

How To Build Android

Introduction

This page describes how to build an Android USB boot image for the MS7724 development board.

Android Userland

Step 1: Setting up Your Machine

Refer to the URL below and follow the step by step instructions in "Setting up your machine" and "Installing Repo" section to set up your local work environment.

<http://source.android.com/source/download.html>

Refer to Failed to get sun-java5-jdk package also if you failed to install package sun-java5-jdk.

Step 2: Initializing a Repo Client

- Create a directory for the Android build tree.

```
$ mkdir mydroid
$ cd mydroid
```

- Initializing a Repo client with a specified tag android-2.1_r1.

```
$ repo init -u git://android.git.kernel.org/platform/manifest.git -b android-2.1_r1
```

Step 3: Initializing settings for MS7724

- Extract MS7724 patches and tools.

```
$ tar xjf (somewhere)/userland/src/android-userland-patches-20100426.tar.bz2
```

- Add local manifest for MS7724.

```
$ cp patches/tools/local_manifest.xml .repo/
```

Step 4: Getting the Files

- Download the Android source files from public Android repositories.

```
$ repo sync
<wait...>
```

Step 5: Add MS7724 patches and setup tools at the top of Android build tree

- Generate default project name and commit hash list. This will create a file named ./patches/tools/projects.list

```
$ ./patches/tools/gen_project_list.py
```

- Generate a new project repository and add it to the projects.list.

```
$ ./patches/tools/gen_repository.py hardware/renesas/shmobile
```

- Apply all patches.

```
$ ./patches/tools/apply_patches.py
```

Step 6: Building the Code

- Build Android userland for MS7724. Also refer to Options for advance setting before build.

```
$ make -j8 TARGET_ARCH=sh TARGET_PRODUCT=ms7724  
<wait...>
```

Step 7: Copy Android root file system to target mount directory

```
$ cp -r out/target/product/ms7724/root/* (target_userland)/  
$ cp -r out/target/product/ms7724/system/* (target_userland)/system/  
$ cp -r out/target/product/ms7724/data/* (target_userland)/data/
```

Android Kernel

Step 1: Use the same toolchain to build userland that was used to build the kernel.

```
$ export PATH=$PATH:(somewhere)/mydroid/prebuilt/linux-x86/toolchain/sh-4.3.3/bin/
```

Step 2: Build the Android Linux kernel.

```
$ tar xjf (somewhere)/kernel/src/android-kernel-src-20100426.tar.bz2  
$ tar xjf sh-2.6.tar.bz2  
$ cd sh-2.6/  
$ cp ../config/sh-2.6_usbboot.config .config  
$ make -j8 ARCH=sh CROSS_COMPILER=sh-linux-gnu- menuconfig  
<Select Exit>  
$ make -j8 ARCH=sh CROSS_COMPILER=sh-linux-gnu-  
<wait...>
```

Step 3: Copy the Android Linux kernel to target mount directory

```
$ cp arch/sh/boot/zImage (target_kernel)/zImage
```

Options

Camera

By default, mt9t112 camera module is required to connect at CAMERA0 connector. If you do not have camera modules and want to use a fake camera, change the following settings and rebuild userland.

```
$ gedit mydroid/vendor/renesas/ms7724/BoardConfig.mk  
change USE_CAMERA_STUB := false to USE_CAMERA_STUB := true  
$ cd mydroid/frameworks/base/  
$ git apply --reverse ../../patches/frameworks_base_0008-change-camera-preview-buffer-to-  
RGB565.patch  
$ make -j8 TARGET_ARCH=sh TARGET_PRODUCT=ms7724 clean  
$ make -j8 TARGET_ARCH=sh TARGET_PRODUCT=ms7724
```

DNS setting

A DNS server IP address is required for web browsing. Add the following setting to vendor/renesas/ms7724/init.r0p7724.rc.

```
setprop net.dns1 <your>.<dns>.<server>.<ipaddress>
```

For example, add OpenDNS to previous script file.

```
setprop net.dns1 208.67.222.222
```

Create USB keyboard's key-layout and key-maps files if necessary.

Add the [hw.keyboards.0.devname].kl and [hw.keyboards.0.devname].kcm to vendor/renesas/ms7724/, modified Androidboard.mk and rebuild userland. You can check [hw.keyboards.0.devname] on Android shell with command 'getprop'.