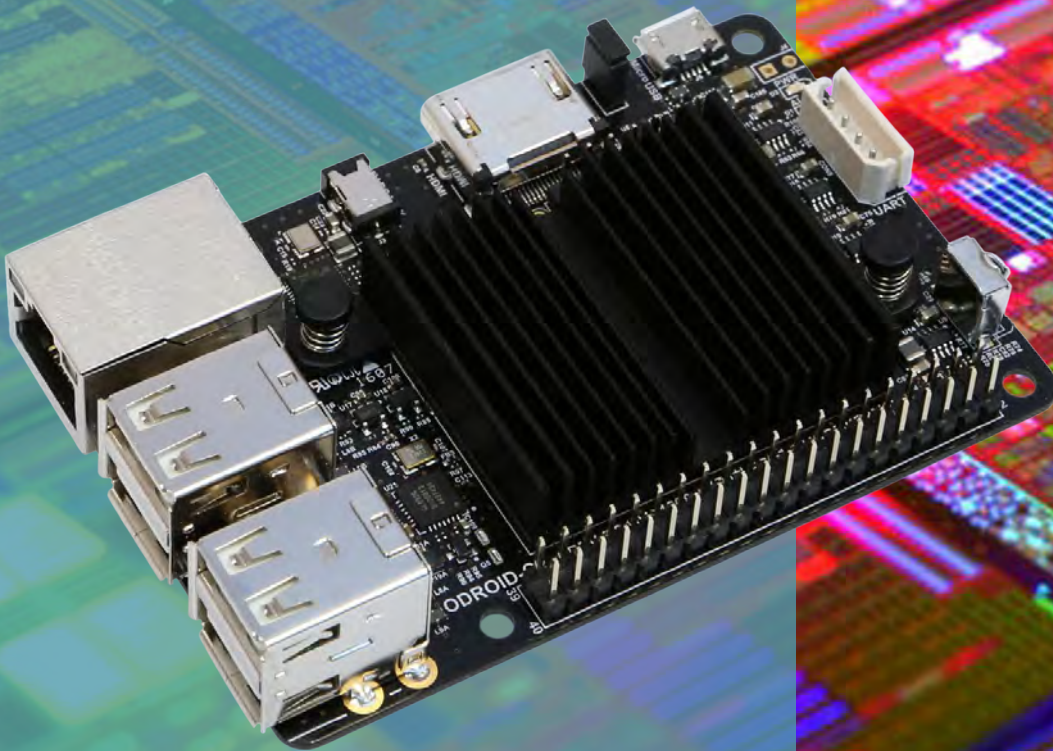




**HARDKERNEL**

**USER MANUAL**



**ODROID-C2**

# Index

## ODROID-C2 Beginner's Guide

<b>Chapter 1 • Welcome</b> .....	1
Differences between a typical PC and a Single Board Computer (SBC) .....	2
Components Included on an SBC .....	2
Block Diagram .....	3
Board Image .....	3
Attaching Peripherals .....	3
Power Supply .....	4
Monitor .....	6
Keyboard and Mouse .....	6
Ethernet and Wifi .....	7
MicroSD Card .....	8
eMMC Module .....	8
LED Status .....	8
Technical Specifications .....	9
Heatsink .....	13
Temperature Measurement .....	13
Frequently Asked Questions .....	15
<b>Chapter 2 • Getting Started</b> .....	20
Home Computing Network .....	20
Preparing the Work Area .....	21
Flashing an Image .....	22
Boot Media .....	22
Windows 7+ .....	25
Linux .....	28
OSX .....	29
Inserting the eMMC Module or SD Card .....	30
Powering Up .....	30
Troubleshooting .....	30
Configuring Linux .....	32
Configuring Android .....	34
Powering Down .....	35



<b>Chapter 3 • Operating Systems</b> .....	36
Ubuntu/Debian .....	36
Linux Basics .....	37
Kernel .....	38
GUI .....	38
720p vs 1080p and 2160p .....	38
Progressive vs Interlaced Video .....	39
Video Downconversion .....	39
Video Upconversion .....	40
HDMI Overscan .....	40
Disabling Monitor Overscan .....	41
Display Setting Button on Remote Control .....	41
Finding Setting in Monitor Menus .....	41
Adjusting Linux Display Output .....	42
Adding an ODROID-VU7 Plus Touchscreen .....	42
Command Line Interface .....	43
Disk Partitions .....	44
Web Browsing .....	44
Kodi (formerly XBMC) .....	45
Office and Productivity Applications .....	46
Music and MIDI .....	46
Experimental Music with the C2 .....	48
Android .....	48
Desktop Environment .....	48
ODROID Utility and Updater .....	49
Setting the Display Resolution .....	49
Kodi .....	50
Netflix .....	50
Gaming .....	50
Music and Midi .....	50
Using Bluetooth Devices with Android .....	51
Adding an ODROID-VU7 Plus Touchscreen .....	51

# Index

## ODROID-C2 Beginner's Guide

<b>Chapter 4 • Hardware Tinkering</b> .....	52
Preparation .....	52
USB / UART Module Kit .....	53
Bluetooth Module 2 .....	58
Tinkering Kit .....	61
ODUINO ONE .....	64
ODROID-SHOW2 .....	66
16x2 LCD I/O Shield.....	68
Weather Board .....	69
USB-DC Plug Cable 2.5x0.8mm .....	71
USB-SPDIF .....	71
USB-CAM 720P.....	73
USB3 / SATA3 HDD/SDD interface kit .....	74
USB3 to SATA Bridge Board .....	77
USB GPS Module .....	78
MyAHRs+ Board .....	80
WiFi Module 4 .....	82
HiFi Shield + .....	83
RTC Shield .....	84
UPS3 .....	86
Xprotolab Plain .....	87
ODROID-VuShell for ODROID-VU7 .....	88
Universal Motion Joypad .....	89
Conclusion .....	90
Additional Resources .....	90

# Credits

Authors: Rob Roy  
Venkat Bommakanti

Art Editor: Bruno Doiche

Assistant Art Editor: Nicole Scott

Contributors: Alan Tung  
Anawat Hokoon  
Patrick Oberli  
Robert Cleere

## 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**

© 2015 Hard Kernel, Ltd. • 704 Anyang K-Center, Gwanyang, Dongan, Anyang, Gyeonggi, South Korea, 431-815  
Hardkernel manufactures the ODROID family of quad-core development boards and the world's first ARM big.LITTLE single board computer.  
Read our monthly magazine at <http://magazine.odroid.com>.  
You can join the growing ODROID community with members from over 135 countries at <http://forum.odroid.com>.  
Explore the new technologies offered by Hardkernel at <http://www.hardkernel.com>.

# Welcome

**C**ongratulations on purchasing the ODROID-C2! It is one of the most powerful low-cost 64-bit Single Board Computers available, as well as being an extremely versatile device. Featuring a fast, quad-core AmLogic processor, advanced Mali GPU, and Gigabit ethernet, it can function as a home theater set-top box, a general purpose computer for web browsing, gaming and socializing, a compact tool for college or office work, a prototyping device for hardware tinkering, a controller for home automation, a workstation for software development, and much more.

Some of the modern operating systems that run on the ODROID-C2 are Ubuntu, Android, and ARCH Linux, with thousands of free open-source software packages available. The ODROID-C2 is an ARM device, which is the most widely used architecture for mobile devices and embedded computing. The ARM processor's small size, reduced complexity and low power consumption makes it very suitable for miniaturized devices such as wearables and embedded controllers.

## *Differences between a typical PC and a Single Board Computer (SBC)*

If you are used to using a standard PC such as an OSX or Windows machine, there are a few small differences to note when transitioning to an ARM device. To begin with, the speed of an ARM processor is not directly comparable to the speed of an Intel processor. Because of the efficiency of the CPU, the ODROID-C2 can give great response time that feels just as fast as using a more expensive computer. The operating systems available for the ODROID-C2 are also highly optimized, and benefit from the expertise of many open-source contributors that continually review each other's work that bring daily improvements to the OS.

In addition, nearly all of the applications available for the ODROID-C2 also have their source code publicly available, which means that you can freely modify and update the applications to fit your specific needs. Program authors often maintain a GitHub repository, where suggestions can be submitted, reviewed and distributed to all of the application's users.

The ODROID-C2 also uses Solid State technology for its storage media, although a conventional hard disk may be used as an auxiliary device. The boot partition can be stored on either a microSD card or the much faster eMMC module, and Hardkernel's products have the unique distinction of supporting removable eMMC modules, so that operating systems may be switched out conveniently and easily. An eMMC module is a type of storage typically used in a smart phone, and is one of the more advanced compact media devices available.

The power consumption of a typical personal computer can be anywhere between 100W and 1000W or more, depending on the peripherals, processor and type of power supply used. However, the ODROID-C2 uses between 5W ~ 10W, greatly reducing your electricity bills, as well as allowing unique power configurations such as compact solar power cells and long-running batteries.

## *Components Included on an SBC*

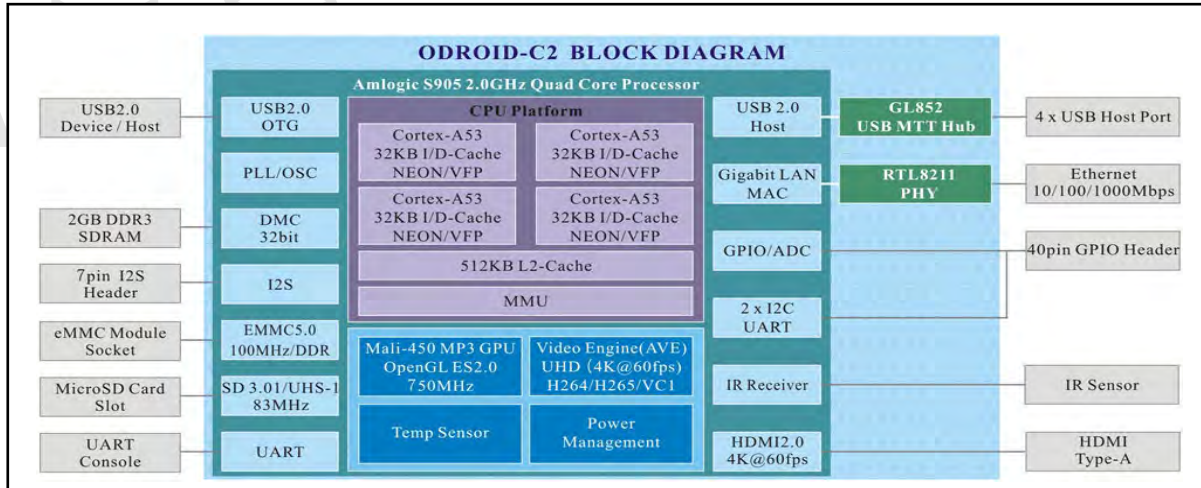
The ODROID-C2 contains many of the same connections as a typical computer, with 4 USB 2.0 ports, a micro-USB OTG port, an Ethernet port that supports Gigabit transfer speeds, a HDMI 2.0 connector for monitors that can display at resolutions up to 4K, and a 5V/2A DC power connector. In addition to these standard inputs, the ODROID-C2 also includes a 40-pin GPIO port, a USB-UART serial console port, an eMMC module connector, and a dedicated slot for a microSD card. For more details, refer to the ODROID-C2 introductory video at <http://bit.ly/1WQ2yQw>.



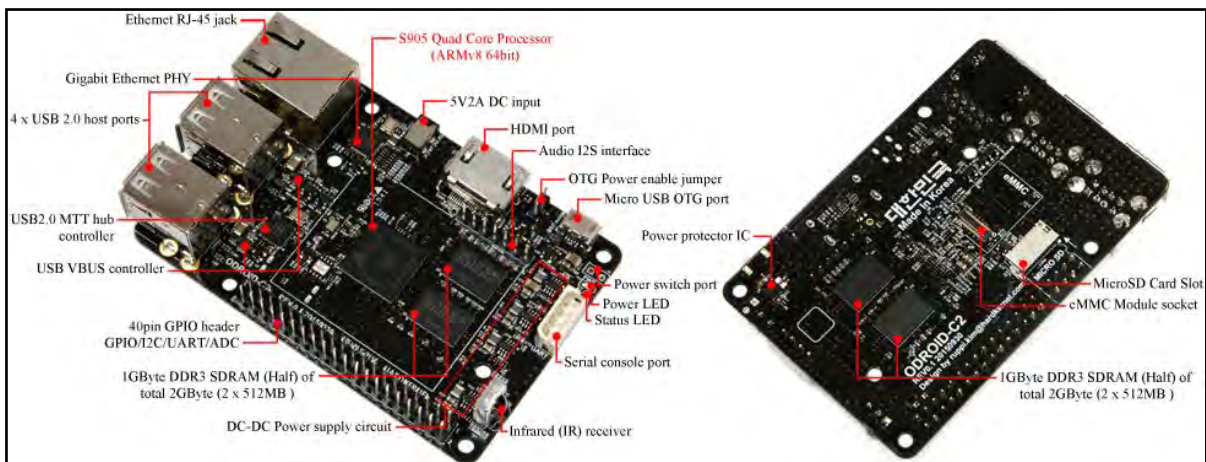
# Chapter 1

## Block Diagram

The following diagram illustrates conceptually how the components of the ODROID-C2 fit together:



## C2 Block Diagram and Annotated Board Image



## Attaching Peripherals



The ODROID-C2 supports the services of numerous peripherals. The image to the left depicts the following peripherals: Power Supply Unit (PSU) with cable, USB / UART debug cable, USB 3 memory card reader, eMMC reader, eMMC card, microSD card, 4k HDMI 2.0 cable, USB / wireless keyboard / mouse dongle, USB touch cable, and Ethernet cable.

## ODROID-C2 and Useful Peripherals



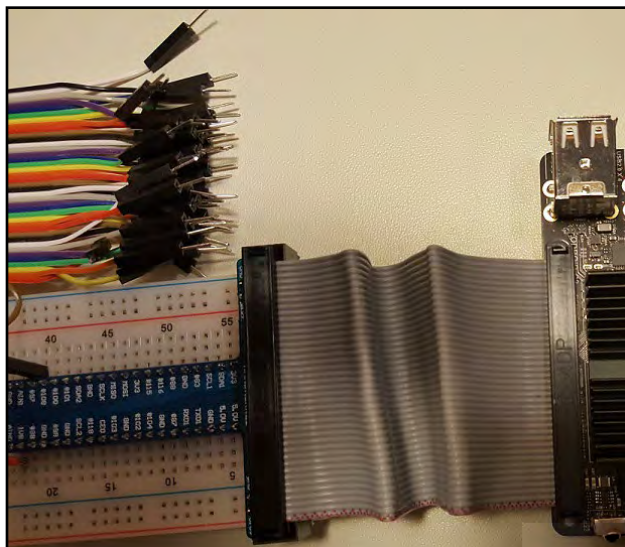
# Chapter 1

Below is shown the 7" ODROID-VU7 Plus HDMI touchscreen monitor supporting 1024x600 pixel resolution and 10-point touch control. The USB cable that provides power and touch related control signals.



## ODROID-VU7 Plus HDMI Monitor

The image below depicts the use of the C Tinkering kit with an ODROID-C2. It is useful to experiment with circuit design and related software monitoring and control, which is detailed in Chapter 4.



## C Tinkering kit for ODROID-C2

### *Power Supply*

The ODROID-C2 board requires a 5V/2A DC power source. The dedicated power connector (miniature barrel jack) can accept a DC plug cable with a plug that has an outer diameter of 2.5mm and an inner diameter of 0.8mm. The plug inner core (center) is positive (attached to the RED wire in the cable) and outer cylinder is negative (attached to BLACK wire in the cable). The ODROID-C2 can be powered using different options, which are outlined below.



## 5V DC 2A PSU

Attach the plug to the power connector on the ODROID-C2. Plug the 2-pin PSU into the power outlet. You may need an adapter to use in your region - such as the Americas. The PSU pictured is available from Hardkernel.



## USB DC plug cable

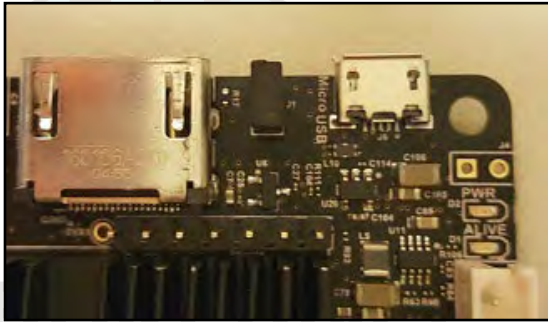
Attach the USB end of this cable to the USB port of a charging device that can generate 2A @5V, and attach the plug to the ODROID-C2. This cable is also available from Hardkernel.



## DC plug cable

If you have a 5V DC 2A PSU which does not have the required plug, you can cut off the plug from such a power supply. Expose about  $\frac{1}{2}$ " of the red/black wires on the psu cable and attach them to the same colored cables of this cable, then solder the joints. You can cover the joint using electrical tape or a heat-shrink wrap. Attach the plug to the ODROID-C2 and insert the PSU pins into a power outlet.

This cable is also available from Hardkernel, and may be paired with the bench power supplies with variable voltage.



## MicroUSB Power

You can alternatively power the ODR0ID-C2 via the micro-USB port using a stable 5V/2A USB charger. This port can be used with Linux Gadget drivers on your host PC, which means that the resources in the ODR0ID-C2 can be shared with typical PCs. You can also add a micro-USB to HOST connector if you need an additional USB host port.

## *Monitor*

The ODR0ID-C2 offers a Type A standard HDMI 2.0 port for connecting an HDMI-compliant monitor. It can support a variety of resolutions up to 4K. It is recommended to use the Hardkernel supplied HDMI 2.0 adapter cable, but many other equivalent high quality standard cables could also work.

## *Keyboard and Mouse*

Nearly all USB HID-compliant keyboard and mouse will work when connected to one of the four USB ports. Support for some wireless keyboards, such as the Logitech K400r, is available by default in the official Hardkernel Ubuntu Mate images.

```
odroid@odroid64:~$ lsusb
Bus 001 Device 004: ID 046d:c52b Logitech, Inc. Unifying Receiver
```

The use of a bluetooth mouse or keyboard requires a bluetooth dongle and, for first pairing, either an USB mouse/keyboard, an SSH access from remote or a working touchscreen. To pair a bluetooth keyboard or mouse via the Linux console, run the following command in a Terminal window:

```
$ sudo hcitool scan
```

Push the Connect button on the bluetooth device and it will appear in the console:

```
Scanning ...
```



# Chapter 1

```
XX:XX:XX:XX:XX:XX    Rapoo E6700
bluez-simple-agent hci0 XX:XX:XX:XX:XX:XX
```

Push the Connect button again, and enter the shown pin on the keyboard, followed by the Enter key. If no pin is shown, try 000000. Then, type the following to trust the device and restart the bluetooth service:

```
$ sudo bluez-test-device trusted XX:XX:XX:XX:XX:XX yes
$ sudo /etc/init.d/bluetooth restart
```

In Android, use the Bluetooth configuration screen in the Settings application to connect to the bluetooth device.

## *Ethernet and WiFi*

The Ethernet port accepts a standard CAT 5E or 6 Ethernet cable, and is capable of up to 1GBps (1000 MB/second) transmission rate. Typically, a working wired ethernet connection creates the eth0 interface, whose details can be obtained with the following command:

```
odroid@odroid64:~$ ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:3f:06:25:57:56
          inet addr:192.168.6.33  Bcast:192.168.6.255
Mask:255.255.255.0
          inet6 addr: fe77::52e:6ea:fb63:53fe/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:25817 errors:0 dropped:0 overruns:0 frame:0
          TX packets:27210 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:14685550 (14.6 MB)  TX bytes:14333588 (14.3
MB)

          Interrupt:40
```

A WiFi USB adapter, such as the official Hardkernel WiFi Module 3, fits in any of the 4 USB ports. With a fully authenticated WiFi connection, the connection details can be found with the following command:

```
odroid@odroid64:~$ ifconfig
enx00a82b000636 Link encap:Ethernet  HWaddr 00:a8:2b:00:06:36
          inet addr:192.168.6.160  Bcast:192.168.6.255
Mask:255.255.255.0
          inet6 addr: fe80::2b6:2cef:ff50:636/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:118 errors:0 dropped:2 overruns:0 frame:0
          TX packets:50 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11313 (11.3 KB)  TX bytes:8948 (8.9 KB)
```

# Chapter 1

The WiFi USB dongle can be enumerated with the following command:

```
odroid@odroid64:~/zBU$ lsusb
Bus 001 Device 005: ID 0bda:8176 Realtek Semiconductor Corp.
RTL8188CUS 802.11n WLAN Adapter.
```

## *MicroSD Card*

Align the metal strips of the microSD card with the pins of the microSD card connector, and slowly push it in until it clicks in place. Be gentle. If you are unable to push it in, it may be misaligned. Recheck and flip the microSD card if you notice a wrong insertion direction. The image on the following page shows a properly mounted microSD card.

## *eMMC Module*

Align the eMMC 5.0 module and the eMMC connector on the ODROID-C2 board, using the white rectangle on the PCB as a guide. The female portion of the eMMC module should line up with the male connector on the board. Slowly push it in until it clicks in place. Be gentle. If you are unable to push it in, it may be misaligned. Recheck and turn the eMMC module if you notice a wrong insertion direction. The writing on the card will be exposed after insertion. The image on the following page shows the eMMC module when mounted.

## *LED status*

The ODROID-C2 includes several LED lights that indicate the status of the device:

### **The red LED**

- Is on when power to the device is available

### **The blue LED**

- Is on (solid light) when the bootloader is running

### **The blue LED**

- Blinks slowly when the kernel is running, like a heartbeat

### **The blue LED**

- Blinks quickly when the kernel is in panic mode



# Chapter 1

## *Technical specifications*

### **Processor**

Amlogic S905 ARM® Cortex®-A53(ARMv8) 1.5Ghz quad-core CPUs, with Mali™-450 GPU (3 Pixel processors + 2 Vertex shader processors)



### **eMMC module socket**

8GB/64GB : Toshiba

16GB/32GB : Sandisk iNAND Extreme

64GB/128GB : Samsung

The eMMC storage access time is 2-3 times faster than the SD card. The storage capacities come in sizes: 8GB, 16GB, 32GB, 64GB and 128GB. Using an eMMC module will increase speed and responsiveness, similar to what one observes in performance improvement, by upgrading a mechanical hard drive (HDD) to a Solid State Drive (SSD) in a typical PC.

### **Micro Secure Digital (microSD) card slot**

There are two methods of storage for the Operating System. The first is to use a microSD card and another is to insert an eMMC module. The latter is normally used for external storage for smartphones and digital cameras. The ODROID-C2 can utilize a newer UHS-1 microSD model, which is about twice as fast as a class 10 card.



**ODROID-C2 closeup with microSD card attached**





## Incorrect usage of microSD card slot

There are some microSD cards which can cause an additional boot delay time of around 30 seconds. According to our testing, most Sandisk microSD cards don't cause a long boot delay. The ODROID-C2 model is compatible with a wider array of microSD cards than earlier ODROID SBC models.

The ODROID-C2 PCB has the outline of a microSD card as a guide to the proper insertion of the card. Also, note that the microSD slot cannot accept an eMMC module mounted on the eMMC module reader.

## 5V 2A DC input

The DC input is for 5V power input, with an inner (positive) diameter of 0.8mm, and an outer negative diameter of 2.5mm. The ODROID-C2 consumes less than 0.5A in most cases, but it can climb to 2A if many passive USB peripherals are attached directly to the main board. Peak momentary consumption may increase to 3A, on a system with too many power hungry peripherals.

## USB host ports

There are four USB 2.0 host ports. You can plug a keyboard, mouse, WiFi adapter, storage or many other devices into these ports. You can also charge your smartphone with it! If you need more than 4 ports, you can use a powered external USB hub to reduce the power load on the main device.

## HDMI 2.0 port

To ensure system longevity and reduce dependency on expensive cables, we used the standard Type-A HDMI 2.0 connector. A high-speed HDMI 2.0 (for 4K resolution) cable may be used with this

port. Lower resolutions can be achieved using the HDMI 1.4a cable.

## **Ethernet RJ-45 jack**

The standard RJ45 Ethernet port for LAN connection supports 10/100/1000 Mbps speed. The green LED Flashes when there is 100 Mbps connectivity, and the yellow LED Flashes when there is 1000 Mbps connectivity.

## **Infrared (IR) receiver**

This is a remote control receiver module that can accept standard 37.9Khz carrier frequency based wireless data in NEC format.

## **MicroUSB OTG port**

You can use the standard micro-USB connector with Linux gadget drivers on your host PC, which means that the resources in the ODROID-C2 can be shared with other PCs. You can also add a micro-USB to HOST connector if you need an additional USB host port. This port can also be used for power input.

## **General Purpose Input and Output (GPIO) ports**

These 40pin GPIO port can be used as GPIO/I2C/SPI/UART/ADC for electronics and robotics. The 40 GPIO pins on an ODROID-C2 are a great way to interface with physical devices like buttons and LEDs using a lightweight Linux controller.

If you are a C/C++ or Python developer, there's a useful library called WiringPi that handles interfacing with the pins. We have already ported the WiringPi v2 library to ODROID-C2, as described in Chapter 4. Note that all of the GPIO ports are 3.3Volt, and the ADC inputs are limited to 1.8Volt. In addition, 7 I2S pins are available.

## **Serial console port**

Connecting to a PC gives access to the Linux console. You can monitor the boot process, or log in to the ODROID-C2 to perform system maintenance. Note that this serial UART uses a 3.3V interface, and it is recommended to use the compatible USB-UART module kit available from Hardkernel. A Molex 5268-04a (2.5mm pitch) is mounted on the PCB, and its mate is Molex 50-37-5043 Wire-to-Board Crimp Housing.

## **RTC (Real Time Clock) backup battery connector**

If you want to add RTC functions for logging or keeping time when offline, just connect the Hardkernel's RTC shield with a Lithium coin backup battery (CR2032 or equivalent) to the appropriate GPIO pins.



## **RTC Shield**

### **Gigabit Ethernet**

The Realtek RTL8211F is a highly integrated Ethernet transceiver that complies with 10Base-T, 100Base-TX, and 1000Base-T IEEE 802.3 standards.

### **USB MTT hub controller**

A GENESYS LOGIC GL852G is used to implement the 4-port Hub function which fully complies with Universal Serial Bus Specification Revision 2.0.

### **USB VBUS controller**

A NCP380 Protection IC is used for the USB power supply from OnSemi.

### **Power switch port**

You can add a slide switch or rocker switch on this port if you want to implement a hardware ON/OFF switch. If this port is open, the power is ON. If this port is closed, the power is OFF.

### **Power supply circuit**

Discrete DC-DC converters LDOs are used for CPU/DRAM/IO power supply.

### **Power protector IC**

The power protection is made possible through the use of the NCP372 over-voltage, over-current, reverse-voltage protection IC from OnSemi.



## *Heatsink*

Electronic components all generate heat while operating, and different components generate different levels of heat. Some components do not require any cooling, while others do. Complex components such as the ODROID-C2 processor may reach temperatures as high as 85°C. At high temperatures, the processor will throttle itself and operate slower so that the temperatures of components do not continue to increase. The ODROID-C2 comes pre-assembled with a black heatsink.



**Heatsink attached to the ODROID-C2**

Heat transfer from components to the surrounding air is related to the surface area available to transfer heat to the surrounding air. The processor of the ODROID-C2 provides a relatively small area to dissipate heat (approximately 1cm x 1cm). The metallic heatsink is much larger and is therefore able to dissipate more heat into the surrounding air than the processor itself.

## *Temperature Measurement*

To help verify that the heatsink is effective, it is useful to get temperature readings during operation. The processor's built-in sensor can be used for this. It would be ideal to have the processor working under a realistic load. It is assumed that your unit has been through the initial setup procedure and is running the latest official Ubuntu image from Hardkernel.

First, install the sysbench utility needed for the test:

```
$ sudo apt-get install sysbench
```

# Chapter 1

Ensure that there are no unnecessary user-initiated applications running. Start two terminal windows. In the first terminal window, type in the following command to obtain the baseline temperature:

```
$ cat /sys/class/thermal/thermal_zone0/temp  
48000
```

If the value has 5 digits, then that value should be divided by 1000 to get the temperature in Celsius. The baseline temperature in this case is 48 degrees Celsius.

In the second terminal window, type in the following command, to increase system load:

```
$ sysbench --test=cpu --num-threads=4 --cpu-max-prime=100000000  
run
```

After waiting at least 5 minutes, re-enter the temperature measure command into the first window:

```
$ cat /sys/class/thermal/thermal_zone0/temp  
59000
```



## Measuring the operating temperature of the ODROID-C2

The temperature is 59 degrees Celsius. The heatsink will be warm to touch as expected. Make sure to touch a grounded metal object first to discharge all static.

## *Frequently Asked Questions*

### **What's an ODROID?**

ODROID means Open + Android. It is a development platform with hardware as well as software.

### **What SoC are you using?**

The SoC is an Amlogic S905 quad-core.

### **What GPU does it include?**

An ARM Mali™-450 GPU (3 Pixel-processors + 2 Vertex shader processors).

### **How does it boot?**

You may boot from either microSD card or eMMC module.

### **Can I buy OS pre-loaded SD card or eMMC?**

Yes, they are available from Hardkernel as well as other distributors.

### **Can I add extra RAM memory?**

No. The RAM is not removable or swappable.

### **How can I get the Ubuntu or Android BSP source code?**

The U-boot, Kernel and OS source code are released via Github from the first shipping date at <http://github.com/hardkernel>.

### **Does it play 1080p/4K content well on Ubuntu Linux?**

Yes, H.265 4K/60FPS and H.264 4K/30FPS video clips are playable with Kodi (formerly XBMC) in most cases.

### **How can I root Android?**

Android for the ODROID is unlocked and rooted by default for development.

### **How can I install the Google Play Store?**

It is very simple. Just download an installer from <http://bit.ly/1rvYvLw>, and follow the instructions inside the application.

### **Can I get PCB layout file and gerber file?**

The ODROID project is not a full open source hardware, and only the schematics are released to the public.

# Chapter 1

## What peripherals are available?

The following peripherals are available from the Hardkernel store and certified distributors: (<http://bit.ly/1hD2dIn>)

- *Power supply: DC 5V/2A*
- *HDMI Type A-A 1.4 and 2.0 (4K) cables*
- *USB-DC Plug Cable 2.5mm*
- *MicroSD 8GB, 16GB cards (with pre-installed Android/Ubuntu OS)*
- *eMMC 8GB, 16GB, 32GB, 64GB (black) module (with a pre-installed Android/Ubuntu OS)*
- *ODROID-VU7 800x480 HDMI touch-sensitive display*
- *ODROID-VU7+ 1024x600 HDMI touch-sensitive display*
- *Clear case (black, blue, clear)*
- *WiFi Modules 0, 3, 4*
- *USB3.0 to SATA Bridge Board*
- *USB OTG cable*
- *USB CAM 720p*
- *HiFi Shield Plus*
- *USB Bluetooth Module 2*
- *USB GPS Module*
- *Micro-USB to USB cable*
- *USB IO Board*
- *Universal Motion Joypad*
- *USB-UART module kit*
- *DC Plug Cable Assembly 2.5mm*
- *ODROID-Show2*
- *CAT6 Ethernet cable*
- *GPIO breakout board with C Tinkering kit*
- *16x2 character LCD shield*
- *myAGRS+*
- *Weather Board 2*
- *USB Audio adapter*
- *VuShell (blue, clear)*
- *Infrared (IR) remote control*
- *Xprotolab Plain*
- *UPS3 Backup Battery*
- *RTC addon module*

## What are the minimum peripherals to run Ubuntu or Android with an ODROID-C2?

The following peripherals are not included with the basic ODROID-C2 board, but may be purchased separately from the Hardkernel store, or from a certified distributor:



# Chapter 1

- *Compatible 5V/2A power supply*
- *HDMI monitor for output device, HDMI Cable and micro-USB cable for touch*
- *Keyboard and Mouse for input device*
- *MicroSD card for the operating system and user data (8GB or higher is required)*
- *MicroSD card reader to install the operating system image*
- *Ethernet cable*

The following additional peripherals are highly recommended:

- *An eMMC module instead of the microSD card due to much higher I/O speed*
- *USB-UART module kit to debug issues that may prevent proper functioning of the unit*

## **I am a platform/OS developer. What should I buy with ODROID-C2?**

Along with those listed earlier, you should also obtain a micro-USB cable for fastboot and ADB interface.

## **I have a USB-Serial converter. Should I buy your USB-UART module kit?**

We strongly recommend using our USB-UART module kit because it includes the proper connector and is voltage matched to work with the ODROID-C2.

## **How do I access the Internet with an ODROID-C2?**

Use a 10/100/1000 LAN Ethernet connection, or purchase the WiFi module kit for a wireless connection.

## **What display can I use?**

There is an HDMI Type-A port on the ODROID-C2. Below is a list of resolutions that are currently supported:

- *3804x2160 (UHD/4K)*
- *1920x1200 (WUXGA)*
- *1920x1080 (1080p)*
- *1280x720 (720p)*
- *720x480 (480p)*
- *720x576 (576p)*
- *1360x768 (768p WXGA)*
- *1280x800 (800p WXGA)*
- *1280x1024 (SXGA)*

- *1024x768 (XGA)*
- *1024x600 (WSVGA: VU7+)*
- *800x600 (SVGA)*
- *800x480 (WVGA: VU7)*
- *640x480 (VGA)*

## **Can I use an HDMI to DVI converter?**

An HDMI-DVI converter may work with many DVI monitors, but a few of them will not work due to compatibility issues. We recommend our 7-inch HDMI LCD kit (ODROID-VU7 or ODROID-VU7+) with a 5-point capacitive touch screen if you want to develop a modern user interface.

## **Is there a touch screen on the 9-inch HDMI LCD Kit?**

No.

## **Is sound over HDMI supported?**

Yes.

## **Is SPDIF and optical pass-through supported?**

It is possible with our USB to SPDIF interface, available at the Hardkernel store, but it only works with Kodi (XBMC) on Ubuntu. The Android platform does not yet support the 5.1 channel pass-through.

## **Is there any analog audio output or input?**

Not on the board itself, but you can use our USB Audio Adapter for analog audio or the HiFi Shield Plus.

## **Is there an I2S port to connect a HiFi DAC?**

No, but an ODROID Hi-Fi Shield may be used with the ODROID-C2.

## **What is an eMMC 5.0 module?**

The eMMC 5.0 module is a NAND flash-based storage IC which is mounted on a custom PCB for easier upgrade/replacement, and is much faster than SD card. The read/write speed of a typical eMMC 5.0 is approximately 125MB/sec, while a microSD UHS-1 card is approximately 15MB/sec and 30MB/sec respectively.

## **Do you have a SATA port?**

No, but you can use Hardkernel's USB3-to-SATA bridge device.

## **Which power adapter should I use?**

The ODROID-C2 consumes less than 0.5A in most cases, but it can

climb to 2A if many passive USB peripherals are attached directly to the main board. It is recommended to use the Hardkernel 5V/2A PSU or USB-to-DC Plug cable with a 5V/2A charger. Due to the limited power output from a computer's USB port, we suggest only powering the ODROID-C2 with a good quality 5V/2A PSU at the least. In some cases, since peak power requirements go slightly beyond 10Watts, it may be safe to even use a compatible quality 5V/3A PSU.

### **What operating system (OS) does it use?**

We recommend Android and Ubuntu as our default distribution. The OS is stored on the SD card or eMMC module.

### **Which Android and Ubuntu version are included?**

Both Android 5.1 Lollipop and Ubuntu 16.04 are available, which both run on the Linux kernel 3.14 LTS.

### **Which OpenGL and OpenCL are included in Android and Ubuntu?**

OpenGL ES 1.1 and 2.0 are included. OpenCL is not supported.

### **Where is the OpenGL ES SDK?**

Refer to the ARM Mali Developers site at <http://bit.ly/1FRJEi0> for information on OpenGL ES.

### **Is the full source code open, and can I build it by myself?**

Yes, the bootloader, kernel and OS platform source code are available. However, the GPU userland drivers are in binary format due to ARM's policies.

### **Do I need to patch for optional accessories that are purchased from the Hardkernel store?**

No, they will work out-of-box with the kernels supplied with the Operating System.

### **Does the device support networking?**

Yes, there is 10/100/1000 RJ45 Ethernet port.

### **Is there WiFi?**

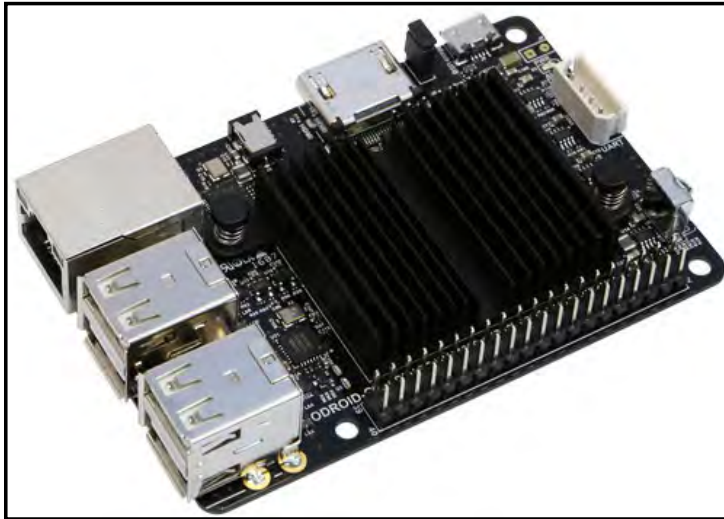
It is available via an optional USB dongle.

### **I still have more questions!**

You may browse <http://bit.ly/1Trq5Ef> and, if in need of additional information, you can also ask questions in our user support forums at <http://forum.odroid.com>, providing sufficient background information.

# Getting Started

Given the introduction to the ODROID-C2 development board and the fact that you obtained it and the needed peripherals, you must be curious to get started with your ODROID-C2 now. To be able to use the computer on the Internet, we first need an idea of how the ODROID-C2 will fit into a home computing network.



## Basic C2 board

### *Home Computing Network*

A typical simplistic home computing network environment includes the following, as shown in the diagram on the following page:

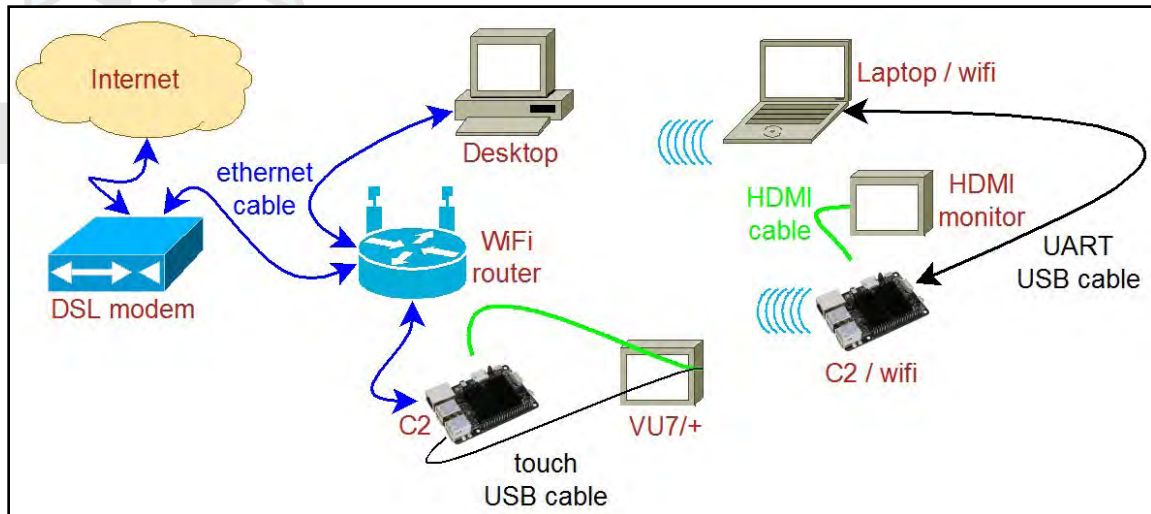
- A Digital Subscriber Line (DSL) modem, typically provided by your Internet Service Provider (ISP), which connects your network to the Internet,
- A single (2.4GHz) / Dual (2.4 and 5GHz) band 802.11b/g/n WiFi router, with at least four (4) Gigabit Ethernet (gigE) ports and an additional Wide Area Network (WAN) port, where the WAN port connects to the DSL modem,
- An Intel/AMD based laptop (running MS Windows 7+ or Ubuntu 16.04 LTS+ Linux), that connects via wired ethernet or WiFi to the router, and
- At least one C2, that connects (wired ethernet or WiFi) to the router, which uses a High-Definition Multimedia Interface (HDMI) monitor (possibly touch-capable, such as an ODROID-VU).

After preparing the boot media, it is theoretically possible to use a C2 in a network disconnected from the Internet. However, many use cases require an always-ON functioning secure connection to the



# Chapter 2

Internet. In either case, it is presumed you have a fully functioning intranet (i.e., the internal network within the home). There are numerous online guides to setting up a secure network and as such, the setup of such a network is beyond the scope of this guide.



## Typical home computing network

### *Preparing the Work Area*

Your work area should be similar to the following:

- A well lit and ventilated area,
- A flat-top desk spanning at least 5' x 2' (preferably 6' x 3'), with a height of 32" or appropriate to optimize posture and minimize overall fatigue,
- A freely available area of non-conductive surface of at least 2' x 2' (or even 3' x 2'), preferably covered by a well-grounded anti-static mat with you wearing the grounding wristband/cable attached to mat. This safeguards the handling and placement of unprotected electronics such as C2 board, etc., on this area,
- A desktop with accessible monitor, keyboard/mouse or a laptop to prepare the boot-media for the C2 and debug the C2 bringup process,
- If needed, a 4-port USB3 (USB2 compatible) hub attached to the computer within the reach of the free area - for easy image flashing using a USB-based SD card or eMMC module reader/writer and attaching the USB UART debug cable, and
- A well-grounded surge-protector with 6 or more outlets, within reach of the free area.

The above setup will go a long way in ensuring a safe workspace for you and your projects involving boards such as the C2.

## *Flashing an Image*

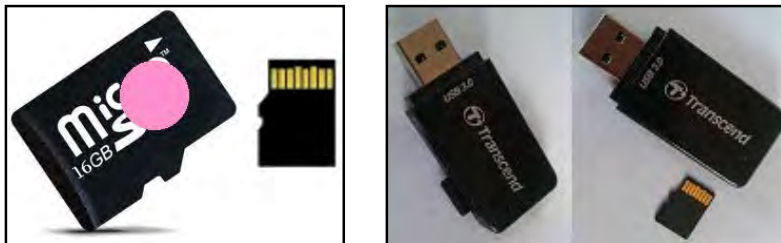
All SBCs require an operating system (OS) for booting. The operating system will be transferred to and reside on boot media, either built on to the board or attachable to the board. The process of placing the operating system on the boot media is termed flashing. Given this process, it is obvious that another computer system will be needed to flash the image on to the boot media, before the C2 can boot up.

The flexible C2 design allows you to use a boot media device that will be attached to it through an appropriate receptacles on the board. If you refer to the annotated board image, you can see that the C2 supports two boot media types: microSD card or an eMMC module. They can be obtained from Hardkernel directly, either with Linux or Android, pre-installed. If you purchased one such device and if it has a factory-installed image, you can skip the flashing steps in this section and proceed to the next section.

## *Boot Media*

### **microSD/microSDHC card**

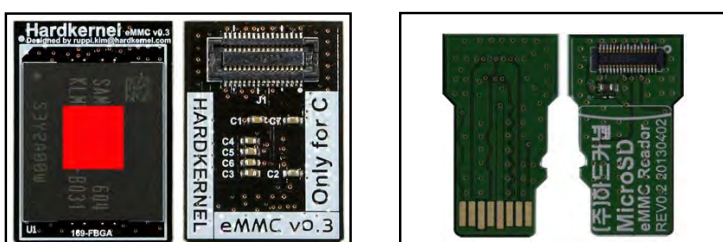
- Slower, less longevity
- 8GB, 16GB, 32GB
- Linux, Android
- Requires USB SD card reader/writer



**16GB Linux MicroSD card and USB reader/writer**

### **eMMC version 5.0 module**

- Faster, more longevity
- 8GB, 16GB, 32GB, 64GB, 128GB
- Linux, Android
- Requires eMMC/micro-USB adapter
- Requires USB reader/writer



**16GB Linux eMMC and adapter**

§ = Available from Hardkernel

# Chapter 2

Development boards such as the ODROID-C2, by definition, will at some point either require an OS upgrade or a simple reflash to revert to a known good base state.



## USB microSD card adapter

No matter which boot media you select for your C2, you will need an additional device to perform the flashing process, which is called a micro-USB card reader / writer. It should be compatible with your host desktop or laptop (flashing computer) and with the OS running on it. Above is shown one such USB3 device. If you recall in the previous section, we recommended the placement of a USB3 hub close to your free work area. This hub, if present, is where you would connect the micro-USB card reader / writer. Now you may see the point of the suggested hub location - basically, it allows for frequent flashing of the boot media and its convenient attachment to the C2. If all points in the connection are USB3 type, you will get the fastest possible read / write speeds. USB2 compatibility will allow one or more points to be of type USB2, but will result in lower speeds.

Start by using the designated non-conductive free work area for all subsequent activities. In the microSD receptacle of the USB SD card reader / writer, look for the metal contact pins. Then, if you are using a microSD card to flash the image, align the metal strips of the microSD card to make contact with the pins. Insert the microSD card into the USB SD card reader / writer. The image below indicates the alignment when using the Transcend USB3 microSD card reader / writer model. The alignment may be different with your reader / writer model.

However, if you are using an eMMC module to flash the image, you first need to attach the eMMC module to the eMMC/microSD USB adapter. Place the eMMC module and the adapter such that the 34 pin female/male connectors overlap and gently press them together.

Align the metal strips of the microUSB adapter to make contact with the pins in the microUSB receptacle of the USB SD card reader / writer. Insert the assembly by pressing firmly on the eMMC module until it clicks into place.



## Preparing the eMMC module

The above image shows the alignment when using the Transcend USB3 microUSB card reader / writer model. The alignment may be different with your own microUSB USB adapter. Next, ensure that the computer you are about to use to flash the image, has a functioning connection to the Internet. Now that you have the USB SD card reader / writer with the boot media inserted, attach the male USB end of the USB SD card reader / writer to the appropriate (USB3 or USB2) female port on the computer/USB3-hub.

The ODROID-C2 may not be able to boot immediately with some models of microUSB cards. However, you can still use any brand or speed of memory card as storage. Sandisk Class 10 or UHS-1 cards are working well with ODROID-C2, and below is a list of the memory cards that have been tested:

### Fully compatible microUSB cards

- SanDisk microUSBHC Memory Card
- SanDisk Ultra microUSBHC/microUSBXC UHS-I Memory Card
- Samsung SDXC EVO Memory Card
- Toshiba Exceria Memory Card

### Partially compatible microUSB cards

The following cards may not be able to boot immediately on ODROID-C2, but instead take about 30 seconds to start the boot up process. Once the boot processing is done, and the operating system is running, they perform normally:

- Transcend microUSBXC Class 10 UHS-I 300x (Premium)



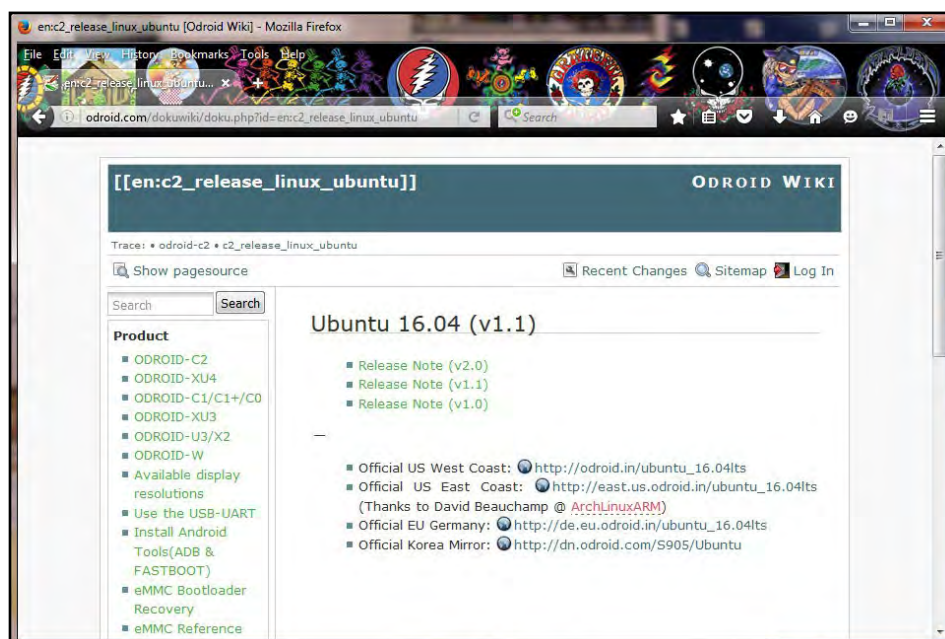
- Transcend microSDHC Class 10 (Premium)
- Transcend microSDHC Class 6 (Standard)
- Samsung microSD Pro Class 10
- Samsung microSDHC EVO Class 10
- Kingston microSD Class 4
- PNY Turbo Performance Class 10

Class 2/4/6 microSD cards have not been tested with the ODROID-C2, therefore it is likely that the ODROID-C2 will not boot properly with them. We strongly recommend to use Class 10/UHS-I microSD cards, and a SanDisk microSD for full compatibility.

## *Flashing an Image on a Microsoft Windows 7+ Computer*

As soon as you attach the microUSB card reader / writer for the first time, the Windows OS will sense the presence of a new USB device through its plug-and-play, (PNP) infrastructure. It will obtain the device identifiers from the device and search through its local database. If found, it will install it for the first time. If not found, it will request you to provide a location for the driver or seek to find it over the Internet. Once the driver is obtained for the first time, it will install it and request permission to reboot the system, which should be allowed. Once installed, it will use the driver on all subsequent sessions.

Once the system is up, launch a web-browser such as firefox and navigate to <http://bit.ly/2b58GEe>. The browser will list all of the available Ubuntu images and the various download locations. You will see a highlighted list as shown in the screenshot below.

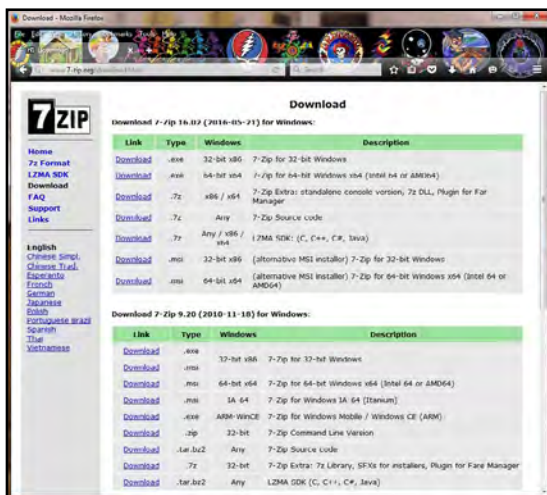


**List of available Ubuntu images.**

# Chapter 2

Each of the entries in the list is also a link. Click the download link for most recent version of the software from the server closest to your location. Note that a compressed image file will be downloaded, so you will need to uncompress this file before flashing it to the boot media. To do so, you will need a specialized utility to uncompress the downloaded file. A reliable utility to uncompress such files is the 7-zip program. Launch a web-browser such as firefox and navigate to <http://www.7-zip.org/download.html>, which will list all of the available installation files for this utility. Click on a link that corresponds to your host computer's OS. Details of your OS can be found via the `msinfo32.exe` or `systeminfo.exe` commands, when run in cmd window.

The Windows OS used for this guide is a 64-bit type. Click the download link that corresponds to the 64-bit Windows OS, in this case, `7z1602-x64.msi`.



## List of 7-zip installation files

After the file is downloaded, run it and select the default options, and wait for 7-zip to finish installing. Next, launch the Windows Explorer application and browse the download directory. Select the downloaded compressed image file and right-click the file with the mouse. Select the 7-zip menu option and under that select the Extract Here option. After the uncompress process, it will result in the image file that can be flashed. In this example, it will be `ubuntu64-16.04lts-mate-odroid-c2-20160525.img`.

Another essential utility is one that will be used to flash the uncompressed image file, such as the Win32DiskImager utility. Hardkernel has an updated version of it that adds a useful step to the flash process, which clears the boot media, prior to writing the image. This version of Win32DiskImager.exe can be downloaded from the following URL:

[http://dn.odroid.com/DiskImager\\_ODROID/Win32DiskImager-odroid-v1.3.zip](http://dn.odroid.com/DiskImager_ODROID/Win32DiskImager-odroid-v1.3.zip)

# Chapter 2

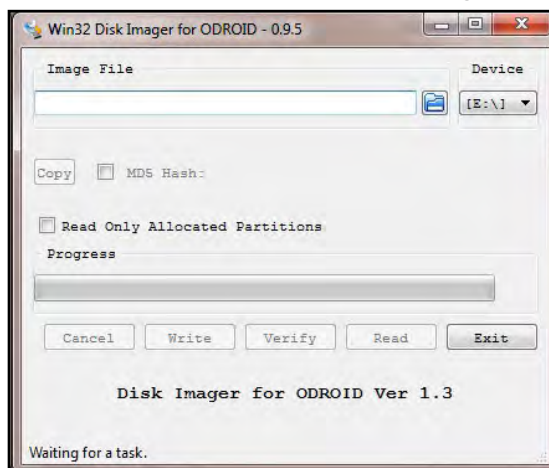
Uncompress the compressed image file using the 7-zip application, which generates the directory win32diskimager2-binary that includes the Win32DiskImager2.exe application. Move the entire directory to C:\Program Files (x86):

```
C:\Program Files (x86)\win32diskimager2-binary\
```

You will need to be logged in to Windows OS as an administrator to move this directory. Then, create a shortcut for the application on the desktop using the following link:

```
C:\Program Files (x86)\win32diskimager2-binary\Win32DiskImager2.exe
```

Next, change the properties of the shortcut (right-click of mouse) so it will run with Administrator privileges. Finally, double-click this shortcut in order to launch the Win32DiskImager2.exe application. Click the folder (directory) icon to browse and select the directory that has the uncompressed image file.



## Flashing an image using Win32DiskImager2

Select the uncompressed image file. Note that the application automatically selects the device (G:) that corresponds to the boot media in the microUSB card reader / writer. The image above shows the configuration. Click on the Write button to start the flash process.

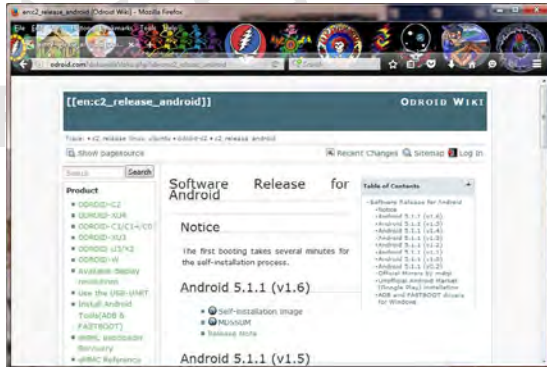
You will be notified of the completion of the flash process. Launch the Windows Explorer application and right-click the device that was used in the flash process by Win32DiskImager2.exe with the mouse and select the Eject option. When permitted, remove the microUSB card reader / writer from the USB port of the computer. Remove the boot media and place it on the non-conductive surface.

The above steps are to flash the Linux image. What if you wish to flash an Android image? While you would download a different compressed

# Chapter 2

file, all subsequent steps would be that same as those listed above.

Launch the browser once again and navigate to <http://bit.ly/2be993R>, then look for the latest Android image and click the link to download it. The screenshot below shows the webpage that lists the Android images.



## List of Android images for the ODRROID-C2

Once the compressed file has been downloaded, uncompress it and flash it, using the steps listed earlier in this section.

### *Flashing an Image on a Linux Ubuntu 16.04 LTS computer*

In your Ubuntu Linux desktop, launch a terminal window. Create a working directory:

```
$ cd ~ && mkdir linux-img && cd linux-img
```

You can download the compressed image using a browser like Firefox for Ubuntu using the same (Windows OS) steps described earlier. If you follow this method, you should use the `mv` command to move the downloaded compressed file to the working dir listed above. Instead, if you wish, you can use a Linux utility like `wget` right from within the working directory:

```
$ wget http://odroid.in/ubuntu_16.04lts/\
ubuntu64-16.04lts-mate-odroid-c2-20160525.img.xz
```

Wait for the download process to complete, then uncompress the file using the following command:

```
$ unxz ./ubuntu64-16.04lts-mate-odroid-c2-20160525.img.xz
```

This will result in an uncompressed image file called `ubuntu64-16.04lts-mate-odroid-c2-20160525.img`. Attach the USB SD card reader / writer with the boot media into an appropriate USB port. After a few moments, check for an entry that corresponds to the SD card or eMMC module using the following command (your results may differ):



# Chapter 2

```
$ df -h
Filesystem      Size  Used Avail   Use% Mounted on
/dev/mmcblk1p2  58G   5.8G  49G    11%  /
...
/dev/sdb1       30G   224K  30G     1%  /media/odroid/BLANK
```

This tells us that the write destination path to the media is `/dev/sdb`. Given this information, we can carefully construct the image copy command using appropriate input and output paths:

```
$ cd ~/linux-img
$ sudo dd \
  if=./ubuntu64-16.04lts-mate-odroid-c2-20160525.img \
  of=/dev/sdb \
  bs=1M conv=fsync
$ sync
```

The `dd` command is very powerful, so use it with a lot of care. If incorrect parameters (especially the `of` parameter) are used here, you could potentially ruin the OS installation of the computer. The `dd` command is hence often referred to as the disk destroyer command.

This step could take a while. So, wait for it to complete before proceeding. Once completed, remove the microUSB card reader / writer from the USB port of the computer. Remove the boot media and place it on the non-conductive surface.

The above steps are to flash the Linux image. To flash an Android image, simply download a different compressed file, and all subsequent steps would be that same as those listed above.

You would need to launch the browser once again and navigate to <http://bit.ly/2cMQ5K1>. Look for the latest Android image and click the corresponding Self-installation image link. In the example case here, we would be downloading the image called `selfinstall-odroidc2-eng-s905_5.1.1_master-230-v1.6.img.xz`.

If you wish to use the `wget` command instead, you can follow the steps listed earlier but using the appropriate Android URL. Once the compressed file has been downloaded, uncompress it and flash it, using the steps listed earlier in this section.

## *Flashing an Image on an OSX Computer*

The procedure for creating an image file using OSX is similar to Linux, with some small differences. First, download and install the `xzutils` package from <http://tukaani.org/xz/>, making sure to select the OSX binaries, and use it to uncompress the image:

```
$ xz -d selfinstall-odroidc2-eng-s905_5.1.1_master-230-v1.6.img.xz
```

# Chapter 2

Other differences are that the block size (bs) parameter for the dd command is in lowercase, and the USB adapter's device name is in the format /dev/diskX. Before writing the image to SD Card or eMMC module using the “dd” utility, run the command “diskutil” to determine the device name, then unmount it:

```
$ diskutil list
$ diskutil unmountdisk /dev/diskX
$ sudo dd of=/dev/diskX bs=1m \
  if=./selfinstall-odroidc-eng-s805_4.4.2_dev_master-195-v1.2.img
```

## *Inserting the eMMC Module or SD Card*

Now that the desired image has been flashed on to the boot media, it's time to insert the boot media into the appropriate port on the ODROID-C2.

Ensure that the work area is clear of unnecessary items. Wear the grounded wristband. Touch a metallic surface, like that of your desktop. Then, open the ODROID-C2 packaging box. Place the anti-static bag containing the ODROID-C2 board, on to the non-conductive work surface. Open the bag and remove the board, holding the edges. Avoid touching any of the components, peripheral connectors, ports, exposed circuits or on board solder. Place the board on the work surface, with the bottom side up. Check the annotated diagrams and follow the instructions from Chapter 1 for inserting the eMMC module or SD card.

## *Powering Up*

Before powering the board, attach all peripherals as described in Chapter 1. Insert the power cable, which will be indicated by the red LED lighting up, and wait for the image to boot to desktop. It will take anywhere from 20 seconds to 2 minutes for the desktop to appear, depending on the operating system and the type of boot media used.

## *Troubleshooting*

### **Red LED not glowing**

If the red LED is not glowing, power is not being supplied to the board. Check the connections, and verify that the plug is inserted properly. Also, make sure that a 5V/2A power supply is being used, and that it is plugged in.

### **Blue LED not flashing or glowing**

If the blue LED is not flashing or glowing at all, something went wrong with the image writing or the card is not correctly inserted. Try to download, extract and write the image again. If that doesn't solve the issue, try downloading and flashing a different image.

# Chapter 2

## Blue LED solid glowing

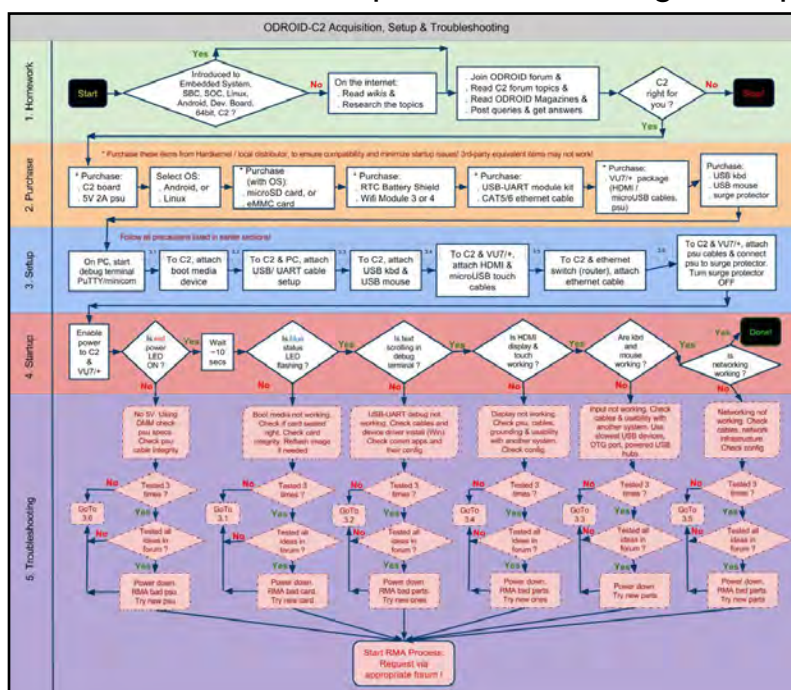
The C2 has panicked or frozen, and a reboot is required.

## Mouse/Keyboard not working

Ensure that it's correctly plugged in, or try a different USB port. You can change it while the C2 is running.

## No desktop image

- Be patient, some SD cards require several minutes to boot and anywhere from 30 seconds to 2 minutes to show the first image. Refer to the microSD card compatibility list above.
- Verify the HDMI cable is correctly plugged in.
- Edit the boot.ini on a PC and select the correct resolution, see Chapter 1 for details.
- Some HDMI cables have compatibility issues, make sure to use an official Hardkernel HDMI cable.
- Some monitors have non-standard EDID functions. In this case, you need to select the non-EDID mode in the ODROID configuration utility.
- Some cable have a shorter plug, and it may be necessary to remove the plastic case for a better connection.
- If there is a USB host port on the monitor, connect a microUSB cable from the ODROID to the monitor, which creates a ground path between the TV and the device.
- Add a short jumper wire between the outer casing of the microHDMI and micro-USB ports to create a ground path.



## Acquisition, setup and troubleshooting

# Chapter 2

- Before requesting an RMA, research the relevant forums and verify that a solution or workaround doesn't already exist for your issue.
- If your issue has not been previously encountered by anyone, create a new issue and post it on the forums at <http://forum.odroid.com>, then carefully read any feedback and follow the recommendations. Provide as many details as possible, indicating not only your software and hardware environment, but also the exact steps to recreate the issue.

## *Configuring Linux*

After booting the ODROID to a Linux desktop, the screen resolution should first be updated to match the refresh rate of the monitor. To do so, edit the `/media/boot/boot.ini` file located on the Linux boot partition using a text editor:

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

After the file has loaded, uncomment the following lines after commenting out the current selections. In the following example, the 1080p resolution at 60hz refresh rate has been selected, which is the default configuration on first boot:

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

# 480 Lines (720x480)
# setenv m "480i60hz" # Interlaced 60Hz
# setenv m "480i_rpt" # Interlaced for Rear Projection Televisions
60Hz
# setenv m "480p60hz" # 480 Progressive 60Hz
# setenv m "480p_rpt" # 480 Progressive for Rear Projection Tele-
visions 60Hz

# 576 Lines (720x576)
# setenv m "576i50hz" # Interlaced 50Hz
# setenv m "576i_rpt" # Interlaced for Rear Projection Televisions
50Hz
# setenv m "576p50hz" # Progressive 50Hz
# setenv m "576p_rpt" # Progressive for Rear Projection Televi-
sions 50Hz

# 720 Lines (1280x720)
# setenv m "720p50hz" # 50Hz
# setenv m "720p60hz" # 60Hz

# 1080 Lines (1920x1080)
# setenv m "1080i60hz" # Interlaced 60Hz
setenv m "1080p60hz" # Progressive 60Hz
```

# Chapter 2

```
# setenv m "1080i50hz" # Interlaced 50Hz
# setenv m "1080p50hz" # Progressive 50Hz
# setenv m "1080p24hz" # Progressive 24Hz

# 4K (3840x2160)
# setenv m "2160p30hz" # Progressive 30Hz
# setenv m "2160p25hz" # Progressive 25Hz
# setenv m "2160p24hz" # Progressive 24Hz
# setenv m "smpte24hz" # Progressive 24Hz SMPTE
# setenv m "2160p50hz" # Progressive 50Hz
# setenv m "2160p60hz" # Progressive 60Hz
# setenv m "2160p50hz420" # Progressive 50Hz with YCbCr 4:2:0 (Re-
quires TV/Monitor that supports it)
# setenv m "2160p60hz420" # Progressive 60Hz with YCbCr 4:2:0 (Re-
quires TV/Monitor that supports it)

### VESA modes ###
# setenv m "640x480p60hz"
# setenv m "800x480p60hz"
# setenv m "800x600p60hz"
# setenv m "1024x600p60hz"
# setenv m "1024x768p60hz"
# setenv m "1280x800p60hz"
# setenv m "1280x1024p60hz"
# setenv m "1360x768p60hz"
# setenv m "1440x900p60hz"
# setenv m "1600x900p60hz"
# setenv m "1680x1050p60hz"
# setenv m "1920x1200p60hz"
```

Supported resolutions are:

- 4K UHD (*2160p*)
- 3440×1440
- 2560×1600
- 2560×1440
- 2560×1080
- 1920×1200 (*WUXGA*)
- 1920×1080 (*1080p*)
- 1280×720 (*720p*)
- 720×480 (*480p*)
- 720×576 (*576p*)
- 1280×800 (*800p for ODROID-VU*)
- 1280×1024 (*SXGA*)
- 1024×768 (*XGA*)
- 800×480 (*WVGA, DVI supported for ODROID-VU7*)
- 640×480 (*VGA*)
- 1680×1050
- 1600×1200 (*on Ubuntu only so far, to be released for Android soon.*)
- 1600×900



- 1440×900
- 1360×768
- 1024×600
- 800×600

*(DVI supported for ODROID-VU7 Plus)*

For more information on configuring the screen resolution, please refer to the ODROID-C2 Wiki at <http://bit.ly/2cUNeli>.

The SSH daemon is enabled by default in the Ubuntu template, and the username:password is odroid:odroid. The root password is also odroid.

Finally, the language and timezone settings may be configured with the following command, after installing the desired language in the “Language Support” settings application:

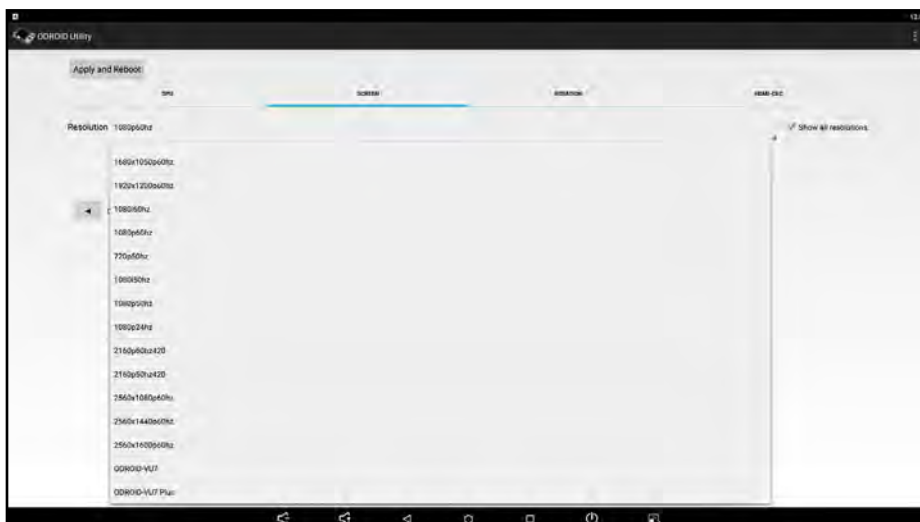
```
$ sudo dpkg-reconfigure locales
```

After the command completes, drag the selected language to the top of the list in the “Language Support” settings. The timezone may be selected by typing the following command:

```
$ sudo dpkg-reconfigure tzdata
```

## *Configuring HDMI Resolution in Android*

Once the Android desktop has loaded, run the ODROID Utility app, which is pre-installed on all Hardkernel Android images, and select the desired CPU frequency, monitor resolution and orientation. Other options may also be selected using the ODROID Utility app, including the CEC options which allow the use of a remote with applications such as Kodi.



**Configuring HDMI resolution with the Android ODROID Utility app**

# Chapter 2

To set the timezone and other configurable options, use the built-in Settings panel that comes with the Android installation.

## *Powering Down*

Shutting down the ODROID-C2 is very important to prevent damage to the microSD card or eMMC module. In Android, powering down is done by selecting the power button icon at the bottom of the desktop. In Linux, powering down may be done by either selecting the shutdown option from the Applications menu, or by typing the following into a Terminal window:

```
$ sudo shutdown -h now
```

Wait until the blue LED goes out, which indicates that all system activity has completed. It is now safe to unplug the ODROID-C2 power supply, remove the boot media, and detach the peripherals.

# Operating Systems

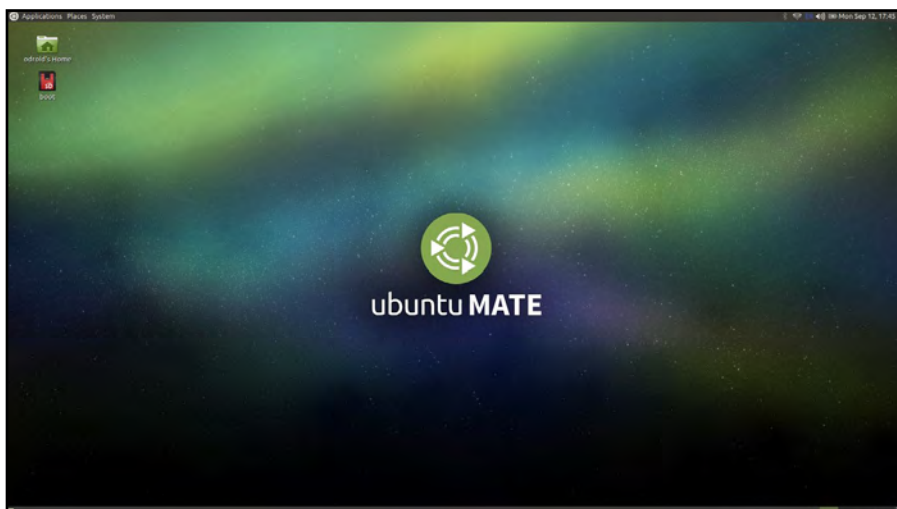
The ODROID-C2 can run many free, full-featured, Linux-based, open-source operating systems. Two of the most popular, Ubuntu and Android, are available for download from Hardkernel's website, while many other flavors of Linux may be compiled from source, such as ARCH Linux, FreeBSD, Fedora, CentOS, OpenSUSE, Slackware, and Mint. All of them have a common customized kernel which allows the operating system to communicate with the ODROID hardware. Hardkernel publishes kernels that are specific to the ODROID architecture, and maintains a repository where they may be downloaded and installed as improvements are made.

## *Ubuntu*

Ubuntu is based on the Debian operating system, and both offer a desktop as well as a large library of applications that may be freely installed, used, and modified. The main GUI tool for obtaining applications on Ubuntu is Synaptic Package Manager, although the command line tool called apt-get is often used by advanced users and scripts in order to streamline the installation process. Applications are stored on servers called repositories, which allow you to receive updates and new versions of software automatically.

## **Booting Up**

Ubuntu can be ordered as a pre-installation on the eMMC or microSD card that comes with the ODROID-C2. Please refer to Chapter 2 for instructions on downloading and flashing Ubuntu onto your C2's boot media. Once the C2 has been powered on, it will boot to a desktop similar to the one shown in the below image.

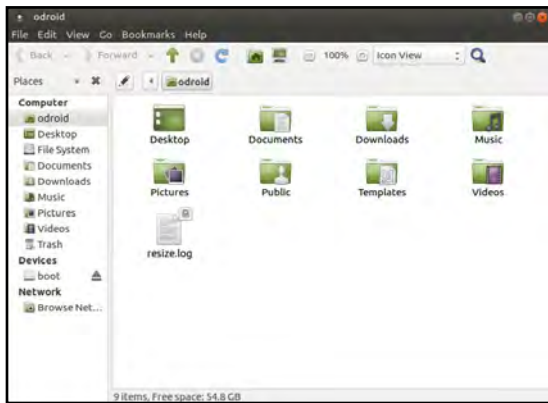


**Ubuntu Desktop**

# Chapter 3

## Linux Basics

Ubuntu and Debian, along with most Linux distributions, have a home folder structure where documents, files and media may be stored for each user of the system. For example, downloaded files from the Internet are automatically saved to the Downloads folder, and the other folders may be used to organize various public and private files such as pictures, video, and word processing documents. There are also many special folders that are normally hidden from view that start with a dot (.), where settings for various applications are stored. To view the Home folder, start the File Manager application from the Applications menu in the bottom left corner of the screen, which will look like the screenshot below.



### Home folder structure as viewed in Ubuntu File Manager

Both Ubuntu and Debian offer many different types of desktop environments, such as Xubuntu, LXDE (Lubuntu), Gnome, KDE (Kubuntu), and Blackbox, to name a few, all of which are interchangeable. Each one includes specific applications that are useful for different types of users. For instance, a typical desktop user who wishes to browse the Internet, use LibreOffice to create, edit and read documents, and listen to music or watch videos might choose Lubuntu, Kubuntu, or Xubuntu since they include a robust Applications menu along with a Software Center that facilitates downloading new programs.



### Home folder structure as viewed in Ubuntu File Manager

If the ODROID-C2 is being used as part of robotics project, a more lightweight environment such as Blackbox may be more suit-



# Chapter 3

able, in order to free up the amount of memory available for the main controller application. Other specialized environments are also available from Synaptic Package Manager, including Ubuntu Studio, which includes many applications that are primarily used for creating and producing music, videos, digital artwork, and photography.

## Kernel

At the heart of the Linux system is the kernel, which is responsible for allowing the desktop environment to communicate with the hardware through a common interface. Each ODROID has its own customized kernel code that is modified, tested and published by Hardkernel. Programmers may also download the source code from the official repository at <http://github.com/hardkernel> and make any modifications or contributions that they wish.

The Linux kernel is compatible with all versions of Linux, so any operating system that has been compiled for the ARM hard-float architecture (ARMHF) may be converted to run on the ODROID by installing an ODROID-C2 kernel and modifying the boot partition. This makes Linux one of the most versatile operating systems available, since it can be configured to run on almost any device, including laptops, desktop PCs, smartphones, and ODROID microcomputers.

## Graphical User Interface (GUI)

All versions of Linux offer a command line interface (CLI), and sometimes include a window manager and desktop environment that is launched on startup. Most modern Linux systems such as Ubuntu use a library called X11 to create a windowing environment and provide graphics libraries so that users may interact with applications using a mouse. The ODROID-C2 includes a Mali 450 Graphics Processor Unit (GPU) which is controlled by X11, in conjunction with the Open Graphics Library (OpenGL), in order to render graphics on a 720p, 1080p or 2160p (4K) monitor. ODROIDs use a subset of the popular OpenGL library called OpenGLES, which is specifically designed to work with ARM processors, especially smartphones. Applications that are written for OpenGLES can use low-level graphics functions on the GPU chip itself in order to quickly and efficiently render graphics, resulting in a much faster and smoother user experience than using the CPU alone.

## 720p vs 1080p and 2160p

The ODROID-C2 model supports 2160p, 1080p and 720p monitor configurations, which are video resolutions commonly supported on LCD TV's and on many LCD computer monitors. 720p is sometimes referred to as HD, 1080p as Full HD, and 2160p (4K) as Ultra HD. The following comparison table lists the native display resolutions

# Chapter 3

as well as some sources that use these resolutions.

Standard	Resolution	Aspect Ratio	Refresh Rate	Typical Sources
2160p (4K)	3840 × 2160	16:9 (1.78:1)	50Hz, 60Hz	HDTV, Blu-ray DVD
1080p	1920 × 1080	16:9 (1.78:1)	50Hz, 60Hz	HDTV, Blu-ray DVD
1080i	1920 × 1080	16:9 (1.78:1)	50Hz, 60Hz	HDTV, Blu-ray DVD
720p	1280 × 720	16:9 (1.78:1)	50Hz, 60Hz	HDTV, miniDV

As a comparison, the standard DVD Video format uses a display resolution of 720 x 480 with a 4:3 aspect ratio (non-widescreen). This resolution is also referred to as D-1 video and is the standard used or miniDV digital camcorders.

## Progressive vs Interlaced Video

2160p (4K), 1080p and 720p are progressive video formats. For these formats, a complete video image is transferred for each frame of the video signal, so that a 60fps video displays 60 complete, full frames per second. This format is visually superior to interlaced video, and in general should be used when supported by your display.

1080i and 720i are interlaced video formats, and video interlacing is the historical standard for broadcast video. In interlacing, two frames (or more correctly, fields) are transferred for each frame of the video signal, so that a 60fps video has 30 complete, full frames per second. The display device sends the two fields to the screen in the horizontal odd and even (or interlaced) pixel positions. Given the visual persistence of the human eye, this reconstruction of full frames is undetectable to the viewer.

In general, faster frame rates result in a visually smoother video playback, and interlacing results in a halving of the native frame/field transfer rate because two fields are required for the full image. Frame rate and visual experience are subjective, however. Traditionally, movies produced with conventional film use a 24 frames per second rate, and many viewers prefer the cinematic experience that 24fps video provides. In fact, much of the professional video produced with high-end gear is actually shot at 60fps or higher, and later converted to video frame rates to provide the look and feel of cinematic film.

## Video Downconversion

When selecting a screen resolution for your C2, you should first consult the specifications for the display that you plan to use. In general, using the native resolution of your display will provide the best performance and quality.

For example, if you use a display that only supports native 720p, and you set your C2 to a 1080p screen resolution, it will probably work, but is not the ideal configuration. The mismatch in resolutions indicates

# Chapter 3

that your display is automatically performing a downconversion of the incoming video signal. Despite appearing to work OK, the C2 will be performing unnecessary work in order to create the 1080p output.

If a 720p video is viewed with this configuration, the C2 will upconvert the original 720p video by extrapolation to produce a 1080p output, only to have that 1080p video downconverted by interpolation back into 720p by the video display. In this case, setting the C2 screen resolution to 720p is the logical choice.

## Video Upconversion

When the C2 is set to a higher output resolution than the source video, an upconversion is performed by extrapolation to produce the higher resolution frame in the framebuffer. The average video play can display videos with a number of different source resolutions. These videos are upconverted by the C2 in order to fit the screen, and the user therefore has a uniform viewing experience.

However, if the C2 is used solely for such purposes as video editing, video effects, or historical restoration, and the source material resolution is lower than the maximum resolution of the video display, the user may wish to setup the video settings differently. In this case, matching both the C2 video settings and the video display settings to the source material's native format would be appropriate. With this configuration, The video playback would provide an image that is true to the original source material, as well as freeing up more of the C2's CPU bandwidth for other processing tasks.

## HDMI Overscan

If the display used on your C2 shows a slight cropping of the visible image on the screen, you may be experiencing overscan. This is not an uncommon problem, and especially so for LCD TV monitors. The fix is usually a simple one, and the underlying issue is most likely due to a setting with the LCD monitor. Some PC Monitors with HDMI inputs will also apply overscan to the HDMI input, assuming that a broadcast TV signal is being used.

## Why Do Monitors Have Overscan?

Monitors that are used for broadcast television usually have overscan enabled by default. This is a normal feature of TV monitors and has been present from the very beginning of television. Overscan is used to crop the edges of the video frame in order to remove any erratic or distorted edges that often exist with broadcast video. To the viewer, this results in a cleaner picture, and the overscan simply isn't noticed. For a computer display however, this can be an issue. For this reason, computer LCD monitors usually do not provide overscan,

and if they do have this feature, it is disabled by default.

## Disabling Monitor Overscan

Prior to attempting to fix overscan using the methods detailed below, first verify that the monitor resolution setting and the C2 screen resolution setting match. Unfortunately, there isn't a standard method or terminology for disabling overscan used by monitor manufacturers, so you may have to search thru the monitor menus a bit until you find the overscan setting.

## Display Setting Button on Remote Control

Using the LCD monitor remote control, first look for a display mode button labeled Display, Screen Mode, marked with a display icon, or marked with |<>| (for wide mode). If your remote has one of these dedicated display mode keys, pressing this key should allow you to cycle through the display modes.

## Finding Setting in Monitor Menus

If you cannot find a display mode button, you will need to enter the display's menu setting mode, usually marked Menu or Settings. You will need to search for the menu item that controls the overscan setting, and on some displays you will need to enable the Advanced mode.

The table below describes several possible menu locations for the overscan setting on different displays:

**Menu > Picture Mode > Aspect ratio**

Change to "Just Scan"

**Menu > Picture > Screen adjustment > Picture Size**

Change to "Screen Fit"

**Menu > View Mode**

Change to "Dot by Dot"

**Menu > Tools**

Change HDMI Source to "PC"

**Menu > Picture > Screen adjustment > Picture Size Screen**

Auto Config

**Menu > AV Preset > HDMI**

Change to "PC"

As you can see, finding the setting for your LCD monitor may take some time to locate. Other terms manufacturers are known to use for overscan disable include HD size, full pixel, unscaled, native, and 1:1.

On some monitor models, one of the HDMI inputs is intended to work with a PC and will disable overscan for that input (often HDMI 2). You can also look at the monitor's HDMI connectors for a label such as PC Input, or read the monitor users manual to find the correct input



to use if applicable.

If you have used one of the above methods to change your LCD monitor's display mode and it does not help, or if your monitor does not provide an overscan adjustment, use one of the following methods to adjust the C2 video output directly.

## Adjusting Linux Display Output

Installing `xrandr` and using the `transform` command can help fix an overscan issue in Ubuntu. Here is a typical adjustment using `xrandr`, which should be typed into a Terminal window:

```
$ xrandr --output HDMI-1 \  
  --transform 1.05, 0, -35, 0, 1.05, -19, 0, 0, 1
```

## Adding an ODROID-VU7 Plus touchscreen

The ODROID-VU7 Plus is a 10-point multitouch display that is capable of 1024x600 resolution. To use the ODROID-VU7 Plus with Linux, connect the USB cable from the VU7 Plus to the ODROID-C2's USB port, then use a Type A HDMI cable to connect the display. Then, edit the `/media/boot/boot.ini` file on the boot partition as detailed in Chapter 2, and uncomment the 1024x600 resolution and DVI mode entries, making sure to comment out the current selections:

```
# Possible screen resolutions  
# Uncomment only a single line! The line with setenv written.  
# At least one mode must be selected.  
  
# 480 Lines (720x480)  
# setenv m "480i60hz" # Interlaced 60Hz  
# setenv m "480i_rpt" # Interlaced for Rear Projection Tele-  
visions 60Hz  
# setenv m "480p60hz" # 480 Progressive 60Hz  
# setenv m "480p_rpt" # 480 Progressive for Rear Projection  
Televisions 60Hz  
  
# 576 Lines (720x576)  
# setenv m "576i50hz" # Interlaced 50Hz  
# setenv m "576i_rpt" # Interlaced for Rear Projection Tele-  
visions 50Hz  
# setenv m "576p50hz" # Progressive 50Hz  
# setenv m "576p_rpt" # Progressive for Rear Projection Tele-  
visions 50Hz  
  
# 720 Lines (1280x720)  
# setenv m "720p50hz" # 50Hz  
# setenv m "720p60hz" # 60Hz  
  
# 1080 Lines (1920x1080)
```

# Chapter 3

```
# setenv m "1080i60hz" # Interlaced 60Hz
# setenv m "1080p60hz" # Progressive 60Hz
# setenv m "1080i50hz" # Interlaced 50Hz
# setenv m "1080p50hz" # Progressive 50Hz
# setenv m "1080p24hz" # Progressive 24Hz

# 4K (3840x2160)
# setenv m "2160p30hz" # Progressive 30Hz
# setenv m "2160p25hz" # Progressive 25Hz
# setenv m "2160p24hz" # Progressive 24Hz
# setenv m "smpte24hz" # Progressive 24Hz SMPTE
# setenv m "2160p50hz" # Progressive 50Hz
# setenv m "2160p60hz" # Progressive 60Hz
# setenv m "2160p50hz420" # Progressive 50Hz with YCbCr 4:2:0
(Requires TV/Monitor that supports it)
# setenv m "2160p60hz420" # Progressive 60Hz with YCbCr 4:2:0
(Requires TV/Monitor that supports it)

### VESA modes ###
# setenv m "640x480p60hz"
# setenv m "800x480p60hz"
# setenv m "800x600p60hz"
setenv m "1024x600p60hz"
# setenv m "1024x768p60hz"
# setenv m "1280x800p60hz"
# setenv m "1280x1024p60hz"
# setenv m "1360x768p60hz"
# setenv m "1440x900p60hz"
# setenv m "1600x900p60hz"
# setenv m "1680x1050p60hz"
# setenv m "1920x1200p60hz"

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

# HDMI DVI/VGA modes
# Uncomment only a single Line! The line with setenv written.
# At least one mode must be selected.
setenv vout "dvi"
# setenv vout "vga"
```

## Command Line Interface

The Linux terminal CLI is the basic method of interacting with the Linux system, normally using a shell called BASH. When a graphical system is running, a Terminal window may be launched in order to issue commands using the keyboard. Many Linux systems also come with the Secure Shell (SSH) server, which allows a command line in-

# Chapter 3

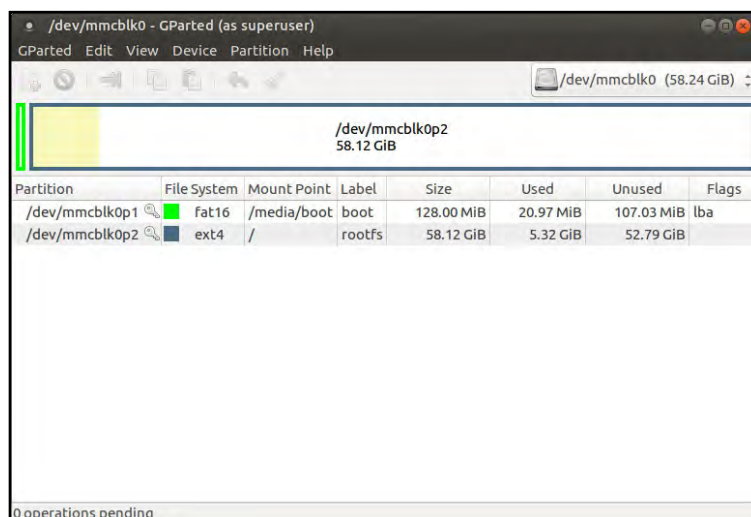
terface to be invoked from a remote computer via Ethernet. Other protocols, such as Virtual Network Computing (VNC) have the ability to project the graphical environment to a remote computer as well.

BASH is a powerful scripting language as well as a means of reading, creating, modifying, and launching files and programs that are stored in the Linux file system. A collection of BASH commands, called a script, can be saved to a file with the extension “.sh” and used as an executable file. A shell script can perform many tasks, such as configuring program variables, launching an application, and copying or moving files.

## Disk Partitions

Linux can read and write to a variety of disk formats, with the most common called EXT3, EXT4, and FAT. If disk compatibility with Windows and OSX is a priority, it is recommended to format external drives as FAT32 which can be read by nearly all operating systems. The root filesystem of Linux is usually in EXT3 or EXT4 format, which can only be read by other Linux systems.

Disk partitions may be easily resized in order to use all of the available space on a disk using a utility called Gparted. The image below shows the Gparted application being used to resize a root partition. On official Hardkernel Ubuntu distributions, the rootfs partition is automatically resized on first boot to use all unused disk space.



**Screenshot of Gparted application**

## Web Browsing

One of the popular uses of modern computers is to browse the World Wide Web, and the quad-core ODROID-C2 excels at delivering a smooth, enjoyable browsing experience. Using the GNOME Software Center, a browser such as Firefox and/or Chromium may be installed, along with open-source versions like Iceweasel.

When watching streaming videos, it is recommended to use the

# Chapter 3

Kodi/XBMC YouTube plugin in order to launch videos in full-screen, since mainstream browsers are not specifically written to use the video decoding chip available on ODROIDS. However, the ODROID-C2 CPU is powerful enough to play standard videos within a browser window at normal size using software rendering.

## Kodi (formerly XBMC)

Kodi is an application which can turn an ODROID-C2 into an amazingly powerful home media center. In fact, using the C2 to run Kodi is the reason why many, if not most, people purchased their C2 in the first place. If you're not familiar with Kodi, here are links to the official Kodi Intro FAQ at <http://bit.ly/1G8wDjD> and the Kodi Wikipedia page at <http://bit.ly/1Ir2z3R>.

The current versions of Ubuntu Linux and Android provided by Hardkernel already have Kodi installed. However, you may wish to re-install or upgrade Kodi in the future when a new release becomes available. Pre-release (beta) versions of Kodi are made available for test, and you can join a group of users that test out new features before the formal release.

## Troubleshooting

If you experience problems with playback of some video formats in Kodi, the first troubleshooting step would be to set video acceleration to “software” by selecting the System menu, the pressing **Settings >Video >Acceleration >Decoding Method >Software**.

## Kodi Upgrade

If you are installing a newer version of Kodi, you do not need to uninstall the current version first. However you should create a Kodi backup just to be safe, which is covered later in this section.

## Kodi Installation

Kodi is pre-installed on the official Hardkernel Ubuntu Mate image. To install Kodi on another Ubuntu 16.04-based distribution, use the command line terminal and enter the following commands. Follow the prompts as you would any other software installation. The official Kodi website at <http://bit.ly/1eZataG> provides more detail on the Linux installation:

```
$ sudo apt-key adv --keyserver keyserver.ubuntu.com \  
  --recv-keys AB19BAC9  
$ sudo wget http://deb.odroid.in/lists/c2-xenial.list \  
  -O /etc/apt/sources.list.d/odroid.list  
$ sudo apt-get update && dist-upgrade  
$ sudo apt-get install kodi
```



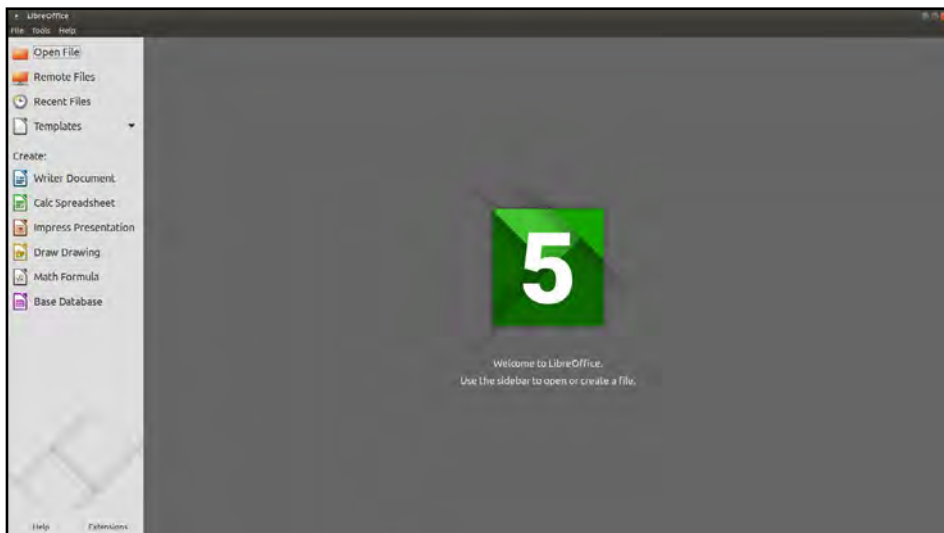
## Kodi Backup

Backing up your data is always a good idea, and Kodi is no exception. Kodi configurations and databases can become quite detailed, but backup and restoration is quite simple. You can perform full or partial backups/restores, as well as copy your complete Kodi setup to another system - even to your PC or laptop.

In order to perform a Kodi backup or restore, you will first need to install the Kodi “Backup AddOn”. More information may be found at <http://bit.ly/1JER8XL>, along with details on performing Kodi backups at <http://bit.ly/1KRNf6K>, and on the Kodi forum at <http://bit.ly/1QGofG>.

## Office and Productivity Applications

LibreOffice is a popular and powerful office productivity suite that includes word processing, spreadsheets, presentations, drawing, and flowcharting applications. It is a free alternative to Microsoft Office, and can be installed on Ubuntu via the GNOME Software Center or Synaptic Package Manager.



**Screenshot of LibreOffice**

## Music and MIDI

Not long after the first personal computers became available, creative individuals began to find inventive ways to use them for music composition and performance. Before long, the need for I/O standardization became clear, and in 1983 the MIDI specification was developed. MIDI (Musical Instrument Digital Interface) gained rapid industry acceptance, and within a few years, a MIDI Interface was found on virtually every sound card and on most PCs with sound chips on the motherboard. This is no longer the case, but the MIDI interface, MIDI instruments, and external MIDI synthesizers are still widely used by musicians and sound engineers. In fact, MIDI is still the de-facto stan-

# Chapter 3

standard instrument interface, and does not show any signs of falling into obscurity as have many early computing standards.

## What is MIDI?

MIDI encompasses not only the MIDI connector hardware specification, but also includes the communication protocol used for instruments and synthesizers, and a MIDI file format specification for MIDI recording and playback. The standard MPU-401 physical MIDI interface is a 5-pin DIN connector. Internal sound cards no longer include this connector, but it can be found on some high-end external USB sound modules. However USB-to-MIDI adapters are inexpensive and widely available. The MIDI Manufacturers Association website at <http://bit.ly/1Gn2cYw> is a good source for additional information.

## Linux Support for Music and MIDI

MIDI adapters, USB Sound Devices, and Music Applications have wide support on Linux. For example, the Ubuntu Studio project is an excellent open-source Linux build that has extensive device support with pre-configured applications. The official Ubuntu Studio site page has an <http://bit.ly/1FFKyvh> which will give you an idea of the range of Music and MIDI applications possible on Linux.

Ubuntu Studio includes some of the most popular audio apps available - including tools for DAW (Digital Audio Workstations) for multi-track mixing, sequencers for MIDI music, drum machines, software synthesizers, and even music creation via programming. Ubuntu Studio is available from the Ubuntu Software Center and Synaptic Package Manager.

## How do I add a MIDI interface to the C2 ?

Three basic options for MIDI are available for the C2:

**USB sound module:** If you wish to use a USB sound module, look for a device that has solid Linux and/or Android device driver support which includes the MPU-401 interface.

**USB-to-MIDI-Adapter:** A number of basic devices are available which have been proven to work on both Android and Linux. Compatible devices will be advertised as a “no device driver required” model.

**Direct UART to MIDI interface:** Since MIDI is an asynchronous serial interface, standard serial ports can often be easily converted to a MIDI interface.

# Chapter 3

## Sound Cards and Devices for MIDI File Playback on the C2

Not all sound cards and modules provide native MIDI file playback capability, including the ODROID USB sound card, but many other USB sound cards do, so make sure to consult the device's specifications. Professional-level USB MIDI synthesizers are also available, including samplers by Roland, Ensoniq, Kurzweil, Roland, and Korg.

## Experimental Music with the C2

Using an ODROID-C2 for computer music opens up possibilities beyond those of traditional computer music performance, productions, and engineering. A MIDI Interface and any number of sensors could be used to create unique musical instruments. For example, a pressure transducer could be used as a MIDI breath controller, ultrasonic transducers could be used to make a Vulcan Lute, a string-less harp, or a super-theremin, or a capacitive touch controller such as the Freescale MPR121. Using conductive paint for the capacitive inputs could inspire even more interesting inventions.

### *Android*

Android is easy-to-use, yet powerful operating system, intended for smartphones, tablets and other portable devices, but also runs very well on the ODROID-C2, not only as an inexpensive testing platform for building Android applications, but also as a set-top box for video and music streaming and playback, a general-purpose operating system capable of web browsing, social networking, remote control of other computers, and much more.

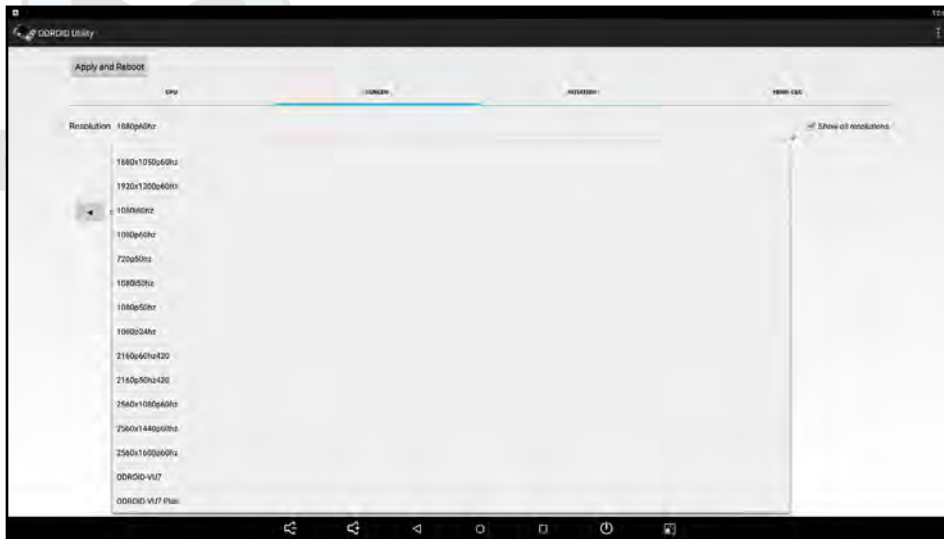


## Desktop Environment

The Hardkernel Android image offers several Android Tablet features, including two pulldown menus at the top of the screen. The top left menu shows notifications and application information. The top right menu offers direct access to the operating system settings. To open the applications menu, click on the circle with the six dots. You will be presented an alphabetical list of all installed apps. The Recent

# Chapter 3

menu will show the recently opened applications. Power Options allows you to reboot, shutdown the device and put it in airplane mode, which disables all wireless functions, including Bluetooth.



## ODROID Utility and Updater

To run the ODROID Utility application, open the applications menu and click on the ODROID Utility icon. When you start the ODROID Utility for the first time, it will ask for superuser permissions, which should be confirmed. This application contains several tabs which can be used to configure various aspects of the hardware:

**CPU** changes the Performance governor settings and can set the Kodi media center to launch at system startup. The Performance governor will give the best speed but consume the most power.

**Screen** allows you to change the resolution as well as stretch and move the screen.

**Rotation** allows you to rotate the screen to either Portrait or Landscape mode.

## Setting the Display Resolution

On the Screen tab of the ODROID Utility application, select the preferred monitor resolution, then click on Apply and Reboot. The new resolution will be active after the reboot has completed.

An overscan adjustment tool is also provided in the ODROID Utility application. To adjust the screen, click on the Screen Tab. You will then see a group of arrows that represents the four screen edges. For the edges that you wish to adjust, click on the corresponding arrow in order to set the number of offset pixels. After adjusting the settings, click on the “Apply and Reboot” button.

## Installing Google Play and applications

The official Hardkernel Android image comes without any Google applications installed. The best way to install them, is to use the

# Chapter 3

“Universal 1-Click GApps Installer for ODROID” application, available at <http://bit.ly/1gkv4PM>.

## Kodi

To install Kodi, you will first need to download the most recent .apk file from the Kodi Download page at <http://bit.ly/1yIrUdd>, as shown in the image below. Find the link to the ARM build, then click on the link to download the .apk installation package. After download, simply click on the .apk file to install. The official Kodi website provides more detail on the Android installation at <http://bit.ly/1FFK7B9>.



## Netflix

Install Netflix from Google Play, then click on its icon, which should be automatically added to the Android desktop. Login with your existing Netflix account or create a new one. When you start a video, it always begins playing at low quality, then switches to higher quality after about 30 seconds, depending on the speed of your Internet connection.

## Gaming

Most Android games are compatible with the ODROID-C2, but some applications that use multitouch gestures won't be playable without a touchscreen. The ODROID-VU, available from the Hardkernel Store, will allow all Android games to be played without the need for an auxiliary controller.

## Music and MIDI

There are a number of Android Apps available for music and MIDI. In order to work correctly with Android, USB MIDI adapters and sound cards typically need to be class-compliant USB MIDI devices that do not require a device driver, and need to be used on a USB connector that supports host mode. For some devices, the USB OTG connection must be used with an adapter.

The MIDI Manufacturers Association webpage at <http://bit.ly/1IzGkvb> is a good source for additional links on this topic. Other good webpages for MIDI and computer music on Android include <http://bit.ly/1FFKoUH> and <http://bit.ly/1IzGumw>.



# Chapter 3

TouchDAW is an excellent and mature app which allows the C2 to be a MIDI Controller for DAW and general-purpose MIDI control on secondary workstations. TouchDAW supports the leading computer music production applications on Windows, IOS, and Linux systems.

## Using Bluetooth Devices with Android

Bluetooth device detection and pairing are very easy with Android using the Settings application. After plugging in a USB bluetooth adapter and powering up the C2, you will need to enable the bluetooth receiver. Open the Settings application, find Bluetooth in the settings list under the Wireless and Networks category, and select ON to enable your USB bluetooth adapter and bluetooth services. To begin pairing your bluetooth device to the C2, turn on your device and put the device into pairing mode. The Bluetooth manager will display a list of detected devices, as well as devices that have been successfully paired.

If your device was detected, but not shown to be a paired device, click on the sliders icon on the right side of the listed device to open the bluetooth settings for this device. Complete the pairing process by entering the pairing code or passkey for the device. If the bluetooth device was not detected, clicking on search for devices will perform a manual scan for any detectable bluetooth devices within range of the C2.

## Adding an ODROID-VU7 Plus touchscreen

To use the ODROID-VU7 Plus with Android, connect the USB cable from the VU7 Plus to the ODROID-C2's USB port, then use a Type A HDMI cable to connect the display. Then, use the ODROID Utility application and select the "ODROID-VU7 Plus" option in the resolution dropdown.

# Hardware Tinkering

Now that you have been introduced to the ODROID-C2 and have become aware of booting it up with a image, it is time to get acquainted with its main purpose of creation – to be able to interact with the external world through the age-old activity of tinkering. While its design and production is a marvel in itself, it really shines in its use to interact with the external world, through a slew of break-out-boards and sensory devices – all under the control of powerful operating systems like Linux or Android.

Here are the devices covered in this section:

- **USB / UART Module Kit, Bluetooth Module 2 and Tinkering Kit**
- **Oduino One and ODROID-SHOW2; Weather Board 2**
- **16x2 LCD I/O Shield and 3.2inch TFT+Touchscreen Shield**
- **DC Plug Cable Assembly 2.5mm and USB-DC Plug Cable 2.5mm x 0.8mm**
- **USB-SPDIF and ODROID USB-CAM 720P**
- **USB GPS Module, WiFi Module 0, 3, 4**
- **USB3 to SATA Bridge Board, USB-CAM 720p, HiFi Shield+**
- **Universal Motion Joypad, myAHRS, USB Audio Adapter**
- **VU7+, VU Shell for VU7+, IR remote Controller**
- **XProto Plain, UPS3, RTC Shield**

## Preparation

Install the official Hardkernel Linux image onto an eMMC module or compatible SD card. After ensuring Internet connectivity, update this image using the following commands, then restart the system:

```
$ sudo apt-get update && sudo apt-get upgrade
$ sudo apt-get dist-upgrade
```

Verify the build information, which should be a similar version or higher to the following:

```
$ uname -a
Linux odroid64 3.14.65-75 #1 SMP PREEMPT Wed Aug 10 12:47:41 BRT
2016 aarch64 aarch64 aarch64 GNU/Linux
```

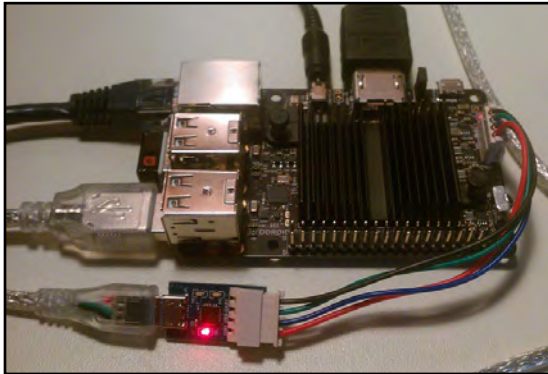
Update the Hardkernel image with the following VNC Server:

```
$ sudo apt-get install x11vnc
```

You can control the ODROID-C2 from another device (via VNC Viewer) on the local network and perform routine checks via the Linux desktop, remotely.

## USB / UART Module Kit

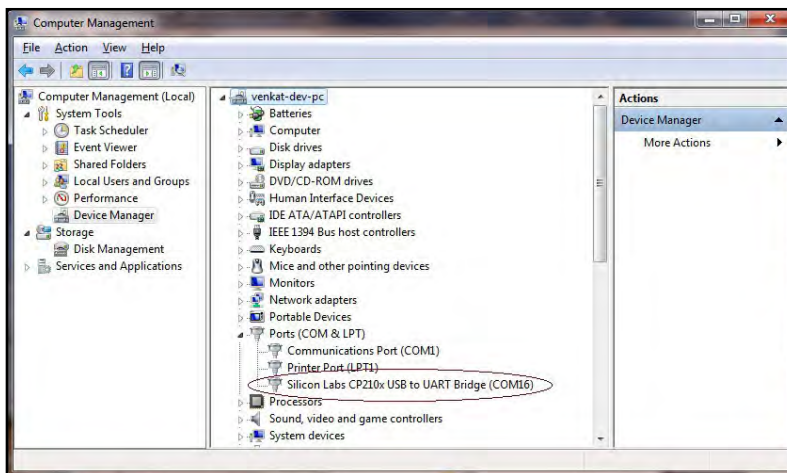
It is beneficial to get a view of the early boot process, especially if you are working with newly added external devices or creating special purpose boot images. Typically these activities can be debugged by viewing their corresponding log entries using the dmesg application. This can be achieved using the USB / UART module and an microUSB male to USB male cable, which is typically used to charge smartphones and tablets. Together, they form the kit shown below:



## USB / UART kit

When using Microsoft Windows 7+ as a host PC system, you will need to install a terminal application such as PuTTY. This can be used to set up commonly used configurations in order to access the C2 either via a serial connection or SSH.

Prior to setting up the kit, power off the ODROID-C2. Attach one end of the USB / UART module to the C2. Attach the other (micro USB) end of the USB / UART module to the USB cable. Attach the free end of the cable to an available USB port of the host PC system, then power on the device.



## USB / UART module on COM16

## Windows Host Setup

The Windows 7+ host PC system will recognize the module as a Silicon Labs CP210x USB to UART Bridge and install the appropriate

# Chapter 4

device driver. After installation has completed, you can verify which COM port the module is associated. Launch the Computer Management utility of Windows 7+, highlight the Device Manager option, and expand the Ports list. In this particular case, as shown in the below example screenshot, the module is installed on COM8.

Launch PuTTY and select the Serial option on the left pane.

Serial line to connect to: COM16

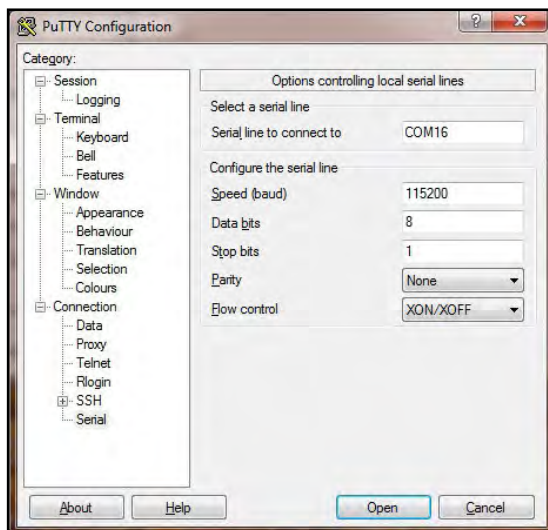
Speed (baud): 115200

Data bits: 8

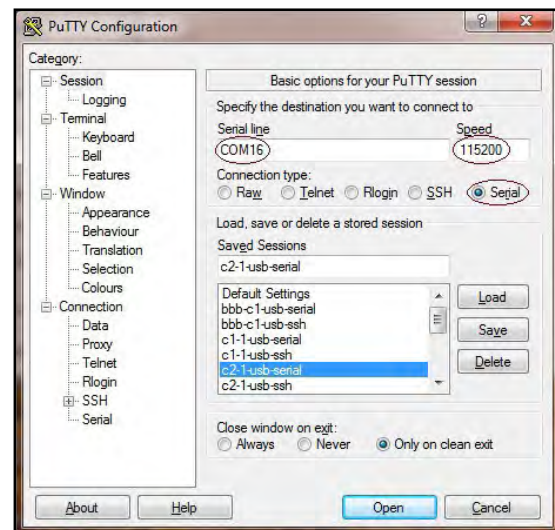
Stop bits: 1

Parity: None

Flow Control: XON/XOFF



Serial configuration in PuTTY



Saved serial configuration in PuTTY

Enter configuration information as listed below, then select the Session option on the left pane. You will see a popup window as indicated in the following image.

Serial line: COM16

Speed: 115200

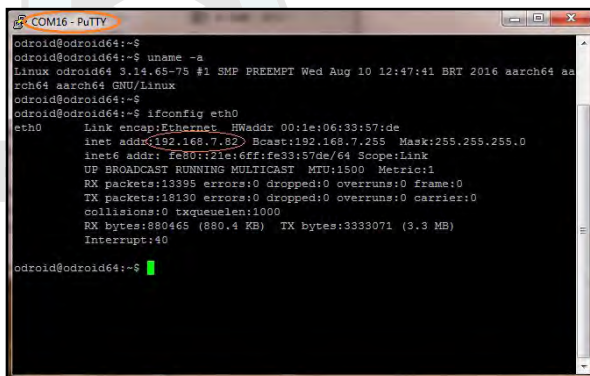
Connection Type: Serial

Give this configuration a name such as c2-1-usb-serial, then save it. This saved configuration can be reloaded and used anytime, as long as the COM port on the host PC system has not changed. Click on the Open button to start a session.

Next, reboot the C2. After a short delay, you will see messages scroll by in the PuTTY console, as shown below. This is when one would notice messages relevant to external devices or software mod-

# Chapter 4

ules. These messages can help with debugging issues that you may encounter. Most of these messages will also be listed through the `dmesg` command.



```
odroid@odroid64:~$  
odroid@odroid64:~$ uname -a  
Linux odroid64 3.14.65-75 #1 SMP PREEMPT Wed Aug 10 12:47:41 BRT 2016 aarch64 aa  
rch64 aarch64 GNU/Linux  
odroid@odroid64:~$  
odroid@odroid64:~$ ifconfig eth0  
eth0  
Link encop:Ethernet HWaddr 00:1e:06:33:57:de  
inet addr:192.168.7.82 Bcast:192.168.7.255 Mask:255.255.255.0  
inet6 addr: fe80::21e:6ff:fe33:57de/64 Scope:Link  
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
RX packets:13995 errors:0 dropped:0 overruns:0 frame:0  
TX packets:18190 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1000  
RX bytes:880465 (880.4 KB) TX bytes:3333071 (3.3 MB)  
Interrupt:40  
odroid@odroid64:~$
```

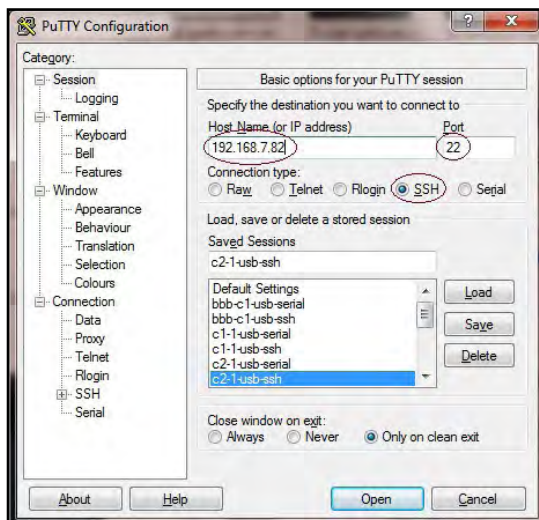
## PuTTY serial console

Close the serial console, then set up the SSH connection option by relaunching the PuTTY application and selecting the Session option on the left pane. For this SSH option to work, the host PC system and the C2 need to be visible to each other on the same local area network (LAN). Refer to the screenshot below for an example PuTTY configuration.

Host name: 192.168.7.82

Port: 22

Connection type: SSH



## Saved SSH configuration in PuTTY

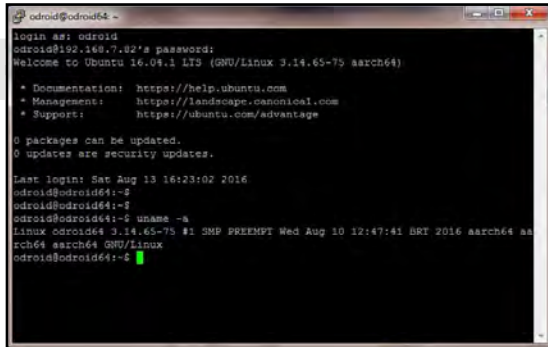
Enter the ODROID-C2's IP address or hostname (if it can be resolved from the host PC system) and a numerical value of 22 for the Port number. Select the Connection Type to be SSH. Enter `c2-1-usb-ssh` for the configuration name and save it. This saved configuration



# Chapter 4

can be reloaded and used anytime, as long as the C2's ip-address and hostname remain the same. Click on the Open button to start a session.

Reboot the C2, and you will then be prompted for the user's ID and password, which are typically "odroid" and "odroid". You will see a screen presented as shown in the image below.



```
odroid@odroid64~$ ssh -p 2222 odroid@192.168.7.02
odroid@odroid64~$
odroid@192.168.7.02's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 3.14.65-75 aarch64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

Last login: Sat Aug 13 16:23:02 2016
odroid@odroid64:~$
odroid@odroid64:~$
odroid@odroid64:~$ uname -a
Linux odroid64 3.14.65-75 #1 SMP PREEMPT Wed Aug 10 12:47:41 BRT 2016 aarch64 aa
rch64 aarch64 GNU/Linux
odroid@odroid64:~$
```

## PuTTY SSH console

You can access the C2 system as if you had locally opened a terminal window. While you cannot view the boot log in real-time, you can run a large number of Linux commands (including dmesg) from this ssh session.

So far, we've addressed the case where the host PC system is a Windows 7+ system. Next, we present information on the use of a Ubuntu desktop Linux host system. Most of the steps apply to a host system running any Linux version, including a server OS. The steps do not address the use of Linux hosted in a virtual environment, such as VirtualBox.

## Linux Host Setup

Access the Linux desktop of the host system and launch a terminal session. Attach the USB cable from the USB / UART setup to the host system. The following commands can be used to verify proper installation:

```
$ lsusb
...
Bus 004 Device 003: ID 10c4:ea60 Cygnal Integrated Products, Inc.
CP210x UART Bridge / myAVR mySmartUSB light
...
```

Using the output above, you get obtain additional details of the USB / UART module, using the command:

```
$ sudo lsusb -D /dev/bus/usb/004/003
Device: ID 10c4:ea60 Cygnal Integrated Products, Inc. CP210x UART
Bridge / myAVR mySmartUSB light
```

# Chapter 4

```
Device Descriptor:
  bLength                18
  bDescriptorType        1
  bcdUSB                  2.00
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass        0
  bDeviceProtocol        0
  bMaxPacketSize0       64
  idVendor                0x10c4 Cygnal Integrated Products, Inc.
  idProduct              0xea60 CP210x UART Bridge / myAVR mySmartUSB
light
  bcdDevice              1.00
  iManufacturer          1 Silicon Labs
  iProduct                2 CP2104 USB to UART Bridge Controller
  iSerial                3 00513B0C
  bNumConfigurations    1
...
```

The TTY port on which the module is detected, can be obtained:

```
$ sudo ls -lsa /dev/tty* | grep USB
0 crw-rw---- 1 root dialout 188, 0 Aug 12 16:48 /dev/ttyUSB0
```

The lockfile location can be checked using the command:

```
$ sudo ls -lsa /var/lock*
0 lrwxrwxrwx 1 root root 9 Aug 12 2016 /var/lock -> /run/lock
```

This information is used to setup the communications between the host system and USB / UART cable attached to the ODROID-C2. The application used to enable communication will be Minicom:

```
$ minicom -version
minicom version 2.6.1 (compiled May 1 2012)
Copyright (C) Miquel van Smoorenburg.
...
```

Launch Minicom using the following command:

```
$ sudo minicom -o -s
```

Here are the four screens you would go through to configure and use minicom in the next page:

The four steps include:

Select the Serial port setup option and you will be presented with screen 2.

In screen 2, type A to select option A. Edit the serial device to that used on your system. In this case it is: `/dev/ttyUSB0`. Ensure the

# Chapter 4

lockfile location is what was obtained earlier. Likewise check option E reflects the information shown in the figure: 115200 8N1. Ensure Hardware Flow Control is set to YES. Then hit the ENTER key to go to the next screen.

In screen 3, select the “Save setup as ...” option to save this config for future use. Enter a filename at the new prompt. Hit ENTER to accept the filename. Then select Exit option to complete the config process.

You will now return to the terminal windows. Reboot the ODROID-C2 and you will observe boot-time information scroll through the minicom (terminal) session.

```
+-----[configuration]-----+
| Filenames and paths          |
| File transfer protocols      |
| Serial port setup            |
| Modem and dialing           |
| Screen and keyboard         |
| Save setup as dfl           |
| Save setup as..             |
| Exit                         |
| Exit from Minicom           |
+-----+-----+

```

Port option

```
+-----+-----+
| A - Serial Device      : /dev/ttyUSB0 |
| B - Lockfile Location  : /var/lock    |
| C - Callin Program    :              |
| D - Callout Program   :              |
| E - Bps/Par/Bits      : 115200 8N1   |
| F - Hardware Flow Control : Yes      |
| G - Software Flow Control : No       |
+-----+-----+
Change which setting?
| Screen and keyboard      |
| Save setup as dfl       |
| Save setup as..        |
| Exit                    |
| Exit from Minicom      |
+-----+-----+

```

Serial port data

```
+-----[configuration]-----+
| Filenames and paths          |
| File transfer protocols      |
| Serial port setup            |
| Modem and dialing           |
| Screen and keyboard         |
| Save setup as dfl           |
| Save setup as..             |
| Exit                         |
| Exit from Minicom           |
+-----+-----+

```

Save setup

```
ve..
Welcome to minicom 2.6.1
OPTIONS: I18n
Compiled on May 1 2012, 13:36:18.
Port /dev/ttyUSB0
Press CTRL-A Z for help on special keys

Broadcast message from root@odroid
(unknown) at 17:19 ...

The system is going down for reboot NOW!
[ 112.436700@0] Restarting system.
[ 112.436803@0] Restarting Linux version 3.10.75-84 (root@xu3-b2) (gcc version
[ 112.436803@0]
QAS:A;SVN:B72;POC:17F;STS:0;BOOT:0;
```

Minicom session

## Bluetooth Module 2

Hardkernel offers a bluetooth adapter called the Bluetooth (4.0) Module 2 which is certified to work with the ODROID-C2 under both Android and Linux, in the following modes:

- |                                  |                |
|----------------------------------|----------------|
| Classic Bluetooth v2.0:          | Android, Linux |
| Bluetooth High Speed v3.0:       | Linux          |
| Bluetooth low energy v4.0 (BLE): | Linux          |



Bluetooth Module 2 adapter and Asus BT400 adapter

# Chapter 4

Ensure the adapter is recognized by the ODROID-C2, by obtaining the USB information of the adapter using the `lsusb` command:

```
$ lsusb
...
Bus 001 Device 005: ID 0a12:0001 Cambridge Silicon Radio, Ltd Blue-
tooth Dongle (HCI mode)
...
```

While some of the required software modules may already exist in your ODROID-C2's installed image, it is helpful to install the following additional modules and utilities, using the following commands:

```
$ sudo apt-get install bluez-dbg bluez-tools bluez-hcidump
$ sudo apt-get install bluez bluez-tools python-bluez
```

Some of them could be used to debug your setup, if you use a non-Hardkernel Bluetooth adapter. You can check the Bluetooth adapter's support for additional features by examining the `dmesg` logs, using the following command:

```
$ dmesg | grep Blue
[ 0.180204] Bluetooth: Core ver 2.18
[ 0.180236] Bluetooth: HCI device and connection manager initialized
[ 0.180250] Bluetooth: HCI socket layer initialized
[ 0.180261] Bluetooth: L2CAP socket layer initialized
[ 0.180284] Bluetooth: SCO socket layer initialized
[ 3.945426] Bluetooth: RFCOMM TTY layer initialized
[ 3.949664] Bluetooth: RFCOMM socket layer initialized
[ 3.954786] Bluetooth: RFCOMM ver 1.11
[ 3.958468] Bluetooth: BNEP (Ethernet Emulation) ver 1.3
[ 3.963723] Bluetooth: BNEP filters: protocol multicast
[ 3.968896] Bluetooth: BNEP socket layer initialized
[ 3.973814] Bluetooth: HIDP (Human Interface Emulation) ver 1.2
[ 3.979677] Bluetooth: HIDP socket layer initialized
```

You can then check the list of the installed Bluetooth modules:

```
$ dpkg -l | grep blue
ii  blueman          ... armhf Graphical bluetooth manager
ii  bluez            ... armhf Bluetooth tools and daemons
ii  bluez-alsa:armhf ... armhf Bluetooth ALSA support
ii  bluez-cups       ... armhf Bluetooth printer driver for CUPS
ii  libbluetooth-dev ... armhf Dev. files for BlueZ Linux Bluetooth lib
ii  libbluetooth3:armhf ... armhf Library to use the BlueZ Linux BT stack
ii  libgnome-bluetooth11 ... armhf GNOME Bluetooth tools - support library
```

With the positive signs observed in the output so far, we can

# Chapter 4

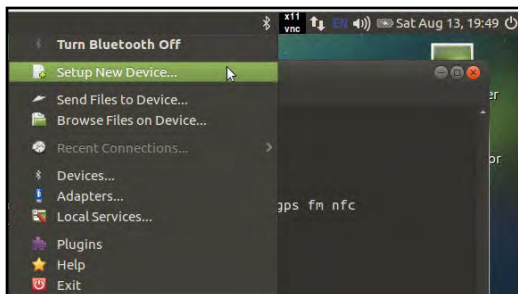
check the presence of the Bluetooth device, which is useful for connection configuration later:

```
$ hcitool dev
Devices:
  hci0      00:1A:7D:DA:71:13

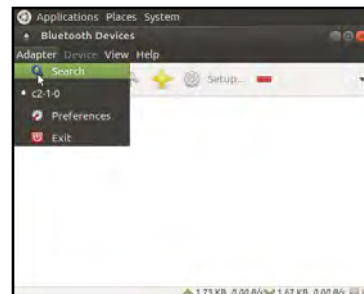
$ hciconfig
hci0: Type: BR/EDR  Bus: USB
      BD Address: 00:1A:7D:DA:71:13  ACL MTU: 310:10  SCO MTU: 64:8
      UP RUNNING PSCAN
      RX bytes:636 acl:0 sco:0 events:41 errors:0
      TX bytes:1224 acl:0 sco:0 commands:41 errors:0

$ sudo rfkill list all
0: hci0: Bluetooth
   Soft blocked: no
   Hard blocked: no
```

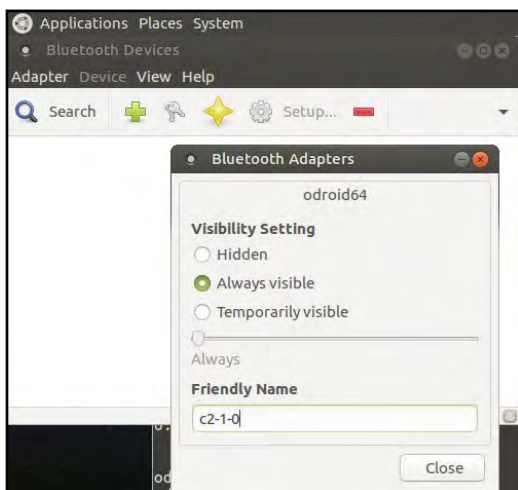
To configure the adapter from the Ubuntu desktop, click on the Bluetooth icon on the top left, then select the Adapter -> Preferences menu item to configure the application. Rename the adapter using a friendly-name such as c2-1-0. You can make the device always visible when other bluetooth devices scan for this adapter.



**Launch Bluetooth Manager**



**Bluetooth Manager configuration**



**Bluetooth Manager config update**

If there is a need to test/debug issues related to the adapter, you



# Chapter 4

can use the command-line bluetooth packet analyzer/sniffer to analyze bluetooth traffic between the ODROID-C2 and a known bluetooth device using the following command:

```
$ sudo hcidump
[sudo] password for odroid:
HCI sniffer - Bluetooth packet analyzer ver 5.37
device: hci0 snap_len: 1500 filter: 0xffffffffffffff
< HCI Command: LE Set Scan Parameters (0x08|0x000b) plen 7
    type 0x01 (active)
    interval 11.250ms window 11.250ms
    own address: 0x00 (Public) policy: All
> HCI Event: Command Complete (0x0e) plen 4
    LE Set Scan Parameters (0x08|0x000b) ncmd 1
    status 0x00
< HCI Command: LE Set Scan Enable (0x08|0x000c) plen 2
    value 0x01 (scanning enabled)
    filter duplicates 0x01 (enabled)
> HCI Event: Command Complete (0x0e) plen 4
    LE Set Scan Enable (0x08|0x000c) ncmd 1
    status 0x00
< HCI Command: LE Set Scan Enable (0x08|0x000c) plen 2
    value 0x00 (scanning disabled)
    filter duplicates 0x00 (disabled)
> HCI Event: Command Complete (0x0e) plen 4
    LE Set Scan Enable (0x08|0x000c) ncmd 1
    status 0x00
< HCI Command: Inquiry (0x01|0x0001) plen 5
    lap 0x9e8b33 len 4 num 0
> HCI Event: Command Status (0x0f) plen 4
    Inquiry (0x01|0x0001) status 0x00 ncmd 1
> HCI Event: Extended Inquiry Result (0x2f) plen 255
    bdaddr D8:B3:77:6F:E2:36 mode 1 clkoffset 0x5d2e class
    0x5a020c rssi -78
    Complete local name: 'OBDII dongle'
    Complete service classes: 0x1105 0x1106 0x110a 0x110c 0x1112
    0x111f 0x112f 0x1200
...
```

## Tinkering Kit

While the power-efficient C2 is a very capable system for software development, it is also very highly suited to prototyping hardware development. Hardkernel has developed the C Tinkering kit with numerous items to get you started, as shown on the next page.

Some of the components in the Tinkering Kit include:

- **Assembled T-breakout PCB - 40Pin GPIO Breakout board**
- **Breadboard - 630 Tie-points with dual power lanes**
- **40pin Ribbon cable - IDC Flat cable 100mm**

# Chapter 4

- 40pin Male-to-Male Dupont jumper Wire 170mm
- Green/Yellow/Red 3mm LEDs – 7 each
- 2 x Photocell (CdS Light sensor), 6 x Tact Switches
- 330 Ω / 10 KΩ 1/6W resistors – 50 each



## Tinkering Kit

ODROID-C2 40pin Layout									
WiringPi GPIO#	Export GPIO#	ODROID-C2 PIN	Label	HEADER		Label	ODROID-C2 PIN	Export GPIO#	WiringPi GPIO#
			3V3	1	2	5V0			
	205	I2CA_SDA	SDA1	3	4	5V0			
	206	I2CA_SCL	SCL1	5	6	GND			
7	249	GPIOX.BIT21	#249	7	8	TXD1	TXD_B	113	
			GND	9	10	RXD1	RXD_B	114	
0	247	GPIOX.BIT19	#247	11	12	#238	GPIOY.BIT10	238	1
2	239	GPIOX.BIT11	#239	13	14	GND			
3	237	GPIOX.BIT9	#237	15	16	#236	GPIOX.BIT8	236	4
			3V3	17	18	#233	GPIOX.BIT5	233	5
12	235	GPIOX.BIT7	#235	19	20	GND			
13	232	GPIOX.BIT4	#232	21	22	#231	GPIOX.BIT3	231	6
14	230	GPIOX.BIT2	#230	23	24	#229	GPIOX.BIT1	229	10
			GND	25	26	#225	GPIOY.BIT14	225	11
	207	I2CB_SDA	SDA2	27	28	SCL2	I2CB_SCL	77	
21	228	GPIOX.BIT0	#228	29	30	GND			
22	219	GPIOY.BIT8	#219	31	32	#224	GPIOY.BIT13	224	26
23	234	GPIOX.BIT6	#234	33	34	GND			
24	214	GPIOY.BIT3	#214	35	36	#218	GPIOY.BIT7	218	27
		ADC.AIN1	AIN1	37	38	1V8	1V8		
			GND	39	40	AIN0	ADC.AIN0		

## GPIO layout for the 40-pin header on the ODROID-C2

The T-breakout board breaks-out all the pins from the 40-pin header onto a solderless breadboard. The pins include, power, GPIO, I2C, and ADC pins. This simplifies prototyping of data acquisition and control projects.

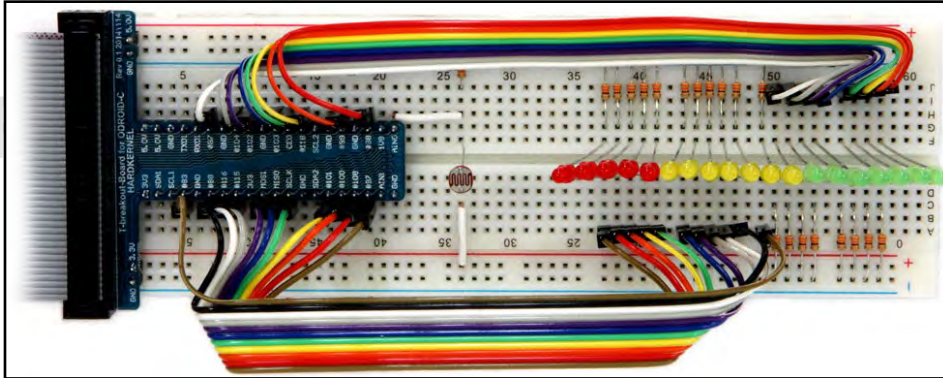
It is advisable to refer to the Amlogic S905 data sheet at <http://bit.ly/2bfeRFF> to fully understand the GPIO designations.

The above image illustrates simplified information on layout and the designation of the 40 pins. Use the data sheet and this information with caution when applying voltages to the pins, to ensure compliance

# Chapter 4

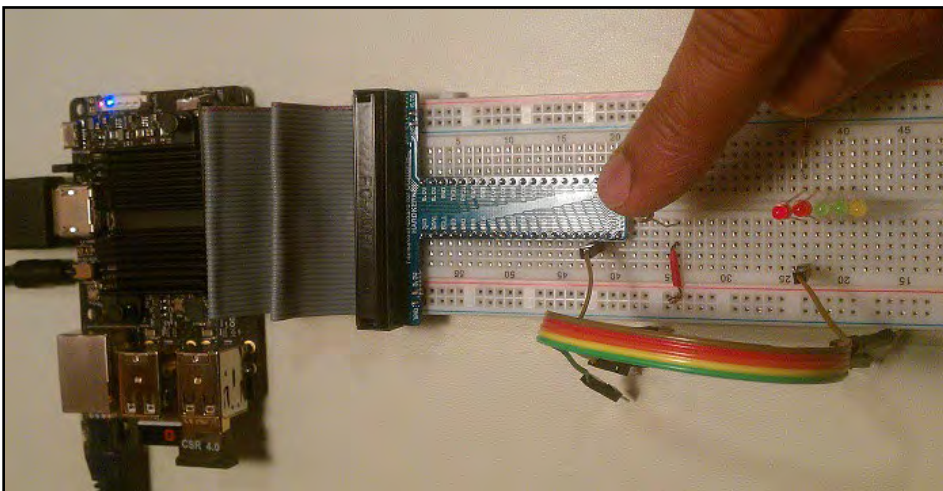
and prevent possible device damage.

The WiringPi library has been ported to work with C2. The chart above illustrates the GPIO pin numbering translation scheme used in the WiringPi library.



## Tinking project example

The image above is a project example using the ODR0ID-C2 and the kit. Details can be found in the related wiki at <http://bit.ly/2ba6h8o>. You can wire up your project to reflect the example shown in the image above, or a simplified version using fewer LEDs that reacts to the same stimuli (light sensor being covered) as shown below.



## Tinking project example

Prior to trying out the example, you will need to build and install the WiringPi library by running the following commands in a Terminal window:

```
$ cd ~  
$ git clone https://github.com/hardkernel/WiringPi  
$ cd WiringPi  
$ ./build
```

# Chapter 4

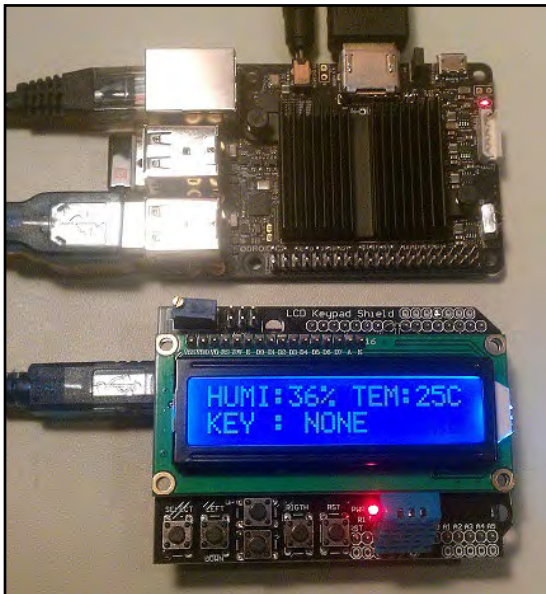
Then, download the C-language sample source code `example-led.c` from the previously mentioned Wiki, to a working directory such as `tkit-example`, then build and run it using the following commands:

```
$ cd ~ && mkdir tkit-example/ && cd tkit-example/  
$ gcc -o example-led example-led.c -lWiringPi -lWiringPiDev -lpthread  
$ sudo ./example-led
```

Note that GPIO access requires root user privilege. Move your finger closer to the light sensor and you should see the LEDs light up.

## ODUINO ONE

The ODROID-compatible Arduino called the ODUINO ONE includes the following components, all packaged into one experimentation unit.



The ODUINO ONE unit

- **Arduino Uno R3**
- **16x2 LCD display**
- **DHT11 one-wire humidity / Temperature sensor**
- **keypad (four directional keys)**

The ONE package can be powered by the C2 using the USB Std A/Std B cable. Connect the cable to the ODUINO ONE and the C2. The ONE goes through the power-up process and displays the ambient humidity and temperature. It can also display the key that gets clicked on the keypad. Some additional information on the ONE package can be obtained using the following commands:



# Chapter 4

```
$ ls -lsa /dev/ttyA*
0 crw-rw---- 1 root dialout 166, 0 May 24 21:05 /dev/ttyACM0

$ lsusb
...
Bus 001 Device 009: ID 2341:0043 Arduino SA Uno R3 (CDC ACM)
...
$ lsusb -D /dev/bus/usb/001/009
Device: ID 2341:0043 Arduino SA Uno R3 (CDC ACM)
Couldn't open device, some information will be missing
Device Descriptor:
  bLength                18
  bDescriptorType        1
  bcdUSB                  1.10
  bDeviceClass            2 Communications
  bDeviceSubClass         0
  bDeviceProtocol         0
  bMaxPacketSize0         8
  idVendor                0x2341 Arduino SA
  idProduct               0x0043 Uno R3 (CDC ACM)
  bcdDevice               0.01
  iManufacturer           1
  iProduct                2
  iSerial                 220
  bNumConfigurations      1
```

The activity on the ONE package also gets transmitted to the ODROID-C2. Hardkernel has provided C sample source code to display this information. You will need to install some pre-requisite software components first:

```
$ sudo apt-get install qt4-default qt4-designer libqwt-dev
```

You can then download the sample (Qt based) source code to the C2:

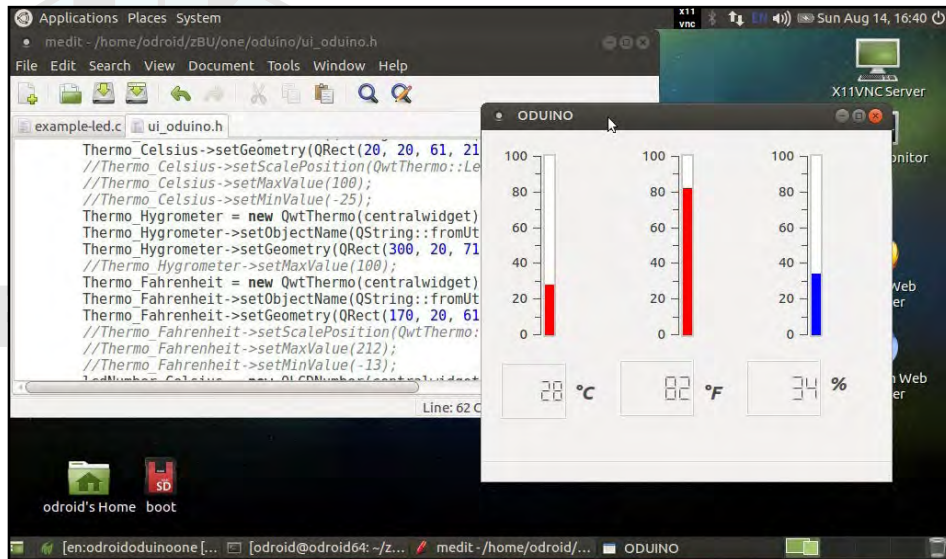
```
$ cd ~
$ wget -c http://dn.odroid.com/ODUINO/source/Ubuntu_Qt/oduino.zip
$ unzip oduino.zip
$ cd oduino
$ mv ui_oduino.h ui_oduino.h-save
$ uic oduin.ui > ui_oduino.h
$ qmake
$ make
```

The generated code will not work as is. After running the make command, you will have to take note of any errors that have to do with undefined attributes. Comment out those lines and rerun the build:

```
$ make
```



# Chapter 4



## ODUINO ONE data captured by application on C2

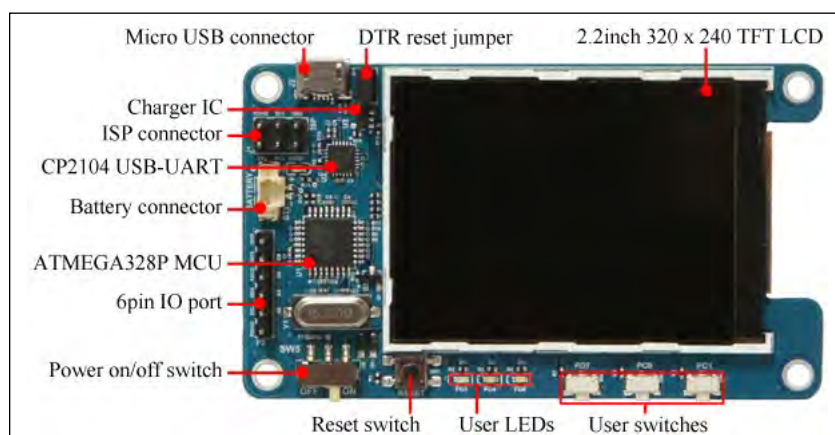
After it builds, you can launch the freshly built application using the following commands:

```
$ cd Linux/  
$ ./oduino
```

The application user-interface should look like the above image.

## ODROID-SHOW2

Hardkernel offers an Arduino-compatible 2.2" 240×320 TFT-LCD display, called ODROID-SHOW2, shown below, that can be used with the ODROID-C2, a PC, and other Hardkernel boards.



## ODROID-SHOW2 annotated image

After attaching the SHOW2 to the ODROID-C2, use the following commands to get details on it:

```
$ lsusb  
...
```

# Chapter 4

```
Bus 001 Device 003: ID 10c4:ea60 Cygnal Integrated Products, Inc.
CP210x UART Bridge / myAVR mySmartUSB light
...

$ sudo lsusb -D /dev/bus/usb/001/003
[sudo] password for odroid:
Device: ID 10c4:ea60 Cygnal Integrated Products, Inc. CP210x UART
Bridge / myAVR mySmartUSB light
Device Descriptor:
  bLength                18
  bDescriptorType        1
  bcdUSB                  2.00
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass         0
  bDeviceProtocol         0
  bMaxPacketSize0        64
  idVendor                0x10c4 Cygnal Integrated Products, Inc.
  idProduct              0xea60 CP210x UART Bridge / myAVR mySmartUSB
light
  bcdDevice               1.00
  iManufacturer           1 Silicon Labs
  iProduct                2 CP2104 USB to UART Bridge Controller
  iSerial                 3 00875559
  bNumConfigurations     1
...

$ ls -lsa /dev/ttyUSB*
0 crw-rw---- 1 root dialout 188, 0 Feb 11 2016 /dev/ttyUSB0
```

Obtain the SHOW sample source code and build a sample using the commands:

```
$ cd ~
$ git clone https://github.com/hardkernel/ODROID-SHOW
$ cd ODROID-SHOW/example/Linux
$ gcc -o status status.c
$ sudo ./status
```

You should see the LCD display similar to the following image:



**ODROID-SHOW showing the sample project startup screen**

Additional details, such as, burning new firmware for the SHOW2 (or SHOW1) and sample applications can be found at <http://bit.ly/1toe7Pl>.

One important aspect of the SHOW2 (or SHOW1) board, is the presence of a Data Terminal Ready (DTR) reset jumper/switch. Its role is important during burning of new firmware. The following precautions need to be followed, regarding the DTR jumper:

- *It must be installed when you upload the firmware.*
- *It must NOT be installed during normal usage mode.*

## 16x2 LCD I/O Shield

Hardkernel offers an I/O shield with a, 16 chars by 2 lines, white on blue, LCD display, tactile switches and I/O pins. Its use requires the WiringPi library, whose installation was discussed earlier. After installing the shield on the C2, the sample source can be obtained from <http://bit.ly/1fbtFlE>.

Copy the sample code using the following commands:

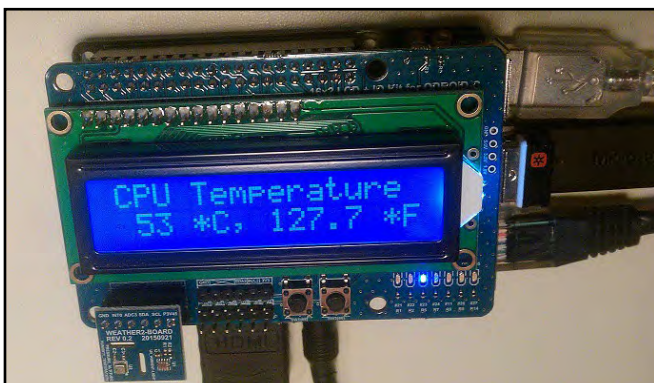
```
$ cd ~ && mkdir lcdio && cd lcdio
$ mv ~/Downloads/lcd_cpuInfo.c .
```

Modify the source code to have CPU information displayed on the LCD:

```
static int DispMode = 2;
```

Then, build and launch the application using the following commands, which should show a display similar to the image below:

```
$ gcc -o lcd_cpuInfo lcd_cpuInfo.c -lWiringPi -lWiringPiDev \
-lpthread
$ sudo ./lcd_cpuInfo
```



16x2 LCD – I/O shield showing some CPU data

## Weather board on 16x2 LCD I/O Shield

The Weather board is an add-on developed by Hardkernel, to be used with either the SHOW2 or 16x2 LCD device. The following steps demonstrate how to get the Weather board to work with the LCD I/O shield mounted on the ODROID-C2. First, enable the I2C module:

```
$ sudo modprobe aml_i2c
$ ls /dev/i2c-*
0 crw----- 1 root root 89, 1 Aug 14 20:06 /dev/i2c-1
0 crw----- 1 root root 89, 2 Aug 14 20:06 /dev/i2c-2
```

Ensure that the wiringpi is installed as previously described. Power down the system, attach the Weather board via the 6 I2C pins to the ODROID-SHOW or 16x2 LCD device, then power it up. Next, download the I2C sample and built it using the commands:

```
$ git clone https://github.com/hardkernel/WEATHER-BOARD.git
$ cd WEATHER-BOARD/c_weather
$ make
```

Run the program to see the weather data on the terminal:

```
$ sudo ./weather_board
===== sll132 =====
UV_index : 0.03
Visible : 360 Lux
IR : 196 Lux
===== bme280 ===== WEATHER-BOARD v1.5
temperature : 35.11 `C
humidity : 35.49 %
pressure : 999.12 hPa
altitude : 209.064621 m
```

## DC plug cable assembly 2.5mm

This accessory, shown in the image below, typically used to provide power to the ODROID-C2, using a special purpose power supply. These power supplies could include bench-top models that can provide clean power at 2~3 Amps, at the least. Use the cable colors to ensure polarities match at either end.



DC plug cable assembly



DC plug cable usage



## USB-DC Plug Cable 2.5x0.8mm

This accessory is typically used to provide power to the C2 using a power supply that has a USB port. These power supplies (or chargers) could include units that convert AC power to DC power transformers or battery packs. The image below shows both the USB-DC plug cable and a typical LiPo battery pack. Ensure that the power supply provides clean power, rated at: 5V, 2.0+ Amps.



USB-DC plug cable



USB-DC plug cable usage

## USB audio adapter

Attach the USB-powered audio adapter to the ODROID-C2. After a few moments, check to see if the device is detected:

```
$ lsusb
...
Bus 001 Device 003: ID 0d8c:000c C-Media Electronics, Inc. Audio
Adapter
...
```

This adapter is based on the CM108AH single-chip USB audio solution, and is capable of stereo output. It includes the dual DAC/headphone amplifier, ADC, microphone booster, PLL, regulator, and USB transceiver.

While the setup can be tested with headphones connected to the headphone port on the adapter, one can also test it with a USB-powered stereo speaker set as shown below.



USB audio adapter with USB-powered speakers

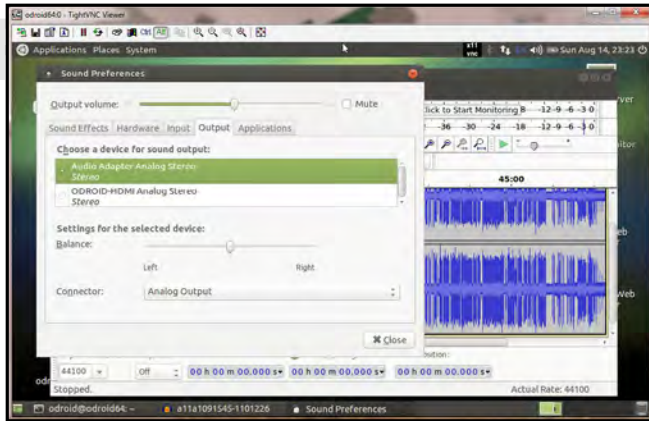


# Chapter 4

Install Audacity to test the setup:

```
$ sudo apt-get install audacity
```

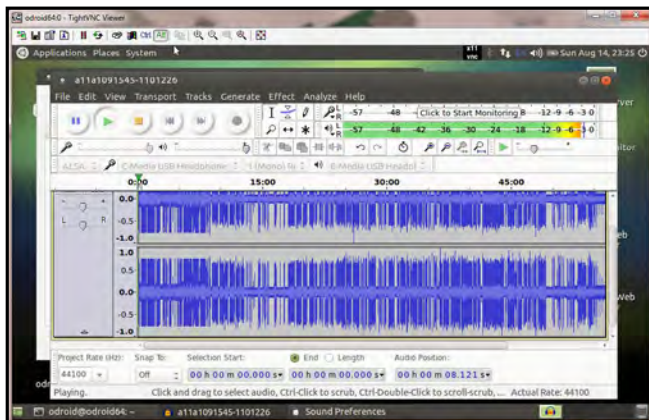
Configure the sound preferences to select the audio adapter by default, instead of the HDMI option.



## Sound preferences

Adjust the volume using speaker icon at the top right of the desktop. Set it to ~25% of the maximum for safety, then start a web browser such as Firefox and access a Youtube link in order to play some video with audio.

Next, download a sample MP3 file. Start Audacity from the Applications > Sound & Video menu of the desktop, open the mp3 file within Audacity and select play in order to listen to it.



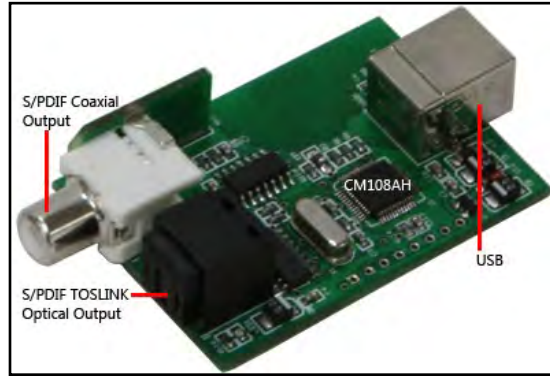
## Audacity screen showing play

## USB-SPDIF

Hardkernel offers an ODROID-C2-compatible S/PDIF (Sony/Philips Digital Interface Format) kit with a USB interface. This allows you to hook up the audio from the ODROID-C2 via USB to an amplifier (A/V receiver), when the HDMI audio out option is not used. The images on the next page show the kit and device internals.



**USB-SPDIF kit**



**USB-SPDIF device internals**

Attach the cable provided in the kit to one end of the device, and the ODROID-C2's USB port to the other end, then reboot the system. You can check to see if the device is detected using the following commands:

```
$ lsusb
...
Bus 001 Device 005: ID 0d8c:013c C-Media Electronics, Inc.  CM108
Audio Controller
...

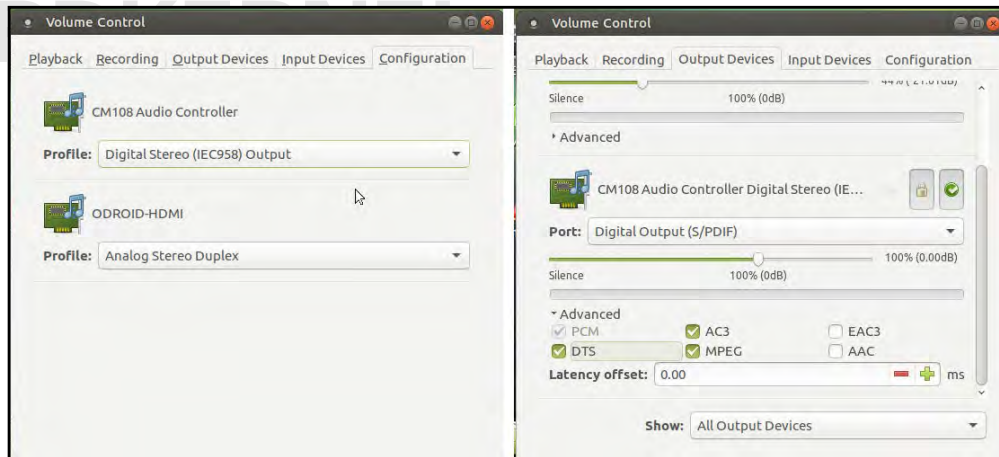
$ sudo lsusb -D /dev/bus/usb/001/005
[sudo] password for odroid:
Device: ID 0d8c:013c C-Media Electronics, Inc.  CM108 Audio Con-
troller
Device Descriptor:
  bLength                18
  bDescriptorType        1
  bcdUSB                  1.10
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass         0
  bDeviceProtocol         0
  bMaxPacketSize0        8
  idVendor                0x0d8c C-Media Electronics, Inc.
  idProduct              0x013c CM108 Audio Controller
  bcdDevice              1.00
  iManufacturer          1 C-Media Electronics Inc.
  iProduct                2 USB PnP Sound Device
  iSerial                 0
  bNumConfigurations     1
...
```

Now that the device is verified to be recognized by the system, configure it by launching the PulseAudio Volume Control configuration option from the desktop. Then, in the Configuration tab, select the Digital Stereo Output profile, select the Output Devices tab and update the configuration for the CM108 device as shown in the image below. Note that TrueHD or DTS-MA pass-through is unsupported.

Connect a TOSLINK optical cable to the SPDIF peripheral and

# Chapter 4

an AV Receiver capable of accepting the optical TOSLINK cable. The AV Receiver is expected to be connected to an appropriate set of speakers. Power up the AV Receiver and select the appropriate input option on the AV Receiver. Launch the browser on the ODROID-C2 and access a youtube video link that has audio included, then press Play and verify that the audio is working properly.



**PulseAudio Volume Control**

**PulseAudio profile**

## USB-CAM 720P

Hardkernel offers a USB-CAM rated @ 720p resolution with up to 30 fps, which is certified to work with the ODROID-C2 as shown below.



**720p 30fps USB-CAM**

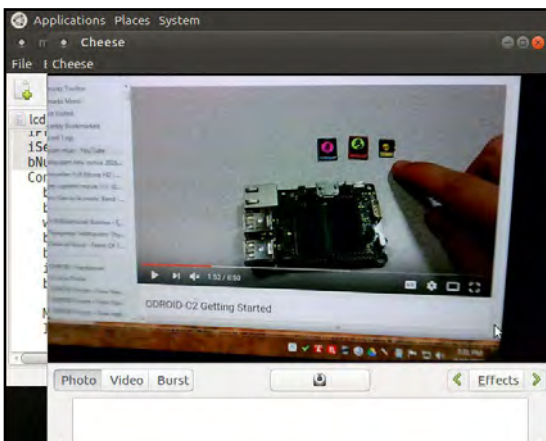
Attach the USB-CAM to an available USB port on the ODROID-C2, and wait for a few moments. Details of the camera can then be obtained using the commands:

```
$ lsusb
Bus 001 Device 005: ID 1b71:0056 Fushicai

$ sudo lsusb -D /dev/bus/usb/001/005
Device: ID 1b71:0056 Fushicai
Device Descriptor:
  bLength                18
  bDescriptorType         1
  bcdUSB                  2.00
  bDeviceClass            239 Miscellaneous Device
```

```
bDeviceSubClass      2 ?
bDeviceProtocol      1 Interface Association
bMaxPacketSize0     64
idVendor             0x1b71 Fushicai
idProduct            0x0056
bcdDevice            0.00
iManufacturer        2 Sonix Technology Co., Ltd.
iProduct             1 USB 2.0 Camera
iSerial              0
bNumConfigurations  1
```

Now, launch the Cheese application from the Applications → Sound & Video menu. Point the camera at a well lit object. You should see a screen as shown below.



**Cheese application**

## USB3 / SATA3 HDD/SSD interface kit

The USB3/SATA3 HDD/SSD interface kit offered by Hardkernel allows a SATA hard drive (HDD) or solid-state drive (SSD) to be attached to the ODROID-C2. The package includes the docking system, a power supply (12V/2A) and a USB3.0 cable, which is compatible with both Android and Linux. To use it, follow these steps:

- Shutdown the ODROID-C2 and turn the power off
- Attach the docking system to the ODROID-C2 with the included USB3 cable
- Insert the SATA storage device (HDD/SSD) with its SATA interface aligned with the receptacle in the docking system
- gently press down the storage device so the SATA connectors connect well and the storage device sits firmly in the docking system



# Chapter 4

- Turn on the power to the ODROID-C2 and let it complete the boot up process.

Next we show the installation of new 1TB HGST Travelstar 2.5" SATA3 HDD suitable for laptops. It works well with SATA3 SSDs too, such as the PNY 240GB XLR8 model, as well as with some SATA 2 desktop-class hard drives.



## Docking system with SATA HDD storage device attached to ODROID-C2

After connecting the kit, you can obtain details about the USB3/SATA3 bridge component used in the docking system with the following commands:

```
$ lsusb
...
Bus 001 Device 004: ID 174c:55aa ASMedia Technology Inc.  ASMedia
2105 SATA bridge
...

$ sudo lsusb -D /dev/bus/usb/001/004
[sudo] password for odroid:
Device: ID 174c:55aa ASMedia Technology Inc.  ASM1051E SATA 6Gb/s
bridge, ASM1053E SATA 6Gb/s bridge, ASM1153 SATA 3Gb/s bridge
Device Descriptor:
  bLength                18
  bDescriptorType         1
  bcdUSB                  2.10
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass         0
  bDeviceProtocol         0
  bMaxPacketSize0        64
  idVendor                0x174c ASMedia Technology Inc.
  idProduct              0x55aa ASM1051E SATA 6Gb/s bridge, ASM1053E
SATA 6Gb/s bridge, ASM1153 SATA 3Gb/s bridge
```



```
bcdDevice          1.00
iManufacturer      2 Asmedia
iProduct           3 ASM1051
iSerial            1 0123456789ABCDEF0124
bNumConfigurations 1
...
```

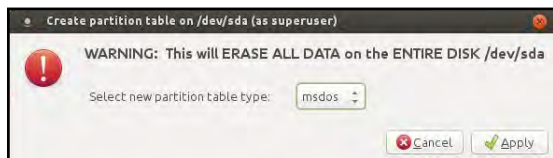
As shown in the output, the docking system uses the ASMedia's ASM1051E single-chip based SuperSpeed USB (USB 3.0) to Serial ATA3 bridge.

A brand new storage device, if unformatted, will not be automatically detected by the docking system. You can follow these steps to use the storage device on an ODROID-C2:

First, install GParted using the command:

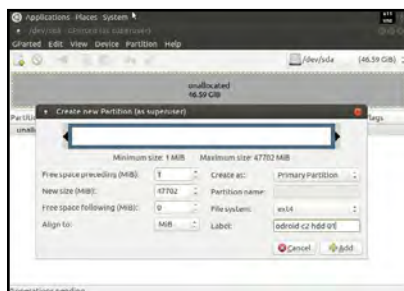
```
$ sudo apt-get install gparted
```

Then start the GParted application from the desktop's System > Administration menu. Select the device, which will typically be named /dev/sda if no other storage devices exist on the system. Highlight the unallocated partition and select the Device > Create Partition Table menu item. You will be presented with a warning as shown below.



## Partition creation warning

Click the Apply button, then highlight the unallocated partition and select the Partition > New menu item. You will be presented with a screen to enter the new partition's information, as shown below:



## New partition information

Enter a label name odroid-c2-hdd-01, and click the Add button. Select the new partition and Click on the check icon, to apply/save all changes.

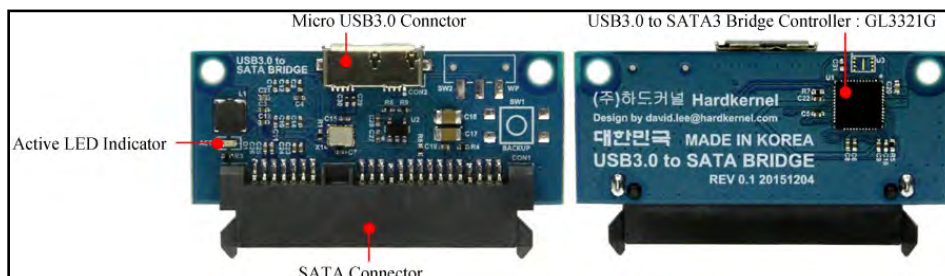
Reboot the system and run the following command to validate the creation of the new partition:

```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            733M  0    733M   0% /dev
tmpfs           172M  7.3M 165M   5% /run
/dev/mmcblk0p2  29G   6.2G 22G   23% /
tmpfs           859M  184K 859M   1% /dev/shm
tmpfs           5.0M  4.0K 5.0M   1% /run/lock
tmpfs           859M  0    859M   0% /sys/fs/cgroup
/dev/mmcblk0p1  128M  22M  107M  17% /media/boot
cgmanagerfs    100K  0    100K   0% /run/cgmanager/fs
tmpfs           172M  4.0K 172M   1% /run/user/118
tmpfs           172M  24K  172M   1% /run/user/1000
/dev/sda1       46G   52M  44G   1% /media/odroid/odroid-c2-
hdd-01
```

You can also verify using the Console application by selecting the Places > Computer menu from the desktop, which should show that the HDD has been installed and mounted, and is ready for use.

## USB3 to SATA Bridge Board

This board contains a USB3.0 to SATA3 controller (Genesis GL3321G) that allows interfacing of a single 2.5" SATA HDD/SDD. While providing power to the storage, it also provides access to the data stored on the attached SATA storage device.



## SATA bridge board

To use the controller, first power down the ODROID-C2, then insert the compatible SATA storage device into the bridge board. Attach the included USB cable to both the board and the ODROID-C2 and power up the SBC.

Details about the accessory can be found using the following commands:

```
$ lsusb
Bus 001 Device 004: ID 05e3:0735 Genesys Logic, Inc.

$ sudo lsusb -D /dev/bus/usb/001/004
Device: ID 05e3:0735 Genesys Logic, Inc.
Device Descriptor:
  bLength           18
  bDescriptorType   1
```

```
bcdUSB                2.10
bDeviceClass          0 (Defined at Interface level)
bDeviceSubClass       0
bDeviceProtocol       0
bMaxPacketSize0      64
idVendor              0x05e3 Genesys Logic, Inc.
idProduct             0x0735
bcdDevice             10.03
iManufacturer         1 USB Storage
iProduct              2 USB3.0 SATA Bridge
iSerial               3 0000000000000003
bNumConfigurations    1
```

...



## SSD used with SATA bridge board

You can use the procedure listed in the previous section to mount the storage device attached. The following command shows how it would appear to the system:

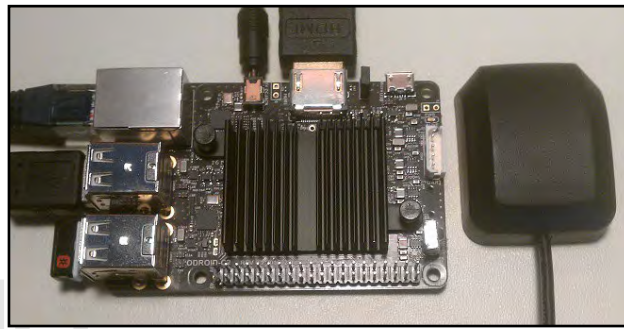
```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
...
/dev/sda1       46G   52M   44G    1% /media/odroid/odroid-c2-hdd-01
```

Once mounted this way, you can proceed to use the SATA storage device.

## USB GPS Module

Hardkernel produces a 5V 0.1A GPS receiver with an USB interface that supports the standard National Marine Electronics Association (NMEA) GPS protocol, as shown next page.

The device uses a Ublox 6010 chipset, with support already built into the official Hardkernel Lubuntu image. One should be able to attach the device to an available USB port on the C2 and start using it right away:



## USB GPS module

```
$ lsusb
...
Bus 001 Device 004: ID 1546:01a6 U-Blox AG
...
$ sudo ls -lsa /dev/ttyA*
0 crw-rw---- 1 root dialout 166, 0 Dec 31 1979 /dev/ttyACM0
```

Now that the relevant TTY port is obtained, you can check to see if the GPS receiver is doing its job, using the following command:

```
$ sudo cat /dev/ttyACM0 | grep GPRMC
$GPRMC,161053.00,A,3719.54074,N,12201.49867,W,0.079,,110415,,
,A*65
$GPRMC,161054.00,A,3719.54074,N,12201.49867,W,0.085,,110415,,
,A*65
$GPRMC,161055.00,A,3719.54074,N,12201.49867,W,0.024,,110415,,
,A*66
...
```

The RMC - NMEA has its own version of essential GPS pvt (position, velocity, time) data, which represents the following information:

```
$GPRMC,123519,A,4807.038,N,01131.000,E,022.4,084.4,230394,003.1,W*6A
```

RMC	Recommended Minimum sentence C
123519	Fix taken at 12:35:19 UTC
A	Status A=active or V=Void.
4807.038,N	Latitude 48 deg 07.038' N
01131.000,E	Longitude 11 deg 31.000' E
022.4	Speed over the ground in knots
084.4	Track angle in degrees True
230394	Date - 23rd of March 1994
003.1,W	Magnetic Variation
*6A	The checksum data, always begins with *

To test the higher level functionality of the GPS dongle, we can use the services of `gpsd`, a service daemon, that monitors one or more GPS modules and makes the pcv (position, course, velocity) data available via the TCP port 2947 of the host system.

# Chapter 4

Install `gpsd` and relevant utilities using the following command:

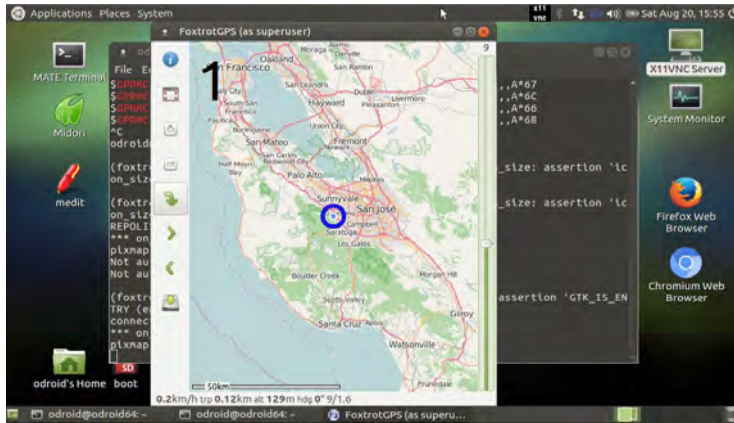
```
$ sudo apt-get install gpsd gpsd-clients foxtrotgps
```

Then, configure `gpsd` using the following command and options, then reboot:

```
$ sudo dpkg-reconfigure gpsd
```

From a terminal window, launch the `foxtrotgps` application using the following command:

```
$ foxtrotgps
```



## Foxtrotgps display

Note that the display of a live map, as shown, requires the presence of a working Internet connection and clear exposure to blue sky for GPS signals. If you wish to see a real-time map while you are driving in a vehicle with this setup, you will need to use your smartphone as a hot-spot and have the ODROID communicate with it via WiFi.

## myAHRS+ board

If you are looking for an Attitude Heading Reference System (AHRS), less influenced by acceleration and magnetic disturbances, the low cost USB2-based myAHRS+ board offered by Hardkernel is a great choice. It also offers an I2C interface, which is useful in an embedded application like Arduino-based projects.

It includes three sensors:

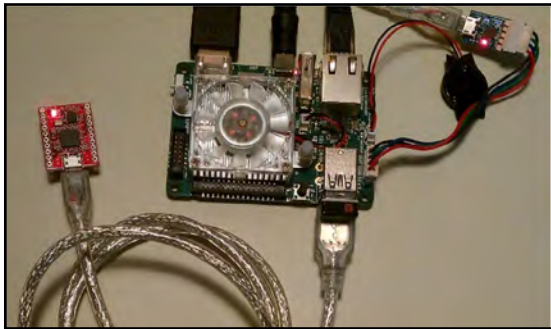
- triple axis 16-bit gyroscope :  $\pm 2000$  dps
- triple axis 16-bit accelerometer :  $\pm 16$  g
- triple axis 13-bit magnetometer :  $\pm 1200$   $\mu$ T



# Chapter 4

Attach the board to the USB2 port on the ODROID-C2. After a few moments, you should observe the following two LEDs:

- Red LED:  
When ON, the myAHRs+ is connected to PC via USB correctly.
- Yellow LED:  
When blinking, it means that myAHRs+ is in normal state.



## myAHRs+ attached to an ODROID-C2

To get the device information, type the following commands:

```
$ lsusb
...
Bus 001 Device 010: ID 0483:5740 STMicroelectronics STM32F407
...

$ sudo lsusb -D /dev/bus/usb/001/010
Device: ID 0483:5740 STMicroelectronics STM32F407
Couldn't open device, some information will be missing
Device Descriptor:
  bLength                18
  bDescriptorType        1
  bcdUSB                  2.00
  bDeviceClass            2 Communications
  bDeviceSubClass         0
  bDeviceProtocol         0
  bMaxPacketSize0        64
  idVendor                0x0483 STMicroelectronics
  idProduct              0x5740 STM32F407
  bcdDevice              2.00
  iManufacturer          1
  iProduct               2
  iSerial                3
  bNumConfigurations     1
...
```

The board can be configured using the steps described at <http://bit.ly/1TGve9g>. Run the following command to view typical

output from the board while moving the board around:

```
$ sudo cat /dev/ttyACM0
$RPY,58,7.95,1.95,79.85*59
$RPY,61,75.92,66.04,132.32*6A
$RPY,62,88.22,59.02,148.94*6B
$RPY,70,-12.06,59.27,48.14*7E
$RPY,85,4.25,22.07,41.41*68
...
```

ODROID-C2 specific sample applications can be obtained from <http://bit.ly/2bFcAjO>. Run the following commands to test an OpenGL sample:

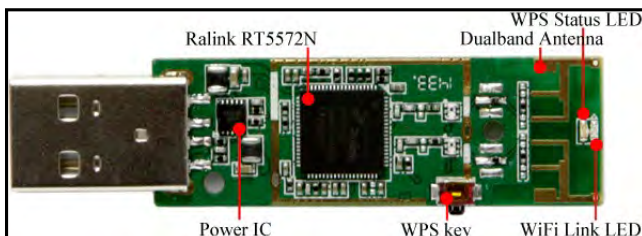
```
$ cd ~ && mkdir ahrs/ && cd ahrs/
$ svn export https://github.com/withrobot/myAHRS_plus/trunk/com-
mon_python
$ cd common_python/basic_example
$ sudo apt-get install python-serial
$ sudo python python_example.py /dev/ttyACM0
```

## WiFi Module 4

The WiFi Module 4 is the latest generation WiFi module offered by Hardkernel. Some of the features that the module supports includes the following:

- IEEE 802.11a/b/g/n WLAN module with on-board 2.4Ghz and 5Ghz Dual band antenna
- USB 2.0 host interface
- LED for WiFi Link Activity and WPS status indicators
- MediaTek(Ralink) RT5572N chipset

Attach the module to the ODROID-C2 as shown in the image below, and wait for a few moments.



## WiFi module and it attached to an ODROID-C2

Details of the device can be found using the following commands:

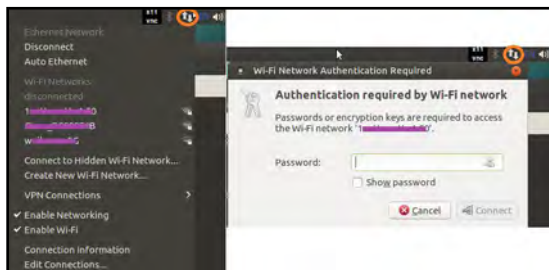
```
$ lsusb
...
Bus 001 Device 014: ID 148f:5572 Ralink Technology, Corp. RT5572
```

# Chapter 4

```
Wireless Adapter
...
$ sudo lsusb -D /dev/bus/usb/001/014
[sudo] password for odroid:
Device: ID 148f:5572 Ralink Technology, Corp. RT5572 Wireless
Adapter
Device Descriptor:
  bLength                18
  bDescriptorType        1
  bcdUSB                  2.00
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass        0
  bDeviceProtocol        0
  bMaxPacketSize0       64
  idVendor                0x148f Ralink Technology, Corp.
  idProduct              0x5572 RT5572 Wireless Adapter
  bcdDevice              1.01
  iManufacturer          1 Ralink
  iProduct               2 802.11 n WLAN
  iSerial                3 1.0
  bNumConfigurations    1
...
```

The module can be configured to use your WiFi network. Click the up-down arrow icon on the top right as shown in the image below. You will first be presented with the option to enable WiFi.

Enable the WiFi adapter, and it will automatically scan and present the various networks that can be accessed. Select your WiFi network, and you will be presented with a login window. Enter the password and click the Connect button. In a few moments, you should be connected to the network. You can then run the `ifconfig` command to obtain connection details, such as IP address and packets transferred. Other WiFi modules offered by Hardkernel can be used using the same steps listed above.



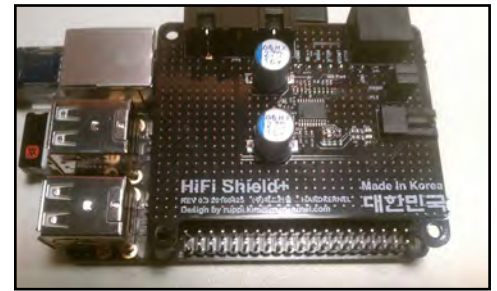
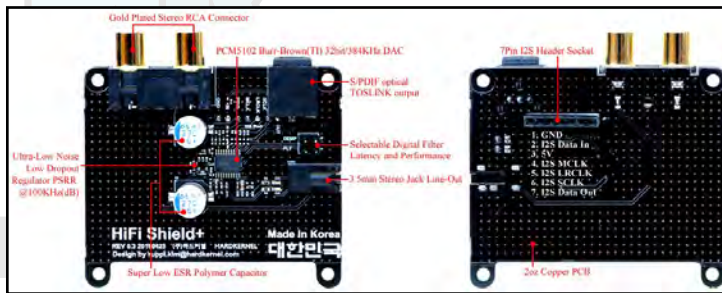
## WiFi configuration

### HiFi Shield+

To use the HiFi Shield+, follow the installation and configuration guide available at <http://bit.ly/2brrGdG>. After installation and power-up, run the following commands to let the system detect the new shield labeled ODR0IDDAC, along with the details of the

# Chapter 4

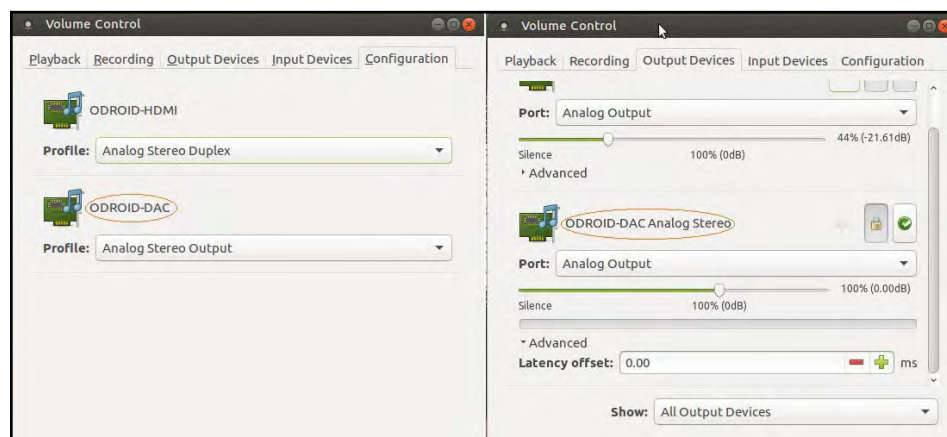
built-in HDMI (ODROIDHDMI):



## HiFi Shield+ in detail and plugged to the ODROID-C2

```
$ sudo modprobe snd-soc-pcm5102
[sudo] password for odroid:
$ sudo modprobe snd-soc-odroid-dac
$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: ODROIDHDMI [ODROID-HDMI], device 0: I2S dit-hifi-0 []
  Subdevices: 0/1
  Subdevice #0: subdevice #0
card 1: ODROIDDAC [ODROID-DAC], device 0: I2S pcm5102-0 []
  Subdevices: 0/1
  Subdevice #0: subdevice #0
```

Following the same steps listed in the S/PDIF section, you can configure the setup using the available Pulse Audio tool, with the S/PDIF cable attached. Next page we demonstrate how the default Analog Audio option is selected. The Digital S/PDIF option may also be selected similarly.

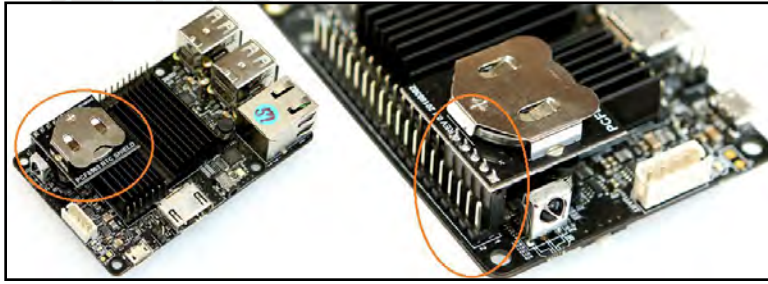


## HiFi Shield+ configuration

### RTC Shield

Like many of the earlier ODROID models, the ODROID-C2 does not come with RTC support on the main PCB. However, Hardkernel has developed a special add-on shield that can be mounted on the available GPIO header, at the designated pins shown below.

# Chapter 4



## RTC shield mounted

The included Lithium-coin battery can keep the RTC chip running up to 4 years without an external power supply, thereby helping with the maintenance of the clock. The official ODRROID-C2 case can be used even with this shield mounted.

After connecting the C2 to the Internet, enable the shield:

```
$ sudo apt install device-tree-compiler
$ sudo fdtput -t s /media/boot/meson64_odroidc2.dtb \
/i2c@c1108500/pcf8563@51 status "okay"
```

Edit the `/etc/modules` file using any text editor:

```
$ sudo vi /etc/modules
```

Then, add the text `aml_i2c` at the end of the file and save it. Similarly, update the `/etc/rc.local` file:

```
$ sudo vi /etc/rc.local
```

Then, add the following command to the bottom of the file, save the file, and reboot the system:

```
$ sudo hwclock -r
```

Ensure the system is able to access the ntp servers on the internet. Run the following command:

```
$ ntpq -p
Remote          refid          st t when poll reach  delay  off-
set  jitter
=====
=====
0.ubuntu.pool.n .POOL.        16 p -   64    0   0.000 0 . 0 0 0
0.001
1.ubuntu.pool.n .POOL.        16 p -   64    0   0.000 0 . 0 0 0
0.001
2.ubuntu.pool.n .POOL.        16 p -   64    0   0.000 0 . 0 0 0
0.001
```



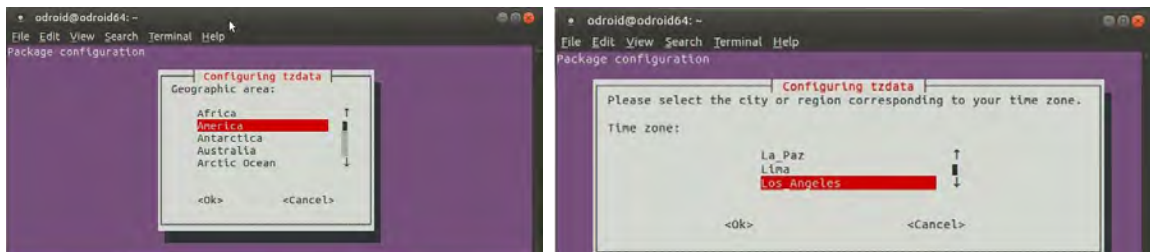
# Chapter 4

```
3.ubuntu.pool.n .POOL.      16 p - 64      0      0.000 0 . 0 0 0
0.001
ntp.ubuntu.com .POOL.      16 p - 64      0      0.000 0 . 0 0 0
0.001
-time-b.timefreq .ACTS.      1  u  227  256  373    70.887  -10.486
2.120
+king.slash31.co 216.218.192.202  2  u   46  128  377    24.454
-0.711  0.322
*time.tritn.com  216.218.192.202  2  u  103  128  377    32.545
-0.605  0.863
-borris.netwurx. 209.242.224.11  2  u  221  256  377    77.102
0.156   0.850
...
```

Next, update the timezone information:

```
$ sudo dpkg-reconfigure tzdata
```

In the first window presented, click on the OK button. In the subsequent windows, enter information relevant to your location and save, as shown.



## Timezone setup

With this connectivity ensured, and timezone setup right, obtain the correct time from any clock, and update the system:

```
$ sudo date -s "20 APR 2016 18:00:00"
$ sudo hwclock -w
```

Reboot and check the timezone and clock settings. It should keep proper time now, syncing with the ntp servers periodically. Detailed steps can be found at <http://bit.ly/2btCAzP>.

## UPS3

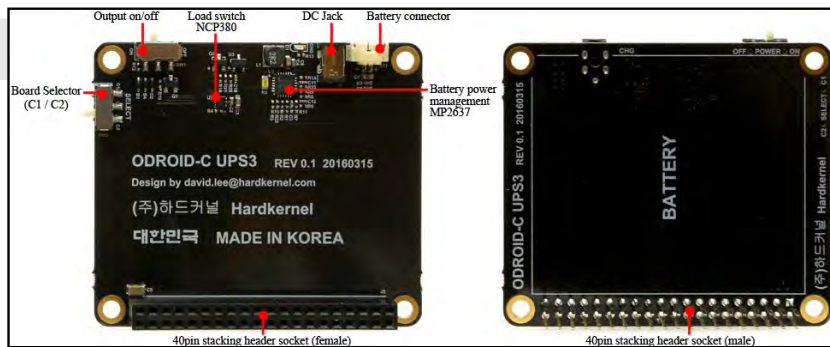
There are many situations when an Uninterrupted Power Supply (UPS) is essential to ensure that the ODROID-C2 is functioning well, even with loss of main power. Hardkernel has developed a 3000mAh UPS kit. This LiPo battery kit, when used in conjunction with the ODROID-C2, charges the included battery pack.

The UPS is designed so that, if the main power source fails, it takes over as the main power source. After the battery has drained to a

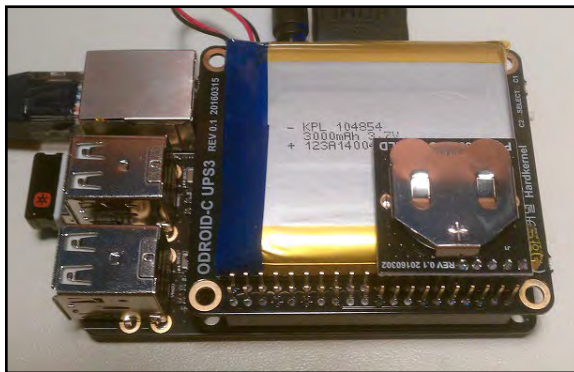
# Chapter 4

low level, it signals the ODROID-C2 through a digital output pin, triggering a graceful shutdown. Since the cache/file operations are likely to be flushed out before the power is turned off, the risk of data loss is low.

Below we illustrate the options of the kit, and how its use can be controlled. Shown below the fully assembled kit mounted on an ODROID-C2. Additional details can be found at <http://bit.ly/2b8XW77>.



## UPS3 PCB



## UPS3 installed

## Xprotolab Plain

Xprotolab Plain is a kit that includes an Oscilloscope, Logic Analyzer, Waveform, Clock generator, 10x hook probe, all in one tiny lightweight and low cost package.



## Xprotolab Plain connected to an ODROID-C2

The main board was developed by Gabotronics, who has provided an open-source QT5-based cross-platform visualization software, which has been ported to the ODROID-C2. To use it with the ODROID-C2, you must remove the Mali graphics package and install

# Chapter 4

the relevant components:

```
$ sudo apt-get remove mali-x11
$ sudo apt-get install qt5-default libqt5serialport5-dev libusb-
1.0-0-dev
$ sudo apt download mali-x11
$ sudo dpkg -i --force-overwrite mali-x11*.deb
$ sudo apt-get install xserver-xorg-video-mali
$ sudo ldconfig
```

Then, install the open-source application using the following commands:

```
$ cd ~
$ git clone https://github.com/ganzziani/xscopes-qt
$ cd xscopes-qt
$ qmake
$ make -j5
```

After connecting the cable as shown in the image on the previous page, launch the application using the following command:

```
$ sudo ./xscope
```

Usage details for the Xprotolab Plain can be found at <http://bit.ly/22xfjzn>.

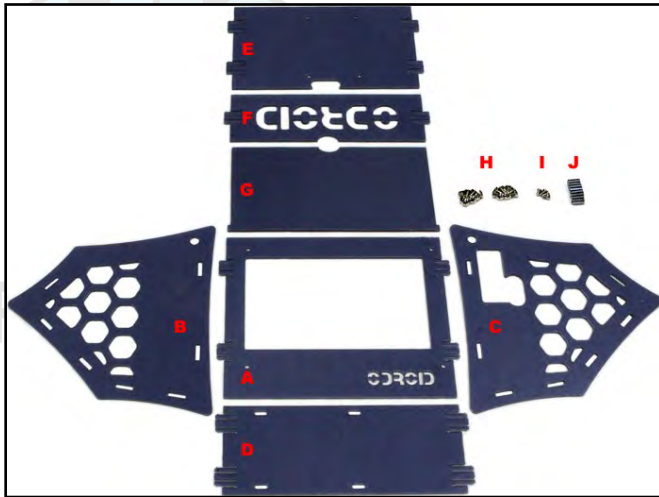


## Xprotolab Plain data acquisition

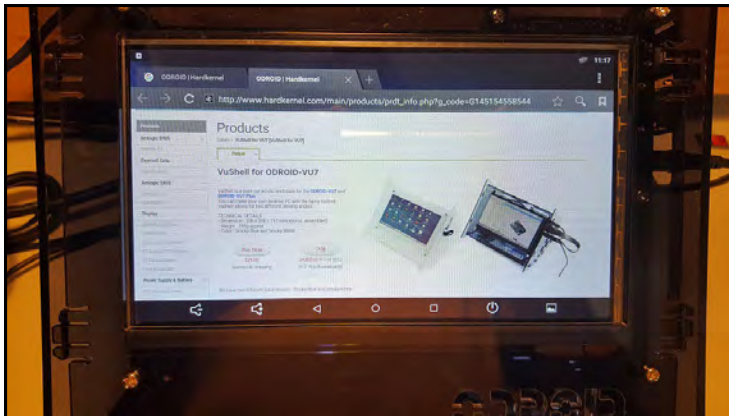
### ODROID-VuShell for ODROID-VU7

The VuShell is a laser-cut acrylic enclosure for the ODROID-VU7 and ODROID-VU7 Plus. With the use of the ODROID-C2, you can create your own desktop PCI. Its design allows for two different viewing angles. The components of the VuShell are shown here.

Detailed assembly instructions can be found at <http://bit.ly/2b81k6a>. The assembled VuShell housing an ODROID-C2 running Android, is shown next.



**ODR0ID VuShell components**



**ODR0ID VuShell assembled with ODR0ID-C2**

## Universal Motion Joypad

The Universal Motion Joypad is a modern game controller that is similar to the steering wheel available for the Nintendo Wii. It is built using the USB IO BOARD and an auxiliary motion sensor, and allows you to play many games available for smartphones that would normally use the phone's gyroscope.

Use the detailed assembly instructions at <http://bit.ly/1S-be46q>, and attach it to the ODR0ID-C2 running the latest Android image in order to play games, as demonstrated in the January 2016 issue of ODR0ID Magazine on page 18 (<http://bit.ly/2byzrQq>).

Before installing games on the ODR0ID-C2, you need to install the Google Play store, which is detailed at <http://bit.ly/2aWS696>. After installing it, you can use your Google account to log into the store and download games. The image below shows a scene from the Beach Buggy Racing game, which is an ideal game for use with the Universal Motion Joypad.

# Chapter 4



**Beach Buggy Racing screenshot**

## *Conclusion*

The ODROID-C2 is compatible with numerous hardware gadgets, and many USB sensors may be used as long as the Linux driver is available. The hardware sold by Hardkernel at <http://bit.ly/2bWxgrK> have the advantage of having pre-configured drivers included with the official Hardkernel disk images. We hope you enjoy tinkering and building your own projects using some of the techniques described here.

## *Additional Resources*

ODROID Forums: <http://forum.odroid.com>

ODROID Magazine: <http://magazine.odroid.com>

ODROID-C2 Wiki: <http://bit.ly/1Trq5Ef>

Android images: <http://bit.ly/2be993R>

Linux images: <http://bit.ly/2b58GEe>

Improved Win32 DiskImager: <http://bit.ly/11YQ7MF>

Hardkernel store: <http://bit.ly/1fbE9ld>