Contents

[I. Host Development System: 2](#_Toc522546745)

[1.1. Ubuntu 14.04-64bit 2](#_Toc522546746)

[II. Build Linux Kernel for azalea\_r1576: 2](#_Toc522546747)

[III. Build Bootloader for azalea\_r1760 4](#_Toc522546748)

[3.1 Build Bootloader: 4](#_Toc522546749)

[3.2 Build early camera: 4](#_Toc522546750)

[IV. dolphin\_plus\_bsp 4](#_Toc522546751)

[4.1. Build U-boot: 4](#_Toc522546752)

[4.2. Build Linux 4.4.120: 5](#_Toc522546753)

# Host Development System:

## Ubuntu 14.04-64bit

* Install additional libraries:

$ sudo apt-get install lib32z1 lib32ncurses5 lib32bz2-1.0

* Install ARM toolchain: arm-2013.11
  + Download:

$ wget https://sourceforge.net/projects/epwa/files/arm-2013.11-33-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2

* + **Install toolchain into /opt/ for AArch32**

$ tar -xvf /home/thinhnt7/Downloads/arm-2013.11-33-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2

# Build Linux Kernel for azalea\_r1576:

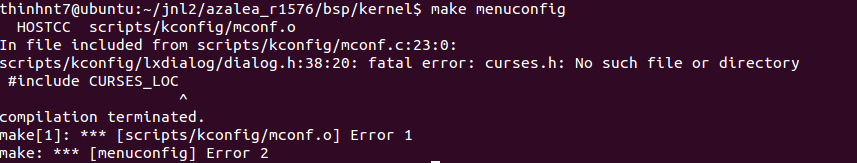
$ cd kernel/

$ export PATH=/opt/arm-2013.11/bin/:$PATH

$ export ARCH=arm CROSS\_COMPILE=/opt/arm-2013.11/bin/arm-none-linux-gnueabi-

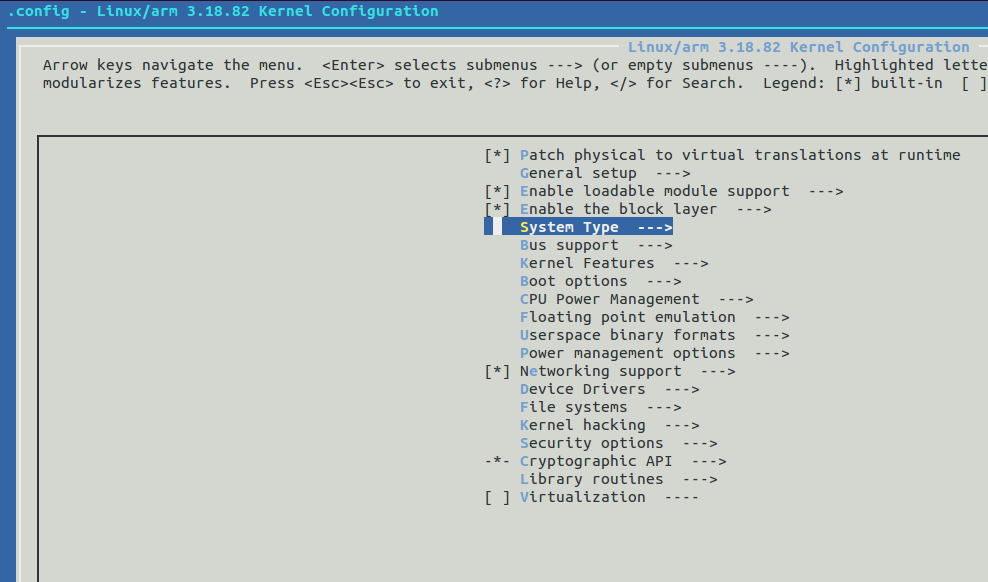
$ make jk\_19common\_fsbios\_defconfig

$ make menuconfig



**->Fix** $ sudo apt-get install libncurses5-dev libncursesw5-dev

$ make menuconfig



The custom configuration will be changed to save into the .config file for compiling.

Create a Linux kernel image:

$ make zImage

# Build Bootloader for azalea\_r1760

## Build Bootloader:

$ export PATH=/opt/arm-2013.11/bin/:$PATH #-> setup evn

$ cd azalea\_r1760/bsp/bootloader

$ make jk\_19m\_2048\_fsbios\_qboot

## Build early camera:

- Download toolchain to compile ARM cortex-M:

$ wget https://sourceforge.net/projects/epwa/files/arm-2013.11-24-arm-none-eabi-i686-pc-linux-gnu.tar.bz2

- Uncompressing them inside the directory "toolchain/gcc"

$ cd ./azalea\_r1760/bsp/bootloader/dev/camera/earlycamera

$ mkdir -p build/toolchain/gcc

$ tar -xvzf arm-2013.11-24-arm-none-eabi-i686-pc-linux-gnu.tar.bz2 buil/toolchain/gcc

- Build: (Notes: There is not bin2hex file for generating early camera file integrated inside bootloader)

$ make clean

$ make

# dolphin\_plus\_bsp

## 4.1. Build U-boot:

- Create a .config to build: notes Default is ARCH64

$ make ARCH=arm tcc803x\_defconfig

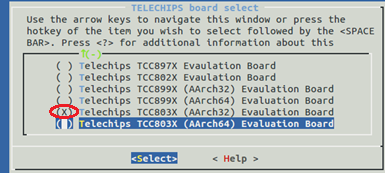
- If you want to change ARCH, follow up

$ make ARCH=arm menuconfig

And choose

ARM architecture --->

[\*] TELECHIPS board select (Telechips TCC803X (AArch32) Evaulation Board) --->



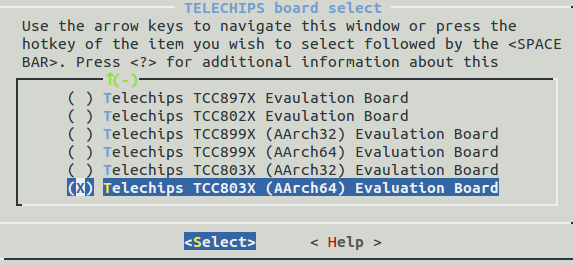
* + - Build for **AArch32**

$ make ARCH=arm CROSS\_COMPILE=arm-none-linux-gnueabi-

Notes: toolchain for AArch32 is arm-none-linux-gnueabi-

* + - Build for **AArch64**:

$ make ARCH=arm menuconfig ->select Arm architecture (AArch64)



**Notes: toolchain** for AArch64 is aarch64-linux-gnu-

$ make ARCH=arm CROSS\_COMPILE= aarch64-linux-gnu-

## 4.2. Build Linux 4.4.120:

* + - Toolchain: aarch64-linux-gnu-

Download toolchain: gcc-linaro-7.3.1-2018.05-i686\_aarch64-linux-gnu.tar.xz

Uncompressing/installing it inside /opt/

* + - Export PATH:

$ export PATH=/opt/gcc-linaro-7.3.1-2018.05-i686\_aarch64-linux-gnu/bin/:$PATH

* + - Create .config file for arm64

$ make ARCH=arm64 tcc803x\_linux\_avn\_defconfig

* + - Change any configuration if you need

$ make ARCH=arm64 menuconfig

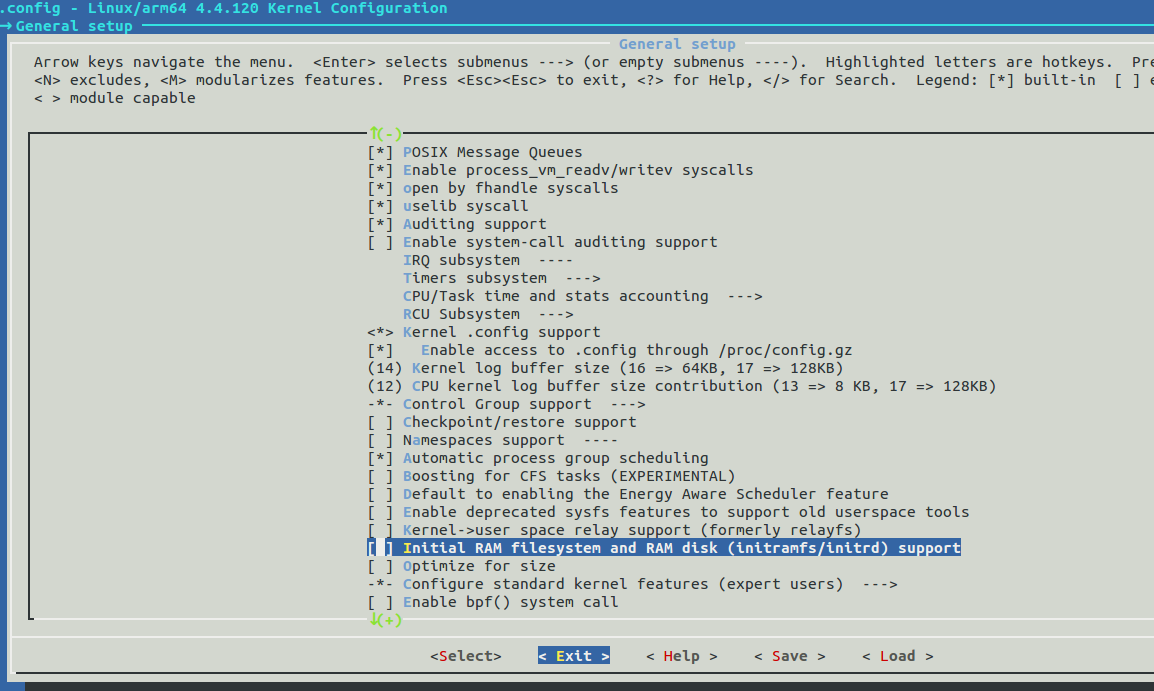
-Build kernel:

$ make ARCH=arm64 CROSS\_COMPILE=aarch64-linux-gnu-

* Error here:



-> How to fix: Enter “space” key to unselect initial RAM



* + - Build again:

Notes: bsp/kernel\_4.4/include/video/tcc/TCC\_HEVCDEC.h must keep as default of original source code.

Build:

$ make ARCH=arm64 CROSS\_COMPILE=aarch64-linux-gnu-

Reference: <https://wiki.linaro.org/HowTo/BuildArm64Kernel>