleccionar el valor Manual en Method. Fija la dirección IP en, por ejemplo, 192.168.1.51, la red en 255.255.255.0 y la puerta de enlace en 192.168.1.1. Finalmente, ajusta el valor para los servidores DNS en 8.8.8.8,8.8.4.4. Guarda los cambios haciendo clic en el botón Save. Para cada una de las 4 tarjetas restantes, asignaremos las direcciones IP estáticas desde 192.168.1.52 hasta 192.168.1.55.

Por último, necesitamos realizar una actualización del sistema para asegurarnos de tener las últimas actualizaciones de software. Haga clic en el menú de Ubuntu en la esquina superior izquierda y elije Control Center en el menú desplegable. En la pantalla resultante, haz clic en el icono Software Updater para llevar a cabo la actualización del sistema.

El último paso es realizar un reinicio para que todos los cambios tengan efecto. En la ventana del terminal, ejecuta el siguiente comando:

```
$ reboot
```

Esto reiniciará de forma segura el dispositivo ODROID-N2.

Realiza los pasos anteriores para cada uno de los dispositivos ODROID-N2 restantes. Ten en cuenta que a medida que reiniciamos cada una de las tarjetas ODROID-N2 restantes, cada una recibirá una dirección IP diferente asignada. Llegados a este punto, deberíamos tener nuestro clúster ODROID-N2 listo para la acción. Para comentarios, preguntas y sugerencias, visita el artículo original en <https://www.polarsparc.com/xhtml/N2-Cluster.html>.