

THE **LINUX** FOUNDATION COLLABORATIVE PROJECTS[Developer Login](#) | [Help](#)

Search site...

[HOME](#)[ABOUT](#) ▼[PROJECTS](#) ▼[SECURITY ADVISORIES](#) ▼

(L)ittle (K)ernel based Android bootloader

First of all thanks to Travis Geiselbrecht, Brian Swetland and Dima Zavin for the initial work on (L)ittle (K)ernel project and open-sourcing the work on kernel.org <<http://kernel.org> (<http://kernel.org/>)>.

Our current version of Android apps processor bootloader is based on this LK work and the Android team at Qualcomm Innovation Center, Inc (QuIC) has added and opensourced features to the project.

We currently support variety of Qualcomm chipsets, including MSM7X25, MSM7X27, MSM7X30 and QSD8X50. The bootloader does the basic task of hardware initialization, reading the Linux kernel & ramdisk from nand device and loading it up to RAM, setting up initial registers and command line arguments for Linux kernel and jumps to the kernel.

In context of support for Qualcomm chipsets and Android, the bootloader currently supports:

- * Variety of nand devices for bootup

- * USB driver to enable upgrading images over usb during development
- * Keypad driver to enable developers enter 'fastboot' mode for image upgrades
- * Display driver for debugging and splash screen
- * Enable Android recovery image and image upgrades

Here is a list of FAQs that we get from OEMs to customize the bootloader for their projects:

Q1) How do I customize physical memory for use by Linux kernel?

A1) We currently have LK project is mapped to bootable/bootloader/lk under the Android tree. The memory tags can be customized in lk/target/<target_name>/atags.c. For example lk/target/msm7630_surf/atags.c has target_atag_mem which has this information for a 256M device. This can be modified to add another tag if OEM device supports additional memory.

Q2) How do I disable 'fastboot' for my commercial device?

A2) Personally, I would like to see commercial devices leave 'fastboot' enabled. It goes in the spirit of Android Open Source Project where end users and developers have access to the source code to allow them to modify it and upgrade their device over usb using 'fastboot'. However, if OEMs or network providers want to lock down 'fastboot' it can be done by disabling the keypad and usb driver in bootloader. This can be customized in the following files:

lk/app/aboot/fastboot.c

lk/app/aboot/aboot.c

Q3) Can we enable splash screen in the bootloader?

A3) In normal bootup, the task of the boot loader should be to perform hardware initializations and jump to Linux kernel as soon as possible. Enabling a display driver and splash screen does go against that goal. However, if an OEM wants to enable splash screen in bootloader, a display driver does exist in boot loader that can be used for splash screen. You will at least have to enable DISPLAY_SPLASH_SCREEN feature. During bootup, the bootloader will read the raw image from 'splash' partition and copy it over to framebuffer. This has been verified for MSM7X27 on LCDC display. OEMs can work through this implementation if they want to enable the same for their own devices.

Q4) How do I customize flash partitions for my device?

A4) The bootloader defines the nand flash partitions and passes them over to Linux kernel. These partitions are currently defined at board_part_list structure in lk/target/<target_name>/init.c. The 'userdata' is defined as a VARIABLE_LENGTH partition, which allows this partition to take up all remaining nand blocks available on the device. This and other partitions can be customized based on OEM's nand device.

Feel free to write if you have additional questions in this area. Also, if you have any MSM specific features or bug fixes, you are welcome to send over the patches so we can integrate it back to our tree.

Tags: [Android bootloader \(/tags/android-bootloader\)](/tags/android-bootloader)
[kernal.org \(/tags/kernalorg\)](/tags/kernalorg)
[Little Kernel \(/tags/little-kernel\)](/tags/little-kernel)
[open source \(/tags/open-source\)](/tags/open-source)
[qualcomm \(/tags/qualcomm\)](/tags/qualcomm)
[Qualcomm chipsets \(/tags/qualcomm-chipsets\)](/tags/qualcomm-chipsets)
[quic \(/tags/quic\)](/tags/quic)