## Yocto build bootloader without spl

Software environment: L5.4.47\_2.2.0 Hardware i.MX8QXPC0 EVK board

To boot a Linux image on i.MX 8QuadMax, i.MX 8QuadXPlus, i.MX 8DXL, and i.MX 8DualX, four elements are needed:

• Bootloader (imx-boot built by imx-mkimage, which is a tool that combines firmware and U-Boot to create a bootloader for i.MX8), which includes U-Boot, Arm Trusted Firmware, DCD file, System controller firmware, and the SECO firmware since B0.

• Arm Cortex-M4 image

• Linux kernel image (Image built by linux-imx)

• A device tree file (.dtb) for the board being used

• A root file system (rootfs) for the particular Linux image

On i.MX 8, the U-Boot cannot boot the device by itself. The i.MX 8 pre-built images or Yocto Project default bootloader is imx-boot for the SD card, which is created by the imx-mkimage. The imx-boot binary includes the U-boot, Arm trusted firmware, DCD file

(8QuadMax/8QuadXPlus/8DXL), system controller firmware (8QuadMax/8QuadXPlus/8DXL), SPL (8M SoC), DDR firmware(8M), HDMI firmware (8M Quad), and SECO firmware for B0 (8QuadMax/8QuadXPlus/8DXL).

In the uuu script we can see the bootloader imx-boot-imx8qxpc0mek-sd.bin-flash is necessary. The default BSP build generate in the yocto project is with the spl, some customer are confused about the how to build the imx-boot-imx8qxpc0mek-sd.bin-flash.

## There is one manually compile way in our i.MX Linux User's Guide:

For the i.mx8qx the bootloader contains atf and u-boot, using the mkimage you can get the flash.bin file. The flash.bin it's the imx-boot-imx8qxpc0mek-sd.bin-flash:

Generate the u-boot.bin first, go to uboot and use

- \$ make imx8qxp\_mek\_defconfig
- \$ make

For i.MX 8QuadXPlus, to build imx-boot image by using imx-mkimage, perform the following steps:

- 1. Copy u-boot.bin from u-boot/u-boot.bin to imx-mkimage/iMX8QX/.
- 2. Copy scfw\_tcm.bin from SCFW porting kit to imx-mkimage/iMX8QX/.
- 3. Copy bl31.bin from Arm Trusted Firmware (imx-atf) to imx-mkimage/iMX8QX/.
- 4. Copy the SECO firmware container image (ahab-container.img) to imx-mkimage/iMX8QX/.
- 5. Run make SOC=iMX8QX flash to generate flash.bin.
- 6. If using OP-TEE, copy tee.bin to imx-mkimage/iMX8QX/ and copy u-boot/spl/u-boot-spl.bin to imx-mkimage/iMX8QX/.

Run make SOC=iMX8QX flash\_spl to generate flash.bin. (This is with spl manually generate)

## But how to modify the yocto config file to build the \*flash with bitbake?

In the conf/loca.conf add MACHINE\_FEATURES\_remove\_imx8qxpc0mek = "optee"

AutoSave 💽 🛱 🏷 🤊 🔻	Yocto build bootloader without spl	₽ Search		Rita Wang 🔗	Ŧ	-	0	×
Open 🔻 🖪	~/ir	*local.conf mx-yocto-bsp/build-xwayland/conf		Save	≡	-		×
MACHINE ??= 'imx8qxpc0mek'								
MACHINE_FEATURES_remove_imx8qx								
DISTRO ?= 'fsl-imx-xwayland'PACKAGE_CLASSES ?= 'package_rpm'								
EXTRA_IMAGE_FEATURES ?= "debug-tweaks"								
USER CLASSES ?= "buildstats image-mklibs image-prelink"								
PATCHRESOLVE = "noop" BØ DISKMON DIRS ??= "\								
STOPTASKS, \${TMPDIR}, 1G, 100								
STOPTASKS, \${DL DIR}, 16, 100								
STOPTASKS, \${SSTATE DIR}, 10								
STOPTASKS,/tmp,100M,100K \								
ABORT, \${TMPDIR}, 100M, 1K \								
ABORT, \${DL_DIR}, 100M, 1K \								
ABORT, \${SSTATE_DIR}, 100M, 1	ιK \							
ABORT,/tmp,10M,1K"								
PACKAGECONFIG_append_pn-qemu-s	ystem-native = " sdl"							
CONF_VERSION = "1"								
DL DIR ?= "\${BSPDIR}/downloads	s/"							
ACCEPT FSL EULA = "1"	18							
<pre># Switch to Debian packaging a PACKAGE_CLASSES = "package_deb</pre>	and include package-management i )"	n the image						
EXTRA IMAGE FEATURES += "packa	age-management"							

After finish compile you can see the imx-boot-imx8qxpc0mek-sd.bin-flash under the deploy file

lrwxrwxrwx 2 nxa07019 nxp	34 Jun 17 15:43 imx-boot -> imx-boot-imx8qxpc0mek-sd.bin-flash
-rw-rr 2 nxa07019 nxp	1200128 Jun 17 15:43 imx-boot-imx8qxpc0mek-sd.bin-flash
-rw-rr 2 nxa07019 nxp	3398656 Dec 29 13:50 imx-boot-imx8qxpc0mek-sd.bin-flash_linux_m4
-rw-rr 2 nxa07019 nxp	1243136 Jun 17 15:43 imx-boot-imx8qxpc0mek-sd.bin-flash_regression_linux_m4
-rw-rr 2 nxa07019 nxp	3355648 Dec 29 13:50 imx-boot-imx8qxpc0mek-sd.bin-flash_spl

But in this method we will disable OPTEE feature, here if we do not want to disable the OPTEE, we can modify the configuration for the IMXBOOT\_TARGETS\_imx8qxpc0mek in the meta-imx/meta-bsp/conf/layer.conf.

IMXBOOT TARGETS SD imx8qxpc0mek = \

"<u>\${@bb.utils.contains('MACHINE\_FEATURES'</u>, 'optee', 'flash', \

'flash flash\_regression\_linux\_m4', d)}"

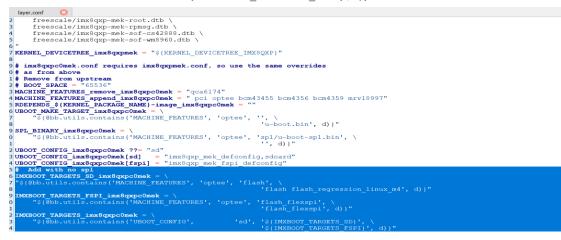
IMXBOOT\_TARGETS\_FSPI\_imx8qxpc0mek = \

"<u>\${@bb.utils.contains('MACHINE\_FEATURES'</u>, 'optee', 'flash\_flexspi', \

'flash\_flexspi', d)}"

"<u>\${@bb.utils.contains('UBOOT\_CONFIG'</u>, 'sd', '\${IMXBOOT\_TARGETS\_SD}', \

'\${IMXBOOT\_TARGETS\_FSPI}', d)}"



When fished compile we can see the imx-boot-imx8qxpc0mek-sd.bin-flash generate under the deploy directory.

lrwxrwxrwx 2 nxa07019 nxp	34 Jun 17 16:49 imx-boot -> imx-boot-imx8qxpc0mek-sd.bin-flash
-rw-rr 2 nxa07019 nxp	1200128 Jun 17 16:49 imx-boot-imx8qxpc0mek-sd.bin-flash
-rw-rr 2 nxa07019 nxp	3398656 Jun 17 16:01 imx-boot-imx8qxpc0mek-sd.bin-flash_linux_m4
-rw-rr 2 nxa07019 nxp	1243136 Jun 17 15:43 imx-boot-imx8qxpc0mek-sd.bin-flash_regression_linux_m4
-rw-rr 2 nxa07019 nxp	3355648 Jun 17 16:01 imx-boot-imx8qxpc0mek-sd.bin-flash_spl

In the yocto generate it is more convenient than the manually compile way.

Hope this can do help for you.