

ODROID

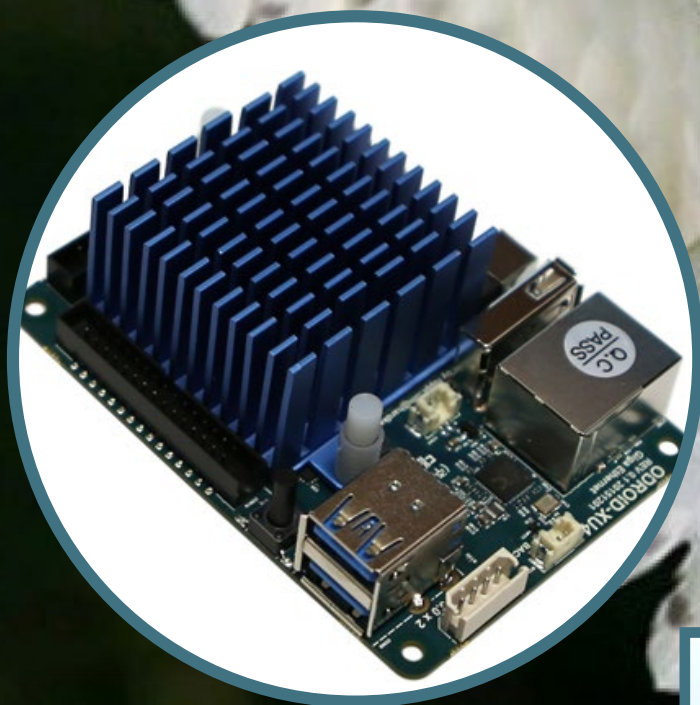
Year Four
Issue #42
Jun 2017

Magazine

Stealth power:

ODROID XU4Q

Our most powerful board, now with 100% silent operation



Getting Kernel Boot
and Panic Messages
With Netconsole

XU4 Case
A Futuristic Design For
Your 3D Printer

What we stand for.

We strive to symbolize the edge of technology, future, youth, humanity, and engineering.

Our philosophy is based on Developers. And our efforts to keep close relationships with developers around the world.

For that, you can always count on having the quality and sophistication that is the hallmark of our products.

Simple, modern and distinctive. So you can have the best to accomplish everything you can dream of.



HARDKERNEL



We are now shipping the ODRROID-C2 and ODRROID-XU4 devices to EU countries! Come and visit our online store to shop!

Address: Max-Pollin-Straße 1
85104 Pförring Germany

Telephone & Fax
phone: +49 (0) 8403 / 920-920
email: service@pollin.de

Our ODRROID products can be found at
<http://bit.ly/1tXPXwe>





When ODRROIDians give feedback, Hardkernel listens! The best example of this is the new ODRROID-XU4Q (Q for Quiet), which is in response to many requests for a silently operating ODRROID. Like the X2, U2, U3, and C2, the new XU4Q comes pre-installed with a heatsink instead of a fan, which does a great job of cooling the device without any fan noise. For high-performance applications, the original XU4 is still the best choice, but if you're just watching videos, browsing the Internet, or creating documents, the quiet version offers the same responsiveness and speed that you've come to expect from the XU series.

The new XU4Q also comes with an updated case, but that hasn't stopped ODRROID fans from building their own optimized cases that are both beautiful and functional. Edward follows up his original case article with more improvements to an artful enclosure that you can 3D print at home, Tobias brings us a detailed review of ExaGear 2.0 along with a surprise game, Marian introduces the Wayland compositor, Adrian describes using NetConsole for system logging, and Jörg continues his series on building a home alarm system with ultra-low power motion sensors.

ODROID Magazine, published monthly at <http://magazine.odroid.com>, is your source for all things ODRROIDian. Hard Kernel, Ltd. • 704 Anyang K-Center, Gwanyang, Dongan, Anyang, Gyeonggi, South Korea, 431-815. Hardkernel manufactures the ODRROID family of quad-core development boards and the world's first ARM big.LITTLE single board computer. For information on submitting articles, contact odroidmagazine@gmail.com, or visit <http://bit.ly/typlmXs>. You can join the growing ODRROID community with members from over 135 countries at <http://forum.odroid.com>. Explore the new technologies offered by Hardkernel at <http://www.hardkernel.com>.

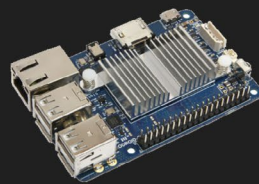


HARDKERNEL

 **ameriDroid.com** High-Performance Embedded Computers Hundreds of products available online for the professional developer and hobbyist alike



ODROID-XU4



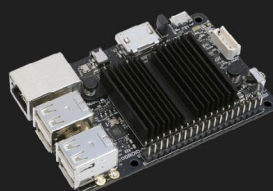
ODROID-C1+



ODROID-C0



OWEN ROBOT KIT



ODROID-C2



VU7 TABLET KIT

Hardkernel's EXCLUSIVE North American Distributor

OUR AMAZING ODROIDIAN STAFF:



Rob Roy, Chief Editor

I'm a computer programmer in San Francisco, CA, designing and building web applications for local clients on my network cluster of ODROIDS. My primary languages are jQuery, Angular JS and HTML5/CSS3. I also develop pre-built operating systems, custom kernels and optimized applications for the ODROID platform based on Hardkernel's official releases, for which I have won several Monthly Forum Awards. I use my ODROIDS for a variety of purposes, including media center, web server, application development, workstation, and gaming console. You can check out my 100GB collection of ODROID software, prebuilt kernels and OS images at <http://bit.ly/1fsaXQs>.



Bruno Doiche, Senior Art Editor

Basically inherited a mutt dog to take care of on weekends whenever he visits his family. Who gets most the most stoked about him when he arrives? The dog, obviously!



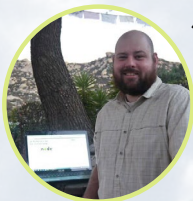
Manuel Adamuz, Spanish Editor

I am 31 years old and live in Seville, Spain, and was born in Granada. I am married to a wonderful woman and have a child. A few years ago I worked as a computer technician and programmer, but my current job is related to quality management and information technology: ISO 9001, ISO 27001, and ISO 20000. I am passionate about computer science, especially microcomputers such as the ODROID and Raspberry Pi. I love experimenting with these computers. My wife says I'm crazy because I just think of ODROIDS! My other great hobby is mountain biking, and I occasionally participate in semi-professional competitions.



Nicole Scott, Art Editor

Nicole is a Digital Strategist and Transmedia Producer specializing in online optimization and inbound marketing strategies, social media management, and media production for print, web, video, and film. Managing multiple accounts with agencies and filmmakers, from web design and programming, Analytics and Adwords, to video editing and DVD authoring, Nicole helps clients with the all aspects of online visibility. Nicole owns an ODROID-U2, a number of ODROID-U3's, and Xu4's, and looks forward to using the latest technologies for both personal and business endeavors. Nicole's web site can be found at <http://www.nicolecscott.com>.



James LeFevour, Art Editor

I'm a Digital Media Specialist who is also enjoying freelance work in social network marketing and website administration. The more I learn about ODROID capabilities, the more excited I am to try new things I'm learning about. Being a transplant to San Diego from the Midwest, I am still quite enamored with many aspects that I think most West Coast people take for granted. I live with my lovely wife and our adorable pet rabbit; the latter keeps my books and computer equipment in constant peril, the former consoles me when said peril manifests.



Andrew Ruggeri, Assistant Editor

I am a Biomedical Systems engineer located in New England currently working in the Aerospace industry. An 8-bit 68HC11 microcontroller and assembly code are what got me interested in embedded systems. Nowadays, most projects I do are in C and C++, or high-level languages such as C# and Java. For many projects, I use ODROID boards, but I still try to use 8bit controllers whenever I can (I'm an ATMEL fan). Apart from electronics, I'm an analog analogue photography and film development geek who enjoys trying to speak foreign languages.



Venkat Bommakanti, Assistant Editor

I'm a computer enthusiast from the San Francisco Bay Area in California. I try to incorporate many of my interests into single board computer projects, such as hardware tinkering, metal and woodworking, reusing salvaged materials, software development, and creating audiophile music recordings. I enjoy learning something new all the time, and try to share my joy and enthusiasm with the community.

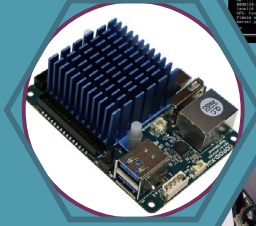
INDEX



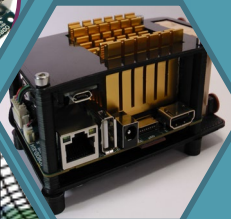
WAYLAND ON XU4: GNOME 3 - 6



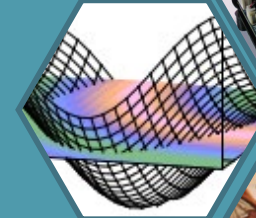
NETCONSOLE - 8



ODROID-XU4Q - 10



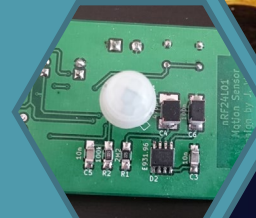
ODROID-XU4 CASE - 14



GNUPLOT - 17



LINUX GAMING - 18



ALARM CENTRAL - 27



ANDROID GAMING - 29



MEET AN ODROIDIAN - 30

GNOME3

WAYLAND ON THE ODROID-XU4

by Marian Mihailescu



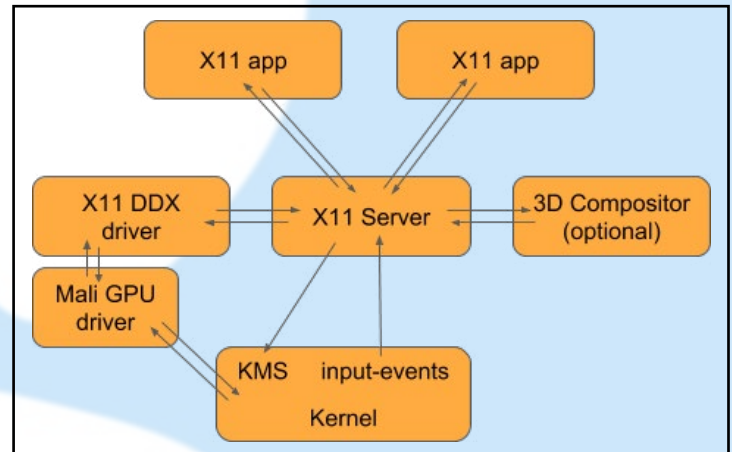
The ODROID-XU4 is the most powerful board manufactured by Hardkernel. The XU4 started its life almost 3 years ago as the more feature-packed but, also more expensive, ODROID-XU3. These boards still top benchmarks and are appealing to users because of features such as a powerful 8-core CPU, OpenGL-ES 3.1, and OpenCL compliant 6-core GPU, gigabit Ethernet, and USB 3.0 capable ports. Its level of performance has allowed many users to replace their daily desktop with an ODROID-XU4. However, there's been one thing that has discouraged a more wide adoption of the XU4 desktop: the desktop environment.

As with most ARM boards, the XU4's GPU provides OpenGL-ES acceleration, a subset of the OpenGL API used for rendering 2D and 3D graphics content. However, most of the Linux desktop environments require the full OpenGL API in order to accelerate the desktop and provide 3D effects. As a result, popular desktop environments either do not run at all, such as Gnome3, or run without acceleration, such as KDE. Typically, Linux distributions provided by Hardkernel or the community are based on 2D-only non-accelerated desktop environments like MATE, LXDE, or Xfce. These environments are coupled with a GPU driver that accelerates OpenGL-ES applications. This solution is sufficient for application-oriented users, for example, Kodi users that are interested mostly in the ODROID as a media center. Unfortunately, this approach is not enough for anyone who wants to use their ODROID for a variety of tasks, or users that want to use their ODROID as a PC replacement.

With the recent developments in software, which include improvements for the Exynos platform in the mainline kernel contributed by Samsung, it is now possible to run the Gnome3 3D desktop environment fully accelerated on the ODROID-XU4. Specifically, patches and additions for the ODROID-XU4 drivers developed by Hardkernel and community, and work done for the Wayland protocol by the freedesktop community, helped make this possible.

Users can download and try Gnome3 on XU4 with the Hipster Stretchy Pants OS image available at <http://bit.ly/2qhFEU2>. However, there are a few caveats. Before we look at them, let's look at the architectural difference between X11 and Wayland.

X11 provides a server to which different clients (applications) can connect. Figure 1 shows a diagram of the X11 system architecture. Events detected by the kernel, such as input events, are sent to

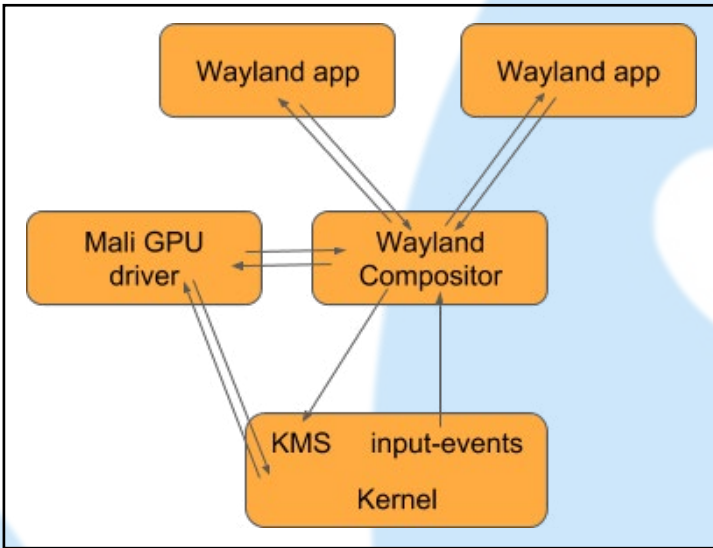


X11 Architecture Diagram

the X11 server, which decides what clients should receive the event. In a 3D environment, because the window on the screen is drawn by a 3D compositor and not by X11 itself, there are issues with passing events, since the X11 server does not know the exact position on screen of the client window which is supposed to receive that event. Also, for OpenGL-ES applications, the X11 server needs a DDX driver component that is able to communicate with the GPU driver. This component, which is not part of the X11 package itself, is dependent on the X11 version. Accordingly, for each X11 version and typically different Linux distributions versions that ship with different X11 versions, Hardkernel needs to provide a specific X11 DDX, a task which makes upgrading and porting to different OS images more difficult than it should be.

Wayland, on the other hand, is actually not a windowing system, but the name of the communication protocol between a display server compositor and its clients. Accordingly, unlike X11, which was the server itself, Wayland is actually the “arrows” shown in Figure 2 between the compositor and the applications. The server that “talks” Wayland, called the Wayland compositor, receives events from the kernel, like the X11 server, but also is able to accurately send these events to the clients, since it can understand the 3D window transformations, unlike X11.

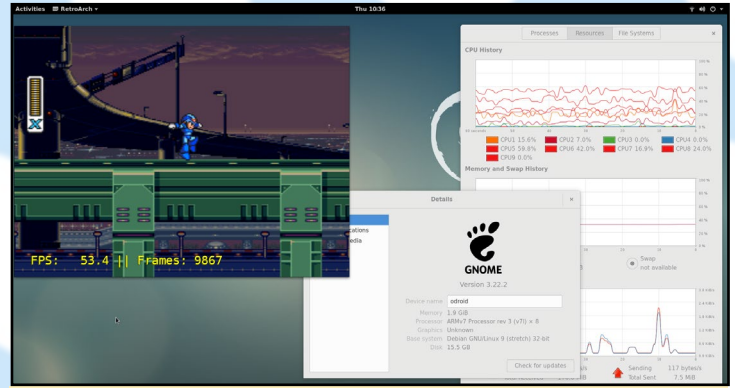
Another difference from X11 is that screen updates are done from within apps, not from the server. This allows only portions of the screen to be updated, where the applications windows are actually being rendered, greatly increasing the performance of the



Wayland Architecture Diagram

desktop environment. With less components than X11 to pass messages to and from, no separate compositor, and no DDX driver, the rendering performance is further enhanced, while maintenance and upgrades are easier to perform.

The biggest caveat for Gnome3 on XU4 at the moment is that, by using Wayland, X11 applications will not run native but through XWayland, a X11 compatibility layer built into Wayland. Since the previous X11 DDX driver is not compatible with XWayland, it also means that X11 OpenGL-ES accelerated applications will fail to run, and require a Wayland compatible version. This may seem

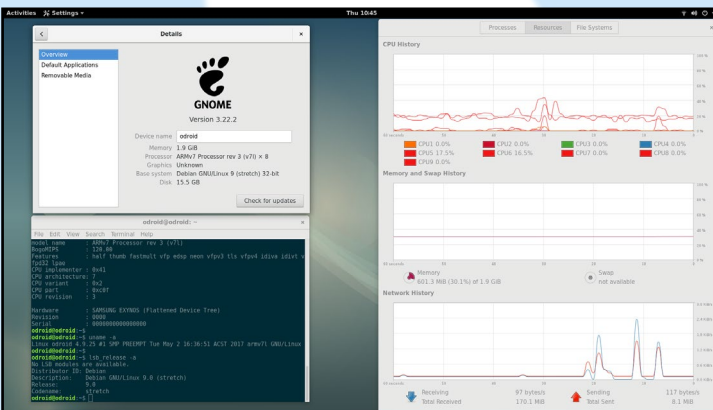


RetroArch is a gaming console emulator that runs very well on Gnome 3

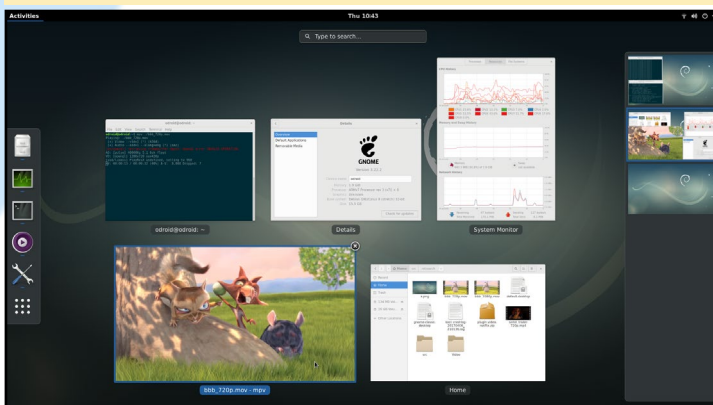
bad, but all GTK3 and QT5 applications are all Wayland compatible, and there have been efforts to port popular applications such as Kodi and retroarch to Wayland.

Currently, Kodi 18 alpha and retroarch 1.5.0 work well on Wayland on ODROID-XU4. LibreOffice is in the process of being ported to Wayland, but in the meantime, it runs very well in XWayland, since it's not OpenGL-ES accelerated. Firefox and Chromium also work well in XWayland, however OpenGL-ES acceleration will not work in Chromium until Google releases a native Wayland version. The GTK3 Epiphany browser is able to run natively under Wayland, however at the time of writing this article, the Mali GPU Wayland driver (r12p0) had a bug which resulted in Epiphany not being able to render OpenGL-ES content. Thus, the second biggest caveat at the moment is the driver quality.

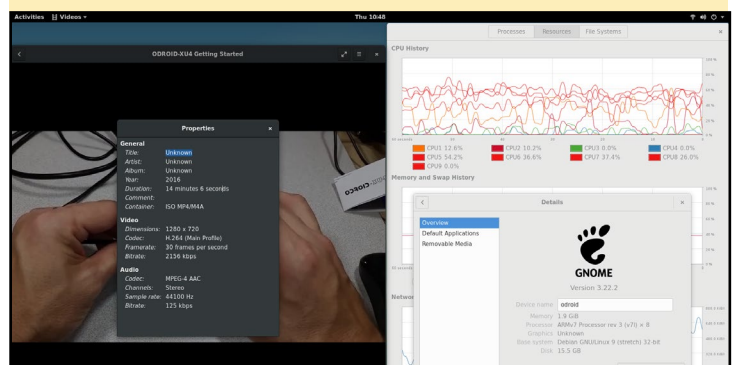
There are still some OpenGL-ES applications such as Gnome Maps and Totem media player that may crash and logout the user from the desktop. Wayland is a new technology, and early adoption is expected to bring bugs and crashes. The only fix for these two issues is time: time for applications to be ported to Wayland, and time for bugs to be fixed. Users of Gnome3 on Wayland are encouraged to report issues so that fixes can be developed faster. The Hipster image currently uses Gnome 3.22, and utilizes only a few patches to make Gnome compatible with the ODROID-XU4. Gnome 3.18 required more than 10 patches, and the upcoming Gnome 3.24 requires only 2 patches. With the power of open-source, very soon we'll be able to run Gnome3 on the ODROID-XU4 with no patching required at all.



Above - Gnome desktop using Wayland
Below - Watching a movie in windowed mode



Youtube videos render smoothly using a Wayland-based browser



GETTING KERNEL BOOT AND PANIC MESSAGES WITH NETCONSOLE

MONITOR YOUR SYSTEM EVEN WITHOUT A SERIAL CONSOLE

by Adrian Popa

Netconsole is the kernel fiddler's best ally. The kernel will panic, but you will certainly not!

There are times when you may want to see what happened to a kernel when it booted or when it crashed unexpectedly. The best option is to have a serial console attached, but if you do not have one, or if you are investigating an issue which may take a long time to reproduce, you can use the netconsole feature. Netconsole is a kernel module that relays kernel messages, which are usually visible in dmesg, to an external system through syslog.

For ODROIDs, netconsole is compiled as a module and is included in the following kernel versions:

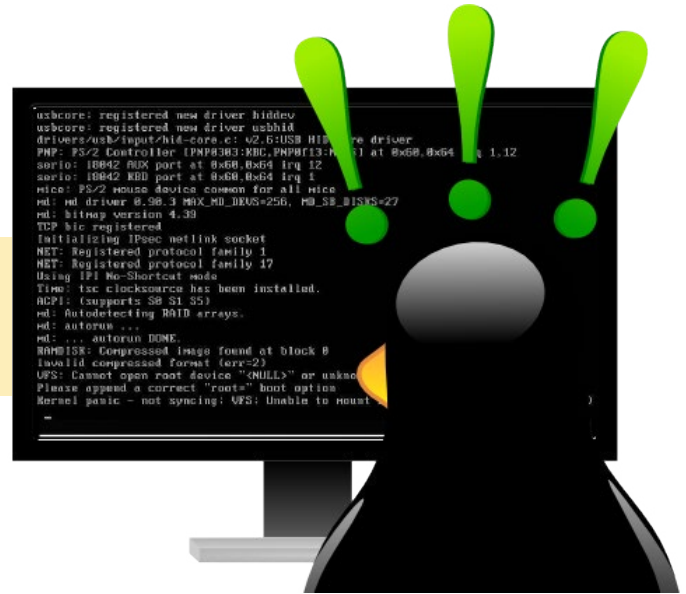
ODROID-C1: 3.10.104-182
ODROID-C2: 3.14.79-112
ODROID-XU3/4: 4.9.30-41

This feature is very useful to have when you experience system lockups without any information in the local logs, since a kernel panic usually is not written to disk to prevent data corruption.

When the kernel starts and the netconsole module is loaded, it needs to be configured with the local system's IP address, local port, interface name and the remote system's IP address, port and MAC address. All this information is needed for the netconsole to work without much network support, which means it is able to send UDP packets even when the network interface is not configured (but it does need to be up).

Transmitter configuration

Let us assume the following addressing for the transmitter and receiver:



TRANSMITTERIP=192.168.228.10
TRANSMITTERPORT=6666
TRANSMITTERINTERFACE=eth0
RECEIVERIP=192.168.228.1
RECEIVERPORT=6666
RECEIVERMAC=44:8A:5B:56:37:5A

In the following examples, replace the names like \$TRANSMITTERIP with the values applicable for your setup. You can find out the mac address of the receiver (it needs to be in capitals!) with the following command:

```
$ arping -c 1 $RECEIVERIP | grep 'reply' \
| cut -d '[' -f 2 | cut -d ']' -f 1
```

The netconsole configuration can be done as part of the boot parameters in /media/boot/boot.ini, or it can be done when the module is loaded. The only difference is that if netconsole is built into the kernel, it would output sooner than when it is loaded as a module, but for our purposes we will configure it as a standard module.

To enable netconsole to be loaded at boot, add it to /etc/modules-load.d/modules.conf:


```
echo 'netconsole' | sudo tee -a \
/etc/modules-load.d/modules.conf
```

Module configuration goes in `/etc/modprobe.d/netconsole.conf`:

```
echo "options netconsole \
netconsole=$TRANSMITTERPORT@$TRANSMITTERIP/\
$TRANSMITTERINTERFACE,$RECEIVERPORT@$RECEIVERIP/\
$RECEIVERMAC" | sudo tee \
/etc/modprobe.d/netconsole.conf
```

It should look like this:

```
options netconsole netconsole=6666@192.168.228.10/eth
0,6666@192.168.228.1/44:8A:5B:56:37:5A
```

Receiver configuration

On the receiver side, you can use an existing syslog server to log the data to a separate log file. It is best if this receiver is always turned on in order not to miss any messages. This tutorial assumes you are running rsyslog, which comes default with Ubuntu.

You will first need to add some custom configuration to rsyslog in order to listen to port 6666 and write data to a log file called `/var/log/netconsole`:

```
$ cat /etc/rsyslog.d/netconsole.conf
$ ModLoad imudp
$ RuleSet remote

/var/log/netconsole.log
& stop

$InputUDPServerBindRuleset remote
$UDPServerRun 6666

$RuleSet RSYSLOG_DefaultRuleset
```

This log file needs to be created and have the right permissions, and afterwards you can restart rsyslog to listen for new messages:

```
$ sudo touch \
/var/log/netconsole.log
$ sudo chown syslog:adm \
/var/log/netconsole.log
$ sudo service rsyslog restart
```

In order for the logs not to get out of hand, it is best if they

are rotated periodically. For this you can use logrotate:

```
$ cat /etc/logrotate.d/netconsole
/var/log/netconsole.log
{
    copytruncate
    rotate 30
    daily
    missingok
    dateext
    notifempty
    delaycompress
    compress
    maxage 31
    postrotate
        invoke-rc.d rsyslog reload > /dev/
null
    endscrip
}
```

Now you're ready to go. If you restart your transmitter ODROID device, you should see the boot log in the `netconsole.log` file:

```
May  9 09:18:22 192.168.228.10 [  0.000000] Ini-
tializing cgroup subsys cpuset
May  9 09:18:22 192.168.228.10 [  0.000000] Ini-
tializing cgroup subsys cpu
May  9 09:18:22 192.168.228.10 [  0.000000] Ini-
tializing cgroup subsys cpuacct
May  9 09:18:22 192.168.228.10 [  0.000000] Linux
version 3.14.79-112 (root@a53_b1) (gcc version 5.4.0
20160609 (Ubuntu/Linaro 5.4.0-6ubuntu1~16.04.1) ) #1
SMP PREEMPT Mon May 8 11:19:14 BRT 2017
May  9 09:18:22 192.168.228.10 [  0.000000] CPU:
AArch64 Processor [410fd034] revision 4
May  9 09:18:22 192.168.228.10 [  0.000000] sec-
mon: share mem setup
...
```

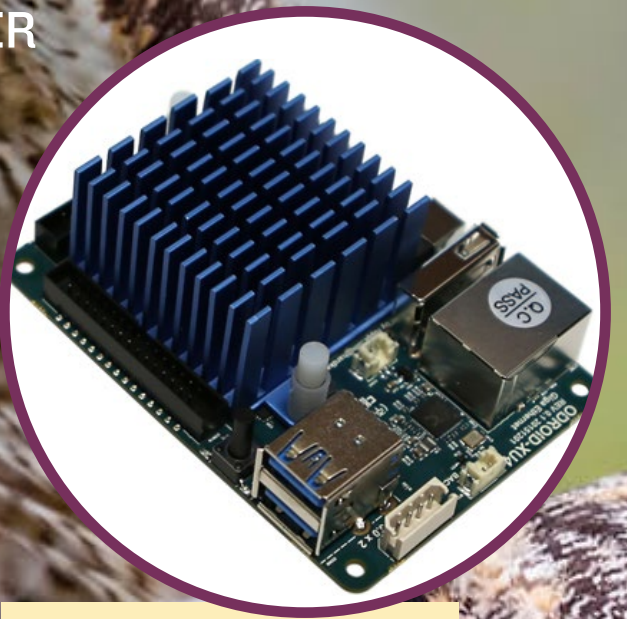
The nice thing is that netconsole will send the whole dmesg buffer when you boot, even if it is initialized later on. The only risk is that if you have an unbootable kernel that cannot load the root filesystem for instance, netconsole will not be of, as it will not initialize. However, for random crashes after boot, it is a must! In case the module can't load, there is likely a configuration problem, and you will see a relevant error message in dmesg.

You can find out more information about netconsole at <http://bit.ly/2r7A1gi>, <http://bit.ly/2r7PtZN>, and <http://bit.ly/2r7SEQS>.

ODROID-XU4Q (Q FOR QUIET)

A COMPLETELY SILENT VERSION OF THE POWERFUL ODROID-XU4 OCTA-CORE COMPUTER

edited by Rob Roy



The ODROID-XU4Q comes with a heat sink pre-installed instead of a fan

Hardkernel, in response to user feedback for silent operation, has introduced a variant of the popular ODROID-XU4, known as the ODROID-XU4Q. Q stands for quiet, which means that it comes with a passive cooler instead of the original cooling fan. Updated plastic cases are also available, which comes with a perforated cutout to make it much easier to create the opening for the tall heat sink.

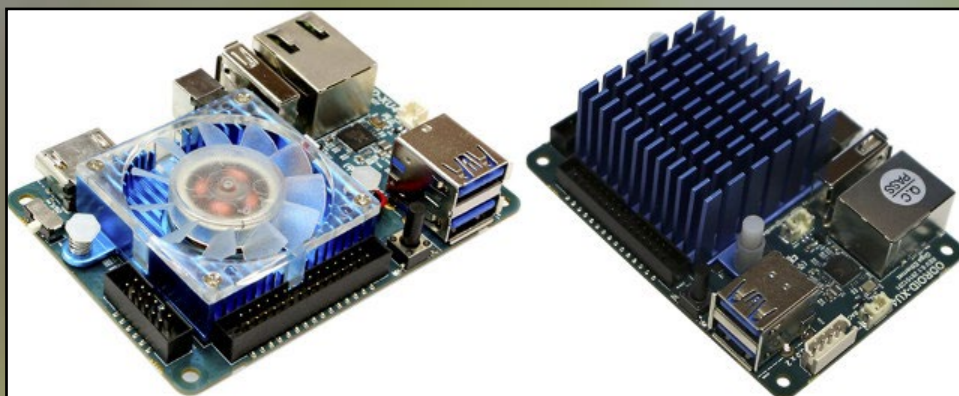
If your application doesn't heavily utilize all eight cores, the passive cooling doesn't cause much thermal throttling. If your application really needs to use full resources, the passive cooling may decrease the performance around 10-20% due to throttling. If your application needs the extra 10-20% of computing power, you will need to use the original ODROID-XU4 with active cooling.

We have already seen the ODROID-XU4 development board price drop to USD\$59 earlier this year, but a frequent complaint about the board was that it requires a fan to operate at full speed, which creates a small amount of noise while the fan turns. To address this issue, Hardkernel offers the ODROID-XU4Q board with exactly the same specifications: a Samsung Exynos 5422 octa-core processor, 2GB RAM, eMMC module support, Gigabit Ethernet, USB 3.0, and HDMI 1.4, except that the fan is replaced by a large heatsink.

Hardkernel has also thoroughly tested both versions in different configurations (<http://bit.ly/2s2ELQK>), such as setting the maximum frequency to 1.8 or 2.0 GHz, and found the ODROID-XU4Q to be slightly slower under high load due to CPU throttling, since the large heat sink does not cool quite as well as the smaller heat sink in combination with a fan. How-

Updated case options for the ODROID-XU4Q





The original ODROID-XU4 (left) vs the new ODROID-XU4Q (right)

ever, in many cases, the difference is minimal, as shown by Antutu results (61,112 vs 60,283 points). Running sysbench at 2.0 GHz showed one of the biggest gaps in performance, with the XU4Q taking 16% more time (420 seconds vs 362 seconds) to complete the tests, as shown in Figures 4a and 4b.

If you run the board at 1.8 GHz, the difference decreases to just 6%. If you are using the ODROID-XU4 as part of a build farm, you may want to keep using the actively cooled version. For example, it takes 25 minutes to build the Linux kernel on the original ODROID-XU4 with cooling fan, but takes 30 minutes on ODROID-XU4Q. If you don't want throttling at all, you need to set the CPU frequency to 1.2 GHz. Hardkernel tested the XU4 and the XU4Q using three categories of benchmarking utilities, with the results shown below:

Sysbench

Sysbench is used to investigate and compare thermal and CPU frequency performance. Overall, the running time increased by 16% at 2GHZ, and 6% at 1.8GHZ when using passive cooling. It was tested using 100,000 iterations, as shown below.

```
$ sysbench --test=cpu --cpu-max-prime=100000 --num-threads=8 run
$ echo 1800000 > /sys/devices/system/cpu/cpu4/cpu-freq/scaling_max_freq
$ cat /sys/devices/system/cpu/cpu4/cpufreq/scaling_max_freq 1800000

#!/bin/sh

echo "Temp, FreqCPU0, FreqCPU4, FreqCPU5, Freq6CPU, FreqCPU7 "
while true :
do
    t=`cat /sys/devices/virtual/thermal/thermal_zone0/temp`
    t1=$(( $t/1000))

    f0=`cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_cur_freq`
    f0_r=$(( $f0/100000))

    f4=`cat /sys/devices/system/cpu/cpu4/cpufreq/scaling_cur_freq`
    f4_r=$(( $f4/100000))

    f5=`cat /sys/devices/system/cpu/cpu5/cpufreq/
```

```
scaling_cur_freq`
f5_r=$(( $f5/100000))

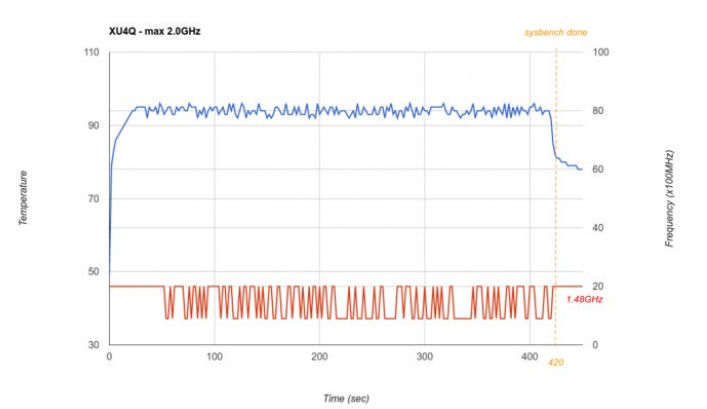
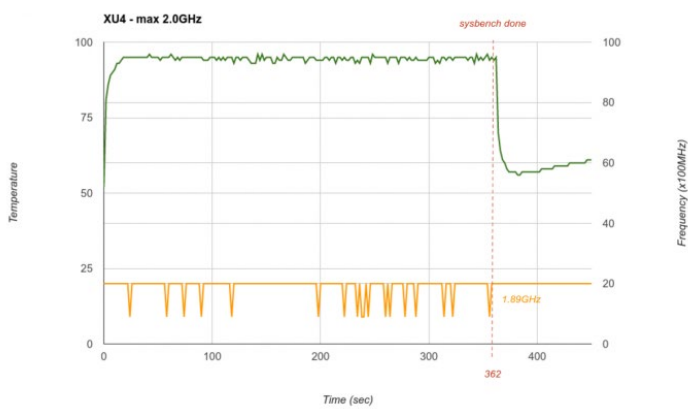
f6=`cat /sys/devices/system/cpu/cpu6/cpufreq/
scaling_cur_freq`
f6_r=$(( $f6/100000))

f7=`cat /sys/devices/system/cpu/cpu7/cpufreq/
scaling_cur_freq`
f7_r=$(( $f7/100000))

echo $t1, $f0_r, $f4_r, $f5_r, $f6_r, $f7_r

sleep 2
```

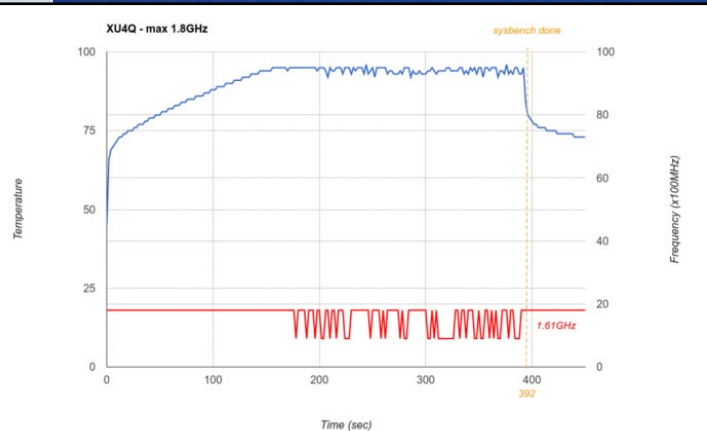
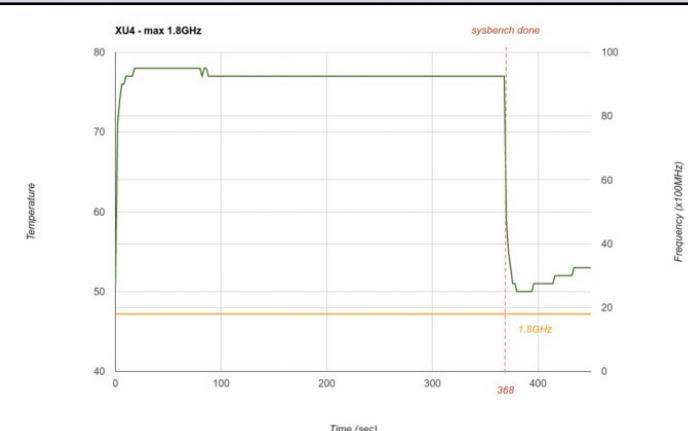
Sysbench test results when the maximum CPU frequency was set to 2.0GHz.



Sysbench results at 2.0 GHZ with the ODROID-XU4 (left) and ODROID-XU4Q (right)

Running Time: about 16% increase - XU4Q / XU4 = 420 / 362 = 116%
Average CPU Frequency: about 22% loss - XU4Q / XU4 = 1.48 / 1.89 = 78%

Sysbench test results when the maximum CPU frequency was set to 1.8GHz.



Sysbench results at 1.8 GHZ with the ODROID-XU4 (left) and ODROID-XU4Q (right)

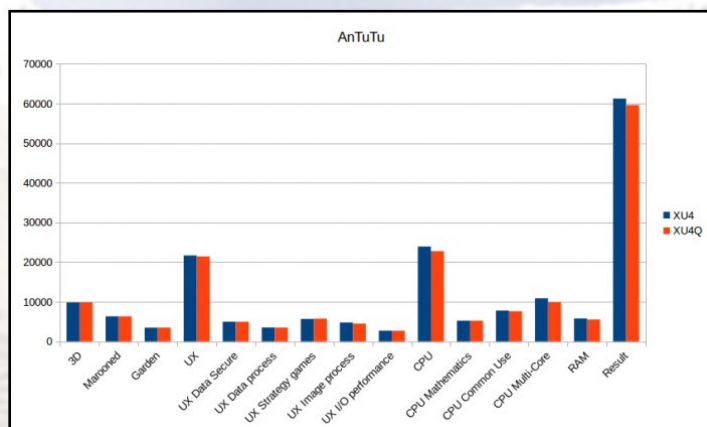
Running Time: about 6% increase - XU4Q / XU4 = 392 / 368 = 106%
Average CPU Frequency: about 11% loss - XU4Q / XU4 = 1.61 / 1.8 = 89%

AnTuTu

The Android app AnTuTu examines a variety of categorical performance metrics, and the difference between the XU4Q and the XU4 is only 2% reduction.

Operating system: LineageOS-14.1 Android 7.1.1 Nougat

Resolution: 1280x720



AnTuTu score metrics graph for the XU4 and the XU4Q

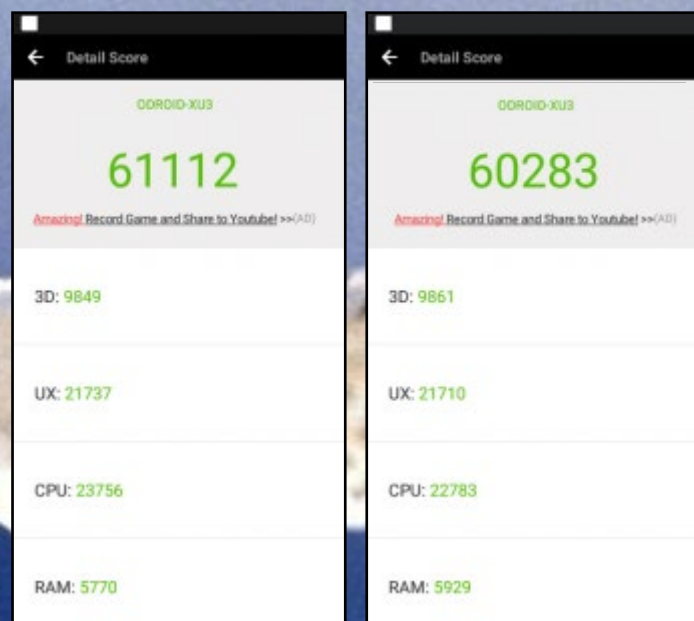
Kernel build

Building a Linux kernel is a good way to apply the highest amount of stress to a computer. The XU4Q increased the build time by about 5m 17s, which represents a 20% increase in execution time. In most situations that are not as stressful as a kernel build, the differences are approximately 10% or lower.

```
$ make odroidxu3_defconfig
$ time make -j8
```

If you have a very CPU-intensive application and need to get the most performance out of your ODROID-XU4, it is recommended to use active cooling, which comes with the original XU4 model. However, if your ODROID-XU4 is being used in situations where the absolute highest performance is not needed, such as web browsing, video rendering, gaming, image editing, or other real-time user applications, the ODROID-XU4Q offers nearly identical octa-core computing power with the additional feature of 100% silent operation.

To purchase the new ODROID-XU4Q, please visit the Hardkernel store at <http://bit.ly/1KhFr6d>.



AnTuTu benchmark results for the ODROID-XU4 and ODROID-XU4Q, respectively

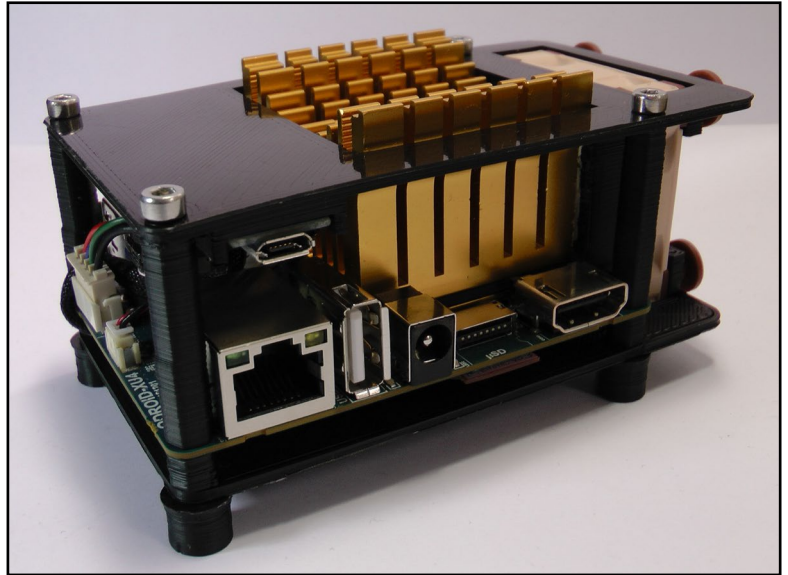
Category	XU4	XU4Q	%
3D	9823	9843.5	0.21
Marooned	6337	6347	0.16
Garden	3486	3496.5	0.30
UX	21681	21443	-1.11
Data secure	4976.5	4975	-0.03
Data process	3517.5	3496.5	-0.60
Strategy games	5677.5	5751.5	1.29
Image process	4787.5	4505	-6.27
I/O performance	2722	2715	-0.26
CPU	23933.5	22774	-5.09
Mathematics	5245	5248.5	0.07
Common use	7798	7621	-2.32
Multi-Core	10890.5	9904.5	-9.96
RAM	5791	5543	-4.47
Result	61228.5	59603.5	-2.73

AnTuTu score metrics for the XU4 and the XU4Q

XU4 CASE

A FUTURISTIC DESIGN FOR YOUR 3D PRINTER PART 2 - PASSIVE VS ACTIVE

by Edward Kisiel



With community input and encouragement, the many changes to the initial prototype of the XU4 Split Airflow Case, covered in the April ODROID Magazine at <http://bit.ly/2qZOGsX>, have been completed with the following improvements:

- Twin enclosed upper and lower ducts with repositioned fan
- EMMC thermally isolated with anytime access
- Case-back heatsink hold-down system
- Fan cable conduit
- UART placement on either side (default above ethernet)
- Recessed screw heads (optional)
- Parametric case with variable height standoffs

The tight tolerance in the upper and lower duct significantly improved airflow through the case. Both the top and bottom ducts work great, and you can feel the airflow coming out much stronger than before the changes.

The idea behind the case-back hold-down system was to provide a more secure heatsink mounting option. There are lots of people experimenting with different heat sinks, but the location of the XU4 mounting holes and the nearby components limits the options. The original spring pins can still be used,

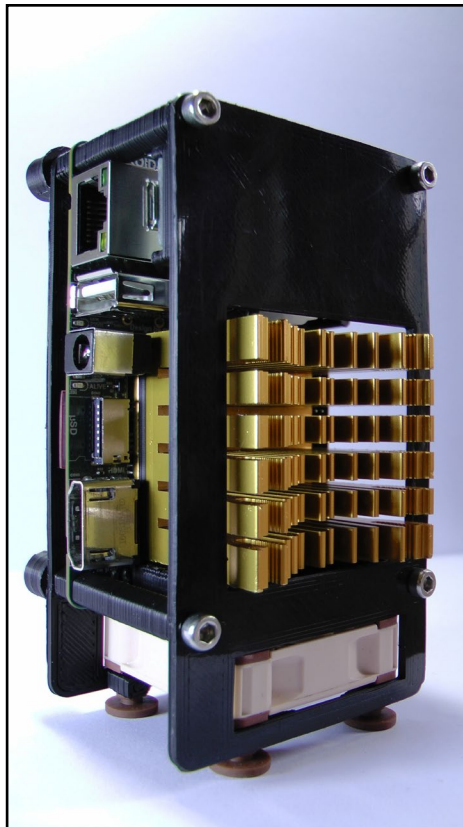


Figure - 1 XU4 Split Airflow Case

but it also provides an upgraded, safer mounting option whether using the stock cooler or experimenting with different heatsinks. Through-bolting from the case bottom, with integrated M3 nuts in the top of the slides, spreads the load across the majority of the PCB. The bottom duct, which surrounds the SOC and support chips, along with a few well placed stanchions, provides rigidity as well as bottom ventilation.

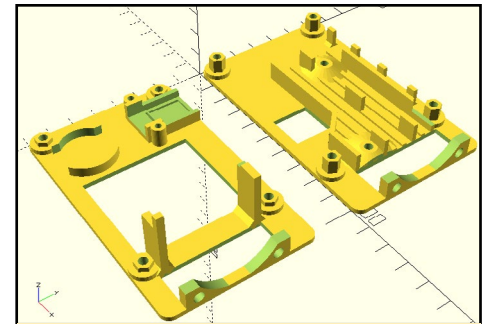


Figure - 2 XU4 Split Airflow Case Design [OpenSCAD Model]

The advantage of this case, from a cooling perspective, is in higher ambient temperature environments and passive and semi-passive configurations, where a well ventilated case with a large heatsink is needed. A cooler overall running system is bound to be more reliable, especially the more demanding the application and environment is. The case-back heatsink hold-down makes new and current cooling options more practical. With that said, adding a copper shim and perch was the next change since the heat sink stays put once tightened down.

Using a copper perch came from the premise that thermal flow can be accelerated thru the shape and mass of a copper slug (2-5mm in height). In essence, it acts as a regulator between the SOC and the aluminum heat sink, leveraging the unique properties of both materials. Bi-metal heat sinks have been in use for a while. Much of what I have seen has been a shim, thin and not very substan-

tial. By having a much thicker piece and less breadth bearing on the heatsink, the rate of flow may be able to be increased to the extent that thermal equilibrium is reached below throttling temperatures. A second premise, that by raising and ventilating under the heatsink, there would be an increase in the cooling area of the SOC. The edges, pins and the bottom of the SOC are directly cooled by the airflow. Additionally, the copper shim and perch are also receiving direct airflow which likely improves efficiency. The increased thermal conveyance and ventilation should have overall positive results. Of course, testing is in order.

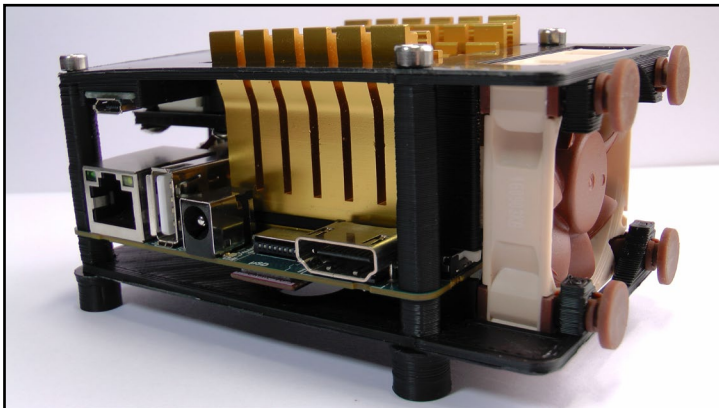


Figure - 3 Side View

Active air cooled benchmark

We evaluated seven different copper shim/perch configurations: five were monoliths and two were stacked configurations. Comparisons were done on both sysbench prime 100K and 300K for short and long term thermal trend and stabilization evaluation. The preparation of the test pieces were as follows: each copper piece was cut and the corners and edges were deburred and chamfered. The faces were sanded with 320, 800 abrasive and finished with #0000 steel wool. All of them were mounted with Antec Formula 6 thermal paste (5.3w/m-k).

Copper Test Pieces						
	Description	Weight (g)	Width (mm)	Depth (mm)	Height (mm)	Orientation
P0	NA	0	0	0	0	NA
P1	3 x 20ga	5	14.5	15.75	2.5	stacked
P2	1 x Rectangle	9	22	16	3.18	Typical
P3	1 x Rectangle	16	14.5	38	3.18	Typical
P4	L Shape	21	29/22	28	3.18	Long side air access
P5	T Shape	24	38/22	16/27	3.18	Long side at rear
P5r	T Shape	24	38/22	16/27	3.18	Rotated 180
P6	Perch + P4	28	29/22	28	6.36	Stacked 14.5x15.75 + P4

Figure - 4 Copper Test Pieces

Even with the improved case-back heatsink hold-down system, the testing was limited by the size and shape of the piece that could be effectively mounted. The longer the piece and the more distant from the SOC, the more unstable and un-

mountable the test piece became. The main cause was insufficient bonding between the copper and the aluminum heatsink. This could be overcome by mechanical fasteners, but that modification was outside the scope of this project. The poor performance of the P5 and P5r was a direct result of poor bonding furthest from the SOC. Other shapes were created, but were not tested because of poor mounting.

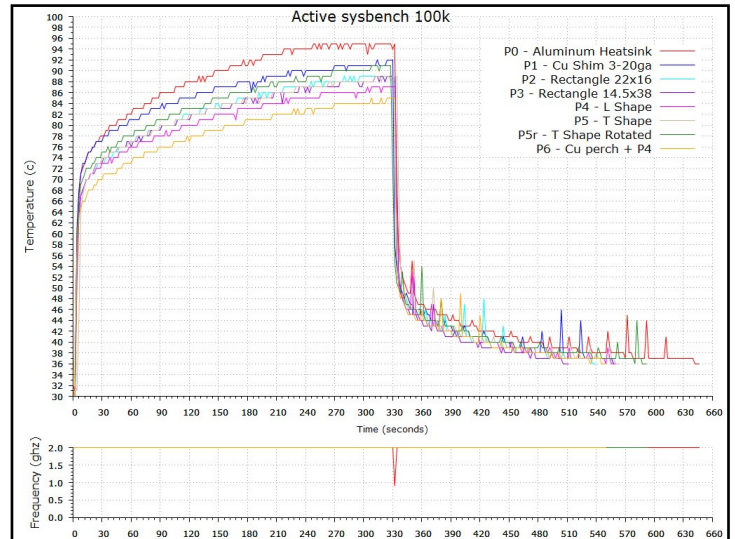


Figure - 5 Active 100k Chart

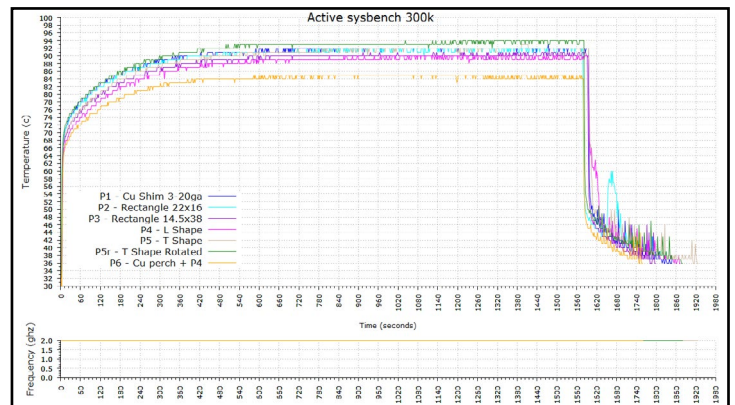


Figure - 6 Active 300k Chart

The other influencing factors were the mass of the piece and the ability of airflow to penetrate under the heatsink. The results of the P3 (dark-violet) and P5 (grey) suffered in part because of the inability of air to get under the heatsink to aid cooling. The test piece P6 (orange), when mounted, provided a gap of over 7mm between the PCB surface and the bottom of the heatsink for airflow. I believe this improved airflow and quick heat-flow away from the SOC, and aided in its superior performance over all of the other test pieces. Due to its restricted perch size, a larger heat differential was maintained which accelerated heat flow to the secondary stage (P4 copper piece), and ultimately to the aluminum heatsink. In essence, it formed a 3 stage cooling pipeline, the first with a short saturation that feeds a larger copper mass that accumulates in proportion to its mass and saturation point. This allows a quick

place to draw heat away from the SOC and perch, while accommodating the aluminum heat sink's slower absorption rate.

Passive vs active cooling comparison

There is always a lot of interest in running an SBC in either a semi-passive or fully passive heatsink mode. For comparison, the P6 copper shim/perch configuration was run passively on the same OS/kernel-setup as the previous actively cooled tests. The Split Airflow Case was in the vertical position for all tests and the ambient temperature was between 69 deg F (20.5 deg C) to 70 deg F (21.11 deg C). The vertical orientation really seemed to work well in a passive configuration and performed likewise. While running, the heatsink was hot with lots of warm air noticeably exiting the top of the case.

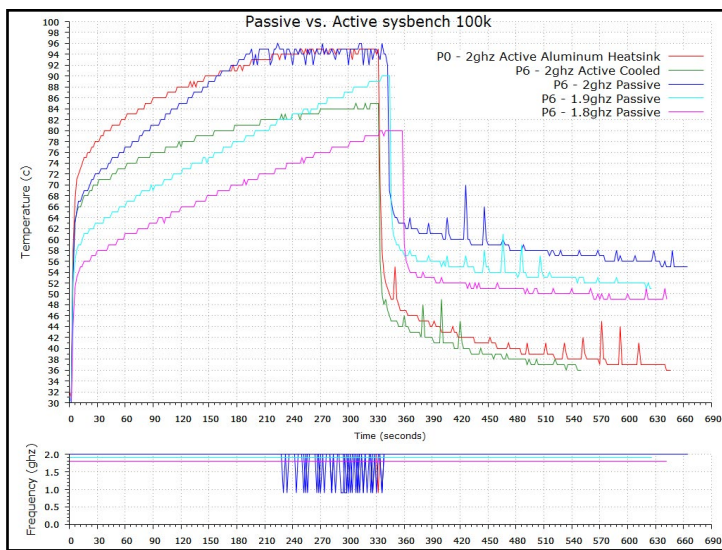
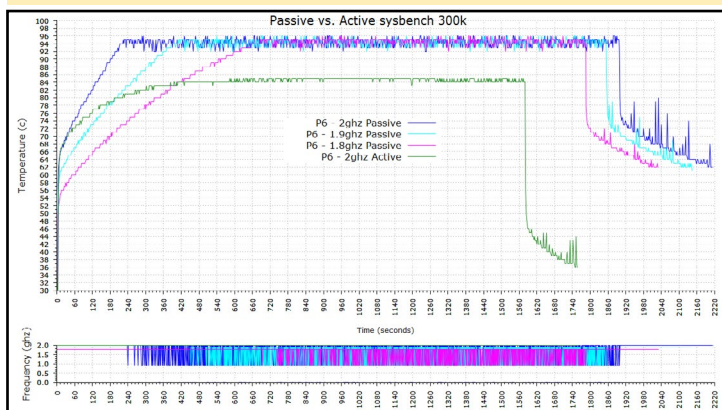


Figure - 7 Passive 100k Chart

Included in the comparison chart was the actively cooled P0 Aluminum heatsink with no copper shim and the actively cooled P6 copper shim/perch configuration. I was impressed that the passive P6 copper shim/perch thermally outperformed the actively cooled P0 Aluminum heatsink and gave a similar result with little performance loss.

As shown in the Passive sysbench 300k Chart above, for

Figure - 8 Passive 300k Chart



passive configurations there was clearly a thermal and performance difference for the 1.8ghz test further out in time. If one only looked at the sysbench 100k test, a different conclusion might have been reached.

Using the case-back hold-down

I ran the sysbench prime 100k every time I remounted the heatsink as a standard practice to verify proper installation. A few times during testing the many heatsink-shim mountings, the heat sink was mounted slightly crooked. A couple of times it was obvious, when sysbench was run, because the temperatures would immediately start to jump by 4-6 degrees at a time. At other times, the temperature overall was just higher, with the rise being faster than expected. In one instance, everything looked good until the temperature reached 80 deg C, when thermal runaway occurred. It started to jump many degrees at a time and hit 104 deg C before I was able to shut it down.

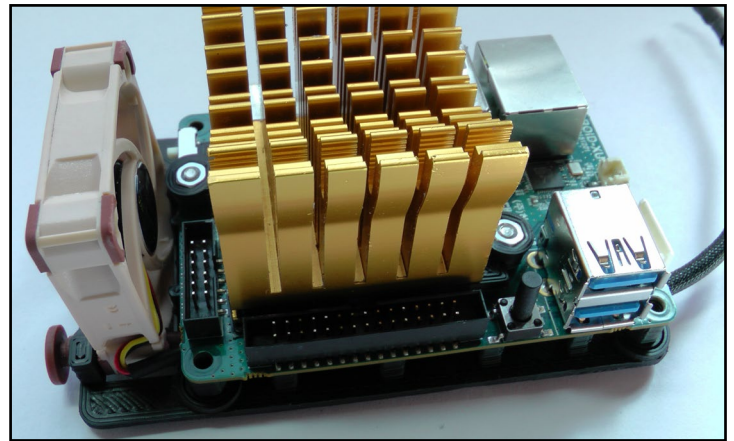


Figure - 9 A closeup of the hold-down

During the few times where there was an issue, simply loosening and redoing the heatsink tightening process fixed the incorrect mounting. It doesn't have to be off very much for a problem to develop. For those who are going to try different heatsink-shim configurations, I would strongly recommend adopting this practice and run sysbench 100k to verify a proper installation before general operation. Pay close attention when you tighten the heatsink down: keep it level and do not overtighten it. I use the edge of the 30 pin connector as a reference to check if the heatsink is level when complete. If something seems off, redo it to ensure proper mounting.

Moving forward

Even though the goals of the XU4 Split Airflow Vertical Case project were reached, there is still more room for improvement. I now think that sustained temperatures in the 70's at 2GHz might be possible. Some ideas are:

- Remove, replace or cut away I2S plastic housing for vastly improved air flow under the heatsink and around

the SOC. A ribbon/wire cable could also be directly soldered and run out the conduit hole meant for the lower duct fan wire. I believe this change could be very fruitful

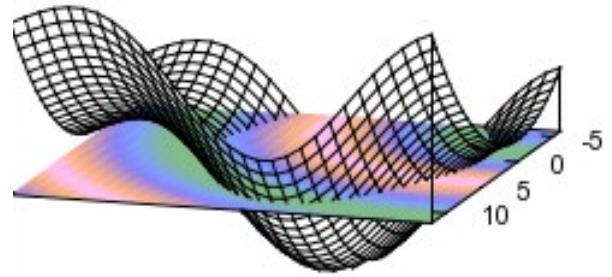
- Capping the heat sink opening or completely enclosing the duct past the heat sink to improve airflow and heat sink efficiency
- A single-point, top hold-down system to address the offset SOC position and improve heat sink mounting options
- A more efficient copper perch/shim design; the possibilities and testing is far from complete in this area
- A custom heatsink and duct
- Adding a stronger fan to induce more airflow, which would be a very easy improvement
- Better Thermal compounds

All of the community input on the ODROID forums paid off, as indicated by the great results. Whether you run an active, passive or semi-passive system, this case and shim/perch configuration can expand your thermal and performance envelope. One lesson learned is that, regardless of the case you have, if you can get a piece of copper under your aluminum heatsink, it can help. Thank you to Ameridroid for making the XU4 Split Airflow Case available to the community through their online store. You can read more about it at the online forum post <http://bit.ly/2rIqacl>. The XU4 Split Airflow Case OpenSCAD design file, the stereolithography file and the GNUPlot scripts are also available for download from the link.

GNU PLOT

CREATE MULTI-FILE CHARTS

by Edward Kisiel



GNUplot is an open source portable command line driven plotting and graphing tool which is available for Linux as well as several other operating systems. This is a quick guide intended to give you an overview on how to create a chart of various systems statistics. GNUPlot version 4.6 is the current stable version, but does not have all of the command line features found in the latest version, GNUplot 5.x. For those who are interested in charting multiple test files, the GNUPlot scripts used to create these charts are available online in the ODROID forum at <http://bit.ly/2qSsgcp>. There are two GNUPlot scripts: `multifile_temp.gpl` and `multicore_temp.gpl`. The scripts can be used on either GNUplot version by simply editing the file and enabling some text variables for the file names; refer to the comments in the files for more details. It can plot up to 8 test files from `watchtemp.sh` or a similar data source on the same chart. The data is formatted as “count, temp, freq1, freq2, freq3, freq4”. For quick command line charting, if a legend’s title is not specified, the name of the file is used by default.

GNUPlot version 5.x use:

```
gnuplot -c multifile_temp.gpl
chart.png testfile1...testfile8
```

GNUPlot version 4.x use:

```
gnuplot multifile_temp
```

The GNUPlot script `multicore_`

`temp.gpl` will plot system temperature, individual A15 core temperatures, GPU temperature and frequency for a single test file from `watchtemp_all.sh` or a similar data source. The data is formatted as “count,temp,freq4,freq5,freq6,freq7,core4,core5,core6,core7,gpu”.

GNUPlot version 5.x use:

```
gnuplot -c multicore_temp.gpl
chart.png testfile
```

GNUPlot version 4.x use:

```
gnuplot multicore_temp
```

To allow for monitoring of the test in progress, in one shell I run the following command:

```
./watchtemp.sh | tee testfile.txt
```

In another shell I run the test:

```
sysbench --test=cpu --cpu-max-prime=100000 --num-threads=8 run
```

I hope the GNUPlot scripts are useful and become part of a standardized way of collecting and displaying test results for the community. There are still some issues on the collection side to be worked out. The latency in recorded time is skewed in its relationship to the sysbench results which is an area that can be improved. Additionally, a standard means to deal with the difference between operating systems and kernel performance needs to be considered.

LINUX GAMING

EXAGEAR DESKTOP 2.0

by Tobias Schaaf

Some time ago, I wrote an article about ExaGear and how to setup different x86 applications on your ARM system like Skype or TeamViewer. This time, I want to talk about the new ExaGear Desktop 2.0 and how you can use it to play some nice Windows and Linux games on your ODROID-XU3/XU4 at a decent speed.

For those who still don't know about ExaGear, it's a software package that allows you to run x86 programs on your ARM hardware. It's a lot faster than using QEMU or emulation in general, and therefore has some interesting use cases. The new version 2.0 is aimed towards gaming, and is a lot faster than earlier versions.

ExaGear has a new optimized version of Wine (based on Wine 2.0), which improves performance a lot, but doesn't support OpenGL, at least not on ODROIDS, but more on that later. The overall performance has also increased. So even without OpenGL support working, I thought it would be really interesting to try getting some games to work and see how they perform. Since CPU power is important for this scenario, I used my ODROID-XU3 for best performance and tried adjusting everything that I could in order to get the games running.

Setting up the environment

For this experience, I used my ODROID GameStation Turbo image for the XU3/XU4, which is based on Debian Jessie, and installed a Debian Jessie Guest environment for ExaGear.



It should also work with Ubuntu, but I'll concentrate on Debian Jessie and will give some hints in case you use Ubuntu instead. I also assume that you already set up the ExaGear environment, and therefore will start right away telling you how to set up your games. For a step-by-step guide to installing ExaGear, please refer to the ODROID Magazine article at <http://bit.ly/2qfFaOk>. I used a lot of CD images, rather than connecting a CD drive to the ODROID, and for this purpose, cdmu, which is a virtual CD drive for mounting CD images, came in very handy. The cdmu package is pre-installed on my ODROID GameStation Turbo image.

Please note that all the steps below should be done within the ExaGear environment, especially installation of new software like programs and drivers. Installing packages also often requires administrator-level permissions, so it should be safe to assume you run these commands as root inside the ExaGear environment.

Game Sources

Normally I don't do much advertisement, but I can highly recommend www.gog.com as a source for games. They provide a lot of older and indie games that may not require 3D acceleration, both for Windows and Linux. The software is optimized for easy in-

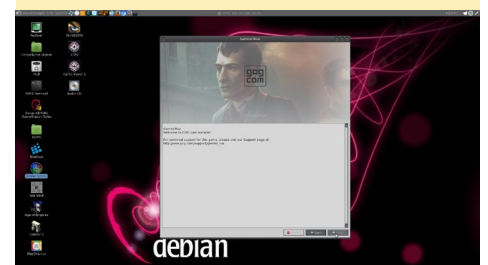
stallation, and has been patched to run on modern operating systems, which is useful for running old Windows 95 or even DOS games using Windows 7. Their Linux software even comes with a graphical installer that allows you to easily install and setup your software. All of the software is DRM free, so a "NoCD" patches are not required, and they are easy to install on your system.

Linux games

There is nothing much that needs to be prepared under Linux, and games that are supposed to run in x86 Linux should be straightforward to install. Just remember that you don't have any 3D support, so any game that requires OpenGL won't work. Even games that are written for OpenGL ES won't work, but hopefully this will change some time in the future.

I installed two games from my GoG library as native Linux applications: Gemini Rue (<http://bit.ly/2qQrH3q>) and Blackwell Deception (<http://bit.ly/2qQrH3q>).

Figure 1 - GoG Linux installer of Gemini Rue with an easy to use graphical interface



ly/2rw0ofs). Although both games are similar, they act a little bit differently.

Gemini Rue runs perfectly fine after some configuration, while Blackwell actually requires OpenGL in some cases in order to work, but let's look into this more deeply.

Gemini Rue

Since this game is available as a native Linux game, it runs out of the box nearly perfectly in ExGear. After installing the game through the graphical installer by downloading the game and running the .sh file with the command "bash <filename>" from the terminal, the game starts in fullscreen mode switching to 640x480 for me, with the picture somewhat distorted.

There is a known issue on the XU3/XU4 that if you change the resolution from within an application to a resolution that is different than the desktop resolution, your picture ends up misplaced and the part that should be all the way to the right is actually all the way to the left, and the left part is in the middle and the middle part on the right. Luckily for us, this can easily be solved either by changing the screen resolution to 640x480 before starting the game (which I don't really like), or by just changing the configuration file of the game.

GoG installs all games in the same folder on our system, which makes it quite easy to find at /home/odroid/GOG Games/Gemini Rue/game/ac-setup.cfg. This actually seems to be the same for all games of Wadjet Eye Games (<http://bit.ly/2qo37mB>), which as a company seems to focus on indie adventure games, which are actually quite nice. It also makes these games rather easy to configure for us.

In this config file we don't need to change much. We set the window mode (windowed) from 0 to 1 and I also set the gfxfilter from StdScale2 to StdScale4. With those settings, I got a nice big window and could play the game



Figure 2 - Playing the indie adventure game Gemini Rue on the ODROID-XU3/XU4 natively under Linux

without any issue. The game ran pretty well for me and I couldn't find any problems with it.

After the configuration, you only need to double click the icon on the desktop and the game will launch just fine. If you want, you can even play the game in fullscreen mode. For this, your TV or monitor needs to support 1280x800 resolution (in StdScale4), and you need to switch to that resolution manually before you start the game as well as making sure that you set "windowed" to 0 in the configuration file.

Blackwell Deception

The entire process of installing and configuring Blackwell Deception is pretty much the same, but this game makes use of OpenGL in some cases, and since ODROIDs do not support OpenGL, the game is rendered in MESA Software OpenGL, which is very slow. In scenes where OpenGL is used a lot, this makes the game drag, but very often when you are inside rooms, the game speed is totally fine. This gives a very mixed experience of ranging between being fully playable in full speed to experiencing scenes where the game plays in slow motion until you exit the scenery.

Reducing the scaling can improve this, and some other games of the Blackwell series are actually working fine by just using 3x scaling, for example, which generally depends on the game. As I mentioned, 3D is not supported, so as long as your games don't use hardware acceleration (for 3D or special effects), you should be fine.

Windows games

Being able to play Windows games on Linux, especially on an ARM board, is a very interesting thing to do, but requires a little bit of preparation first to get things to work properly.

I'm using an application called "PlayOnLinux" for this, since it allows us to install and configure games individually as well as manage different Wine versions simultaneously. You can also use desktop icons rather than using command line tools to get everything to work. It's a very powerful tool which I also use on my Linux laptop to get many Windows games to work under Linux.

Wine

There is a lot of misunderstanding going around what "Wine" is. It stands for Wine Is Not an Emulator.

Many people think you only need Wine to run Windows games and programs on any system, but that is totally wrong (!) and often something that is misunderstood by many people. Wine does NOT emulate a Windows PC. It is a rebuild of the Windows API, which means that if a Windows program has a function that says "open a new window", this function is being translated to Linux, and Linux is told to open a window. This is done for everything a Windows program is doing, like opening a file, drawing a picture, rendering 3D graphics playing sounds, and so on. Since it directly starts a Windows program and just translates the code, the code must actually be able to run on the system itself.

Consequently, the main issue with compatibility is that Windows 32-bit x86 code only runs on a 32-bit x86 processor, and you cannot run an x86 binary on an ARM system. This doesn't work under Linux either. You can not just install x86 Skype directly on ARM and expect it to work, since the binary was not made for ARM.

This situation also applies to Wine. If you install Wine on an armhf image,

you would need an armhf Windows binary to make use of Wine directly. Since there is no such things as an armhf windows binary, just having Wine on ARM is not enough. That's why we use ExaGear, which emulates an x86 environment on ARM, and with that we can now run x86 binaries using x86 Wine.

Installing PlayOnLinux inside ExaGear

The installation is slightly different depending on what OS you use, either Debian or Ubuntu, so here are the steps you need to setup your system:

Debian

```
# activate all OS repos
sed -i "s/main/main contrib non-free/" /etc/apt/sources.list
# update system
apt-get update && apt-get upgrade && apt-get dist-upgrade
# install MS Core Fonts
apt-get install ttf-mscorefonts-installer
# install PlayOnLinux (this will take a long time)
apt-get install playonlinux net-cat gettext wine
```

Ubuntu

```
# activate all OS repos
sed -i "s/main/main multiverse restricted/" /etc/apt/sources.list
# update system
apt-get update && apt-get upgrade && apt-get dist-upgrade
# install MS Core Fonts
wget http://ftp.de.debian.org/debian/pool/contrib/m/msttcorefonts/ttf-mscorefonts-installer_3.6_all.deb
dpkg -i ttf-mscorefonts-installer_3.6_all.deb
# install PlayOnLinux (this will take a long time)
```

```
apt-get install playonlinux net-cat gettext wine
```

The Wine version actually being installed comes from ExaGear itself, which is optimized for ARM and offers very good performance. If you use a Raspberry Pi instead, you can even use an experimental OpenGL driver and have hardware acceleration. Unfortunately, as mentioned before, that does not work on ODROIDs, and in fact actually causes some issues that we need to address later. We can still benefit from the much higher CPU power of the XU3/XU4 compared to the Raspberry Pi, and with that can get some things to work as well on our ODROIDs.

PlayOnLinux

PlayOnLinux is a frontend to manage your "wine bottles", which are separate instances of Wine environments which you can configure individually and even run on different Wine versions. PlayOnLinux helps you with installing the software and drivers needed to run them, as well as configure these installations individually, such as screen resolutions, wine version, and the Windows API version.

From here, you can install appli-

cations easily either by using prebuilt scripts, or by clicking through some simple menus. I can only suggest using PlayOnLinux rather than directly using Wine. You still have the option to directly using Wine as is anyway, but PlayOnLinux offers additional options for you.

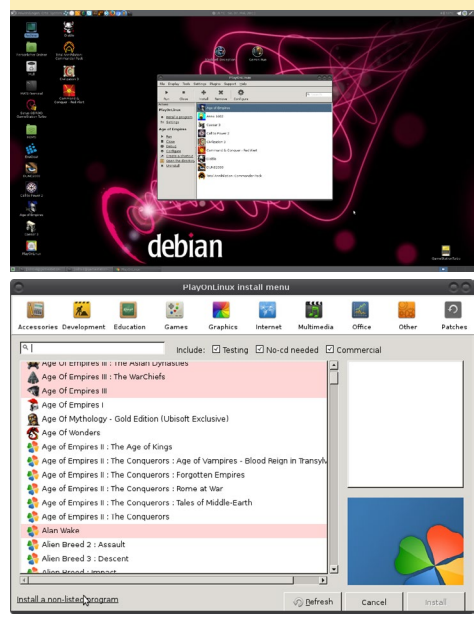
There are some basic stuff that you should know about PlayOnLinux and how to manage your virtual drives.

PlayOnLinux installations

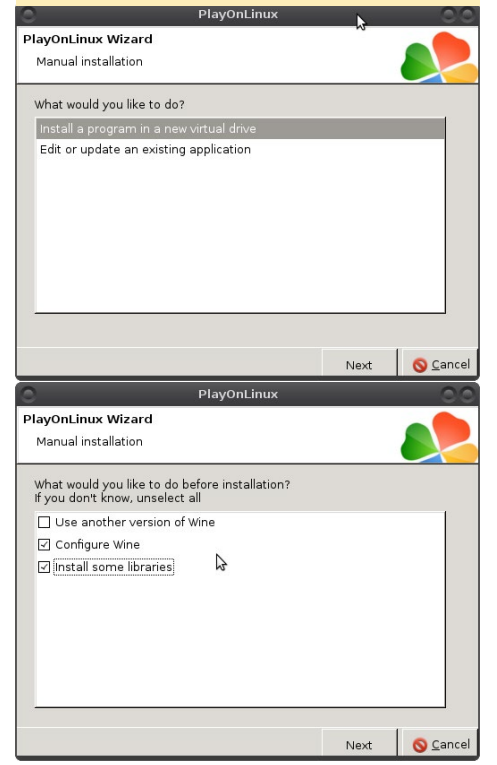
To install a new game or program, you just press the big install button at the top of the menu. A new window will open, and you can select what you want to install. You can install some programs or games from pre-written scripts that guide you through the entire install process very easily, or you can choose to install a non-listed program and select the options manually depending on your needs.

If you install the game from a pre-made script, you often only need to adjust minor things like the Wine version

Figures 3 and 4 - PlayOnLinux on the ODROID for managing your Windows games in wine bottles



Figures 5 and 6 - Installing a game either from pre-made scripts (list on the top) or through setup manager (on the bottom)

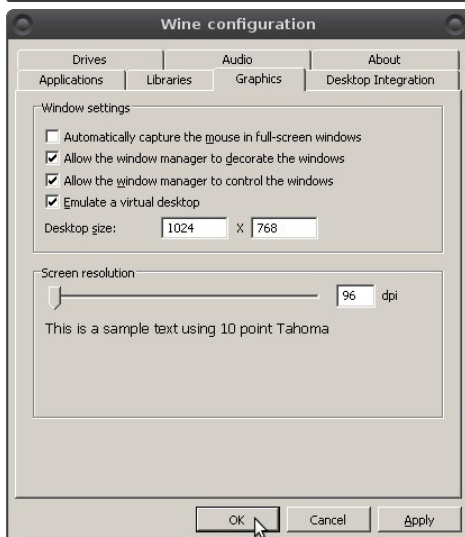
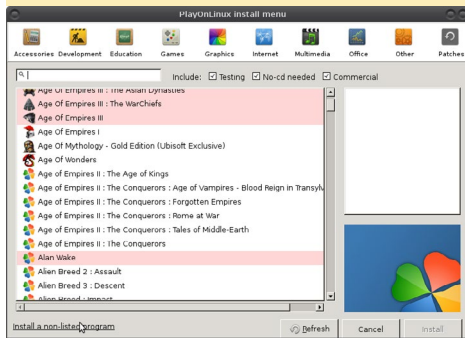


you want to run this on, because these scripts were made with a particular version of Wine which might be very outdated by now. Therefore, you only have to select a different version of Wine and be done after that (see configuration later). If you install a non-listed program, you have to make a couple of choices about how and what you want to install.

The first question is whether you want to install a program in a new virtual drive (wine bottle) or in an pre-existing one. I prefer having individual drives for each game, unless they are tiny games like Bejeweled or something similar, which I normally put in the same virtual drive, at least on my laptop. This makes things easier if you have to configure them differently, and doesn't mess up other already-working games because you are trying to fix a non-working game.

After you create a new drive, or selected an existing one, you are asked if you want to change some options before

Figures 7 and 8 - Selecting the Wine configuration, version and other things before installing a new game/program



you start installing your application. With the “Use another version of Wine” option, you can select a different version of Wine that you want to use to run your application. By default, this is always the System version that comes with your OS. On ExaGear, this is version 2.0 with the ARM optimizations, which should be a very good default. On my laptop I often change this, since even on Ubuntu 16.04, the default is still version 1.6 of Wine, while currently version 2.7 is already available in PlayOnLinux at the time of this article.

The “Configure Wine” option allows you to set up options for Wine, like the API version such as Windows 95 or Windows XP, or if you want to use a virtual desktop instead of running it directly on the Linux desktop. This is probably a very good thing to do, since many games run in a low resolution, and switching resolution on the fly does not work properly in all cases. To select this option, switch to the Graphics tab and enable the “Emulate a virtual desktop” checkbox. Good options to start with are 800x600 or 1024x768, since these are very common resolutions for older games. 1024x768 is actually nice if you run a 1080p resolution, while 800x600 looks really good in 720p.

Through the “Install some libraries” option, you pre-install libraries that you know are required to run your games properly, like “.Net framework”, “DirectX”, or just “dsound”, which are very commonly needed. To start, I suggest just setting up the virtual desktop and being done with it, since everything else can be done later if necessary.

After you're done setting up your environment, the system will ask where it should install from. By default, it will scan for a CD that has an AutoStart on it and ask if it should start this. If it can't find a similar file, it will ask you to select a setup/install file manually. This is where cdmu comes in handy, since it allows you to mount your images from CDs and DVDs from which you can in-

stall your games, but also installing GoG setup files should work fine.

The installation of software is straight forward, same as it would be on a Windows machine. After the installation, PlayOnLinux will automatically scan the new virtual drive for all .exe files that it may use to create a shortcut for you on the desktop. You select the start program of the game and give it a name. PlayOnLinux will repeat the process of you selecting additional .exe files for shortcuts until you say you have enough.

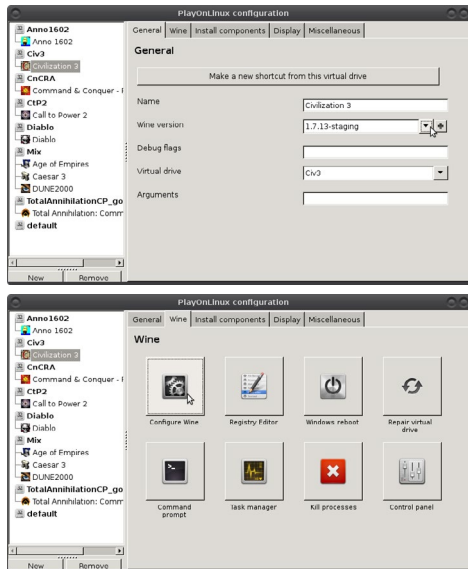
After that, you have a shortcut for your games and programs directly on the desktop as well as in PlayOnLinux itself.

Since PlayOnLinux is a Python-based frontend, ExaGear does not recognize that it needs to start the links in an x86 environment if you double click the icons on the desktop. This can easily be fixed by adding “exagear --” at the beginning of the command line of the icon on your desktop. Just right click it, select properties and type “exagear --” (without quotes) in front of the Command: line of the desktop shortcut. That's all that needs to be done to start the games directly from the desktop.

Configurations

After your installation, it will probably be necessary to occasionally configure your games and programs if something is missing, or you just want to try new settings. To do so, select the game or program you want to configure in your PlayOnLinux main window and click on the Configure button on top of the window. A new window will open on the left side, where you can see all of your Virtual Drives and the shortcuts you created in them. You can switch between them by just pressing on either the game or the name of the virtual drive to configure this particular drive. Please note that Configuration changes are always done for the ENTIRE virtual drive, not only for a single program/game in it, which is why I said it's better to have one virtual drive per game/program.

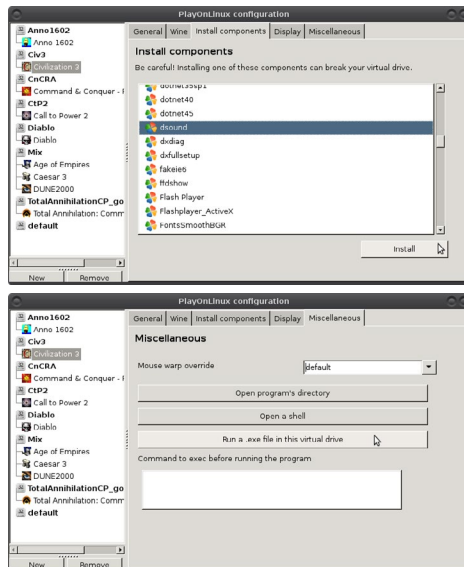
On the “General” tab, you can create new shortcuts in case you missed one or change your Wine version that you want to use with this virtual drive. On the “Wine” tab you can start certain tools for your Wine environment. “Configure Wine” allows you to set up your Windows API version, or virtual desktop settings. “Registry Editor” allows editing of the Windows registry, and “Command Prompt” opens a cmd shell inside your virtual drive from which you can type your Windows/DOS commands. These are probably the only buttons you need, and you might not even need them at all.



Figures 9 and 10 - Change your Wine version, make a new shortcut or configure your Wine environment inside PlayOnLinux

The “Install Components” tab is quite interesting. It allows you to decide what drivers you want to install in your Wine environment. Some might know this from “winetricks”, but this is like a graphical tool for winetricks where you can select what kind of drivers you want to install. If you want to install DirectX, DirectPlay, or even Xvid codecs, they are often just one click away inside the “Install Components” tab.

However, it seems that not all of the components are working 100%. For example, MS Core Fonts and some video codecs install scripts don't seem to work properly, but it's hard to find the ones



Figures 11 and 12 - Install components and drivers needed for your applications and running new .exe files inside virtual drives

that are not working, since most are.

The last tab that's probably important is the “Miscellaneous” tab from which you can choose to run an .exe file inside the virtual drive environment. This can be useful if you want to install patches for certain games.

Installing additional Wine versions

This is another very important feature and you can reach it on the main screen of PlayOnLinux by clicking on “Tools -> Manage Wine versions”. Here you can install and remove different versions of Wine, since not every program may work with the system version of Wine that comes with ExaGear. Some games that you install via pre-made scripts inside PlayOnLinux, will also install their own Wine version, which you might not want to use, so you can just remove them here after you've switched to the version that you want to use. Since you now know the basics, you are ready to play Windows games on PlayOnLinux inside ExaGear, running on your ODROID-XU3/XU4 ARM board.

Windows games on Linux

Not every game you want to run un-

der Linux in Wine will work. Even on my laptop, I often struggle to get some games to work. So If you try a game and it fails to run, don't give up, especially on ODROIDs, where we are very limited due to the missing OpenGL drivers. We have to be a little patient and find the games that do not require hardware acceleration to run properly. To make it easier for you to start, I made a list of games that I got to work, as well as any issues I encountered and how I solved them.

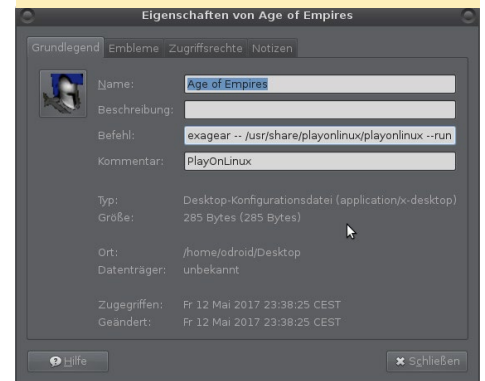
Keep in mind that there are often different version of games out there. I don't always use the CD version to get things to work, or you may have to use a “NoCD” patch. If you feel uncomfortable with these solutions, there are other ways as well, but I often chose the easiest solution.

Age of Empires

Age of Empires is actually rather easy to get to work. If you look around, you will probably quickly find ready-to-use .zip files that you only need to extract and have a fully installed version of Age Of Empires with the expansion pack. What I did is was create a new virtual drive by clicking on “Configure” and then “New” in the lower left corner, created a new drive, gave it the name “AoE1”, and selected “System” as the Wine version.

After that was done, I just extracted the .zip file to \$HOME/.PlayOnLinux/wineprefix/AoE1/drive_c/, then clicked

Figure 13 - Adding “exagear --” to the shortcut to start the game right from the desktop

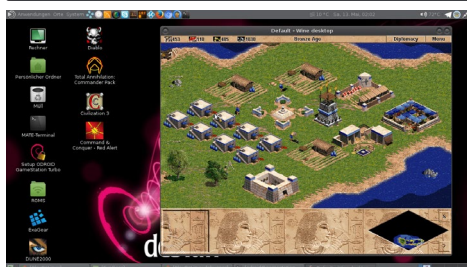


the on the new drive in the configuration screen and selected “Make a new shortcut from the virtual drive”. I chose EMPIRESX.EXE, which is the start binary for the extension pack of Age of Empires 1, and also includes the original missions. I set the name for the desktop shortcut to “Age of Empires” rather than “EMPIRESX”, and altered the desktop icon to start in an ExaGear environment by adding “exagear -- “ in front of the command.

After that, I configured Wine to use a virtual desktop of 1024x768 since it’s the highest resolution the original Age of Empires supports. Under “Install components”, I installed dsound, because without it, you won’t have any sounds at all. If you don’t want to use .zip files that you extract manually, there’s an installer for Age of Empires I – Gold and the original Age of Empires for PlayOnLinux, which should allow you to install your original CDs. Unfortunately, I wasn’t able to get the music to work on this game (see the “Known issues” section below)

The game comes in three resolutions: 640x480, 800x600 and 1024x768. While 1024x768 is a nice high resolution, it is rather tiny on a 1080p

Figures 14 and 15 - Running Age of Empires in all its glory here in 800x600 on a 720p desktop resolution



TV, which is visually stressful. If you don’t run the game in fullscreen on a 1024x768 desktop, I’d suggest running the game in 800x600 in on a 720p desktop. The picture will fill the most of the screen and is easier on the eyes, and looks really gorgeous.

Caesar III

Caesar III pretty much works the same as Age of Empires, except that it runs even better. After you install dsound, which seems to be needed for every virtual drive, all of the features are available: sound, music, and video. The game also offers different resolutions, with 1024x768 being the highest, but I think it’s best played in 800x600 on a 720p desktop, since there’s a lot of text to read. I really like the game, and I know there’s an opensource remake, but being able to play the original in all of its glory and with videos, music and everything is just the right experience.

The game can also be found as ready-to-use zip files that you only need to



Figure 16 - Same as Age of Empires Caesar III run beautifully on the XU3/XU4 and it’s very addictive to play

extract, but as previously mentioned, PlayOnLinux offers installers for the CD version as well as the GoG version. Just make sure that you select “System” as the correct Wine version after installation.

Dune 2000

This one was a little bit surprising for me, since this game not only has some very nice effects, lots of explosions and bullet effects and such, but it also offers

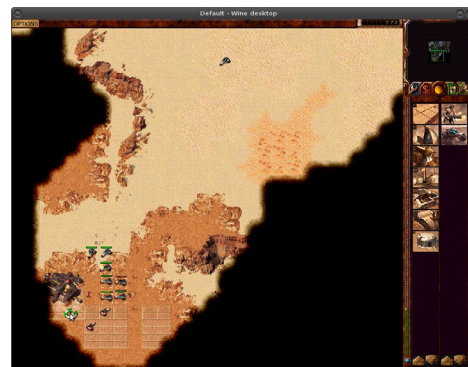
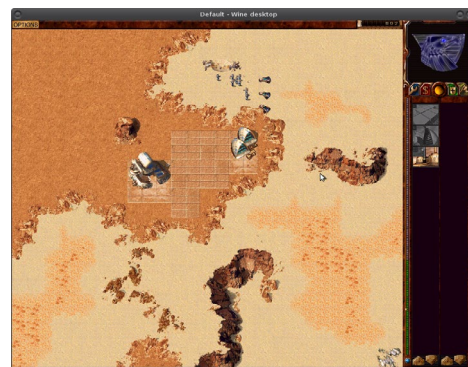


Figure 17 and 18 - Dune 2000 in high resolution on the XU3/XU4. Although this game is rather old, it’s still very fun to play and looks really good!

full motion videos (fmv) which tell the background story. Once again, get the zip, or install from CD, although this time I really recommend using the .zip file. I actually had a hard time getting the CD to work, especially with all the music and videos, so you would need a NoCD patch with this. The zip file worked right out of the box, and everything was already included. Nothing extra should be needed except for the dsound component. Originally the game was designed to only run in 640x480, which is rather tiny for ODROIDS. Luckily, there is a high resolution patcher, which allows for many different resolutions for the game to run on, except that the videos which will still be 640x480. You can download the high resolution patch from <http://bit.ly/2rVGqly>.

Use the option to run an .exe file in the “Miscellaneous” tab of the configuration menu and select the high resolution patcher. 800x600 or 1024x768 should work fine, and I played the game in 1024x768 since the game does not

have much text to read anyway. It's fun to play, although it feels slightly too slow.

Anno 1602

Anno 1602 really made me fear the words "endless game", after I played it as a teenager with some friends in multiplayer mode. After endless hours we completely eradicated all enemy parties, all pirates and had all of the islands to ourselves, but, as expected, the endless game did not end!

This game is a little bit more tricky to get to work. I used my CD from the original German version (not even the Kings Edition). I made a image from my CD and mounted it on the ODROID using cdemu with the following command:

```
$ cdemu load 0 ANNO1602.cue
```

I then used the predefined script inside PlayOnLinux to install the game automatically. I changed the Wine version to system and installed dsound as usual. I also set the virtual desktop to 1024x768 and started the game. It seemed to work fine, since I saw the introduction and heard sound, but the introduction froze at some point, and I needed to use the escape key to skip it. Reading through the information about this game under Wine, I found that the SMACKER32.DLL is not working correctly and needs to be replaced to fix the video issue. I copied the SMACKER32.DLL from Caesar III, and ever since the movies worked fine.

When I tried to start the game, I was told I need the CD to play it.

Figure 19 - Thanks to the zoom function in Anno 1602 this game looks really good, even in 1024x768



This was troublesome, since the guides I found regarding that topic did not work. Although the introduction video was played from the CD, the game itself wouldn't start. I ended up using a "NoCD" patch, which still required the use of the CD for music and video, but this allowed me to play the game. Even without the CD mounted, the game will still work, just without music and videos.

Civilization 3

Civilization 3 is available from GoG at <http://bit.ly/2r2Gzts>. Installing the GoG game is actually quite easy. Just create a new virtual drive, dsound, virtual desktop, select the setup.exe from GoG, let it install, and you're done. It was a little bit tricky to find the correct binary to start the game, since "Civ3Launcher.exe" was actually not working, but "Civ3Conquests.exe" worked fine for me. The game is fixed to 1024x768 resolution, so it might not work if you run your desktop at 720p. Apparently the GoG version is the best you can get. Reading through the topic on WineHQ, it can be rather complicated to get this game to work if you have the CD version of it, and involves NoCD patches, copying files, renaming files, making sure files can not be deleted, and so on. It seems to have some minor issues with the sound, which is a known issue for this game, but disabling and re-enabling the sound effects and music should fix this

Figure 20 - Civilization 3 on the XU3/XU4 has nice graphics and nice animation, and is probably the best Civilization game you can play on the ODROID



issue, even if just temporarily.

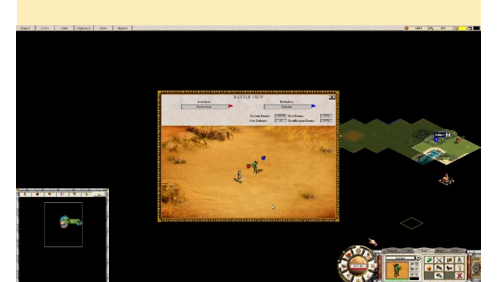
The game runs really well, and you can really see the improvements over the predecessors as well as what made the successors like Civilization 4 so great. You should really try this game!

Call to Power 2

Speaking of predecessors; Call to Power 2 is a "Civilization like" game, and I want to look into it because of some interesting issues and features. I have the CD image of this game, and mounting it in cdemu installation is straightforward, although you don't have the luxury of a pre-made script from PlayOnLinux, but it's really not needed. Strangely enough, this game does not require dsound installed, but if you install it anyway, it doesn't harm anything. However, this game requires MicroSoft Core Fonts to be installed, and this is where it gets a little tricky. Apparently the installer of PlayOnLinux for the corefonts is broken, so we need to do it the "old fashioned way", which means we need to install them via winetricks. To do so, open the configuration menu for the virtual drive, switch to the "Miscellaneous" tab and select "Open a shell". This will launch a terminal window from within the virtual drive.

The rest is actually quite easy. Just type "winetricks corefonts", press Enter, wait for the process to finish, then you can close the window. This allows you to start the game. Without the corefonts installed, the game will crash right after the loading screen. All videos are broken at the moment, and I couldn't find a way to get them to work, but it seems no one has done that yet.

Figure 21 - Call to Power II in 1920x1080 full-screen-mode on the ODROID-XU3



Despite the videos, the game works, even if it's a little less attractive this way. What makes this game interesting is that you can select nearly any resolution you want and yes, that means you can play this game on 1080p full-screen mode on the ODROID-XU3/XU4.

If that's the way you want to play it, that's up to you, and running the game in full-screen mode gives you a little bit more extra performance, which is why you can actually use 1920x1080. I haven't played this game a lot, so I don't know how it will perform in later stages with bigger armies and huge maps. The fighting animations were quite nice, and it might be interesting to see how far we can push the XU3/XU4 with this game.

One last game

Yes there is one more game that I want to talk about a little bit more in detail, and I hope that you're going to like it. The game only runs in 640x480, so using 720p display resolution is probably a good idea unless you want a tiny window on your screen. You need to run the game in window mode, or set your display resolution to 640x480 for full-screen-mode. The game is installed from CD or an image mounted via cdemu. You will need the CD/Image whenever you play, but it doesn't require a NoCD patch.

The game requires dsound as usual, but also needs some extra configuration. Go to the configuration screen of the virtual drive and click on "Display", then change the Direct Draw Renderer from default to GDI. After that, one more step is needed, since the direct draw rendering library of the game needs to be replaced.

After you install the game, you need to download a fix from <http://bit.ly/2r49kIv>. Extract the ddraw.dll and wine3d.dll into the game folder, overwriting the ddraw.dll already inside the folder. This will allow you to play the game with music, video, and gameplay working, except for the menu. The



Figures 22 and 23 - Diablo I one of the most iconic games of all times running on the ODROID-XU3/XU4 and everything just plays like in 1997

menu will be missing some elements, but you should be able to get everything done anyway, like creating a new character, loading a game and so on. You may wonder what this game is, and Figures 22 and 23 will give you an answer to that!

If you never played Diablo before, now is your chance to replay some gaming history right on your ODROID. Find out why this game defined the hack-and-slash RPG genre.

Debugging and tips

On the main window of PlayOnLinux, you can start a game in debug mode by clicking on the Debug text on the left side, next to your game/program list. Or you can select "Tools -> PlayOnLinux debugger" from the menu, which will give some information about why a game did not start or is having issues. This doesn't mean you will always be able to fix these issues, nor to identify them, but it's a good place to start investigating.

Besides that, there is WineHQ,

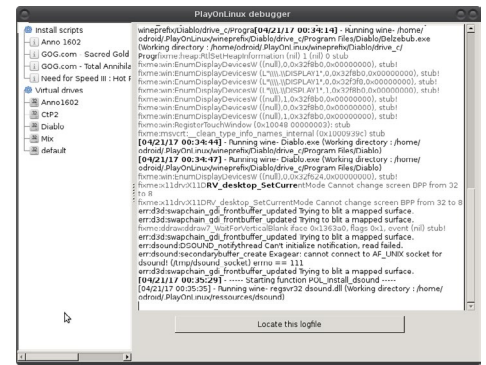


Figure 24 - The message "err:dsound:DSOUND_notifythread Can't initialize notification" indicates that you forgot to install dsound

which is a forum about Windows games on Linux using Wine. If you want to find out why a certain game is not working under Wine, or what drivers you need to install, you should google "WineHQ <game-name>" (ex. "winehq diablo"), and you will get more detailed information about the game and how to configure it under Wine.

Known Issues, limitations and last thoughts

Apparently MIDI does not work inside the Exagear environment. It works outside of it, but inside Exagear, no process can access the MIDI device. Maybe that is something that can be fixed in the next release of ExaGear.

As I mentioned at the start of the article, OpenGL doesn't work, and some games may require it but don't use it heavily, so you may need to switch your Wine version to something different than "System", since this version won't even allow MESA Software GL to be used. Keep in mind that other Wine versions may be a lot slower.

Some games may not work even if they don't use OpenGL, but exhibit other issues. For example, the game Airline Tycoon Evolution runs perfectly fine, but has one issue. The fonts do not work in 24 or 32-bit color and you need to switch to 16-bit, but that's not supported, at least not on the ODROID-XU3/XU4. Although the game would

normally work fine, you can't play it because all of the text is invisible. 16-bit Windows games (Windows 95) might also not work, and they typically crash for me when I try to install them, so you might be further limited in your game choices.

I got a couple of other games to work such as Total Annihilation, but that game has a few issues with the sound. Sometimes it is necessary to turn sounds or music off in order to play some games. Generally, GoG games are the easiest to get to work. They are already DRM-Free and don't require a NoCD patch, and are pre-optimized to run on modern Windows systems.

There aren't many games that do not require 3D support. You have probably already seen a pattern in that adventure and strategy games are the most likely to work, and maybe some older RPG games from the Windows 98 era from around 1996-2001 as well

Trying out games and failing can be a little frustrating, and ExaGear is not very fast with its I/O operations, which means that the access on disk is rather slow, so it takes a while to install, setup, and test your games. Be patient and you will find some really nice Windows games that will work on your ODROID. Hopefully someday we can use 3D acceleration as well, and will have even more games to play.

There is no cow level!



ODROID

DIFFERENT BOARDS FOR DIFFERENT PROJECTS



Single Board Computers

Add-On Boards

Cables

Cameras

Cases

Connectivity

Connectors

Coolers

Development

Display

Memory

Power & Battery

Sensors

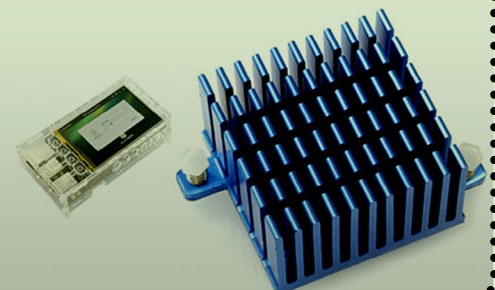
Sound

Visit the
NEW
store
for all
your
ODROID
needs!

odroidinc.com

MANY KINDS OF
ACCESSORIES

Multiple accessory options gives users the ability to create all-in-one, integrated projects such as: tablets, game consoles, infotainment systems, and embedded systems.



ALARM CENTRAL - PART 2

ULTRA-LOW POWER MOTION SENSORS

by Jörg Wolff

This is part two of my project Alarm Central for the ODROID-C1 running Android. The project consists of the Alarm Central Android app, ultra low power window sensors, and ultra low power motion sensors. The sensors communicate with the ODROID-C1 via nRF24L01 2.4GHz modules. In this article, I will explain how to make ultra low power motion sensors.



Figure 1 - nRF24 as Motion Sensor decoration

RF24 Motion Sensor

Based on an ATtiny84 processor, I designed a small board which contains an Elmos E931.96 motion controller, a PerkinElmer Pir LHi968, a connector for the nRF24L01, a connector for flashing the processor, and some additional parts. I ordered the printed board through Itead Studio. The soldering of the components takes about 45 minutes per sensor.

Parts List

- Printed board
- ATtiny 84A-SSU SO-14



Figure 2a - nRF24 Motion Sensor PCB - Top View

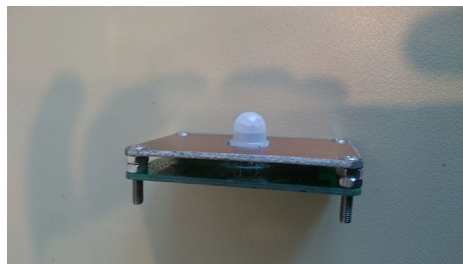


Figure 2b - nRF24 Motion Sensor PCB - Side View

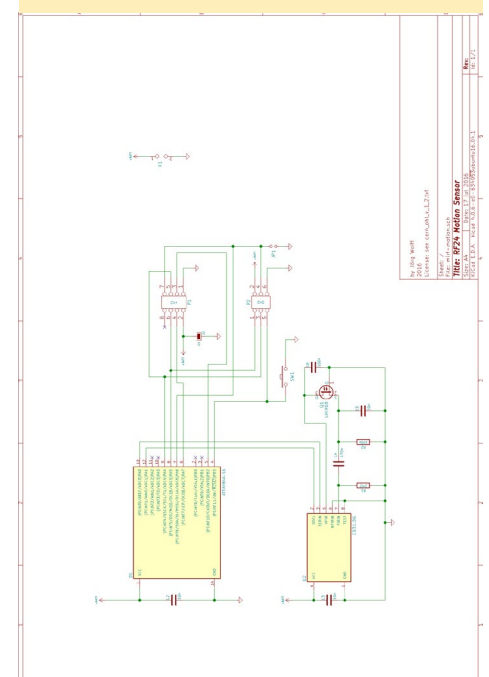
- NRF24L01 module
- Motion Controller E931.96
- Pir LHi968
- Resistor 100k SMD1206
- Resistor 2M2 SMD1206
- Capacitor 22u/16V 4.3mm x 4.3mm
- Capacitor x 3 10n SMD1206
- Capacitor x 1 100n SMD1812
- Capacitor x 1 470n SMD1812
- Pin strip 2 x 3 2.54mm
- Female strip 2 x 4 2.54mm
- Screw Terminal 2 pin Grid 5.0mm
- Fresnel Lense Ø11.8mm
- Battery Holder 2 x AA
- Push Button SMD 6 x 6 (optional)



The board's dimensions are 40mm x 60mm (1.57" x 2.36"). Using 3000mAh capacity AA lithium batteries, the estimated lifetime of the sensor is about 5 to 8 years, depending on how often motion is detected. The ATtiny's firmware designed so that it goes to sleep for 4 seconds, then wakes up, sending a payload of 20 bytes to the Alarm Central. If motion is detected, the ATtiny will also wake up and send the payload. The total average current is about 32µA. To reach this low current, I disabled the ATtiny's Brown-Out Detection.

A freshly flashed sensor board will send data unencrypted with the node

Diagram 1 - Motion Sensor circuit diagram



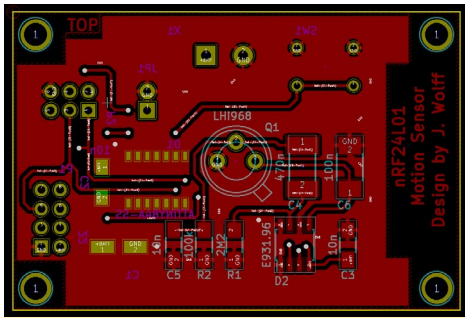


Figure 3 - nRF24 Motion Sensor PCB - top side

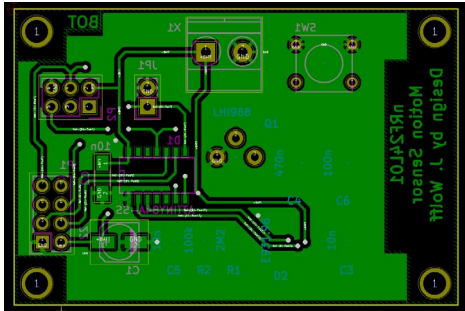


Figure 4 - nRF24 Motion Sensor PCB - bottom side

number of 255. Alarm Central receives the message, then does auto node numbering and sends the AES key. This will only happen if Alarm Central is unlocked and the user authenticated. For a short time, communication is open. If the board has been rebooted, the sensor will again, for a short time, send an initial payload of about 500ms, with a notification that it is already configured. Alarm Central, if unlocked and authenticated, can reconfigure the sensor during this window. This makes it possible to reconfigure the sensor (e.g. if the AES key has changed), without having to re-flash a new firmware.

The code can be found on Github at <http://bit.ly/2qEUcjW>, and the KiCad project can be found at <http://bit.ly/2q6PLf1>.

I made the case from a cardboard tube with an outer diameter of 75mm, cut to 40mm in length with a chop saw. I then glued some pressed cardboard to the inside of the tube using wood glue.

I painted the surface with about four coats of transparent acrylic paint, and the paint entered the pores of the case material helps to ensure its stability. Af-

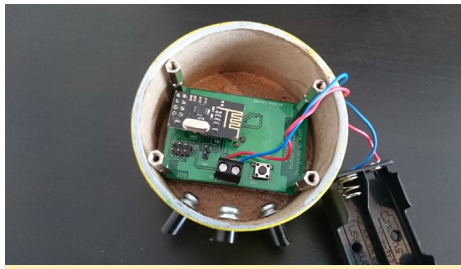


Figure 5 - nRF24 Motion Sensor case inside with board

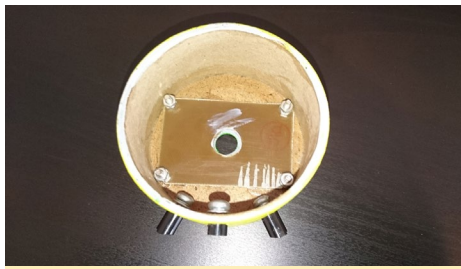


Figure 6 - nRF24 Motion Sensor case inside without board

ter each coat, I polished the surface with fine sandpaper, using an acrylic filler to smooth the surface. This way, I got a case with a fine surface, which I then

Figure 7 - Motion Sensor case unpainted



Table 1 - Example of plain text payload from sensor

00	01	02	03	04	05	06	07	08	09
node	aes_key	aes_key	aes_key	aes_key	aes_key	aes_key	aes_key	aes_key	aes_key
1	xx	xx	xx	xx	xx	xx	xx	xx	xx

10	11	12	13	14	15	16	17	18	19
aes_key	aes_key	aes_key	aes_key	aes_key	aes_key	aes_key	crc	crc	dummy
xx	xx	xx	xx	xx	xx	xx	0	0	0

Table 2 - Example of plain text initial payload from Alarm Central

00	01	02	03	04	05	06	07	08	09
node	type	v_bat	v_bat	value	info	crc	crc	dummy	dummy
255	1	220		0	255	nn	nn	0	0

10	11	12	13	14	15	16	17	18	19
dummy	dummy	dummy	dummy	dummy	dummy	dummy	dummy	dummy	dummy
0	0	0	0	0	0	0	0	0	0



Figure 8 - Motion Sensor is ready

passed on to my wife, who was responsible for the art design.

Inside the case, I glued an epoxy plate of the same dimensions as the PCB and spacer, as well as three housing feet, as shown in the image. The back side is covered by an epoxy plate.

```
Node: 255
type: 1 - window
2 - motion
v_bat: 220
info: former received node
255 - if freshly flashed
```

```
Node: 1 -> 254
aes_key: 16 bytes 128bit
crc: 2 bytes
```

Example of crc16 calculation on the Java side:

00	01	02	03	04	05	06	07	08	09
node	type	v_bat	v_bat	value	info	crc	crc	dummy	dummy
1	1	2995		0	0	xx	xx	0	0

10	11	12	13	14	15	16	17	18	19
dummy	dummy	dummy	dummy	dummy	dummy	dummy	dummy	dummy	dummy
0	0	0	0	0	0	0	0	0	0

Table 3 - Example of encrypted payload from sensor

```

/*
 * Calculate a CRC16 XModem, Init-
 * value 0h
 * */
public int calculateCRC(byte[]
bytes) {
    int crc16 = 0x0000; // ini-
    tial value
    int polynomial = 0x1021;
    for (byte b : bytes) {
        //bytesCRC = bytesCRC +
Integer.toHexString(b);
        for (int i = 0; i < 8;
i++) {
            boolean bit = ((b
>> (7-i) & 1) == 1);
            boolean c15 = ((crc16
>> 15 & 1) == 1);
            crc16 <<= 1;
            // If coefficient of
bit and remainder polynomial = 1
xor crc with polynomial
            if (c15 ^ bit) crc16
^= polynomial;
        }
    }
    crc16 &= 0xffff;
    return crc16;
}

```

```

copyOfRange(array.array(), 0,
17);
int crc = calculateCRC(bytes);
array.putShort(17, Short.
reverseBytes((short) (crc &
0xFFFF)));

```

Example of crc16 calculation on the ATtiny side:

```

uint16_t crc = 0;
for (cnt1 = 0; cnt1 < 17; cnt1++)
{
    crc = _crc_xmodem_update(crc,
data_out.as_crc.crc[cnt1]);
}

```

```

Node:      1 ... 254 - former
received node
type: 1 - window
2 - motion
v_bat:     battery voltage in
mV
value:     0 - open (window),
stillness (motion)
1 - closed (window), motion (mo-
tion)
info: former received node
crc: CRC16 Xmodem of bytes 0 ... 5
of payload

```

To reverse the received crc16 (posi-
tion 6 and 7 in the payload):

```

short crc = Short.
reverseBytes(array.getShort(6));

```

To reverse the crc16 before sending
to sensor (position 17 and 18 in the pay-
load):

```

byte[] bytes = Arrays.

```

CROSSY SPACE

A JOURNEY WHERE NO CROSSY GAME HAS GONE BEFORE

by Bruno Doiche

Buckle up! Get ready to jump
from planet to planet with
Crossy Space.

On this simple but
addictive game, you
need to take your
starship as far away
as you can, but for
that you will need,



as in all twitch-based
games, your sharpest reflexes ever, and
also you will need to master the abil-
ity to do tangents. It sounds simple
enough, right? Well, good luck!

<https://play.google.com/store/apps/details?id=com.irrationaldesign.orbitinspace>



**You will suffer and you will rage, but you
will feel the bliss when you get your own
Millenium Falcon!**

MEET AN ODROIDIAN

DAVIDSON BRAGA DE SOUZA (@SNK)

edited by Rob Roy (@robroy)

Please tell us a little about yourself.

I am 23 years old and, since the age of 10, I have been fascinated with computers and systems. I am a public servant, and live alone, away from my relatives, in Serra / ES - Brazil. I have a degree in Law (lawyer) and Environmental Management. From the age of 12, I began to delve deeper into the field to which I am passionate and work until today in the free time. I remember when I went to the lan houses just to learn and watch the technicians tinkering. Little by little, I perfected myself and doing training courses, but most of my knowledge is self-taught

Whom do you admire in the world of technology?

I think I have a lot of people that I admire, but someone who really catches my eye is Bill Gates, both for having revo-

lutionized the way we practice computer science today, and his philanthropic projects.

What attracted you to the ODROID platform?

I originally only knew about the Raspberry Pi project and did not know about ODROIDS. As I did more research on the Internet, I discovered other boards and fell in love with the ODROID products and platform. Aside from the support being great, the boards are of impeccable quality and reliability! Hardkernel has a very active forum, even though it seems “abandoned” in terms of infrastructure. Their boards are not as cheap as the Raspberry Pi, but, we need to take into consideration that ODROIDS offer a lot in terms of the quality of construction and performance of the system. It’s something that really makes the cost vs. benefit of this platform worth it. I’ve even recommended several friends to buy their boards from the HardKernel.

How do you use your ODROIDS?

I use my ODROIDS to browse the web, as a media center and mainly as a file server. I have not yet managed to buy a Cloudshell, but I will still fulfill that dream! If I had known of its utility, I would have bought it much earlier.

Which ODROID is your favorite and why?

I already had an ODROID-C2 and now I have an XU4. Sure enough, XU4 is my favorite board, due to its great performance and perfect construction. Its use as a media center and file server is much better than having a dedicated NAS, which would be more expensive and have a higher energy consumption. Internet browsing on it is pretty smooth, but, there are still some points of improvement such as system optimizations. Since it is a development board, these are things that we have to continue working on and solving so that we can develop a well-optimized and fluid system for general use.

What innovations would you like to see in future Hardkernel products?

In my humble opinion, with the ODROID-C1+ and C2, I think the board hardware is great. For the XU line, I believe that it could use an updated SOC from Samsung and an infrared port, but that’s all. The products are high quality and, in relation to addons, Hardkernel is to be con-



Davidson enjoys spending time with his sister



Davidson enjoying nature with his father

gratulated, since it is always updating its peripherals and launching new ones, such as the recently released Cloud-Shell 2.

What hobbies and interests do you have in addition to computers?

I really like photos and hardware. I'm in love with the environment, and I appreciate a good beer!

What advice would you give to someone who wants to know more about programming?

If a particular area interests you, go ahead and study it. Knowledge is something that no one will ever take away from you. If you know someone who is wiser than you, accept criticism and grow with them. All knowledge must be shared. Programming is an art. It is a very relaxing process, and the result is even more pleasurable to see. Seeing someone using software programmed by you is an indescribable sensation! If you want, go in search of your dreams.

Thank you for the opportunity to participate in this interview in one of the most read magazines related to the ODROID world, and certainly the best!



ODROID Magazine is on Reddit!



ODROID Talk Subreddit

<http://www.reddit.com/r/odroid>

