

Setting up Linux on ODROID-X/[Exynos4412](#)

## Setting up Linux on ODROID-X/Exynos4412

---

### Contents

1. [Setting up Linux on ODROID-X/Exynos4412](#)
  1. [Serial Port Setup](#)
    1. [Connecting the Serial Cable](#)
    2. [Minicom Configuration](#)
  2. [Cross Compiler Setup](#)
  3. [Building the System](#)
    1. [Getting the Sources](#)
    2. [Compiling U-Boot](#)
    3. [U-Boot Fusing](#)
    4. [Creating a RootFS](#)
    5. [Compiling the Kernel](#)
    6. [Configuring U-Boot](#)
    7. [Configuring Debian](#)

## Serial Port Setup

---

### ■ Connecting the Serial Cable

The Serial-to-USB cable that came with the ODROID-X is powered by the USB port so do the following:

1. Plug the USB side of the dongle to your host machine
2. Plug the serial connector onto the ODROID-X's UART connector

At this point, the ODROID-X I was playing with was still not powered but the USB-serial was up and running on my host machine under `/dev/ttyUSB0`.

### ■ Minicom Configuration

Below are the parameters to set in Minicom. Obviously other serial programs can be happily used as well.

```
Serial Port : /dev/ttyUSB0
Baud Rate  : 115200
Parity     : 8bit
Hardware Flow Control : No
Stop bit   : 1bit
```

You can configure the settings in Minicom by using the following:

Toggle line numbers

```
$ minicom -s
```

## Cross Compiler Setup

---

I downloaded the CodeSourcery Lite cross compiler from the website below:

<https://sourcery.mentor.com/sgpp/lite/arm/portal/release1294>

You have to give your name, email and location and they will mail you a link. Sheesh.

I downloaded the "IA32 GNU/Linux TAR" since installers tend to suck. I extracted it over into my /opt directory:

Toggle line numbers

```
$ cd /opt
$ tar -xvf /home/mvanga/arm-2010q1-188.tar.bz2
```

The kernel and U-Boot cross compilation scripts make use of the CROSS\_COMPILE variable as a prefix to the tools in order to call cross-compilers properly.

Normally you add /location/you/extracted/tools/to/bin to the PATH variable and during compile give the CROSS\_COMPILE environment variable as the tool name prefix (eg. "arm-none-linux-gnueabi-").

I just set up two aliases in my .bashrc to switch between cross and host toolchains. Much easier for me this way:

Toggle line numbers

```
alias setcc='export CROSS_COMPILE=/opt/arm-2010q1/bin/arm-none-eabi-'
alias unsetcc='export CROSS_COMPILE='
```

## Building the System

---

### Getting the Sources

The company behind the ODROID-X provides a board support package for the ODROID-X that contains all the source code you need (including U-boot and the Linux kernel sources). You can grab that from here:

[http://com.odroid.com/sigong/nf\\_file\\_board/nfile\\_board\\_view.php?keyword=&tag=&bid=112](http://com.odroid.com/sigong/nf_file_board/nfile_board_view.php?keyword=&tag=&bid=112)

You need to enter a CAPTCHA to download the thing and it's 2GB. Let's just say they don't have the fastest servers (mine took 6 hours to download).

The board support package provides the U-Boot sources, the kernel sources as well as Android sources for the board.

### Compiling U-Boot

U-Boot is the bootloader we'll use to boot the kernel. To compile it, untar it from the board support

package archive and navigate to the source directory. Before we compile, we must make sure to set our cross compiler by calling the alias we defined in our `.bashrc` file:

Toggle line numbers

```
$ cd uboot
$ setcc                # set the cross compiler here
$ make mrproper        # clean the directory, just in case
$ make smdk4412_config # this config works with the ODROID-X
$ make -j8
```

NOTE: DO NOT DO A 'make distclean'. This will remove the proprietary bootloader files. You will have to extract it again in this case.

The config file used is that of the Samsung Development Kit for the Exynos 4412 Quad-core processor (the one on the ODROID-X).

## U-Boot Fusing

The boot process on the Exynos 4412 (and other Exynos processors from what I've read) is a bit contrived. The board boots in the following sequence:

1. Proprietary boot loader from first 8KB of SD card
2. U-Boot stage 1 bootloader (we must generate this)
3. Proprietary ARM security extension (TrustZone) initialization
4. U-Boot main loader

The above need to be fused into a binary blob that go at the start of the SD card immediately after the MBR/Partition table. The board support package provides number 1 and 3.

The Hardkernel guys have also provided a script that generates this stuff automatically in the `uboot` folder:

Toggle line numbers

```
$ cd sd_fuse
$ ./sd_fuse_4412.sh /dev/sdX
```

Where `sdX` is the SD card drive. Obligatory: DO NOT ERASE YOUR MAIN DISK (You need to be root for this)

We should now have a bootable SD card with U-Boot on it! Stick it into the SD card slot on the board and plug in the power.

GOTCHA: There is a power button on the board that needs to be held for 2-3 seconds to actually boot up the board. Simply plugging in the power does not start the boot process. The same applies for reset; the reset button needs to be held for a few seconds to reboot the board.

Here is the log from the first boot of the board:

OK

U-Boot 2010.12 (Sep 22 2012 - 22:57:04) for ODROID4412

```

CPU: S5PC220 [Samsung SOC on SMP Platform Base on ARM CortexA9]
APLL = 1000MHz, MPLL = 800MHz
DRAM: 1023 MiB

PMIC VERSION : 0x12, CHIP REV : 1

BL1 version: N/A (TrustZone Enabled BSP)

Checking Boot Mode ... SDMMC
REVISION: 1.1
MMC Device 0: 7695 MB
[ERROR] response timeout error : 00000104 cmd 1
FAIL: waiting for status update.
MMC Device 1: 0 MB
MMC Device 2 not found
*** Warning - using default environment

ModeKey Check... run normal_boot
Net: No ethernet found.
Hit any key to stop autoboot: 0
ODR0ID4412 #

```

At this point, we can do whatever we want with the SD card partitions and setting up the kernel. I decided to try it the way recommended by the Hardkernel guys.

So now the U-Boot install can set up the remaining partitions on the SD card (the `sd_fuse` script simply sticks a bootloader on the card and screws up the partition info). To do this run the following at the U-Boot prompt:

```

ODR0ID4412 # fdisk -c 0
fdisk is completed

partition #    size(MB)    block start #    block count    partition_Id
1             5508       4336673         11281523       0x0C
2             896        138897          1836527        0x83
3            1024       1975424         2098888        0x83
4             128        4074312         262361         0x83
ODR0ID4412 #

```

Next, we must format the first partition created above (type 0x0c is Windows FAT type):

```

ODR0ID4412 # fatformat mmc 0:1
Start format MMC&d partition&d ...
Partition1: Start Address(0x422c21), Size(0xac2473)
size checking ...
Under 8G
write FAT info: 32
Fat size : 0x2b09
Erase FAT
region.....
.....
Partition1 format complete.
ODR0ID4412 #

```

I formatted the remaining three partitions as EXT4 partitions on my host PC:

Toggle line numbers

```
$ mkfs.ext4 /dev/sdb2
$ mkfs.ext4 /dev/sdb3
$ mkfs.ext4 /dev/sdb4
```

Mount /dev/sdb3 (randomly selected here):

Toggle line numbers

```
$ mount /dev/sdb3 /media
```

## ■ Creating a RootFS

To create a base Debian root filesystem, I did the following:

Toggle line numbers

```
$ mkdir ~/deb
$ cd ~/deb
$ debootstrap --verbose --arch armhf --foreign wheezy ./rootfs
http://ftp.de.debian.org/debian
```

The architecture is armhf (for ARM processing with hardware floating point unit). The output directory is the SD card.

The suite must be set to 'wheezy' as it is the only one which supports the armhf architecture. If you want an older version, change the armhf to armel in the multistrap command.

To preserve file permissions when copying to the SD card, we can tar and untar the whole thing:

Toggle line numbers

```
$ cd rootfs
$ tar -cvzf ../rootfs.tgz *
$ cd /media/
$ tar -xvf ~/deb/rootfs.tgz
```

Remember to sync the filesystem to make sure everything is copied:

Toggle line numbers

```
$ sync
$ umount /media
```

## ■ Compiling the Kernel

Once again, we use the files from the board support package to get our kernel up and running. Untar the kernel.tar.gz file and proceed:

Toggle line numbers

```
$ tar xvf kernel.tar.gz
$ cd kernel_4412
$ setcc                # Set cross compiler here
$ make ARCH=arm odroidx_defconfig
```

Now before we compile, I ran into one issue. The DRM driver expects a full blown toolchain, but we're using a bare-metal toolchain which doesn't define the symbol '`linux`'. In its absence, the toolchain assumes a FreeBSD host OS. BAD! Here is a quick fix to solve this:

On line 39 of `include/drm/drm.h` change:

```
#if defined(linux)
```

to the following:

```
#if defined(KERNEL) || defined(linux)
```

Since I'm using the defconfig provided by Hardkernel, I'm not messing with the config just yet. In the future, however, we can just get rid of the DRM driver since we have no display anyway.

Anyway, now just compile:

Toggle line numbers

```
$ make ARCH=arm -j8
```

To make the U-Boot image, first export the tools directory within the U-boot source folder into the PATH variable:

Toggle line numbers

```
$ export PATH=$PATH:/home/mvanga/uboot_4412/uboot/tools/
$ make ARCH=arm zImage
```

Now just write the zImage to a specific offset on the SD card.

Toggle line numbers

```
$ dd iflag=dsync oflag=dsync if=./arch/arm/boot/zImage of=/dev/sdX
seek=1057
```

Apparently the bootcmd in U-Boot loads the kernel from an offset of 1057 on the SD card. This is actually quite weird and in the future, we will just load it from the filesystem (eg. `/boot`)

## ■ Configuring U-Boot

So now if we boot up the board, we should have a nice Linux kernel booting up and...kernel panicking. This is because we have to specify where it can find the rootfs.

Boot into U-Boot and set the 'bootargs' variable, which is the one used to pass command line options to the kernel during boot. At the U-Boot console, type the following:

```
ODROID4412 # setenv bootargs root=/dev/mmcblk0p3 console=tty1
console=ttySAC1,115200n8 rootwait rw init=/bin/sh noinitrd
ODROID4412 # saveenv
```

```
Saving Environment to SMDK bootable device...  
done  
ODROID4412 #
```

Above, I set the init to the bash shell as we still need to configure Debian. Since I used the 3rd partition on the SD card, I used mmcblk0p3 above. The tty console must be set correctly to ttySAC1 or you will get no output after the inittab run which sets them up.

## ■ Configuring Debian

So simply boot into the root shell we specified with our init argument and run the second stage of the debootstrap command:

Toggle line numbers

```
$ export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:  
/bin  
$ mount -t proc nodev /proc  
$ /debootstrap/debootstrap --second-stage
```

If it complains, there is a detailed log in "/debootstrap/debootstrap.log". I, for example, had to manually remove /var/run folder due to unclean shutdowns.

Also remember to set the root password:

Toggle line numbers

```
$ passwd root
```

And we're done! Reboot and login 

Below is the complete boot log for the ODROID-X board I used:

```
OK
```

```
U-Boot 2010.12 (Sep 22 2012 - 22:57:04) for ODROID4412
```

```
CPU: S5PC220 [Samsung SOC on SMP Platform Base on ARM CortexA9]  
APLL = 1000MHz, MPLL = 800MHz  
DRAM: 1023 MiB
```

```
PMIC VERSION : 0x12, CHIP REV : 1
```

```
BL1 version: N/A (TrustZone Enabled BSP)
```

```
Checking Boot Mode ... SDMMC
```

```
REVISION: 1.1
```

```
MMC Device 0: 7695 MB
```

```
[ERROR] response timeout error : 00000104 cmd 1
```

```
FAIL: waiting for status update.
```

```
MMC Device 1: 0 MB
```

```
MMC Device 2 not found
```

```
ModeKey Check... run normal_boot
Net:  No ethernet found.
Hit any key to stop autoboot:  0
reading kernel..device 0 Start 1057, Count 16384
MMC read: dev # 0, block # 1057, count 16384 ... 16384 blocks read: OK
completed
reading RFS..device 0 Count 17441, Start 2048
MMC read: dev # 0, block # 17441, count 2048 ... 2048 blocks read: OK
completed
Boot with zImage
Wrong Ramdisk Image Format
[err] boot_get_ramdisk

Starting kernel ...

Uncompressing Linux... done, booting the kernel.
[    0.000000] Booting Linux on physical CPU 0
[    0.000000] Linux version 3.6.0-rc6 (mvanga@wks-40-13) (gcc version
4.4.1 (Sourcery G++ Lite 2010q1-188) ) #1 SMP PREEMPT Sun Sep 23
00:12:53 CEST 2012
[    0.000000] CPU: ARMv7 Processor [413fc090] revision 0 (ARMv7),
cr=10c5387d
[    0.000000] CPU: PIPT / VIPT nonaliasing data cache, VIPT aliasing
instruction cache
[    0.000000] Machine: ODROID-X
[    0.000000] cma: CMA: reserved 16 MiB at 6e800000
[    0.000000] Memory policy: ECC disabled, Data cache writealloc
[    0.000000] CPU EXYNOS4412 (id 0xe4412211)
[    0.000000] S3C24XX Clocks, Copyright 2004 Simtec Electronics
[    0.000000] s3c_register_clksrc: clock armclk has no registers set
[    0.000000] s3c_register_clksrc: clock audiocdclk has no registers
set
[    0.000000] audiocdclk: no parent clock specified
[    0.000000] EXYNOS4: PLL settings, A=1000000000, M=800000000,
E=96000000 V=108000000
[    0.000000] EXYNOS4: ARMCLK=1000000000, DMC=400000000,
ACLK200=160000000
[    0.000000] ACLK100=100000000, ACLK160=160000000, ACLK133=133333333
[    0.000000] sclk_pwm: source is ext_xtal (0), rate is 24000000
[    0.000000] sclk_csis: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_csis: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_cam0: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_cam1: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_fimc: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_fimc: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_fimc: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_fimc: source is xusbxti (1), rate is 1500000
[    0.000000] sclk_fimd: source is mout_mpll_user (6), rate is 50000000
[    0.000000] sclk_mfc: source is mout_mfc0 (0), rate is 50000000
[    0.000000] PERCPU: Embedded 7 pages/cpu @c0dee000 s7744 r8192 d12736
u32768
[    0.000000] Built 1 zonelists in Zone order, mobility grouping on.
Total pages: 255746
[    0.000000] Kernel command line: root=/dev/mmcbk0p3 rootfstype=ext4
console=tty1 console=ttySAC1,115200n8 rootwait rw init=/sbin/init
noinitrd
```



```
[ 0.000000] PID hash table entries: 4096 (order: 2, 16384 bytes)
[ 0.000000] Dentry cache hash table entries: 131072 (order: 7, 524288
bytes)
[ 0.000000] Inode-cache hash table entries: 65536 (order: 6, 262144
bytes)
[ 0.000000] Memory: 48MB 216MB 743MB = 1007MB total
[ 0.000000] Memory: 999524k/1015908k available, 31644k reserved,
269312K highmem
[ 0.000000] Virtual kernel memory layout:
[ 0.000000]     vector : 0xffff0000 - 0xffff1000   (   4 kB)
[ 0.000000]     fixmap : 0xffff0000 - 0xffffe000   ( 896 kB)
[ 0.000000]     vmalloc : 0xf0000000 - 0xff000000   (240 MB)
[ 0.000000]     lowmem  : 0xc0000000 - 0xef800000   ( 760 MB)
[ 0.000000]     pkmap   : 0xbfe00000 - 0xc0000000   (   2 MB)
[ 0.000000]     modules : 0xbf000000 - 0xbfe00000   (  14 MB)
[ 0.000000]     .text   : 0xc0008000 - 0xc052c448   (5266 kB)
[ 0.000000]     .init   : 0xc052d000 - 0xc0556e40   ( 168 kB)
[ 0.000000]     .data   : 0xc0558000 - 0xc05a8ec0   (  324 kB)
[ 0.000000]     .bss   : 0xc05a8ee4 - 0xc05e1c20   (  228 kB)
[ 0.000000] SLUB: Genslabs=11, HWalign=64, Order=0-3, MinObjects=0,
CPUs=4, Nodes=1
[ 0.000000] Preemptible hierarchical RCU implementation.
[ 0.000000] NR_IRQS:549
[ 0.000000] sched_clock: 32 bits at 200 Hz, resolution 5000000ns,
wraps every 4294967291ms
[ 0.000000] Console: colour dummy device 80x30
[ 0.000000] console [tty1] enabled
[ 0.045000] Calibrating delay loop... 1992.29 BogoMIPS (lpj=4980736)
[ 0.045000] pid_max: default: 32768 minimum: 301
[ 0.045000] Mount-cache hash table entries: 512
[ 0.045000] CPU: Testing write buffer coherency: ok
[ 0.045000] CPU0: thread -1, cpu 0, socket 10, mpidr 80000a00
[ 0.045000] hw perfevents: enabled with ARMv7 Cortex-A9 PMU driver, 7
counters available
[ 0.045000] Setting up static identity map for 0x403f4c58 -
0x403f4cb0
[ 0.045000] L310 cache controller enabled
[ 0.045000] l2x0: 16 ways, CACHE_ID 0x4100c4c8, AUX_CTRL 0x7e470001,
Cache size: 1048576 B
[ 0.070000] CPU1: Booted secondary processor
[ 0.090000] CPU1: thread -1, cpu 1, socket 10, mpidr 80000a01
[ 0.090000] CPU1: Unknown IPI message 0x1
[ 0.100000] CPU2: Booted secondary processor
[ 0.120000] CPU2: thread -1, cpu 2, socket 10, mpidr 80000a02
[ 0.120000] CPU2: Unknown IPI message 0x1
[ 0.130000] CPU3: Booted secondary processor
[ 0.150000] CPU3: thread -1, cpu 3, socket 10, mpidr 80000a03
[ 0.150000] CPU3: Unknown IPI message 0x1
[ 0.150000] Brought up 4 CPUs
[ 0.150000] SMP: Total of 4 processors activated (7969.17 BogoMIPS).
[ 0.150000] devtmpfs: initialized
[ 0.155000] dummy:
[ 0.155000] NET: Registered protocol family 16
[ 0.160000] DMA: preallocated 256 KiB pool for atomic coherent
allocations
[ 0.170000] hw-breakpoint: found 5 (+1 reserved) breakpoint and 1
```

```
watchpoint registers.
[ 0.170000] hw-breakpoint: maximum watchpoint size is 4 bytes.
[ 0.170000] S3C Power Management, Copyright 2004 Simtec Electronics
[ 0.170000] EXYNOS4x12 PMU Initialize
[ 0.170000] EXYNOS: Initializing architecture
[ 0.180000] bio: create slab <bio-0> at 0
[ 0.180000] hdmi_5v: 5000 mV
[ 0.180000] usbcore: registered new interface driver usbfs
[ 0.180000] usbcore: registered new interface driver hub
[ 0.180000] usbcore: registered new device driver usb
[ 0.180000] s3c-i2c s3c2440-i2c.0: slave address 0x10
[ 0.180000] s3c-i2c s3c2440-i2c.0: bus frequency set to 97 KHz
[ 0.180000] max77686 0-0009: device found
[ 0.180000] max77686 0-0009: irq is not specified
[ 0.185000] VDD_ALIVE_1.0V: 1000 mV
[ 0.185000] VDDQ_M1M2_1.8V: 1800 mV
[ 0.190000] VDDQ_M0_1.8V: 1800 mV
[ 0.190000] VDDQ_MMC2_2.8V: 2800 mV
[ 0.195000] VDDQ_MMC13_1V8: 1800 mV
[ 0.200000] VDD_MPLL_1.0V: 1000 mV
[ 0.200000] VDD_VPLL_1.0V: 1000 mV
[ 0.205000] VDD10_HDMI_1.0V: 1000 mV
[ 0.205000] VDD_VTCORE_1.0V: 1000 mV
[ 0.210000] VDDQ_MIPIHSI_1.8V: 1800 mV
[ 0.210000] VDD18_ABB1_1.8V: 1800 mV
[ 0.215000] VDD33_UOTG_3.3V: 3300 mV
[ 0.215000] VDDQ_C2C_1.8V: 1800 mV
[ 0.220000] VDD18_ABB2_1.8V: 1800 mV
[ 0.225000] VDD10_HSIC_1.0V: 1000 mV
[ 0.225000] VDD18_HSIC_1.8V: 1800 mV
[ 0.230000] VDDQ_CAM_1.8V: 1800 mV
[ 0.230000] VDD_LD018_1.8V: 1800 mV
[ 0.230000] VDD_VTCAM_1.8V: 1800 mV
[ 0.235000] VDD_LD020_1V8: 1800 mV
[ 0.235000] VDD_SDCARD_3.3V: 3300 mV
[ 0.240000] VDD_LD022_2.8V: 2800 mV
[ 0.240000] VDD_TOUCH_2.8V: 2800 mV
[ 0.245000] VDD_TOUCHLED_3.3V: 3300 mV
[ 0.245000] VDDQ_LCD_1.8V: 1800 mV
[ 0.250000] VDD_MOTOR_3.0V: 3000 mV
[ 0.250000] VDD_MIF_1.0V: 800 <--> 1050 mV at 1100 mV
[ 0.250000] VDD_ARM_1.3V: 800 <--> 1350 mV at 1000 mV
[ 0.255000] VDD_INT_1.0V: 800 <--> 1150 mV at 1000 mV
[ 0.255000] VDD_G3D_1.0V: 850 <--> 1200 mV at 1000 mV
[ 0.260000] VDDQ_CKEM1M2_1.2V: 1200 mV
[ 0.260000] VDD_INL_1.35V: 1350 mV
[ 0.265000] VDD_INL_2.0V: 2000 mV
[ 0.265000] VDD_BUCK8_2.8V: 2800 mV
[ 0.270000] VDD_BUCK9_1.2V: 1200 mV
[ 0.270000] s3c-i2c s3c2440-i2c.0: i2c-0: S3C I2C adapter
[ 0.270000] s3c-i2c s3c2440-i2c.1: slave address 0x10
[ 0.270000] s3c-i2c s3c2440-i2c.1: bus frequency set to 97 KHz
[ 0.270000] s3c-i2c s3c2440-i2c.1: i2c-1: S3C I2C adapter
[ 0.270000] s3c-i2c s3c2440-i2c.3: slave address 0x10
[ 0.270000] s3c-i2c s3c2440-i2c.3: bus frequency set to 97 KHz
[ 0.270000] s3c-i2c s3c2440-i2c.3: i2c-3: S3C I2C adapter
```

```
[ 0.270000] s3c-i2c s3c2440-i2c.7: slave address 0x10
[ 0.270000] s3c-i2c s3c2440-i2c.7: bus frequency set to 97 KHz
[ 0.270000] s3c-i2c s3c2440-i2c.7: i2c-7: S3C I2C adapter
[ 0.270000] s3c-i2c s3c2440-hdmiphy-i2c: slave address 0x10
[ 0.270000] s3c-i2c s3c2440-hdmiphy-i2c: bus frequency set to 97 KHz
[ 0.270000] s3c-i2c s3c2440-hdmiphy-i2c: i2c-8: S3C I2C adapter
[ 0.270000] Linux video capture interface: v2.00
[ 0.270000] Advanced Linux Sound Architecture Driver Version 1.0.25.
[ 0.270000] cfg80211: Calling CRDA to update world regulatory domain
[ 0.270000] Switching to clocksource mct-frc
[ 0.280000] NET: Registered protocol family 2
[ 0.280000] TCP established hash table entries: 131072 (order: 8,
1048576 bytes)
[ 0.280000] TCP bind hash table entries: 65536 (order: 8, 1310720
bytes)
[ 0.280000] TCP: Hash tables configured (established 131072 bind
65536)
[ 0.280000] TCP: reno registered
[ 0.280000] UDP hash table entries: 512 (order: 2, 24576 bytes)
[ 0.280000] UDP-Lite hash table entries: 512 (order: 2, 24576 bytes)
[ 0.285000] NET: Registered protocol family 1
[ 0.285000] bounce pool size: 64 pages
[ 0.295000] msgmni has been set to 1458
[ 0.295000] io scheduler noop registered
[ 0.295000] io scheduler deadline registered
[ 0.295000] io scheduler cfq registered (default)
[ 0.295000] start plist test
[ 0.300000] end plist test
[ 0.300000] s3c-fb exynos4-fb.0: start latency exceeded, new value
875 ns
[ 0.300000] s3c-fb exynos4-fb.0: state restore latency exceeded, new
value 35375 ns
[ 0.320000] Console: switching to colour frame buffer device 170x48
[ 0.335000] s3c-fb exynos4-fb.0: window 0: fb
[ 0.340000] dma-pl330 dma-pl330.0: Loaded driver for PL330
DMAC-267056
[ 0.340000] dma-pl330 dma-pl330.0:  DBUFF-32x4bytes Num_Chans-8
Num_Peri-32 Num_Events-32
[ 0.345000] dma-pl330 dma-pl330.1: Loaded driver for PL330
DMAC-267056
[ 0.345000] dma-pl330 dma-pl330.1:  DBUFF-32x4bytes Num_Chans-8
Num_Peri-32 Num_Events-32
[ 0.345000] dma-pl330 dma-pl330.2: Loaded driver for PL330
DMAC-267056
[ 0.345000] dma-pl330 dma-pl330.2:  DBUFF-64x8bytes Num_Chans-8
Num_Peri-1 Num_Events-32
[ 0.420000] Serial: 8250/16550 driver, 4 ports, IRQ sharing disabled
[ 0.420000] exynos4210-uart.0: ttySAC0 at MMIO 0x13800000 (irq = 84)
is a S3C6400/10
[ 0.420000] exynos4210-uart.1: ttySAC1 at MMIO 0x13810000 (irq = 85)
is a S3C6400/10
[ 1.300000] console [ttySAC1] enabled
[ 1.305000] exynos4210-uart.2: ttySAC2 at MMIO 0x13820000 (irq = 86)
is a S3C6400/10
[ 1.315000] exynos4210-uart.3: ttySAC3 at MMIO 0x13830000 (irq = 87)
is a S3C6400/10
```

```
[ 1.330000] brd: module loaded
[ 1.340000] loop: module loaded
[ 1.340000] ehci_hcd: USB 2.0 'Enhanced' Host Controller (EHCI)
Driver
[ 1.345000] s5p-ehci s5p-ehci: S5P EHCI Host Controller
[ 1.350000] s5p-ehci s5p-ehci: new USB bus registered, assigned bus
number 1
[ 1.360000] s5p-ehci s5p-ehci: irq 102, io mem 0x12580000
[ 1.375000] s5p-ehci s5p-ehci: USB 2.0 started, EHCI 1.00
[ 1.375000] usb usb1: New USB device found, idVendor=1d6b,
idProduct=0002
[ 1.385000] usb usb1: New USB device strings: Mfr=3, Product=2,
SerialNumber=1
[ 1.390000] usb usb1: Product: S5P EHCI Host Controller
[ 1.395000] usb usb1: Manufacturer: Linux 3.6.0-rc6 ehci_hcd
[ 1.400000] usb usb1: SerialNumber: s5p-ehci
[ 1.405000] hub 1-0:1.0: USB hub found
[ 1.410000] hub 1-0:1.0: 3 ports detected
[ 1.415000] ohci_hcd: USB 1.1 'Open' Host Controller (OHCI) Driver
[ 1.420000] exynos-ohci exynos-ohci: Already power on PHY
[ 1.425000] exynos-ohci exynos-ohci: EXYNOS OHCI Host Controller
[ 1.430000] exynos-ohci exynos-ohci: new USB bus registered, assigned
bus number 2
[ 1.435000] exynos-ohci exynos-ohci: irq 102, io mem 0x12590000
[ 1.500000] usb usb2: New USB device found, idVendor=1d6b,
idProduct=0001
[ 1.505000] usb usb2: New USB device strings: Mfr=3, Product=2,
SerialNumber=1
[ 1.515000] usb usb2: Product: EXYNOS OHCI Host Controller
[ 1.520000] usb usb2: Manufacturer: Linux 3.6.0-rc6 ohci_hcd
[ 1.525000] usb usb2: SerialNumber: exynos-ohci
[ 1.530000] hub 2-0:1.0: USB hub found
[ 1.535000] hub 2-0:1.0: 3 ports detected
[ 1.535000] usb3503: USB HUB driver init
[ 1.540000] usb3503: usb3503_probe:375
[ 1.545000] usb3503: usb3503_set_mode: mode = 1
[ 1.655000] usb3503: reg_write: write E7, req:03, val:03
[ 1.655000] usb3503: usb3503_register_read: reg: a read failed
[ 1.660000] usb3503: reg_update: update 0A, req:0c, val:ee
[ 1.665000] usb3503: PDS update fail err = -6
[ 1.670000] usb3503: usb3503_probe: probed on hub mode
[ 1.675000] i2c-core: driver [usb3503] using legacy suspend method
[ 1.685000] i2c-core: driver [usb3503] using legacy resume method
[ 1.690000] s3c-rtc s3c64xx-rtc: rtc disabled, re-enabling
[ 1.695000] s3c-rtc s3c64xx-rtc: rtc core: registered s3c as rtc0
[ 1.700000] s5p-hdmiphy 8-0038: probe successful
[ 1.715000] s5p-hdmi exynos4-hdmi: probe successful
[ 1.720000] Samsung TV Mixer driver, (c) 2010-2011 Samsung
Electronics Co., Ltd.
[ 1.725000] s5p-mixer s5p-mixer: probe start
[ 1.730000] s5p-mixer s5p-mixer: resources acquired
[ 1.735000] s5p-mixer s5p-mixer: added output 'S5P HDMI connector'
from module 's5p-hdmi'
[ 1.740000] s5p-mixer s5p-mixer: module s5p-sdo is missing
[ 1.750000] s5p-mixer s5p-mixer: registered layer graph0
as /dev/video0
```

```
[ 1.755000] fb1: registered frame buffer emulation for /dev/video0
[ 1.760000] s5p-mixer s5p-mixer: registered layer graph1
as /dev/video1
[ 1.765000] fb2: registered frame buffer emulation for /dev/video1
[ 1.775000] s5p-mixer s5p-mixer: registered layer video0
as /dev/video2
[ 1.780000] fb3: registered frame buffer emulation for /dev/video2
[ 1.785000] s5p-mixer s5p-mixer: probe successful
[ 1.790000] cpuidle: using governor ladder
[ 1.795000] sdhci: Secure Digital Host Controller Interface driver
[ 1.800000] sdhci: Copyright(c) Pierre Ossman
[ 1.805000] s3c-sdhci exynos4-sdhci.2: clock source 2: mmc_busclk.2
(500000000 Hz)
[ 1.815000] mmc0: no vmmc regulator found
[ 1.845000] mmc0: SDHCI controller on samsung-hsmmc [exynos4-sdhci.2]
using ADMA
[ 1.850000] usbcore: registered new interface driver usbhid
[ 1.855000] usbhid: USB HID core driver
[ 1.875000] mmc0: new high speed SDHC card at address b368
[ 1.880000] mmcblk0: mmc0:b368 00000 7.51 GiB
[ 1.880000] max98090 1-0010: revision 0x42
[ 1.885000] usb 1-2: new high-speed USB device number 2 using
s5p-ehci
[ 1.885000] [MAX98090] max98090_set_record_main_mic(150)
[ 1.900000] mmcblk0: p1 p2 p3 p4
[ 1.905000] hkdk-snd-max89090 hkdk-snd-max89090: max98090-aif1 <->
samsung-i2s.0 mapping ok
[ 1.915000] hkdk-snd-max89090 hkdk-snd-max89090: max98090-aif1 <->
samsung-i2s.0 mapping ok
[ 1.925000] TCP: cubic registered
[ 1.925000] NET: Registered protocol family 17
[ 1.930000] NET: Registered protocol family 15
[ 1.935000] NET: Registered protocol family 35
[ 1.940000] VFP support v0.3: implementor 41 architecture 3 part 30
variant 9 rev 4
[ 1.945000] Registering SWP/SWPB emulation handler
[ 1.950000] VDD_G3D_1.0V: incomplete constraints, leaving on
[ 1.955000] VDDQ_LCD_1.8V: incomplete constraints, leaving on
[ 1.965000] s3c-rtc s3c64xx-rtc: setting system clock to 2000-01-01
07:42:32 UTC (946712552)
[ 1.975000] ALSA device list:
[ 1.975000] #0: hkdk-max98090
[ 2.015000] usb 1-2: New USB device found, idVendor=0424,
idProduct=3503
[ 2.015000] usb 1-2: New USB device strings: Mfr=0, Product=0,
SerialNumber=0
[ 2.025000] hub 1-2:1.0: USB hub found
[ 2.030000] hub 1-2:1.0: 3 ports detected
[ 2.315000] usb 1-2.1: new high-speed USB device number 3 using
s5p-ehci
[ 2.365000] EXT4-fs (mmcblk0p3): mounted filesystem with ordered data
mode. Opts: (null)
[ 2.370000] VFS: Mounted root (ext4 filesystem) on device 179:3.
[ 2.385000] devtmpfs: mounted
[ 2.385000] Freeing init memory: 164K
[ 2.420000] usb 1-2.1: New USB device found, idVendor=0424,
```

```
idProduct=9514
[ 2.425000] usb 1-2.1: New USB device strings: Mfr=0, Product=0,
SerialNumber=0
[ 2.430000] hub 1-2.1:1.0: USB hub found
[ 2.435000] hub 1-2.1:1.0: 5 ports detected
INIT: version 2.88 booting
[ 2.730000] usb 1-2.1.1: new high-speed USB device number 4 using
s5p-ehci
[info] Using makefile-style concurrent boot in runlevel S.
[ 2.835000] usb 1-2.1.1: New USB device found, idVendor=0424,
idProduct=ec00
[ 2.840000] usb 1-2.1.1: New USB device strings: Mfr=0, Product=0,
SerialNumber=0
[....] Starting the hotplug events dispatcher: udevd[ 3.145000]
udev[1301]: starting version 175
. ok
[....] Synthesizing the initial hotplug events...[ 3.395000]
s5p-mixer s5p-mixer: start latency exceeded, new value 667 ns
[ 3.400000] s5p-mixer s5p-mixer: state restore latency exceeded, new
value 40250 ns
[ 3.410000] s5p-mixer s5p-mixer: stop latency exceeded, new value
1542 ns
[ 3.415000] s5p-mixer s5p-mixer: start latency exceeded, new value
875 ns
[ 3.420000] s5p-mixer s5p-mixer: state save latency exceeded, new
value 17083 ns
done.
[....] Waiting for /dev to be fully populated...done.
[....] Activating swap...done.
[ 4.810000] EXT4-fs (mmcblk0p3): re-mounted. Opts: (null)
[....] Checking root file system...fsck from util-linux 2.20.1
/dev/mmcblk0p3: clean, 12001/65664 files, 74934/262144 blocks
done.
[ 5.000000] EXT4-fs (mmcblk0p3): re-mounted. Opts: (null)
[....] Creating compatibility symlink from /etc/mtab
to /proc/mounts. ... (warning).
[....] Cleaning up temporary files... /tmp. ok
[....] Activating lvm and md swap...done.
[....] Checking file systems...fsck from util-linux 2.20.1
done.
[....] Mounting local filesystems...done.
[....] Activating swapfile swap...done.
[....] Cleaning up temporary files.... ok
[....] Setting kernel variables ...done.
[....] Configuring network interfaces...Internet Systems Consortium DHCP
Client 4.2.2
Copyright 2004-2011 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Cannot find device "eth0"
Bind socket to interface: No such device
Failed to bring up eth0.
done.
[....] Cleaning up temporary files...[ 7.900000] [MAX98090]
max98090_set_playback_speaker_headset(111)
```



```
[ ok
INIT: Entering runlevel: 2
[info] Using makefile-style concurrent boot in runlevel 2.
[....] Starting enhanced syslogd: rsyslogd. ok
[....] Starting periodic command scheduler: cron. ok

Debian GNU/Linux wheezy/sid wks-40-13 ttySAC1

wks-40-13 login: root
Password:
Linux wks-40-13 3.6.0-rc6 #1 SMP PREEMPT Sun Sep 23 00:12:53 CEST 2012
armv7l

The programs included with the Debian GNU/Linux system are free
software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@wks-40-13:~# cat /proc/cpuinfo
Processor       : ARMv7 Processor rev 0 (v7l)
processor       : 0
BogoMIPS       : 1992.29

processor       : 1
BogoMIPS       : 1992.29

processor       : 2
BogoMIPS       : 1992.29

processor       : 3
BogoMIPS       : 1992.29

Features        : swp half thumb fastmult vfp edsp neon vfpv3 tls
CPU implementer : 0x41
CPU architecture: 7
CPU variant     : 0x3
CPU part        : 0xc09
CPU revision    : 0

Hardware        : ODROID-X
Revision       : 0000
Serial         : 0000000000000000
```

Setting up Linux on ODROID-X/Exynos4412 (last edited 2012-10-18 17:56:06 by mvanga)