U-boot

You **MUST** build u-boot on your Host PC.
This guide doesn't support native build.

Toolchain

Click one of the site to download toolchain to build U-boot. Please note that this toolchain is for U-boot, there is another toolchain to build Linux kernel.

- Download #1
- Download #2

Once the download is done, extract the tarball to **/opt/toolchains/**.

```
$ sudo mkdir -p /opt/toolchains
$ sudo tar xJvf gcc-linaro-arm-none-eabi-4.8-2014.04_linux.tar.xz -C
/opt/toolchains/
```

In order to add the toolchain path to PATH, paste below lines to **$HOME/.bashrc**.

```
export PATH=/opt/toolchains/gcc-linaro-arm-none-
eabi-4.8-2014.04_linux/bin:$PATH
```

You can apply the change if you login again or import to apply this change, login again or evaluate **$HOME/.bashrc** with source command.

```
$ source ~/.bashrc
```

You can check if the toolchain installed above works properly while checking the version of toolchain. If you can find **gcc version 4.8.3 20140401 (prerelease)** at the end of the line, the toolchain is well installed.

```
$ arm-none-eabi-gcc -v
Using built-in specs.
COLLECT_GCC=arm-none-eabi-gcc
COLLECT_LTO_WRAPPER=/opt/toolchains/gcc-linaro-arm-none-
eabi-4.8-2014.04_linux/bin/../libexec/gcc/arm-none-eabi/4.8.3/lto-wrapper
Target: arm-none-eabi
Configured with: /cbuild/slaves/oorts/crosstool-ng/builds/arm-none-eabi-
linux/.build/src/gcc-linaro-4.8-2014.04/configure --build=i686-build_pc-
linux-gnu --host=i686-build_pc-linux-gnu --target=arm-none-eabi --
prefix=/cbuild/slaves/oorts/crosstool-ng/builds/arm-none-eabi-linux/install
--with-local-prefix=/cbuild/slaves/oorts/crosstool-ng/builds/arm-none-eabi-
linux/install/arm-none-eabi --without-headers --with-newlib --enable-
```

```
threads=no --disable-shared --with-pkgversion='crosstool-NG
linaro-1.13.1-4.8-2014.04 - Linaro GCC 4.8-2014.04' --with-
bugurl=https://bugs.launchpad.net/gcc-linaro --disable-__cxa_atexit --with-
gmp=/cbuild/slaves/oorts/crosstool-ng/builds/arm-none-eabi-linux/.build/arm-
none-eabi/build/static --with-mpfr=/cbuild/slaves/oorts/crosstool-
ng/builds/arm-none-eabi-linux/.build/arm-none-eabi/build/static --with-
mpc=/cbuild/slaves/oorts/crosstool-ng/builds/arm-none-eabi-linux/.build/arm-
none-eabi/build/static --with-isl=/cbuild/slaves/oorts/crosstool-
ng/builds/arm-none-eabi-linux/.build/arm-none-eabi/build/static --with-
cloog=/cbuild/slaves/oorts/crosstool-ng/builds/arm-none-eabi-
linux/.build/arm-none-eabi/build/static --with-
libelf=/cbuild/slaves/oorts/crosstool-ng/builds/arm-none-eabi-
linux/.build/arm-none-eabi/build/static --enable-lto --enable-linker-build-
id --enable-libmudflap --disable-libgomp --enable-libssp --disable-
libstdcxx-pch --enable-multilib --enable-languages=c,c++,fortran --with-
multilib-list=aprofile
Thread model: single
gcc version 4.8.3 20140401 (prerelease) (crosstool-NG
linaro-1.13.1-4.8-2014.04 - Linaro GCC 4.8-2014.04)
```

Checkout & compile

You can checkout **U-boot** source tree from **Hardkernel's Github**.

```
$ git clone https://github.com/hardkernel/u-boot.git -b odroidc-v2011.03
```

Before you compile **U-boot**, you must configure for **ODROID-C1** with following command, and then compile.

For Ubuntu and Android, the different config option is used.

Ubuntu

```
$ cd u-boot
$ make odroidc_config
$ make
```

Android 5.1.1 and 4.4.4

```
$ cd u-boot
$ make odroidc_rev2_config
$ make
```

Android 4.4.2

```
$ cd u-boot
$ make odroidc_config
$ make
```

Installation

The bootloader for **ODROID-C1** is consisted with two part, **1st stage bootloader** + **U-boot**. You can find **u-boot.bin** once you compile **U-boot**, as well as **bl1.bin.hardkernel** in the directory **sd_fuse/**. Both binaries must be installed in a card (eMMC or MicroSD).

- The 1st stage bootloader, **bl1.bin.hardkernel**, is provided as prebuilt binary only.
- Bootloader signing process is not necessary for **ODROID-C1** at all.

Installation to blank card

We provide the script, **sd_fuse/sd_fusing.sh**, this helps you to install the bootloader into your blank card eMMC or MicroSD.

1. Insert your card to USB card reader and attach to USB host port of your desktop.
2. Check the device path of your USB card reader.
3. Install the bootloader binaries using **sd_fuse/sd_fusing.sh**.

```
$ cd sd_fuse
$ ./sd_fusing.sh <device/path/of/your/card>
```

Installation using fastboot

If you can boot your **ODROID-C1** already and want to install a new **u-boot.bin** built by you. Fastboot helps you to install a **u-boot.bin** into your board.

```
$ sudo fastboot flash bootloader sd_fuse/u-boot.bin
```

If installation is done, you care reboot your **ODROID-C1** with fastboot.

```
$ fastboot reboot
```

Updating from Linux on your ODROID-C1

You also can update U-boot from Linux on the ODROID-C1 with **dd** command. After building U-boot, copy **u-boot.bin** into your **ODROID-C1**. Then do the command below in order to flash your U-boot

image to MicroSD or eMMC.

```
$ sudo dd if=u-boot.bin of=/dev/mmcblk0 bs=512 seek=64
$ sudo sync
```