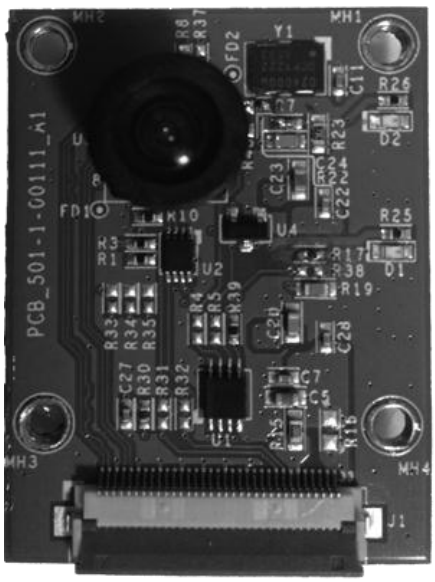
radiumboards.com

MIPI Camera Board
for
Wandboard



System Reference Manual

Revision 1.2

Copyright Notice

Legal Disclaimer

The information contained in this document is subject to change without notice. The information in this document is provided for informational purposes only. RadiumBoards specifically disclaims all warranties, express or limited, including, but not limited, to the implied warranties of merchantability and fitness for a particular purpose, except as provided for in a separate software license agreement.

Radium Boards

RadiumBoards is a unique website providing complete board and software solutions addressing a broad range of markets including Security and Surveillance, Networking, Wireless, Video, Audio, Automotive, Mobile Device and IoT (Internet of Things).

OEMs and Systems Integrators can shop for complete assemblies, with firmware, BSP and applications ready to integrate into your own enclosures.

ODMs can shop for reference designs complete with full BSP (Board Support Package) and application support. Radium Boards can also provide any level of customization to the hardware or software required by the ODM to help differentiate in their markets.

Electronics hardware and software hobbyists, experimenters and educators now have access to complete high performance platforms for application in an infinite range of projects.

Correspondence

Corporate Office
B-22, Infocity Sector-34,
Gurgaon-122001, Haryana, India
Tel No: +91 124 4284250

US Office
2025 Gateway Place, Suite 465,
San Jose, California 95110
Tel No: +1 408 807 3951

E-MAIL
info@radiumboards.com

WEBSITE
www.radiumboards.com

VVDN Technologies

VVDN Technologies Pvt Ltd. is a sibling company of RadiumBoards and is responsible for the design and development of all products sold through the RadiumBoards brand.

Founded in 2007, VVDN is a technology innovation and development company providing a broad spectrum of services and technology expertise to our core domains. VVDN provides “Concept to Customer” services at any point in the development cycle, as well as full turnkey solutions.

WEBSITE
www.vvdntech.com

Revision History

Revision History tracks the changes made for each revision of this document.

Date	Rev.No	Description	By
	1.1	• Initial Release	RadiumBoards
6 Feb 2014	1.2	• Provides full Android Source Code with Patches applied • Added SD Card Creation from prebuilt using Ubuntu • Updated fstab.wandboard contents • /boot directory created	RadiumBoards

Contents

List Of Figures

List Of Tables

1 MIPI Camera board for Wandboard Quad Overview

1.1 Descriptions

The MIPI Camera board for Wandboard provides a portable camera solution for Wandboard. The Camera board carries a 1.26 Mega pixels camera sensor and transmits captured image data to the Freescale i.MX6Q processor via MIPI (Mobile Industry Processor Interface). Camera board connects with Wandboard via 33 position FPC cable which adds good flexibility in using the camera at any angle.

1.2 Package Contents

The final packaged product will contain the following items:

1. MIPI Camera board for Wandboard Quad
2. FPC Cable
3. A Software CD, containing:
    - Full Android source code for MIPI Camera Board support
    - SD Card pre built binary
    - Patch files applied on the code
    - System Reference Manual

Note:
Patches for MIPI camera board support are already applied on the code. RadiumBoards provide these patches to developers for reference purpose only. The source code that we provide along the release was downloaded from the Android repo following the instruction given at http://www.wandboard.org/index.php/downloads . The main remote repository may get updated at any time. As of now the MIPI camera board support is not added in the main repository, therefore we provide no guarantee that these patches can be applied successfully on the source that is being downloaded from the remote repository or any other local source code. Once we add support in the remote Android repository, we would be updating it on our website.

1.3 Getting Started

For testing the MIPI Camera board, onboard HDMI of the Wandboard is used. So a HDMI supported monitor and a HDMI cable is needed. For powering up the Wandboard, use a 5V DC power supply with a minimum current sourcing capability of 3A.
Now follow the instructions below to start using the MIPI Camera board for Wandboard.

- Connect the MIPI Camera board with Wandboard using the FPC cable provided along the package. Ensure the cable is inserted and locked properly.
- Insert the uSD card (containing Android image with MIPI Camera board support) in to Wandboard. There are two uSD slots present in the Wandboard Quad. So ensure the uSD card is inserted in the correct slot. (See Figure 8).
- Power up the board and press the RESET switch on the Wandboard. Wandboard is now booting up. It will take about 3 to 4 minutes for the first time boot. After boot process is completed; you can see the Android Lock Screen.
- Unlock the screen by dragging the lock icon towards right side. For this you need a USB mouse.
- Now you can see the Android Home screen. Click  on applications icon.
- Select  Camera Application.
- Now you are able to see the 720p video.
- Adjust the focus by turning the lens, if required.

2 Features & Specifications

This section covers the specifications of the MIPI Camera Board for Wandboard and provides a high level description of the major components and interfaces that make up the board. Table 1 shows the overall product features.

Camera Sensor	1/6-Inch 720p High-Definition Aptina MT9M114
Video	720p HD at 30 fps
Focus	Manual
Data Interface	MIPI (1 lane)
Power	3.3 V and 5 V from Wandboard through FPC
Power Indicator	Two power LEDs
Connector	33 pin FPC Connector

Table 1: Specifications of MIPI Camera Board for Wandboard

2.1 Key Components

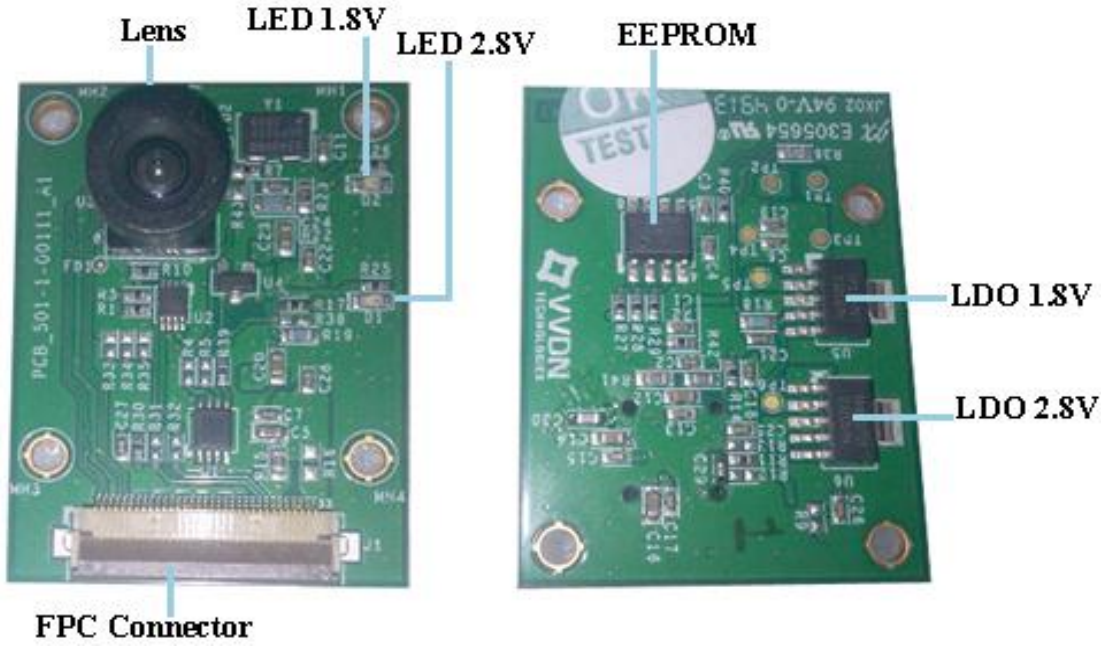


Figure 1: Front and Back of Board showing key compnoents

2.2 Camera Sensor

The MIPI Camera board for Wandboard uses Aptina MT9M114 camera sensor, which is a 1/6-Inch 720p High-Definition (HD) System-On-A-Chip (SOC) Digital Image Sensor.

MT9M114 features an integrated image processor to process the acquired image then transmits the processed data over MIPI interface. The data is transmitted in YUV 4:2:2 format (565RGB, 444RGB and Bayer formats are also supported by the sensor).Following are some feature highlights of MT9M114 camera sensor:

- 720p HD video at 30 fps

- Active-pixel array of 1296H x 976V
- Superior low-light performance
- Ultra-low-power
- Electronic rolling shutter (ERS), progressive scan
- Automatic image correction and enhancement

2.3 Lens

MIPI Camera board for Wandboard uses manual focus lens with good field of view to capture scenes from shorter distances.

2.3.1 Optical Specification

Focus type	Manual
Effective Focal Length (EFL)	1.70 mm
Back Focal Length	2.34 mm
Flange Back Length	1.68 mm
Field of View	135º
Optical Distortion	< -3.5%

Table 2: Optical Specifications

2.4 Connector

MIPI Camera board has a 33 pin FPC connector for connecting it with the Wandboard. FPC cable connection helps the board to be placed at any angles.

2.5 Power Indicators

The MIPI Camera Board for Wandboard features two LEDs to indicate that power rails 1.8V and 2.8V.

2.6 Sensor Board

Sensor board is made up of good quality FR4 material.

2.6.1 Mechanical Specifications

Size	30mm x 40 mm
Layers	4
PCB Thickness	1.6mm
RoHS Compliant	Yes

Table 3: Mechanical Specifications of MIPI Camera board for Wandboard

2.6.2 Electrical Specifications

Specification	Min	Type	Max	Unit
Power				
DC Input Voltage		3.3		V
		5		V
Environmental				
Temperature Range	0		+85º	

Table 4: Electrical Specifications of MIPI Camera board for Wandboard

2.7 Software Interface

Software interface is via Linux-V4L2 layer. MIPI Camera board driver is compatible to V4l2.

2.8 Compatible media

Camera board can be connected with any MIPI interface supported processors.
Note: MIPI Camera board supports only one lane MIPI.

3 System Architecture & Design

This section provides a high level description of the design of the MIPI Camera Board for Wandboard along with its overall architecture.

3.1 System Block Diagram

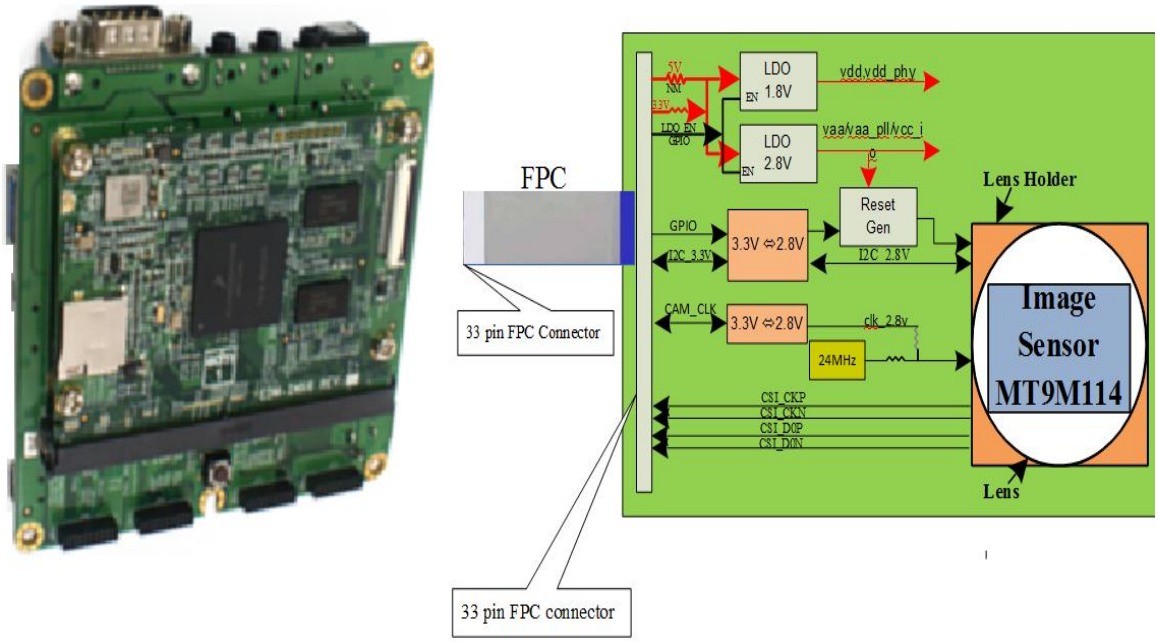


Figure 2: *High Level Block Diagram of HD Camera Board for Wandboard*

3.2 Camera Sensor

The camera sensor has one I2C bus (CAM_I2C_SDA and CAM_I2C_SCL) which is

connected with host processor I2C bus(I2C2_SDA and I2C2_SCL). MIPI data and clock interface between processor and sensor carries the video. Camera sensor is permanently mounted on the board, so chances of disconnection are reduced and at the same time provides good mobility. The sensor receives its master clock (CAM_MCLK) either directly from the Wandboard or from the onboard 24Mhz crystal. To select clock from Wandboard, mount resistor R43, R4 and unmount resistor R7. For selecting 24 Mhz crystal clock source, mount resistor R7 and unmount R43, R4. Camera hardware RESET signal (RESET#) can be generated either from the onboard RESET generator IC (APX809-26SAG) or by configuring the GPIO6 of host processor (Freescale i.MX6Q).

3.3 MIPI Interface

Camera sensor sends frames to the host processor via MIPI. Sensor has 1 lane MIPI data interface (CSI_D0P and CSI_D0M) and MIPI clock interface (CSI_CLK0P and CSI_CLK0M). MIPI clock synchronises the MIPI data. Both MIPI clock and data signals are differential signals.

3.4 Power Supply

The MIPI Camera board for Wandboard generates 1.8V and 2.8V power supplies. Two low-dropout (LDO) voltage regulators TPS73701 are used to regulate VCC_1V8 and VCC_2V8 power rails. The LDO's are ON when their enable pins are pulled HIGH and go to shutdown mode when the enable inputs are LOW. The power leds indicates the state of these LDOs. The LDO enable pin is controlled by GPIO signal (GPIO4).

3.5 Voltage Translator

Voltage translations are required on the Camera board to bring the signals from Wandboard down to the voltage level of camera sensor. These signals include the I2C bus signals (I2C2_SCL and I2C2_SDA), GPIO4(camera RESET#), and GPIO_3_CLKO2.

4 Mechanical Information

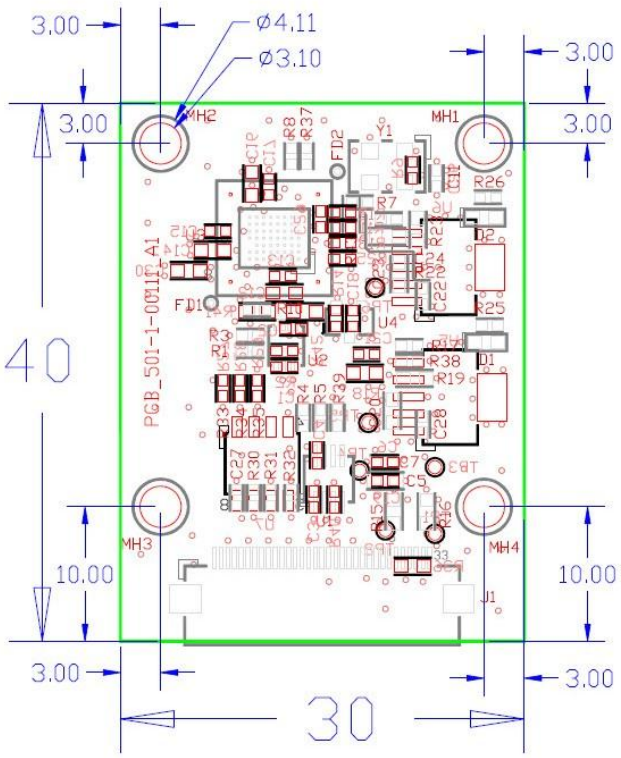


Figure 3: Detailed dimensions of MIPI Camera Board for Wandboard.

Note: All dimensions of Figure 3 are in mm

5 Source build

This section describes how to build Android source code for Wandboard Quad on an Ubuntu 12.04 (64 bit) host machine.
Note: Don't copy-paste the commands in the documents as it may end up in errors

5.1 Installing build tools

Host machine has to be setup for building the source code. Ubuntu has to be updated before proceeding to install the required packages.

```
sudo apt-get update
```

This will update the Ubuntu and after completing update, restart the Ubuntu host machine to apply the updates properly.
Install the required packages for building android from the Ubuntu terminal window.

sudo apt-get install git-core gnupg flex bison \
gperf build-essential zip curl libc6-dev \
libncurses 5-dev x11proto-core-dev \
libx11-dev:i386 libreadline6-dev:i386 \
libgl1-mesa-dev g++-multilib mingw32 \
openjdk-6-jdk tofrodos \
python-markdown libxml2-utils xsltproc \
zlib1g-dev \
ia32-libs u-boot-tools minicom lib32ncurses5-dev \
uuid-dev liblzo2-dev

Android build requires Python (ver 2.6-2.7) and JDK6 and Make (ver 3.81-3.8 2) tool. These packages are installed by default with the Ubuntu 12.04 64bit release. Please note the possible versions of the same. User can check the installed version of **python**, **JDK** (java) and **make** from the terminal. See Figure 4 for commands and its output.

```
[redacted]-Vostro-2520:~$ python --version
Python 2.7.3
[redacted]-Vostro-2520:~$ make --version
GNU Make 3.81
Copyright (C) 2006  Free Software Foundation, Inc.
This is free software; see the source for copying conditions.
There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A
PARTICULAR PURPOSE.

This program built for x86_64-pc-linux-gnu
[redacted]-Vostro-2520:~$ java -version
java version "1.6.0_27"
OpenJDK Runtime Environment (IcedTea6 1.12.6) (6b27-1.12.6-1ubuntu0.12.04.4)
OpenJDK 64-Bit Server VM (build 20.0-b12, mixed mode)
```

Figure 4: Snapshot of commands and its output

5.2 Setting up Android Source Code

Source code is released as 'tar.bz2' along with the DVD. Copy the entire contents of the DVD in to a location in your local HDD. Let this be the release directory(<release_dir>).

1. Make a directory **wandboard** in the home directory and move the source tar.bz2 file to it.
    - mkdir ~/wandboard
    - mv <release_dir>/src/Android/AndroidJB4.2.2DVDv2.tar.bz2

2. Change the working directory to ~/wandboard
    - cd ~/wandboard

3. Extract the source tar file from the terminal. This may take a while.
    - tar -xjf AndroidJB4.2.2DVDv2.tar.bz2

5.3 Compilation steps

Compilation can now be started by executing the following commands

- cd ~/wandboard/myandroid/

- . build/envsetup.sh
- lunch

Here we will get a menu, select 'wandboard-userdebug'

- make

Note: compilation may take 8+ hours

5.4 Compiling kernel alone (optional)

This is an option for compiling the kernel alone, if user wants to do modification in kernel and test it.

- cd ~/wandboard/myandroid/kernel_imx
- export ARCH=arm
- export CROSS_COMPILE=~/wandboard/myandroid/prebuilts/gcc/linux-x86/arm/arm-eabi-4.6/bin/arm-eabi-
- make wandboard_android_defconfig
- make uImage

With a successful build, the generated kernel image is at ~/wandboard/myandroid/kernel_imx/arch/arm/boot/uImage
Note: User can use new uImage by copying it to 'BOOT' partition of the uSD card.

6 Booting Wandboard Quad

Wandboard Quad can be booted from uSD card.

6.1 SD card deployment

Requirements:

- micro SD card of minimum capacity 4GB.
- Gparted (or any other similar application) for partitioning SD card.

6.1.1 Gparted installation in Ubuntu

- sudo apt-get install gparted

6.1.2 uSD card layout for deployment

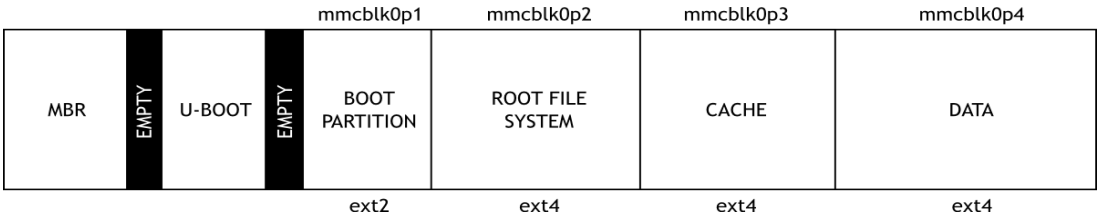


Figure 5: SD-Card Layout for Deployment

Use Gparted to create partition on SD card as shown Figure.5. Use the same partition label, size and file-system as show in Figure.6. DATA partition can vary in size according to the user requirements.

- Insert uSD card into the host machine (PC). Use uSD to USB converter.
- Open gparted
- Select the uSD card device from combo box at top right corner of Gparted window

Note: to open 'Gparted' window: sudo gparted

After creating partitions and exiting gparted, remove the SD card and reinsert to mount it on the file-system

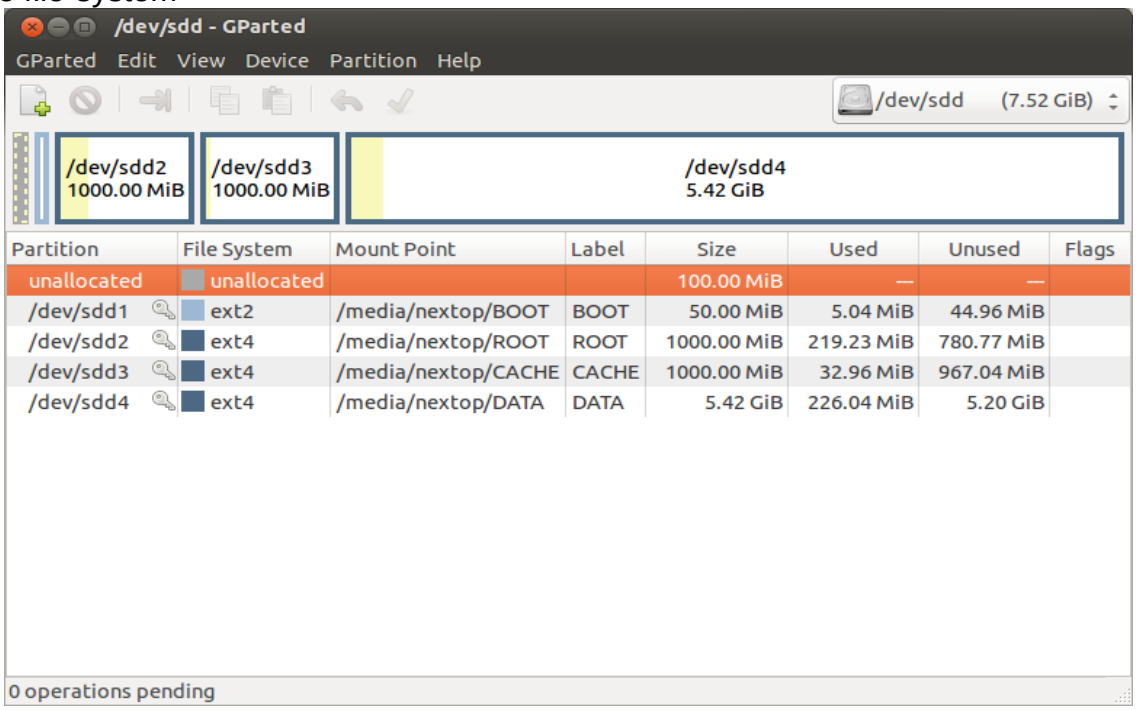


Figure 6: Gparted Partition Editor screen shot

6.1.3 U-Boot installation

Successful compilation of the source code will form U-Boot image "u-boot-6quad.bin" under "**out/target/product/wandboard/ " directory.**
Install u-boot-6quad.bin using the command:

- cd ~/wandboard/myandroid/**out/target/product/wandboard/**
- sudo dd if=u-boot-6quad.bin of=/dev/sdd bs=1k seek=1 skip=1

Note: Change device node (colored in blue) appropriately.

6.1.4Kernel installation

- cd ~/wandboard/myandroid/
- sudo cp kernel_imx/arch/arm/boot/uImage /media/BOOT/
- sync

Note: Change the mount point (colored in blue) appropriately.

6.1.5File-system installation

1. Create a new directory (say "*deploy*").

- cd ~/wandboard/myandroid
- mkdir deploy
- cd deploy

2. Copy the contents of the *root/* folder into it and adjust the permissions.

- cp -a ../out/target/product/wandboard/root/* .
- sudo chown -R 2000.root *
- sudo chmod 644 *.rc
- sudo chmod 644 *.prop

3. Create a tarball with all the correct file stats for the system partition.

- cd ../
- build/tools/mktarball.sh out/host/linux-x86/bin/fs_get_stats out/target/product/wandboard/ system system.tar system.tar.bz2
  Note: Above command has to be entered in one line and there should be a space between "build/….stats" and "out…tar.bz2"

4. Untar the system.tar.bz2 to deploy directory

- sudo tar -xpvf system.tar.bz2 -C deploy/

5. The file deploy/fstab.wandboard should now be adjusted to represent the new partition layout. See Figure.7. for a snapshot of the file contents.
The content of the file *fstab.wandboard* should be:

- /dev/block/mmcblk0p1 /boot ext4 nosuid,nodev,nodiratime,noatime,nomblk_io_submit,noauto_da_alloc, errors=panic wait
- /dev/block/mmcblk0p4 /data ext4 nosuid,nodev,nodiratime,noatime,nomblk_io_submit,noauto_da_alloc,errors=panic wait
- /dev/block/mmcblk0p3 /cache ext4 nosuid,nodev,nomblk_io_submit wait

```
# Android fstab file.
#<src>                                                  <mnt_point>         <type>
<mnt_flags>                                                                 <fs_mgr_flags>
# The filesystem that contains the filesystem checker binary (typically /system) cannot
# specify MF_CHECK, and must come before any filesystems that do specify MF_CHECK

/dev/block/mmcblk0p1    /boot    ext4    nosuid,nodev,nodiratime,noatime,nomblk_io_submit,noauto_da_alloc,errors=panic    wait
/dev/block/mmcblk0p4    /data    ext4    nosuid,nodev,nodiratime,noatime,nomblk_io_submit,noauto_da_alloc,errors=panic    wait
/dev/block/mmcblk0p3    /cache   ext4    nosuid,nodev,nomblk_io_submit   wait
```

Figure 7: Snapshot of fstab.wandboard

6. Adjust file *init.rc* to get read-write access on root partition. Change line 148,

- mount rootfs rootfs / ro remount

to

- mount rootfs rootfs / rw remount

7. Create 'boot' directory inside 'deploy' directory

- cd~/wandboard/myandroid/deploy
- mkdir boot

8. Copy '*deploy*' directory into ROOT partition of uSD card.

- Change the mount point appropriately
- sudo cp -a ~/wandboard/myandroid/deploy/* /media/ROOT/
- sync

6.2Booting from uSD card

1. Insert the uSD card in to the Wandboard.
2. Power on (5V/3A).
3. Press the RESET button on Wandboard.

Note: There are two uSD card slots for the Wandboard Quad. So insert the uSD card in the appropriate slot as shown in the Figure.7.

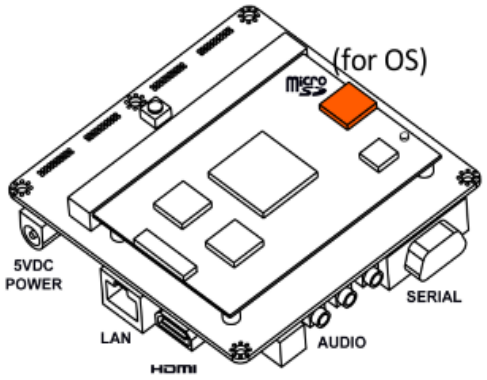


Figure 8: *Wandboard uSD card slot for OS*

7 Testing Video Using Android default camera Application

Follow the “getting started” section (section 1.3 above) to test video

8 Creating Bootable uSD card From uSD Card Image

RadiumBoards provide uSD card image of Android Distribution with MIPI camera Board support for Wandboard. User can find this image in release DVD.

8.1 Using Ubuntu

1. Insert an SD card of minimum capacity 4GB into Ubuntu host machine.
2. Enter the following commands in a terminal window. Assuming the user has copied the release DVD into some directory (<release_dir>) in the local HDD.

- mkdir ~/JB
- cd ~/JB
- sudo cp <release_dir>/bin/image/AndroidJB4.2.2_imagev2.tar.bz2 ./
- tar -xjf AndroidJB4.2.2_imagev2.tar.bz2
- sudo dd if=AndroidJB4.2.2_imagev2.img of=/dev/sdb bs=512
- sync

Note: Change device node (colored in blue) appropriately.