

SCM-i.MX 6 Series Yocto Linux User's Guide

1. Introduction

The NXP SCM Linux BSP (Board Support Package) leverages the existing i.MX 6 Linux BSP release L4.1.15-2.0.0. The i.MX Linux BSP is a collection of binary files, source code, and support files that can be used to create a U-Boot bootloader, a Linux kernel image, and a root file system. The Yocto Project is the framework of choice to build the images described in this document, although other methods can be also used.

The purpose of this document is to explain how to build an image and install the Linux BSP using the Yocto Project build environment on the SCM-i.MX 6Dual/Quad Quick Start (QWKS) board and the SCM-i.MX 6SoloX Evaluation Board (EVB).

This release supports these SCM-i.MX 6 Series boards:

- Quick Start Board for SCM-i.MX 6Dual/6Quad (QWKS-SCMIMX6DQ)
- Evaluation Board for SCM-i.MX 6SoloX (EVB-SCMIMX6SX)

Contents

1. Introduction.....	1
1.1. Supporting documents	2
2. Enabling Linux OS for SCM-i.MX 6Dual/6Quad/SoloX ..	2
2.1. Host setup	2
2.2. Host packages	2
3. Building Linux OS for SCM i.MX platforms	3
3.1. Setting up the Repo utility	3
3.2. Installing Yocto Project layers	3
3.3. Building the Yocto image	4
3.4. Choosing a graphical back end	4
4. Deploying the image	5
4.1. Flashing the SD card image	5
4.2. MFGTool (Manufacturing Tool)	6
5. Specifying displays	6
6. Reset and boot switch configuration	7
6.1. Boot switch settings for QWKS SCM-i.MX 6D/Q ..	7
6.2. Boot switch settings for EVB SCM-i.MX 6SoloX ..	8
7. SCM uboot and kernel repos.....	8
8. References.....	8
9. Revision history	9



1.1. Supporting documents

These documents provide additional information and can be found at the NXP webpage ([L4.1.15-2.0.0 LINUX DOCS](#)):

- *i.MX Linux® Release Notes*—Provides the release information.
- *i.MX Linux® User's Guide*—Contains the information on installing the U-Boot and Linux OS and using the i.MX-specific features.
- *i.MX Yocto Project User's Guide*—Contains the instructions for setting up and building the Linux OS in the Yocto Project.
- *i.MX Linux® Reference Manual*—Contains the information about the Linux drivers for i.MX.
- *i.MX BSP Porting Guide*—Contains the instructions to port the BSP to a new board.

These quick start guides contain basic information about the board and its setup:

- *QWKS board for SCM-i.MX 6D/Q Quick Start Guide*
- *Evaluation board for SCM-i.MX 6SoloX Quick Start Guide*

2. Enabling Linux OS for SCM-i.MX 6Dual/6Quad/SoloX

This section describes how to obtain the SCM-related build environment for Yocto. This assumes that you are familiar with the standard i.MX Yocto Linux OS BSP environment and build process. If you are not familiar with this process, see the *NXP Yocto Project User's Guide* (available at [L4.1.15-2.0.0 LINUX DOCS](#)).

2.1. Host setup

To get the Yocto Project expected behavior on a Linux OS host machine, install the packages and utilities described below. The hard disk space required on the host machine is an important consideration. For example, when building on a machine running Ubuntu, the minimum hard disk space required is about 50 GB for the X11 backend. It is recommended that at least 120 GB is provided, which is enough to compile any backend.

The minimum recommended Ubuntu version is 14.04, but the builds for dizzy work on 12.04 (or later). Earlier versions may cause the Yocto Project build setup to fail, because it requires python versions only available on Ubuntu 12.04 (or later). See the Yocto Project reference manual for more information.

2.2. Host packages

The Yocto Project build requires that the packages documented under the Yocto Project are installed for the build. Visit the Yocto Project Quick Start at www.yoctoproject.org/docs/current/yocto-project-qs/yocto-project-qs.html and check for the packages that must be installed on your build machine.

The essential Yocto Project host packages are:

```
$ sudo apt-get install gawk wget git-core diffstat unzip texinfo gcc-multilib build-essential
chrpath socat libstd1.2-dev
```

The i.MX layers' host packages for the Ubuntu 12.04 (or 14.04) host setup are:

```
$ sudo apt-get install libstdc++6 xterm sed cvs subversion coreutils texi2html docbook-
utils python-pysqlite2 help2man make gcc g++ desktop-file-utils libgl1-mesa-dev libglu1-mesa-
dev mercurial autoconf automake groff curl lzop asciidoc
```

The i.MX layers' host packages for the Ubuntu 12.04 host setup are:

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

The i.MX layers' host packages for the Ubuntu 14.04 host setup are:

```
$ sudo apt-get install u-boot-tools
```

The configuration tool uses the default version of `grep` that is on your build machine. If there is a different version of `grep` in your path, it may cause the builds to fail. One workaround is to rename the special version to something not containing “`grep`”.

3. Building Linux OS for SCM i.MX platforms

3.1. Setting up the Repo utility

Repo is a tool built on top of GIT, which makes it easier to manage projects that contain multiple repositories that do not have to be on the same server. Repo complements the layered nature of the Yocto Project very well, making it easier for customers to add their own layers to the BSP.

To install the Repo utility, perform these steps:

1. Create a `bin` folder in the home directory.

```
$ mkdir ~/bin (this step may not be needed if the bin folder already exists)
$ curl http://commondatastorage.googleapis.com/git-repo-downloads/repo > ~/bin/repo
$ chmod a+x ~/bin/repo
```

2. Add this line to the `.bashrc` file to ensure that the `~/bin` folder is in your PATH variable:

```
$ export PATH=~/bin:$PATH
```

3.2. Installing Yocto Project layers

All the SCM-related changes are collected in the new `meta-nxp-imx-scm` layer, which is obtained through the Repo sync pointing to the corresponding `scm-imx` branch.

Make sure that GIT is set up properly with these commands:

```
$ git config --global user.name "Your Name"
$ git config --global user.email "Your Email"
$ git config --list
```

The `NXP Yocto Project BSP Release` directory contains the `sources` directory, which contains the recipes used to build, one (or more) build directories, and a set of scripts used to set up the environment. The recipes used to build the project come from both the community and NXP. The Yocto Project layers are downloaded to the `sources` directory. This sets up the recipes that are used to build the project. The following code snippets show how to set up the SCM L4.1.15-2.0.0_ga Yocto environment for the SCM-i.MX 6 QWKS board and the evaluation board. In this example, a directory called `fsl-arm-yocto-bsp` is created for the project. Any name can be used instead of this.

3.2.1. SCM-i.MX 6D/Q quick start board

```
$ mkdir fsl-arm-yocto-bsp
$ cd fsl-arm-yocto-bsp
$ repo init -u git://git.freescale.com/imx/fsl-arm-yocto-bsp.git -b imx-4.1-krogoth -m scm-
imx-4.1.15-2.0.0.xml
$ repo sync
```

3.2.2. SCM-i.MX 6SoloX evaluation board

```
$ mkdir my-evb_6sxscm-yocto-bsp
$ cd my-evb_6sxscm-yocto-bsp
$ repo init -u git://git.freescale.com/imx/fsl-arm-yocto-bsp.git -b imx-4.1-krogoth -m scm-
imx-4.1.15-2.0.0.xml
$ repo sync
```

3.3. Building the Yocto image

Note that the quick start board for SCM-i.MX 6D/Q and the evaluation board for SCM-i.MX 6SoloX are commercially available with a 1 GB LPDDR2 PoP memory configuration.

This release supports the *imx6dqscm-1gb-qwks*, *imx6dqscm-1gb-qwks-rev3*, and *imx6sxscm-1gb-evb*. Set the machine configuration in MACHINE= in the following section.

3.3.1. Choosing a machine

Choose the machine configuration that matches your reference board.

- *imx6dqscm-1gb-qwks* (QWKS board for SCM-i.MX 6DQ with 1 GB LPDDR2 PoP)
- *imx6dqscm-1gb-qwks-rev3* (QWKS board Rev C for SCM-i.MX 6DQ with 1GB LPDDR2 PoP)
- *imx6sxscm-1gb-evb* (EVB for SCM-i.MX 6SX with 1 GB LPDDR2 PoP)

3.4. Choosing a graphical back end

Before the setup, choose a graphical back end. The default is X11.

Choose one of these graphical back ends:

- X11
- Wayland: using the Weston compositor
- XWayland
- FrameBuffer

Specify the machine configuration for each graphical back end.

The following are examples of building the Yocto image for each back end using the QWKS board for SCM-i.MX 6D/Q and the evaluation board for SCM-i.MX 6SoloX. Do not forget to replace the machine configuration with what matches your reference board.

3.4.1. X11 image on QWKS board Rev C for SCM-i.MX 6D/Q

```
$ DISTRO=fsl-imx-x11 imx6dqscm-lgb-qwks-rev3 source fsl-setup-release.sh -b build-x11
$ bitbake fsl-image-gui
```

3.4.2. FrameBuffer image on evaluation board for SCM-i.MX 6SX

```
$ DISTRO=fsl-imx-fb MACHINE=imx6sxscm-lgb-evb source fsl-setup-release.sh -b build-fb-
evb_6sxscm
$ bitbake fsl-image-qt5
```

3.4.3. XWayland image on QWKS board for SCM-i.MX 6D/Q

```
$ DISTRO=fsl-imx-xwayland MACHINE=imx6dqscm-lgb-qwks source fsl-setup-release.sh -b build-
xwayland
$ bitbake fsl-image-gui
```

3.4.4. Wayland image on QWKS board for SCM-i.MX 6D/Q

```
$ DISTRO=fsl-imx-wayland MACHINE=imx6dqscm-lgb-qwks source fsl-setup-release.sh -b build-
wayland
$ bitbake fsl-image-qt5
```

The *fsl-setup-release* script installs the *meta-fsl-bsp-release* layer and configures the `DISTRO_FEATURES` required to choose the graphical back end. The `-b` parameter specifies the build directory target. In this build directory, the *conf* directory that contains the *local.conf* file is created from the setup where the `MACHINE` and `DISTRO_FEATURES` are set. The *meta-fslbsp-release* layer is added into the *bblayer.conf* file in the *conf* directory under the build directory specified by the `-e` parameter.

4. Deploying the image

After the build is complete, the created image resides in the `<build directory>/tmp/deploy/images` directory. The image is (for the most part) specific to the machine set in the environment setup. Each image build creates the U-Boot, kernel, and image type based on the `IMAGE_FSTYPES` defined in the machine configuration file. Most machine configurations provide the SD card image (*.sdcard*), *ext4*, and *tar.bz2*. The *ext4* is the root file system only. The *.sdcard* image contains the U-Boot, kernel, and *rootfs*, completely set up for use on an SD card.

4.1. Flashing the SD card image

The SD card image provides the full system to boot with the U-Boot and kernel. To flash the SD card image, run this command:

```
$ sudo dd if=<image name>.sdcard of=/dev/sd<partition> bs=1M && sync
```

For more information about flashing, see “Preparing an SD/MMC Card to Boot” in the *i.MX Linux User's Guide* (document [IMXLUG](#)).

4.2. MFGTool (Manufacturing Tool)

MFGTool is one of the ways to place the image on a device. To download the manufacturing tool for the SCM-i.MX 6D/Q and for details on how to use it, download the SCM-i.MX 6 Manufacturing Toolkit for Linux 4.1.15-2.0.0 under the "Downloads" tab from www.nxp.com/qwks-scm-imx6dq. Similarly, download the manufacturing tool for the SCM-i.MX 6SoloX evaluation board under the "Downloads" tab from www.nxp.com/evb-scm-imx6sx.

5. Specifying displays

Specify the display information on the Linux OS boot command line. It is not dependent on the source of the Linux OS image. If nothing is specified for the display, the settings in the device tree are used. Find the specific parameters in the *i.MX 6 Release Notes L4.1.15-2.0.0* (available at [L4.1.15-2.0.0 LINUX DOCS](#)). The examples are shown in the following subsections. Interrupt the *auto-boot* and enter the following commands.

5.1.1. Display options for QWKS board for SCM-i.MX 6D/Q

HDMI display

```
U-Boot > setenv mmcargs 'setenv bootargs console=${console},${baudrate} ${smp}
root=${mmcroot} video=mxcfb0:dev=hdmi,1920x1080M@60,if=RGB24'
U-Boot > run bootcmd
```

5.1.2. Display options for EVB for SCM-i.MX 6SX

Note that the SCM-i.MX 6SX EVB supports HDMI with a HDMI accessory card (MCIMXHDMICARD) that plugs into the LCD connector on the EVB.

Accessory boards:

- The LVDS connector pairs with the NXP MCIMX-LVDS1 LCD display board.
- The LCD expansion connector (parallel, 24-bit) pairs with the NXP MCIMXHDMICARD adapter board.

LVDS display

```
U-Boot > setenv mmcargs 'setenv bootargs console=${console},${baudrate} ${smp}
root=${mmcroot} ${dmfc} video=mxcfb0:dev=ldb,1024x768M@60,if=RGB666 ldb=sep0'
U-Boot > run bootcmd
```

HDMI display (dual display for the HDMI as primary and the LVDS as secondary)

```
U-Boot > setenv mmcargs 'setenv bootargs console=${console},${baudrate} ${smp}
root=${mmcroot} video=mxcfb0:dev=hdmi,1920x1080M@60,if=RGB24
video=mxcfb1:dev=ldb,LDBXGA,if=RGB666'
U-Boot > run bootcmd
```

LCD display

```
u-boot > setenv mmcargs 'setenv bootargs ${bootargs}
root=${mmcroot} rootwait rw video=mxcfb0:dev=lcd,if=RGB565'
u-boot> run bootcmd
```

6. Reset and boot switch configuration

6.1. Boot switch settings for QWKS SCM-i.MX 6D/Q

There are two push-button switches on the QWKS-SCMIMX6DQ board. SW1 (SW3 for QWKS board Rev B) is the system reset that resets the PMIC. SW2 is the i.MX 6Dual/6Quad on/off button that is needed for Android.

There are three boot options. The board can boot either from the internal SPI-NOR flash inside the SCM-i.MX6Dual/6Quad or from either of the two SD card slots. The following table shows the switch settings for the boot options.

Table 1. Boot configuration switch settings

Boot from top SD slot (SD3)

Switch	D1	D2	D3	D4	D5	D6	D7	D8
SW3 ¹	ON	ON	OFF	OFF	ON	OFF	OFF	ON

Boot from bottom SD slot (SD2)

Switch	D1	D2	D3	D4	D5	D6	D7	D8
SW3 ¹	ON	ON	OFF	ON	OFF	OFF	OFF	ON

Boot from internal SPI NOR

Switch	D1	D2	D3	D4	D5	D6	D7	D8
SW3 ¹	ON	ON	OFF	X	X	ON	ON	OFF

Default

Switch	D1	D2	D3	D4	D5	D6	D7	D8
SW3 ¹	ON	ON	OFF	OFF	ON	OFF	OFF	ON

1. SW1 is for QWKS board Rev B and SW3 is for QWKS board Rev C

Table 2. DDR memory map configuration settings

Dip Switch #	2	3
Single DDR channel	OFF	OFF
4KB interleaving mode	OFF	ON
Fixed 2x32 map	ON	OFF
Default (Fixed mode)	ON	OFF

6.2. Boot switch settings for EVB SCM-i.MX 6SoloX

This table shows the jumper configuration to boot the evaluation board from the SD card slot SD3.

SW2							
D1	D2	D3	D4	D5	D6	D7	D8
Off							
SW3							
D1	D2	D3	D4	D5	D6	D7	D8
Off	Off	On	On	Off	Off	Off	Off
SW4							
D1	D2	D3	D4	D5	D6	D7	D8
Off	On	Off	Off	Off	Off	On	Off

7. SCM uboot and kernel repositories

The kernel and uboot patches for both SCM-i.MX 6 QWKS board and evaluation board are integrated in specific git repositories. Below are the git repos for SCM-i.MX 6 uboot and kernel:

uBoot repo: <http://git.freescale.com/git/cgit.cgi/imx/uboot-imx.git>

SCM Branch: scm-imx_v2016.03_4.1.15_2.0.0_ga

kernel repo: <http://git.freescale.com/git/cgit.cgi/imx/linux-imx.git>

SCM branch: scm-imx_4.1.15_2.0.0_ga

8. References

1. For details about setting up the Host and Yocto Project, see the *NXP Yocto Project User's Guide* (document [IMXLXOCTOUG](#)).
2. For information about downloading images using U-Boot, see “Downloading images using U-Boot” in the *i.MX Linux User's Guide* (document [IMXLUG](#)).
3. For information about setting up the SD/MMC card, see “Preparing an SD/MMC card to boot” in the *i.MX Linux User's Guide* (document [IMXLUG](#)).

9. Revision history

Table 3. Revision history

Revision number	Date	Substantive changes
L3.14.52_1.1.0-ga	02/2017	Initial release
L4.1.15-2.0.0_ga	04/2017	Document type changed to "User's Guide" Document title updated Release number changed to L4.1.15-2.0.0 throughout the document Sections 3.2, 3.2.1, and 3.2.2 updated New bullet added in section 3.3.1 Headings 3.4.1, 3.4.2, 3.4.3 and 3.4.4 updated Section 4.2 updated Table 1: Switch numbering changed, footnote added Section 7: removed completely and replaced with new content

How to Reach Us:

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address:

nxp.com/SalesTermsandConditions.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, Freescale, and the Freescale logo are trademarks of NXP B.V. All other product or service names are the property of their respective owners.

ARM, the ARM Powered logo, and Cortex are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved.

© 2017 NXP B.V.

Document Number: SCMIMX6LRNUG

Rev. L4.1.15-2.0.0-ga

04/2017

