freescale™
飞思卡尔

Hands-on Workshop: Developing on Quick Start Board for i.MX Processors

APF-CON-T0701

Rui Yang

May 2013

Topics

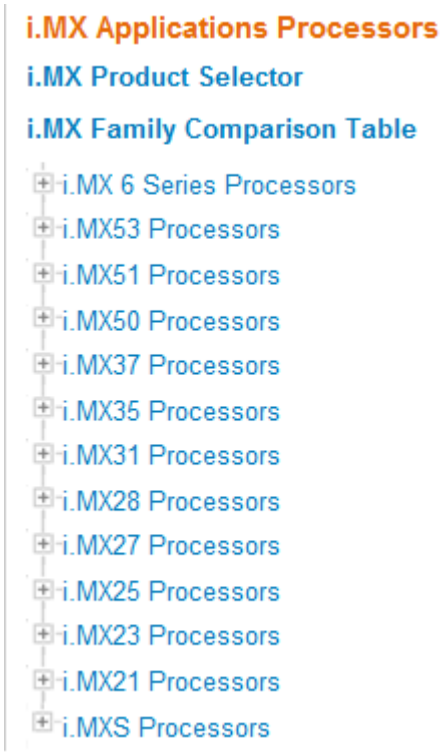
- **Get i.MX materials from Freescale website**
  - i.MX6 Series Processor / Reference Solutions Material
- **Setup building environment and compiling**
  - Android / Linux solution
- **Download compiled images into target board**
- **Design Consideration**
  - Hardware Design

freescale™
飞思卡尔

Get i.MX materials from Freescale website

Get i.MX materials from Freescale website

- From the following web address:
- http://www.freescale.com/webapp/sps/site/homepage.jsp?nodeId=018rH3ZrDR
- You can get all i.MX application processors' family list:



You can get detailed information about each i.MX application processors after clicking into it.

This section use i.MX 6 Series Processors for example.

freescale™ 飞思卡尔

i.MX6 Series and Reference Solutions

- Check available reference solutions for i.MX6:

Choose suitable processor for your production in i.MX 6 Series :

Product (# of Parts) Compare	Datasheet Part Data	Order	Description	Product Page Status	Core Type	Core: Operating Frequency (Max) (MHz)	Core: Number of cores (Spec)	Ambient Operating Temperature (Min-Max) (°C)	Cache (KB)	L2 Cache (Max) (KB)	Internal RAM (KB)	External Memory Supported	Serial Interface Type	Video/Display features
i.MX6D (10)		Distributor Buy Direct Samples	i.MX 6Dual Family of Applications Processors	Active	ARM Cortex A9	1000 850	2	-40 to 125 -40 to 105 -20 to 105	32	1000	256	DDR3 DDR3L FLASH LPDDR2	SDIO eSDFI I2C UART USB I2S ESAI	HD1080p Video Decode HD1080p Video Encode HD1080p Video Transcode HDMI 1.4 Image Pre and Post Processor LCD Controller OpenCL OpenGL ES 2.0 OpenVG 1.1 OpenGL ES 1.1
i.MX6DL (7)		Distributor Buy Direct Samples	i.MX 6DualLite Family of Applications Processors	Active	ARM Cortex A9	800 1000	2	-40 to 105 -40 to 125 0 to 95 -20 to 105	32	512	128	DDR3 DDR3L FLASH LPDDR2	SDIO eSDFI I2C UART USB I2S ESAI	HD1080p Video Decode HD1080p Video Encode HD1080p Video Transcode HDMI 1.4 Image Pre and Post Processor LCD Controller OpenGL ES 1.1

i.MX6Q Reference Solutions

SABRE Platform for Smart Devices Based on the i.MX 6 Series

SABRE Board for Smart Devices Based on the i.MX 6 Series

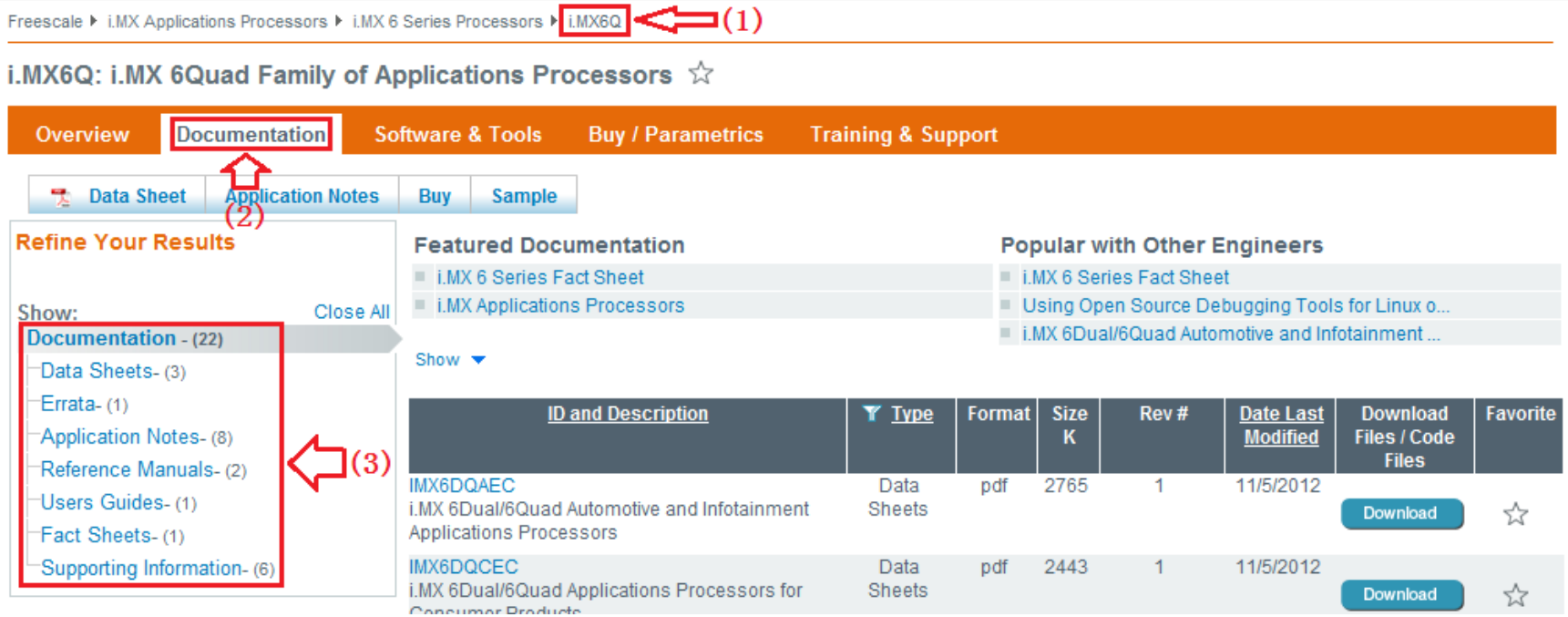
- Select proper reference solutions for your production

SABRE for Automotive Infotainment Based on the i.MX 6 Series

i.MX 6SoloLite Evaluation Kit

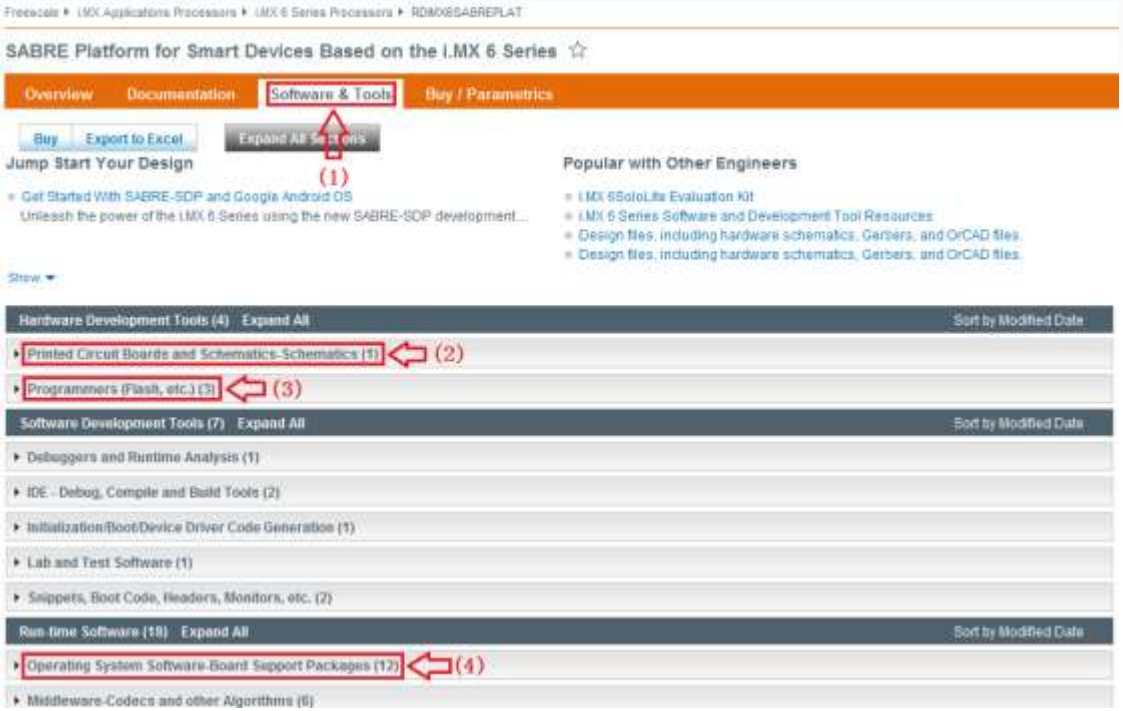
freescale™ 飞思卡尔

i.MX6Q Processor Related Material



1. Select i.MX6Q in family tree and click into it;
2. Choose and press “Documentation” in option bar;
3. Get i.MX6Q Application Processor related material;

i.MX6 Reference Solution Related Material



1. Select “SABRE Platform for Smart Devices”, and choose “Software & Tools”;
2. Hardware Material (Schematic, Gerber, OrCAD..etc);
3. Manufacturing tool;
4. Software Material (Android, Linux..etc)

freescale™
飞思卡尔

Setup Building Environment and Compiling

Using Android Solution

Preparation

Download Ubuntu 10.04 (Lucid) 64-bit Desktop from:
http://releases.ubuntu.com/lucid/

The Sun JDK is no longer in Ubuntu's main package repository.
Download latest Oracle/Sun JDK 6 binary release from:
http://www.oracle.com/technetwork/java/javase/downloads/index.html

Java SE 6 Update 43
This release brings in security features and bug fixes. Oracle strongly recommends that all Java SE 6 users upgrade to this release. Learn more

JDK DOWNLOAD

JDK 6 Docs

JRE DOWNLOAD

JRE 6 Docs

Product / File Description	File Size	Download
Linux x86	65.43 MB	jdk-6u43-linux-i586-rpm.bin
Linux x86	68.45 MB	jdk-6u43-linux-i586.bin
Linux x64	65.65 MB	jdk-6u43-linux-x64-rpm.bin
Linux x64	68.7 MB	jdk-6u43-linux-x64.bin
Solaris x86	68.35 MB	jdk-6u43-solaris-i586.sh
Solaris x86 (SVR4 package)	119.92 MB	jdk-6u43-solaris-i586.tar.Z

freescale™ 飞思卡尔

Setup Building Environment

Install the Ubuntu 10.04 64 Bit Desktop.

Install the following packages for essential Android build, uImage And uboot format support, building mtd-util, file comparison, storage partition:

```
$ sudo apt-get install git-core gnupg flex bison gperf build-essential \
  zip curl zlib1g-dev libc6-dev lib32ncurses5-dev ia32-libs \
  x11proto-core-dev libx11-dev lib32readline5-dev lib32z-dev \
  libgl1-mesa-dev g++-multilib mingw32 tofrodos python-markdown \
  libxml2-utils xsltproc
$ sudo apt-get install uboot-mkimage
$ sudo apt-get install uuid-dev liblzo2-dev
$ sudo apt-get install meld gparted
```

Install and setup jdk-6u43-linux-x64.bin, then verify Java version.

```
$ sudo chmod +x jdk-6u43-linux-x64.bin
$./jdk-6u43-linux-x64.bin
$ echo 'export PATH=Your Path/jdk1.6.0_43/bin:$PATH' >> ~/.bashrc
$ java -version
java version "1.6.0_43"
Java(TM) SE Runtime Environment (build 1.6.0_43-b01)
Java HotSpot(TM) 64-Bit Server VM (build 20.14-b01, mixed mode)
```

NOTE: You can get detailed install guide in "buildenv.pdf" from our Disty.

freescale™ 飞思卡尔

Get Source Code (Android/Kernel)

Get Android source code from Goodle repo, then retrieve ALSA source code:

```
$ cd ~
$ mkdir myandroid
$ cd myandroid
$ curl https://dl-ssl.google.com/dl/googlesource/git-repo/repo > ./repo
$ chmod a+x ./repo
$ ./repo init -u https://android.googlesource.com/platform/manifest -b android-4.0.4_r1.1
$ cp /opt/imx-android-13.4.1/code/13.4.1/default.xml .repo/manifests/default.xml
$ ./repo sync
$ cd myandroid/external
$ git clone git://android.git.linaro.org/platform/external/alsa-lib.git
$ cd myandroid/external
$ git clone git://android.git.linaro.org/platform/external/alsa-utils.git
$ cd myandroid/hardware
$ git clone git://android.git.linaro.org/platform/hardware/alsa_sound.git
```

Get Kernel source code from Freescale's git:

```
$ cd myandroid
$ git clone git://git.freescale.com/imx/linux-2.6-imx.git kernel_imx
$ cd kernel_imx
$ git checkout imx-android-13.4.1
```

NOTE: Assume you unpack "imx-android-13.4.1.tar.gz" to "/opt/imx-android-13.4.1/"

freescale™ 飞思卡尔

Get Source Code (U-Boot/i.MX Patch)

Get U-Boot source code from Freescale's git:

```
$ cd myandroid/bootable
$ mkdir bootloader
$ cd bootloader
$ git clone git://git.freescale.com/imx/uboot-imx.git uboot-imx
$ cd uboot-imx
$ git checkout imx-android-13.4.1
```

Add patch code for i.MX:

```
$ cd myandroid/bootable
$ cd ~/myandroid
$ source /opt/imx-android-13.4.1/code/13.4.1/and_patch.sh
$ c_patch /opt/imx-android-13.4.1/code/13.4.1 imx_13.4.1
If everything is OK, "c_patch" will generate the following output to indicate successful
patch:
**********************************************************************
Success: Now you can build the Android code for FSL i.MX platform
**********************************************************************
```

NOTE: You can get detailed steps from "/imx-android-13.4.1/doc/Android_User_Guide.pdf"

freescale™ 飞思卡尔

Build Images

Build U-Boot image (i.MX 6Quad SABRE SD for example):

```
$ cd ~/myandroid/bootable/bootloader/uboot-imx
$ export ARCH=arm
$ export CROSS_COMPILE=~/myandroid/prebuilt/linux-x86/toolchain/arm-eabi-4.4.3/bin/arm-eabi-
$ make distclean
$ make mx6q_sabresd_android_config
$ make
```

Build Kernel image:

```
$ cd ~/myandroid/kernel_imx
$ make distclean
$ make imx6_android_defconfig
$ make uImage
```

Build boot.img (uImage + uRamdisk):

```
$ cd ~/myandroid
$ source build/envsetup.sh
$ lunch sabresd_6dq-user
$ make bootimage
```

Build Android Image

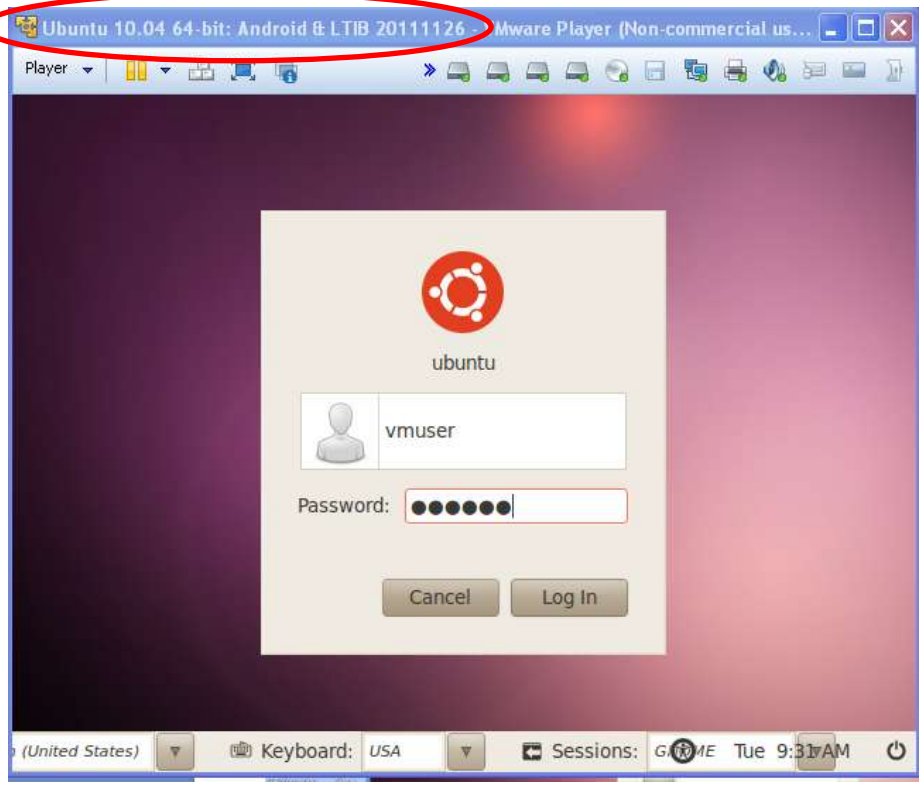
```
$ cd ~/myandroid
$ source build/envsetup.sh
$ lunch sabresd_6dq-user
$ make
```

freescale™ 飞思卡尔

Useful Tips:

- We provide reference virtual build host running on VMware-player-5.0.0, all the building environment are ready.
- Uncompress ubunut_64-bit_1004_20121126_android.7z to PC, Install
- VMware-player-5.0.0 and open this image.

Username: vmuser
Password: vmuser



freescale™ 飞思卡尔

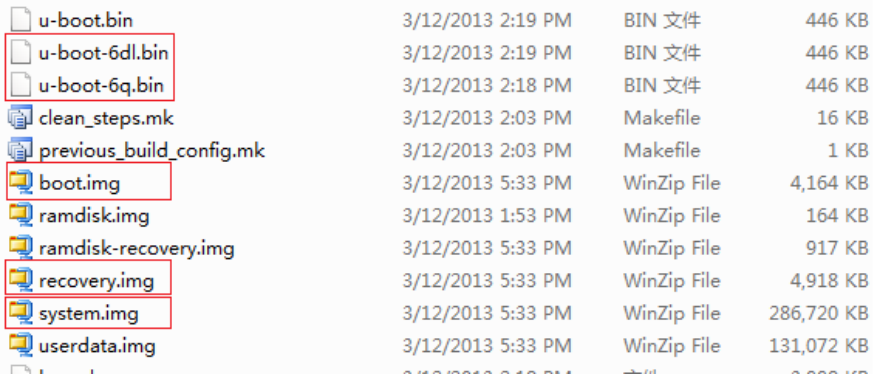
freescale™
飞思卡尔

Download Android Images into Target Board

Using SABRE Platform for Smart Devices

Compiled Images

- After building, you can get the following images located in “myandroid/out/target/product/sabresd_6dq”:



u-boot.bin	3/12/2013 2:19 PM	BIN 文件	446 KB
u-boot-6dl.bin	3/12/2013 2:19 PM	BIN 文件	446 KB
u-boot-6q.bin	3/12/2013 2:18 PM	BIN 文件	446 KB
clean_steps.mk	3/12/2013 2:03 PM	Makefile	16 KB
previous_build_config.mk	3/12/2013 2:03 PM	Makefile	1 KB
boot.img	3/12/2013 5:33 PM	WinZip File	4,164 KB
ramdisk.img	3/12/2013 1:53 PM	WinZip File	164 KB
ramdisk-recovery.img	3/12/2013 5:33 PM	WinZip File	917 KB
recovery.img	3/12/2013 5:33 PM	WinZip File	4,918 KB
system.img	3/12/2013 5:33 PM	WinZip File	286,720 KB
userdata.img	3/12/2013 5:33 PM	WinZip File	131,072 KB

- **u-boot-6dl.bin/u-boot-6q.bin**: bootloader, start offset is 0, max size is 1MB;
- **boot.img**: android image which stores kernel and ramdisk together, partition name is “Boot”, start offset is 8MB, max size is 8MB;
- **recovery.img**: boot.img format, which stores kernel and ramdisk, partition name is “Recovery”, start offset is following “Boot”, max size is 8MB;
- **system.img**: android EXT4 system files, partition name is “System”, start offset is following “Recovery”, max size is 512MB;

freescale™ 飞思卡尔

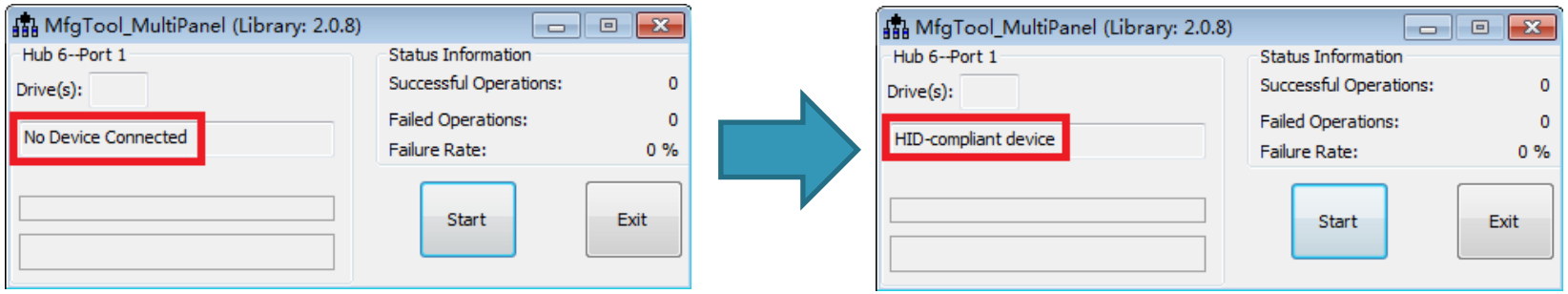
Download Preparation

- Setup download environment as following:

freescale™ 飞思卡尔

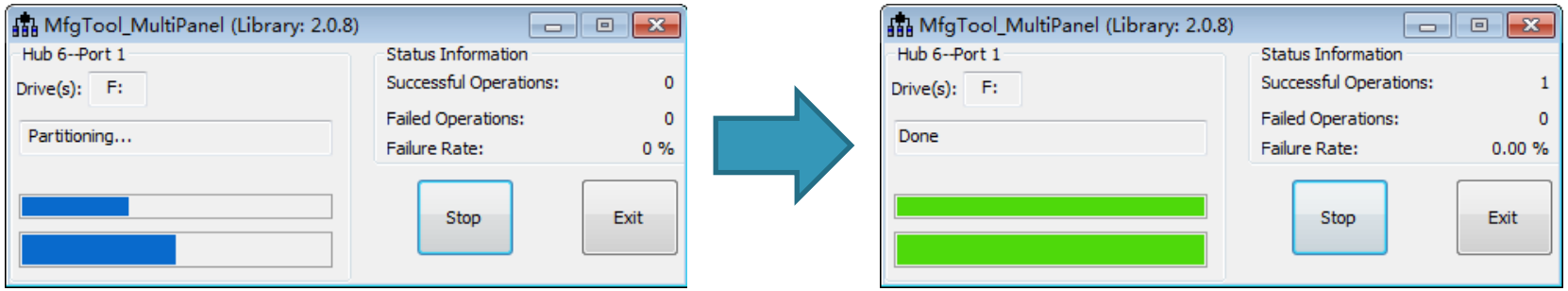
Download Images by Mfgtools

1. Copy such four images into “\Mfgtools-Dir\Profiles\MX6Q Linux Update\OS Firmware\files\android”;
2. Change the SABRE SD SW6 (boot) to 00001100 (from 1-8 bit) to enter USB OTG download mode;
3. Power on the board. Using USB cable on the SABRE SD OTG port, connect your Windows PC to SABRE SD;
4. MfgTool will detect SABRE board, the status will change as following:



Download Images by Mfgtools

5. Click “Start” to start image downloading;
6. During the downloading process, the status bar of MfgTool will show the downloading status;
7. The downloading is complete when MfgTool show “green progress bar” as following:



8. Change Boot Switch (SW6) to 11100110 (from 1-8 bit), make SABRE boot from eMMC;

NOTE: You can get detailed steps from “/imx-android-13.4.1/doc/ Android_Quick_Start_Guide.pdf”

Useful Tips:

1. There are three hardware displays supported in SDP: two LVDS display panels and HDMI output, you can set different U-Boot environment parameters for display output as following:

```
LVDS Display Single Display on LVDS1 display:
U-Boot > setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0 vmalloc=400M
init=/init video=mxcfb0:dev=ldb,LDB-XGA,if=RGB666,bpp=16 video=mxcfb1:off video=mxcfb2:off
fbmem=10M fb0base=0x27b00000

HDMI Display Single Display:
U-Boot > setenv bootargs console=ttymxc0,115200 androidboot.console=ttymxc0 vmalloc=400M
init=/init video=mxcfb0:dev=hdmi,1920x1080M@60 video=mxcfb1:off video=mxcfb2:off fbmem=28M

LVDS&HDMI Display Dual Display enable LVDS1 and HDMI output dual display feature:
U-Boot > setenv bootargs console=ttymxc0,115200 init=/init rw video=mxcfb0:dev=ldb,LDBXGA,
if=RGB666,bpp=16 fb0base=0x27b00000 video=mxcfb1:dev=hdmi,1920x1080M@60 fbmem=10M,28M
video=mxcfb2:off vmalloc=512M androidboot.console=ttymxc0

LVDS&HDMI&LVDS Display Triple Display enable LVDS1, HDMI output, and LVDS0 triple display:
U-Boot > setenv bootargs console=ttymxc0,115200 init=/init rw video=mxcfb0:dev=ldb,LDBXGA,
if=RGB666,bpp=16 fb0base=0x27b00000 video=mxcfb1:dev=hdmi,1920x1080M@60 fbmem=10M,28M
vmalloc=512M androidboot.console=ttymxc0
```

2. You can follow below steps to copy files into “/system” partition:

```
$ mount -t ext4 -o rw,remount /dev/block/mmcblk0p5 /system
$ busybox cp /mnt/sdcard/files /system/media/
$ mount -t ext4 -o ro,remount /dev/block/mmcblk0p5 /system
```

freescale™
飞思卡尔

Setup Building Environment and Compiling

Using Linux Solution

LTIB Introduction

- Linux Target Image Builder (LTIB) is a tool created by Freescale that is used to build Linux target images, composed of a set of packages:
  - A mechanism to deliver Linux board support packages (BSPs)
  - A wrapper around tool chains and standard Linux commands (cp, make, objcopy, tar, gcc, …)

- LTIB Packages for i.MX6Q SABRE Board:
  - Toolchain for the ARM® Cortex™- A9 CPU
  - Linux Kernel 3.0.35
  - Uboot 2009.08
  - Base tools: BusyBox, Dropbear, ...
  - and many more …

freescale™ 飞思卡尔

Setup LTIB Environment

Install the Ubuntu 10.04 64 Bit Desktop.

Install the following packages:

```
$ sudo apt-get install gettext libgtk2.0-dev rpm bison m4 libfreetype6-dev
$ sudo apt-get install libdbus-glib-1-dev liborbit2-dev intltool
$ sudo apt-get install ccache ncurses-dev zlib1g zlib1g-dev gcc g++ libtool
$ sudo apt-get install uuid-dev liblzo2-dev
$ sudo apt-get install tcl dpkg
$ sudo apt-get install ia32-libs libc6-dev-i386 lib32z1
```

This package is used for uImage and uboot format support:

```
$ sudo apt-get install uboot-mkimage
```

These two packages are recommended to help the daily work, “meld” for file comparison and “gparted” for storage partition:

```
$ sudo apt-get install meld gparted
```

NOTE: You can get detailed steps from “Setting_up_LTIB_Host_L3.0.35_1.1.0.pdf”

freescale™ 飞思卡尔

Install and Configure LTIB

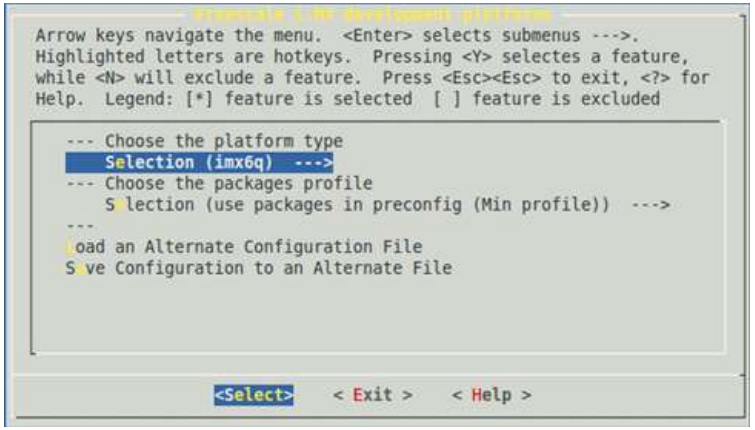
Install LTIB package, not as root, in a location such as /home/user/:

```
$ tar zxvf L3.0.35_1.1.0_121218_source.tar.gz
$ ./L3.0.35_1.1.0_121218_source/install
```

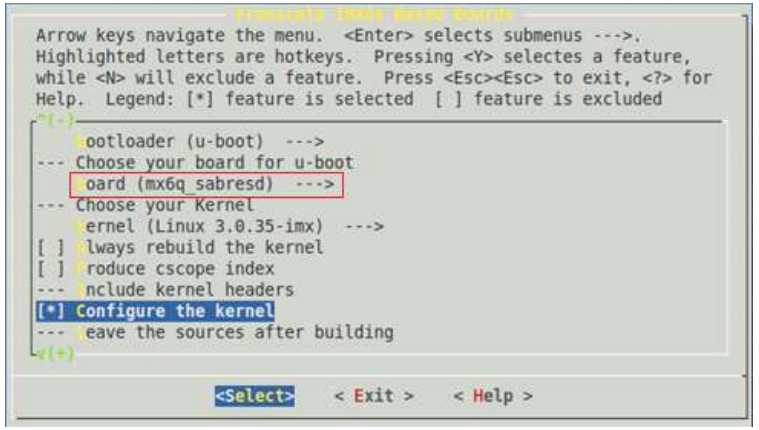
Configure and build LTIB:

```
$ cd <LTIB directory>
$ ./ltib -m config

1). The LTIB menu will appear as following;
2). If it has not been selected already,
select:
   --- Choose the platform type
        Selection (imx6q) --->
3). Exit
4). Save
```

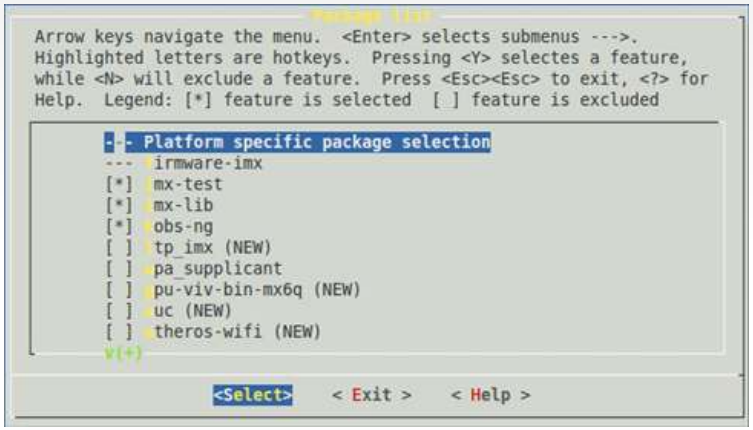



```
5). Select
   --- Choose your board for u-boot
        board (mx6q_sabresd) --->
6). Using the spacebar, select:
   [*] Configure the Kernel
7). Don't exit yet...
```

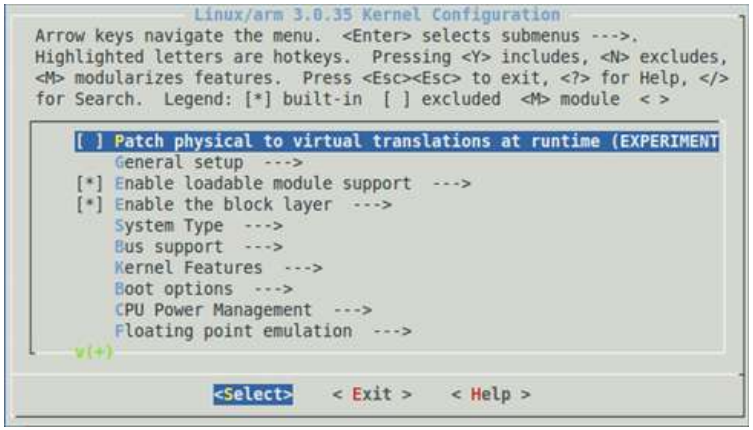


Install and Configure LTIB

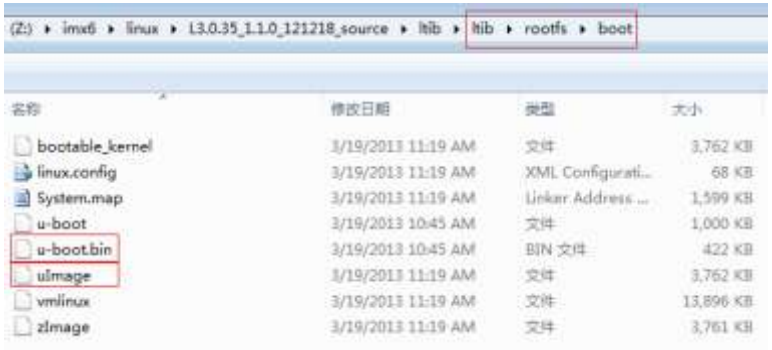
8). Select Package List;
    A Linux system is comprised of two main entities:
     -- Kernel
     -- File System
9). The Package List is what produces the File System.



10). Kernel Configuration;
11). Text messages will scroll by until the Kernel Configuration menu appears;
12). Exit from the kernel config;
13). The kernel now builds.



After LTIB building, you can get the U-boot and Kernel images:



freescale™ 飞思卡尔

freescale™
飞思卡尔

Download Linux Images into Target Board

Using SABRE Platform for Smart Devices

Download Images by Mfgtools

- You should prepare Rootfs for Mfgtools download as following:

```
$ cd <your rootfs directory>
$ sudo -s
$ tar -cjf ../rootfs.tar.bz2 ./*
```

- Copy “u-boot.bin”, “uImage”, “rootfs.tar.bz2” into i.MX6Q Linux MFGtool profile folder.
  - *(~\Mfgtools-Rel-1.1.0_121218_MX6Q_UPDATER\Profiles\MX6Q Linux Update\OS Firmware\files)*

- Change the SABRE SD SW6 (boot) to 00001100 (from 1-8 bit) to enter USB OTG download mode, then download Linux images by Mfgtoos at the same way as Android;

- Change Boot Switch (SW6) to 01000010 (from 1-8 bit), make SABRE boot from SD card slot3;

Download Images in Linux Host – (1)

Insert one SD card into your Linux host PC, and it will recognize your SD, In this example, the device node assigned is “/dev/sdb”:

```
$ cat /proc/partitions
 major minor  #blocks   name
   8     0    78125000  sda
   8     1    75095811  sda1
   8     2    1         sda2
   8     5    3028221   sda5
   8     32   488386584 sdc
   8     33   488386552 sdc1
   8     16   3921920   sdb
   8     18   3905535   sdb1
```

Copying Boot Loader Image into SD card:

```
$ sudo dd if=u-boot-mx6q-sabresd.bin of=/dev/sdb bs=512 seek=2 skip=2 conv=fsync
```

Copying Kernel Image into SD card:

```
$ sudo dd if=uImage of=/dev/sdb bs=512 seek=2048 conv=fsync
```

freescale™ 飞思卡尔

Download Images in Linux Host – (2)

Create a partition for Root File System:

```
$ sudo umount /dev/sdb
$ sudo fdisk /dev/sdb
  u [switch the unit to sectors instead of cylinders]
  d [repeat this until no partition is reported by the 'p' command ]
  n [create a new partition]
  p [create a primary partition]
  1 [the first partition]
  16384 [starting at offset sector #16384, i.e. 8MB, which leaves enough space for the
kernel, the boot loader and its configuration data]
  <enter> [using the default value will create a partition that spans to the last sector
of the medium]
  w [this writes the partition table to the medium and fdisk exits]
$ sudo mkfs.ext4 /dev/sdb1
```

Copy target file system into the partition in SD card:

```
$ mkdir /home/user/mountpoint
$ sudo mount /dev/sdb1 /home/user/mountpoint

$ gunzip rootfs.ext2.gz
$ mount -o loop -t ext2 rootfs.ext2 /home/user/rootfs

$ cd /home/user/rootfs
$ sudo cp -rpa [A-z]* /home/user/mountpoint
$ sudo umount /home/user/mountpoint
```

Insert this SD card into SABRE board slot3 and boot from it.

freescale™ 飞思卡尔

Modify U-Boot Environment

Create U-Boot environment commands to send display out through LVDS connected to DISP0 (default) from SD card:

```
U-Boot > setenv loadaddr 0x10800000
U-Boot > setenv bootargs_base 'console=ttymxc0,115200'
U-Boot > setenv bootargs_base_lvds 'video=mxcfb0:dev=ldb,LDB-XGA,if=RGB666'
U-Boot > setenv bootargs_mmc 'root=/dev/mmcblk1p1 rootwait rw ip=none rootfstype=ext4'
U-Boot > setenv bootargs ${bootargs_base} ${bootargs_base_lvds} ${bootargs_mmc}
U-Boot > setenv bootcmd_mmc 'mmc dev 2;mmc read ${loadaddr} 0x800 0x2000;bootm'
U-Boot > setenv bootcmd 'run bootcmd_mmc'
U-Boot > saveenv
```

For SDP, the LVDS is connected to DISP1, thus, modify the command as follows:

```
U-Boot > setenv bootargs_base_lvds 'video=mxcfb0:dev=ldb,LDB-XGA,if=RGB666 ldb=sin1'
```

Need to add “rootfstype=xxx” to eliminate the time to determine the file system type of the root file system.

NOTE: refer to “SABRE_SD_Release_Notes_L3.0.35_1.1.0.pdf” for the details about the “Kernel Boot Parameters”.

freescale™ 飞思卡尔

Ubuntu Booting on SDP

- After download, the Ubuntu boot from SD card slot3 on SABRE Platform for Smart Devices as following:

freescale™
飞思卡尔

Hardware Design Consideration

Power Management

- You should consider the following parts in power design:
  1. Voltage range of each power domain;
  2. Maximal current consuming of each power domain;
  3. System power up sequence;
- The following are typical Max Power Measurement Results on SD Board:

Supply Domain	Voltage (V)	Linux - ER1205 - on SD Board[1]	
		P (mW)	I (mA)
VDDARM_IN	1.37	2068.7	1510 (1625 max[2])
VDDSOC_IN	1.37	1555	1135 (1250 max[2])
VDDHIGH_IN	2.78	236.3	85
Total Power (without DDR3 I/O + Memories)		**3860**	
DDR3 I/O +[3] Memories	1.5	1995	1330 (1390 max[2])
Total Power		5855	

i.MX6Q Power Domain

Symbol	Usage	Voltage	Max Current	Generated	Sequence
VDDARM_IN	ARM Core Power	1.05~1.5V, 1.375V	2500 mA	PF0100 SW1A/B	1
VDDARM23_IN					
VDDSOC_IN	IP Power(VPU, GPU…etc)	1.275~1.5V, 1.375V	1750 mA	PF0100 SW1C	1
VDDHIGH_IN	Internal Regulator	2.8~3.3V, 2.8V	100 mA	PF0100 VGEN5	---
VDD_SNVS_IN	Backup Battery	2.8~3.3V, 3.0V	400 µA	PF0100 VSNVS	0
USB_OTG_VBUS	USB Supply Voltages	4.4~5.25V, 5.0V	600 mA	PF0100 SWBST	---
USB_H1_VBUS					
NVCC_DRAM	DDR I/O supply	LPDDR2: 1.14~1.3V, 1.2V DDR3: 1.425~1.575V, 1.5V DDR3_L:1.283~1.45V,1.35V	2500 mA	PF0100 SW3A/B	---
NVCC_RGMII	RGMII I/O Power	1.15~2.625V, 1.5V			
NVCC_EIM0,1,2 NVCC_ENET NVCC_GPIO NVCC_LCD NVCC_NANDF NVCC_SD2/3 NVCC_JTAG	GPIO Power Supply	1.65~3.6V, 1.8/ 2.8/ 3.3 V	2000 mA	PF0100 SW2	---

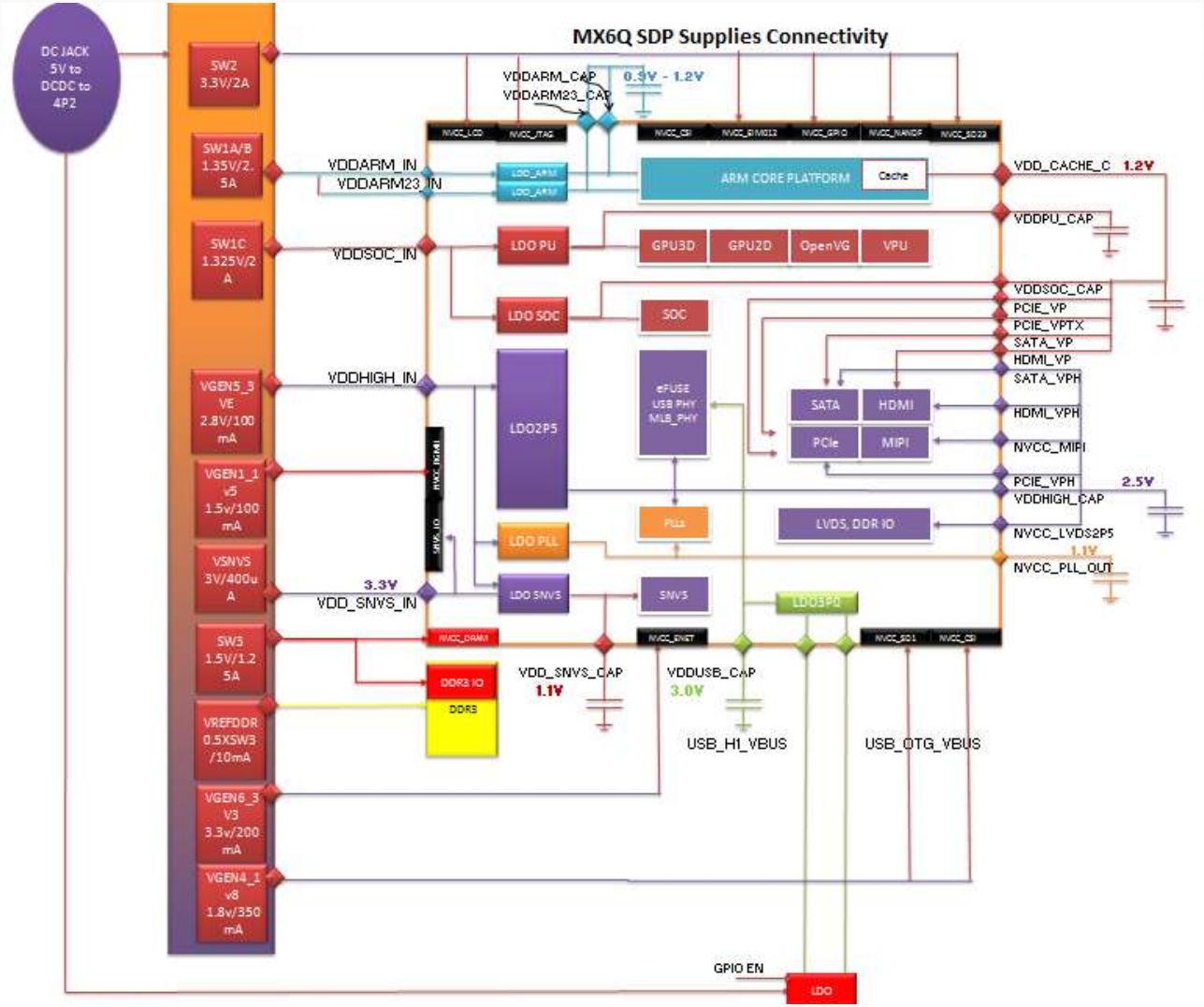
freescale™ 飞思卡尔

i.MX6Q Internal Regulators

- i.MX6 series contain 7 internal regulators, it simplify the power supply scheme of the system;
- The following domains are supplied by internal regulators:

Symbol	Usage	Voltage	Generated
NVCC_LVDS2P5	LVDS	2.25~2.75V, 2.5V	i.MX VDDHIGH_CAP
NVCC_MIPI	MIPI	2.25~2.75V, 2.5V	i.MX VDDHIGH_CAP
HDMI_VP	HDMI Supply Voltages	0.99~1.3V, 1.1V	i.MX VDDSOC_CAP
HDMI_VPH		2.25~2.75V, 2.5V	i.MX VDDHIGH_CAP
PCIE_VP	PCIe Supply Voltages	1.023~1.3V, 1.1V	i.MX VDDSOC_CAP
PCIE_VPH		2.325~2.75V, 2.5V	i.MX VDDHIGH_CAP
PCIE_VPTX		1.023V~1.3V, 1.1V	i.MX VDDSOC_CAP
SATA_VP	SATA Supply Voltages	0.99~1.3V, 1.1V	i.MX VDDSOC_CAP
SATA_VPH		2.25~2.75V, 2.5V	i.MX VDDHIGH_CAP

i.MX6 SDP Power Connectivity



i.MX6 SDP Power Design

	Voltage	Power Up Sequence	Current Drawn (mA)	SYS 4V2 Current (mA)	NOTES
SW1A	1.375	1	2155	1001	
SW1B					
SW1C	1.375	2	1590	739	
SW2	3.3	5	653	728	
SW3A	1.5	3	1500	760	
SW3B					
SW4	3.15	6	200	213	
SWBST	5.0	13	300	507	
VGEN1	1.5	9	100	0	Supplied from SW4
VGEN2	1.5	10	250	0	Supplied from SW4
VGEN3	2.8	11	70	66	
VGEN4	1.8	12	310	189	
VGEN5	2.8	10	75	71	See Note on Page 20
VGEN6	3.3	8	160	178	
VSNVS	3.0	0	0.2	0	
VREFDDR	0.75	3	10	3	
Total System Current Requirements:				4454	

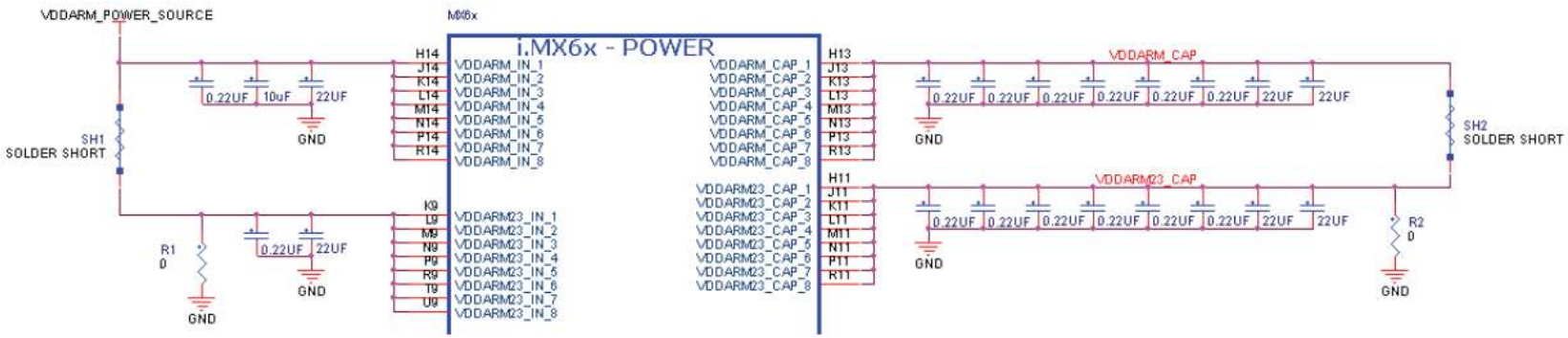
Typical Power Requirements

Voltge	Rail Name	Block	Generated By	Current Capability (mA)	NOTES
5.0	PMIC_5V	USB	PF0100 SWBST	600	
		LVDS1			
		HDMI			
	AUX_5V	SATA	MAX8815	1000	
		LVDS0			
		CAN			
3.3	GEN_3V3	EMMC	PF0100 SW2	2000	NVCC_LCD NVCC_EIM0/1/2 NVCC_GPIO NVCC_SD2/3 NVCC_NANDF NAND_JTAG
		SD3			
		NOR			
		SATA			
		LVDS			
		HDMI			
		MIPI			
		mPCIe			
		SENSORS			
	VGEN6_3V3	ETH	PF0100 VGEN6	200	NVCC_ENET
3.15	AUX_3V15	EXP HDR	PF0100 SW4	1000	Supplies: VGEN1 VGEN2
		TOUCH			
		GPS			
2.8	VDDHIGH_IN	IMX6	PF0100 VGEN5	100	
	VGEN3_2V5	CAMERA	PF0100 VGEN3	100	
2.5	GEN_2V5	SATA	IMX6 VDDHIGH_CAP	TBD	NVCC_MIPI
		HDMI			
		MIPI			
		mPCIe			
1.8	GEN_1V8	AUDIO	PF0100 VGEN4	350	NVCC_SD1 NVCC_CSI
		CAMERA			
		ACC			
1.5	VGEN2_1V5	CAMERA	PF0100VGEN2	250	
	VGEN1_1V5	GPS	PF0100 VGEN1	100	
		mPCIe			
	DDR_1V5	DDR	PF0100 SW3A/B	2500	
1.375	VDDCORE	ARMCORE	PF0100 SW1A/B	2500	
	VDDSOC	VDDSOC	PF0100 SW1C	1750	
0.75	VREFDDR	DDR	PF0100 VREFDDR	10	

System Power Rails

Pin-to-Pin Compatible Design

All-In-One Circuit:



Power Connections for i.MX6 Series:

	i.MX 6Quad	i.MX 6Dual	i.MX 6DualLite	i.MX 6Solo
SH1	Shorted	Open	Shorted	Shorted
SH2	Shorted	Open	Shorted	Shorted
R1	Open	Shorted	Open	Open
R2	Open	Shorted	Open	Open

Capacitor Placement

- One 22 µF bulk capacitor should be connected to each of these on-chip LDO regulator outputs:
  - VDDARM_CAP
  - VDDARM23_CAP
  - VDDSOC_CAP
  - VDDPU_CAP
- A 22 µF bulk capacitor must be placed as near as possible with pins/vias. ***The distance should be less than 50mil between bulk cap and VDDxx_CAP pins***. Decoupling capacitors such as 0.1 µF or 0.22 µF should also be used.

- It is highly recommended that ***the user places the decoupling and bulk capacitors of the power domains on the bottom layer of the hardware design***, directly underneath the associated package contacts.

Related Materials:

- **AN4397.pdf:**
  - Common Hardware Design for i.MX 6Dual/6Quad and i.MX 6Solo/6DualLite
- **AN4509.pdf:**
  - i.MX 6Dual/6Quad Power Consumption Measurement
- **IMX6DQ6SDLHDG.pdf:**
  - Hardware Development Guide for i.MX 6Quad, 6Dual, 6DualLite, 6Solo
  - Families of Applications Processors
- **IMX6DQCEC.pdf:**
  - i.MX 6Dual/6Quad Applications Processors for Consumer Products Data Sheet

freescale™ 飞思卡尔


freescale™
飞思卡尔