Embedded Systems
iMX6 Factory Programming and Software Upgrade Strategy

001-011-081 Rev A

Executive Summary

This document provides a high-level description of the system configuration and install/update process for on-board iMX6 flash.

Table of Contents

1. Partition Table

1.1. Bootloader Partition

The EMMC Flash has a bootloader partition built in. This partition is independent of the device's primary flash space. The U-Boot binary can be burned directly into this section.

1.2. Boot Partition

This FAT formatted partition contains the Linux kernel binary (zImage), device tree binary (dtb), and boot script (boot.scr).

1.3. RootFS Partitions 1 and 2

These ext4 formatted partitions are used for the root file system. However, only one is in use at a time; the second is used during the software upgrade process.

1.4. NVM Partition

This ext4 formatted partition is used to store logs and settings that will persist through a software update.

2. Factory Programming

2.1. MFGTOOL

The iMX6 MFG Tool utilizes a USB-OTG interface to communicate with the iMX6 serial bootloader ROM. It is used to upload a Linux kernel into RAM, which in turn is used to configure the flash partitions and install the initial BSP.

2.2. Workflow

- Boot board to MFGTOOL Linux image
- Create four partitions
- Copy U-Boot binary, Linux binary and device tree, rootfs tarball, and boot script to nvm partition
- Write the U-Boot binary to the bootloader partition
- Copy zImage, dtb, and boot.scr to the boot partition
- Extract the rootfs tarball to the rootfs1 partition
- Install serial number(s) and MAC address(es)
- Remove files from nvm partition

3. Software Upgrade

3.1. Signed Images

System images are deployed as a tarball signed with a 4096-bit public/private key pair. By deploying the units with a public key, we can ensure only software signed by our private key can be installed by the application.

Installation methods are numerous and application specific; any physical mechanism that can be used to transfer files to the system can be used to install updates. Syncroness will work with our customers to ensure the application update process fits their use case.

3.2. Upgrade Process

- Verify image signature
- Extract new rootfs to unused rootfs partition
- Modify boot.scr to boot with the alternate rootfs
- Replace the kernel and device tree binary (if included)
- Replace the bootloader (if necessary)
- Reboot the system