WT-EVK6ULX Kernel Compilation Method

Revise history:

Version	Date	Log
V1.00	2019/01/10	Create
V1.01	2019/01/16	Publish V1.01

Contents

1．Construction of Development Environment

Create a work directory:
mkdir /home/industio_work
cd /home/industio_work

1.1、Get source file

- Download uboot source:
  Download link: git clone http://git.freescale.com/git/cgit.cgi/imx/uboot-imx.git
  After download, enter the source directory "uboot-imx":
  cd uboot-imx
  Checkout v2016 branch:
  git checkout imx_v2016.03_4.1.15_2.0.0_ga
  Download uboot patch:
  wget https://github.com/industio/WT_EVK6ULX/raw/master/UBOOT-PATCH/0002-update-CONFIG_ENV_OFFSET-0x280000.patch
  wget https://github.com/industio/WT_EVK6ULX/raw/master/UBOOT-PATCH/0001-Industio-evk-board-first-commit-uboot.patch

  Merge patch:
  git am -s 0001-Industio-evk-board-first-commit-uboot.patch
  git am -s 0002-update-CONFIG_ENV_OFFSET-0x280000.patch
- Download kernel source file
  git clone http://git.freescale.com/git/cgit.cgi/imx/linux-imx.git
  Enter kernel directory : linux-imx
  cd linux-imx
  Checkout branch:
  git checkout imx_4.1.15_2.0.0_ga
  Download kernel patch:
  wget https://github.com/industio/WT_EVK6ULX/raw/master/KERNEL-PATCH/0001-Industio-evk-board-first-commit-kernel.patch
  Merge patch:
  git am -s 0001-Industio-evk-board-first-commit-kernel.patch

1.2、Get the development SDK

Enter the work directory:
cd /home/industio_work
Download the SDK :
Download link:https://releases.linaro.org/components/toolchain/binaries/4.9-2017.01/arm-

linux-gnueabihf/gcc-linaro-4.9.4-2017.01-i686_arm-linux-gnueabihf.tar.xz

Decompression:

```
sudo tar xjvf gcc-linaro-4.9.4-2017.01-i686_arm-linux-gnueabihf.tar.xz -C /opt/industio
```

Add environment variables to PATH:

Method 1:

Configure environment variables and edit configuration scripts,

```
#vi environment-setup_hf
GCC_PATH=/opt/industio/gcc-linaro-4.9.4-2017.01-i686_arm-linux-gnueabihf
GCC_CC=arm-linux-gnueabihf

export ARCH=arm
export CROSS_COMPILE=$GCC_CC-
export PATH=$GCC_PATH/bin:$GCC_PATH/bin/$GCC_CC:$PATH
```

Run: `source environment-setup_hf`

Method 2: add the environment variables to profile

eg:

`vi .profile` add the following contents to the end of the file.

```
export PATH=$PATH:/opt/industio/gcc-linaro-4.9.4-2017.01-i686_arm-linux-gnueabihf/bin
export ARCH=arm
export CORESS_COMPILE= arm-linux-gnueabihf
```

Note: If this method is used to compile QTs through qmake, make, it is necessary to ensure that the current system does not have qmake.

Check whether the environment is in effect:

```
which arm-linux-gnueabihf-gcc
```

2. Compile u-boot and kernel

2.1、compile u-boot

Enter uboot-imx:

```
cd /home/industio_work/uboot-imx
```

- For Nandflash version:

```
make mx6ull_14x14_evk_nand_defconfig
make
```

- For EMMC version:

```
make mx6ull_14x14_evk_emmc_defconfig
make
```

Final Generation: u-boot.imx

Note: If you want to compile the generated files into a directory, the method is as follows (for

example, to generate the current “build” directory), here take Nand version as an example:

```
make mx6ull_14x14_evk_nand_defconfig O=build
make O=build
```

2.2、Compile kernel

```
cd /home/industio_work/linux-imx
make    imx6ull_evk_defconfig
make    -j4
```

(J4 represents multithreaded compilation, and 4 is the number of host kernels)

Final Generation : zImage and dtb

zImage directory：arch/arm/boot/zImage

nand version dtb file directory：arch/arm/boot/dts/imx6ull-14x14-evk-gpmi-weim.dtb

emmc version dtb file directory：arch/arm/boot/dts/imx6ull-14x14-evk-emmc.dtb

Note: If you want to compile the generated files into a directory, the method is as follows (for example, to generate the current “build” directory),

```
make imx6ull_evk_defconfig O=build
make O=build
```

3. Update u-boot and kernel

3.1 update u-boot

Firstly, put the u-boot.imx file generated at 2.1 is placed to the SD card root directory, insert the SD card into the EVK board and mounted. Mount to the /mnt directory for example. Enter the /mnt directory and then execute the following commands::

3.1.1 For Nandflash version

```
mount -t debugfs debugfs /sys/kernel/debug
flash_erase /dev/mtd0 0 0
kobs-ng init -x -v --chip_0_device_path=/dev/mtd0 u-boot.imx
```

3.1.2 For EMMC version:

```
dd if=/dev/zero of=/dev/mmcblk1 bs=1k seek=768 conv=fsync count=8
dd if=u-boot.imx of=/dev/mmcblk1 bs=1k seek=1 conv=fsync
```

3.2 update kernel

3.2.1 For Nandflash version:

After Kernel compilation is completed, executing make Zi generates boot_wdk.img and boot_evk.dtb files in the /home/industio_work/linux-imx directory. Put these two files in the

root directory of TF card, insert it into the EVK board, and power on, kernel files will be automatically updated.

3.2.2 For EMMC version:

After compiling Kernel, executing `make Zi` generates imx6ull-14x14-evk.dtb and zImage in the `/home/industio_work/linux-imx` directory , puts the file on TF or U disk and boot up the board:

First mount EMMC to /mnt directory

```
mount /dev/mmcblk1p1 /mnt
```

- TF card:

Mount tf card:

```
mount /dev/mmcblk0p1 /sdcard
cp /sdcard/zImage /mnt -f
cp /sdcard/imx6ull-14x14-evk.dtb /mnt -f
```

- Udisk:

```
mount /dev/sda1 /udisk
cp /udisk/zImage /mnt -f
cp /udisk/imx6ull-14x14-evk.dtb /mnt -f
```

Umount:

```
umount /mnt
umount /sdcard
umount /udisk
```

3.3 Use mfgtools

Download to nandflash :

Modify the config file, open cfg.ini,configure

```
name = NAND Flash
folder=6ull-evk-nand
```

```
[profiles]
chip = Linux

[platform]
board = sabreauto

[LIST]
name = NAND Flash
#name = SDCard

[variable]
board = 14x14evk

#sdcard
mmc = 0

#emmc
#mmc = 1

#sxuboot=sabresd
sxuboot=sabreauto
#sxdtb=sdb
sxdtb=sabreauto
7duboot=sabresd
7ddtb=sdb
6uluboot=14x14evk
#6uldtb=14x14
6uldtb=14x14-evk
6ulldtb=14x14
ldo=evk
plus=
lite=1
initramfs=fsl-image-mfgtool-initramfs-imx_mfgtools.cpio.gz.u-boot
seek = 1
sxnor=qspi2
7dnor=qspi1
6ulnor=qspi1
nor_part=0
nand=nand
nanddtb=gpmi-weim
part_uboot=0
part_kernel=1
part_dtb=2
part_rootfs=3

folder=6ull-edk-nand
#folder=6ull-edk-emmc
```

Note: Short J2 before booting, connect Mrico USB to PC

Download to emmc or SD card:

Modify config file, open cfg.ini, configure:

name = SDCard

#name = NAND Flash

#sdcard

mmc = 0

#emmc

mmc = 1

folder=6ull-evk-emmc

```
[profiles]
chip = Linux

[platform]
board = sabreauto

[LIST]
#name = NAND Flash
name = SDCard

[variable]
board = 14x14evk

#sdcard
#mmc = 0

#emmc
mmc = 1

#sxuboot=sabresd
sxuboot=sabreauto
#sxdtb=sdb
sxdtb=sabreauto
7duboot=sabresd
7ddtb=sdb
6uluboot=14x14evk
#6uldtb=14x14
6uldtb=14x14-evk
6ulldtb=14x14
ldo=evk
plus=
lite=1
initramfs=fsl-image-mfgtool-initramfs-imx_mfgtools.cpio.gz.u-boot
seek = 1
sxnor=qspi2
7dnor=qspi1
6ulnor=qspi1
nor_part=0
nand=nand
nanddtb=gpmi-weim
part_uboot=0
part_kernel=1
part_dtb=2
part_rootfs=3

#folder=6ull-edk-nand
folder=6ull-edk-emmc
```