

LIRC Setup for IR Blaster with Ubuntu 16.04

Install LIRC Package

```
$ sudo apt-get install lirc
```

While installing lirc, you will be asked about the device type of IR receiver and transmitter. Since we will set up later manually, just select **None**.

Configuring LIRC and the driver module

lirc_odroid module driver is already included by default.

Now you need to set up hardware.conf and lircd.conf first and then probe the modules with some parameters.

LIRC Configuration

hardware.conf

hardware.conf is generated by default when lirc is installed and in this conf file, devices and drivers information is supposed to be specified.

```
$ sudo vi /etc/lirc/hardware.conf
```

Specify the kernel driver module and its node.

```
#Chosen IR Transmitter
TRANSMITTER="odroid blaster"
TRANSMITTER_MODULES="lirc_odroid lirc_dev"
TRANSMITTER_DRIVER=""
TRANSMITTER_DEVICE="/dev/lirc0"
TRANSMITTER_SOCKET=""
TRANSMITTER_LIRCD_CONF=""
TRANSMITTER_LIRCD_ARGS=""
```

Please refer to this link for the whole contents of hardware.conf.

[hardware.conf](http://wiki.odroid.com/hardware.conf)

lircd.conf

lircd.conf is also generated automatically via the Ubuntu LIRC package scripts. It includes signal timing information and provides a mapping from button presses to key symbols. You can modify this file based on your custom remote controller.

```
$ sudo vi /etc/lirc/lircd.conf
```

This reference lircd.conf is generated to support Hardkernel IR remote controller.

[lircd.conf](#)

<https://www.hardkernel.com/shop/ir-remote-controller/>

```
begin remote
  name odroid
  bits 16
  . . . . .
  . . . . .
```

Install module and Run lirc service

All necessary configuration is done, you need to install lirc_odroid module and set some parameters.

```
# The usage of lirc_odroid module
lirc_odroid gpio_out_pin=${gpio_number} invert=[0 or 1] softcarrier=[0 or 1]
```

Examples

ODROID-N2

Supposing to use Pin#15 of ODROID-N2 (GPIO #483),

```
$ service lirc stop
$ modprobe lirc_dev
$ modprobe lirc_odroid gpio_out_pin=483 softcarrier=1 invert=1
(automatically, service lirc will be started)
```

ODROID-C4

Supposing to use Pin#13 of ODROID-C4 (GPIO #480),

```
$ service lirc stop
```

```
$ modprobe lirc_dev
$ modprobe lirc_odroid gpio_out_pin=480 softcarrier=1 invert=1
(automatically, service lirc will be started)
```

ODROID-C2

Supposing to use Pin#11 of ODROID-C2 (GPIO #247),

```
$ service lirc stop
$ modprobe lirc_dev
$ modprobe lirc_odroid gpio_out_pin=247 softcarrier=1 invert=1
(automatically, service lirc will be started)
```

ODROID-C1/C1+

Supposing to use Pin#7 of ODROID-C1/C1+ (GPIO #83),

```
$ service lirc stop
$ modprobe lirc_dev
$ modprobe lirc_odroid gpio_out_pin=83 softcarrier=1 invert=1
(automatically, service lirc will be started)
```

ODROID-XU4

Supposing to use Pin#26 of ODROID-XU4 (GPIO #24),

```
$ service lirc stop
$ modprobe lirc_dev
$ modprobe lirc_odroid gpio_out_pin=24 softcarrier=1 invert=1
(automatically, service lirc will be started)
```

If you have the warning **No such device** during **modprobe lirc_odroid**, check if the `gpio_out_pin` is already assigned to another purpose and change to another pins.

Send IR code using 'irsend'

If `lirc_odroid` module has been loaded and `lirc` service is running normally, you can get the key list as following.

Please make sure the remote name is same with the **name** described in **/etc/lirc/lircd.conf**.

```
$ irsend LIST odroid ""
irsend: 00000000000009966 KEY_LEFT
irsend: 0000000000000837c KEY_RIGHT
irsend: 000000000000053ac KEY_UP
irsend: 00000000000004bb4 KEY_DOWN
irsend: 0000000000000738c KEY_ENTER
irsend: 000000000000041be KEY_HOME
irsend: 000000000000011ee KEY_MUTE
irsend: 0000000000000a35c KEY_MENU
irsend: 000000000000059a6 KEY_BACK
irsend: 0000000000000817e KEY_VOLUMEDOWN
irsend: 000000000000001fe KEY_VOLUMEUP
irsend: 00000000000003bc4 KEY_POWER
```

irsend is basic LIRC program to send the key code.

```
$ irsend SEND_ONCE odroid KEY_ENTER
```

For more detailed information, please refer to this link.

<http://www.lirc.org/html/irsend.html>

Reference Configuration Files

hardware.conf

[hardware.conf](#)

```
# /etc/lirc/hardware.conf
#
#Chosen Remote Control
REMOTE="None"
REMOTE_MODULES=""
REMOTE_DRIVER=""
REMOTE_DEVICE=""
REMOTE_SOCKET=""
REMOTE_LIRCD_CONF=""
REMOTE_LIRCD_ARGS=""

#Chosen IR Transmitter
TRANSMITTER="odroid blaster"
TRANSMITTER_MODULES="lirc_odroid lirc_dev"
TRANSMITTER_DRIVER=""
TRANSMITTER_DEVICE="/dev/lirc0"
TRANSMITTER_SOCKET=""
TRANSMITTER_LIRCD_CONF=""
```

```
TRANSMITTER_LIRCD_ARGS=""

#Disable kernel support.
#Typically, lirc will disable in-kernel support for ir devices in order
to
#handle them internally. Set to false to prevent lirc from disabling
this
#in-kernel support.
#DISABLE_KERNEL_SUPPORT="true"

#Enable lircd
START_LIRCD="true"

#Don't start lircmd even if there seems to be a good config file
#START_LIRCMD="false"

#Try to load appropriate kernel modules
LOAD_MODULES="true"

# Default configuration files for your hardware if any
LIRC_CMD_CONF=""

#Forcing noninteractive reconfiguration
#If lirc is to be reconfigured by an external application
#that doesn't have a debconf frontend available, the noninteractive
#frontend can be invoked and set to parse REMOTE and TRANSMITTER
#It will then populate all other variables without any user input
#If you would like to configure lirc via standard methods, be sure
#to leave this set to "false"
FORCE_NONINTERACTIVE_RECONFIGURATION="false"
START_LIRC_CMD=""
```

lircd.conf

[lircd.conf](#)

```
begin remote

name odroid
bits 16
flags SPACE_ENC|CONST_LENGTH
eps 30
aeps 100

header 9000 4500
one 563 1688
zero 563 564
ptrail 563
```

```
pre_data_bits 16
pre_data 0x4DB2
repeat 9000 2250
gap 100000
toggle_bit_mask 0x0
  begin codes
    KEY_LEFT 0x9966
    KEY_RIGHT 0x837C
    KEY_UP 0x53AC
    KEY_DOWN 0x4BB4
    KEY_ENTER 0x738C
    KEY_HOME 0x41BE
    KEY_MUTE 0x11EE
    KEY_MENU 0xA35C
    KEY_BACK 0x59A6
    KEY_VOLUMEDOWN 0x817E
    KEY_VOLUMEUP 0x01FE
    KEY_POWER 0x3BC4
  end codes
end remote
```

To support both of IR Tx and Rx simultaneously

```
#Chosen Remote Control
REMOTE="None"
REMOTE_MODULES="meson-ir"
REMOTE_DRIVER=""
REMOTE_DEVICE="/dev/lirc0"
REMOTE_SOCKET=""
REMOTE_LIRCD_CONF=""
REMOTE_LIRCD_ARGS="--uinput"

#Chosen IR Transmitter
TRANSMITTER="Home-brew (odroid gpio)"
TRANSMITTER_MODULES="lirc_odroid lirc_dev"
TRANSMITTER_DRIVER=""
TRANSMITTER_DEVICE="/dev/lirc1"
TRANSMITTER_SOCKET=""
TRANSMITTER_LIRCD_CONF=""
TRANSMITTER_LIRCD_ARGS=""
...
...
```

From:

<http://wiki.odroid.com/> - **ODROID Wiki**

Permanent link:

http://wiki.odroid.com/common/application_note/gpio/lirc_gpio_blaster/irblaster_lirc_setup_ubuntu16.04

Last update: **2020/04/23 08:09**

