linaro™

Arm Solutions at Lightspeed

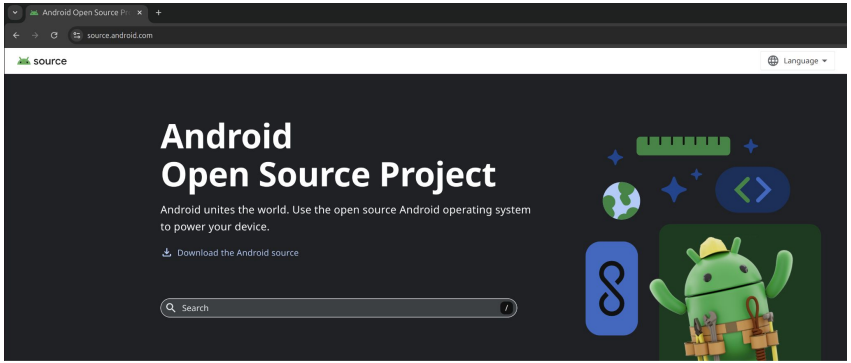
Keeping up with The Android Open Source Project (AOSP)

Amit Pundir, Sumit Semwal

aosp-devroom, FOSDEM'25

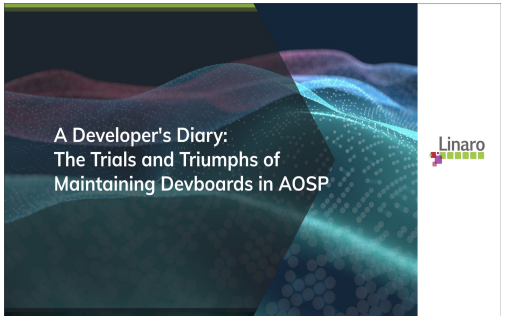
Agenda!

- What's new in the Android Open Source Project
- Keeping up with the AOSP development



Flashback: Linaro, Devboards and AOSP!

- Linaro has been a long-time advocate of AOSP development boards
  - Linaro Android team supports AOSP on a variety of member devboards, and performs extensive testing coverage
  - We have been talking about the importance of devboards in AOSP for years
    - At EOSS ‘23, Prague, we talked about the pain-points and benefits of maintaining devboards in AOSP and keeping up with the AOSP on devboards.



EOSS, Prague ‘23: Maintaining Devboards in AOSP, Linaro

- In this session we will look at some of the features landed in AOSP since then.

What's new in the AOSP?

- Generic Bootloader Library (GBL)
  - LPC: Android Generic Boot Loader
  - Generic Bootloader library

- Simplified booting from external storage (androidboot.boot_part_uuid)
  - Find the boot device by partition UUID

What's new in the AOSP?

- Page-agnostic builds (4k, 16K, ..)
  - Google dev blog: Adding 16 KB Page Size to Android
  - Results in 5-10% performance boost at about 9% space cost

- ACK DDK
  - LPC: Simplified Driver Development with DDK v2
  - The DDK helps Vendors develop GKI modules for seamless integration with the GKI build.
    - Ensures correctness. I.e. correct toolchain use (compilers, linkers, flags, etc.)
    - correct visibility of resources provided by the GKI kernels (such as headers, Makefiles).

What's new in the AOSP?

- GKI Protected and Unprotected Kernel Modules and DLKM partitions
  - Protected GKI modules
    - Modules signed using build time key pair
    - Can't be overridden by Vendor/unsigned modules
  - Unprotected GKI modules
    - GKI kernel modules which can be replaced by Vendor's driver with upstream backported features to let them incorporate new upstream features after the KMI freeze.
    - Treated just like Vendor kernel modules and are restricted to KMI symbols.
  - How to implement a GKI module partition

- Parallel kernel module loading (androidboot.load_modules_parallel)
  - How to enable asynchronous probing

What's new in the AOSP?

- ashmem-compat patches
  - Introduce ashmem compatability layer for memfd, allowing applications that depend on ashmem's ioctls to work seamlessly with memfd fds.

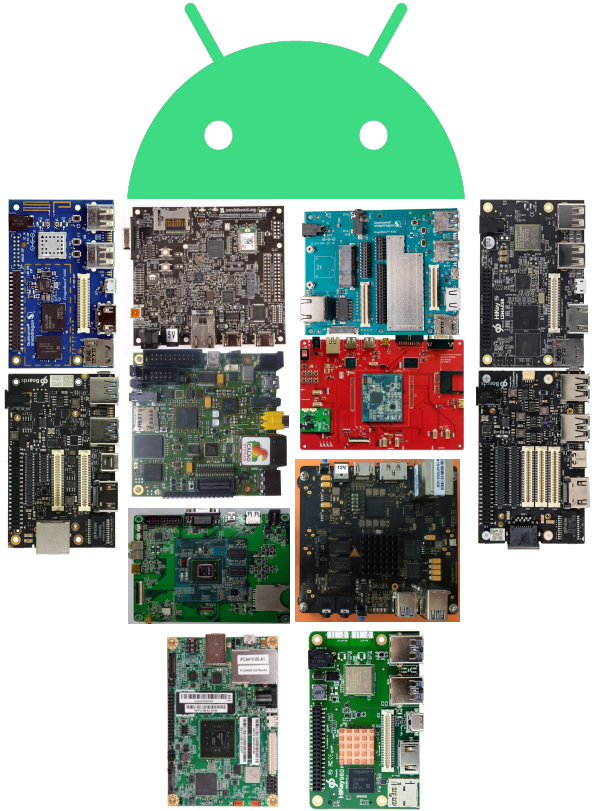
- GKI 16-6.12 errata
  - https://source.android.com/docs/core/architecture/kernel/android16-6.12-errata

What's new in the AOSP?

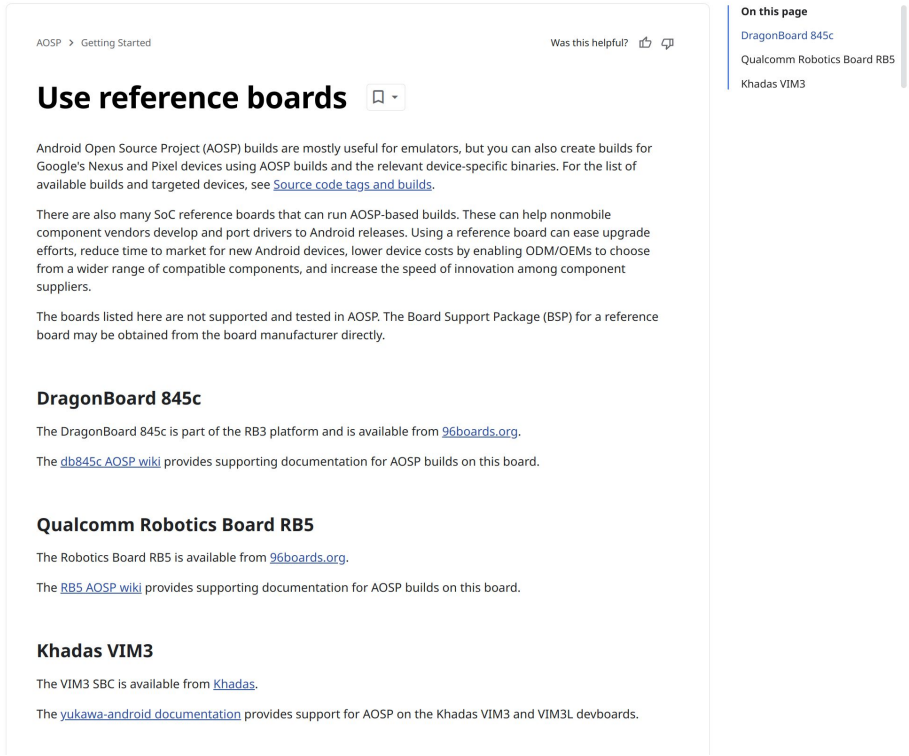
- AVF and pKVM
  - AVF overview
  - SMMU v3 support in pKVM
  - Various AVF how-tos around custom VMs, updatable VMs, Device assignment
- Trunk stable

Keeping up with the AOSP!

- Maintaining development devices in the AOSP repo
- Maintaining AOSP on the development devices



Maintaining development devices in the AOSP!



AOSP > Getting Started

Was this helpful?

Use reference boards

Android Open Source Project (AOSP) builds are mostly useful for emulators, but you can also create builds for Google's Nexus and Pixel devices using AOSP builds and the relevant device-specific binaries. For the list of available builds and targeted devices, see Source code tags and builds.

There are also many SoC reference boards that can run AOSP-based builds. These can help nonmobile component vendors develop and port drivers to Android releases. Using a reference board can ease upgrade efforts, reduce time to market for new Android devices, lower device costs by enabling ODM/OEMs to choose from a wider range of compatible components, and increase the speed of innovation among component suppliers.

The boards listed here are not supported and tested in AOSP. The Board Support Package (BSP) for a reference board may be obtained from the board manufacturer directly.

DragonBoard 845c

The DragonBoard 845c is part of the RB3 platform and is available from 96boards.org.

The db845c AOSP wiki provides supporting documentation for AOSP builds on this board.

Qualcomm Robotics Board RB5

The Robotics Board RB5 is available from 96boards.org.

The RB5 AOSP wiki provides supporting documentation for AOSP builds on this board.

Khadas VIM3

The VIM3 SBC is available from Khadas.

The yukawa-android documentation provides support for AOSP on the Khadas VIM3 and VIM3L devboards.

On this page

DragonBoard 845c

Qualcomm Robotics Board RB5

Khadas VIM3

Maintaining development devices in the AOSP!

- Keeping up with subtle AOSP breakages
  - Not a good time/day/week to sync AOSP sources
    - Transient build or runtime failures due to async project updates
    - Caught up in between cross framework changes
  - Core framework changes
    - Mostly HIDL to AIDL transition
    - Example: **TARGET_USES_VULKAN := true** breakage on some Vulkan drivers
  - Keeping up with .mk / .bp / .bazel build configurations
    - Keeping track of soong changes
    - Move to Bazel seem to be limited to GKI/GBL repo only

Maintaining development devices in the AOSP!

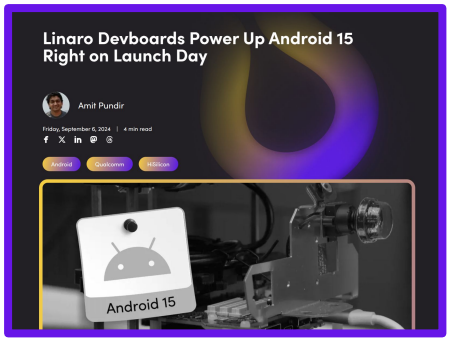
- Tracking upcoming features
  - Generic Bootloader Library (GBL)
    - Enabled GBL for one of our supported devices in AOSP
  - 16K page_size builds
    - Enabled 16K page_size aligned builds for dragonboards in AOSP
    - Workaround a bootloader crash by setting **INTERNAL_VENDOR_BOOTIMAGE_ARGS += --pagesize 4096**
  - Find the boot device by partition UUID
    - Simplifies booting AOSP by finding boot device based on a partition UUID (androidboot.boot_part_uuid) instead of a /sysfs path
    - UUID parsing is case sensitive, it needs to be set in the lower-case
  - Drop ashmem in favour of memfd (WIP)
    - Switch to memfd by setting sys.use_memfd property true
    - selinux=permissive because vendor_init is restricted to set the system property

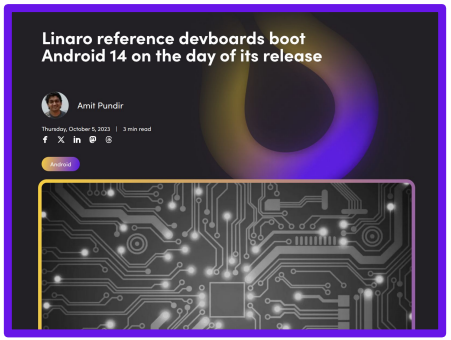
Maintaining development devices in the AOSP!

- Tracking relevant upstream projects (drm_hwcomposer, libdrm, linux, mesa3d, minigbm, u-boot, ...)
  - Occasional upstream Mesa build breakages
    - Recent build aosp/upstream-main build breakages due to gfxstream cross project changes
  - Linux breakages
    - Due to Android Common Kernel (GKI) changes
      - Tracking core ACK patches for breakages (ashmem, overlayfs, dm-default-key)
    - DT nodes / sysfs nodes (e.g. androidboot.boot_devices) do not count as stable interfaces
    - If it is not reproducible on Debian/Fedora/OE then is it even a valid bug?
    - Good luck with a firmware bug, worse if it is an OEM signed firmware
    - LTS breakages due to AUTOSEL cherry-picks

Maintaining development devices in the AOSP!

- One of the many benefits of well maintained devices in AOSP is to have a Day-0 boot of Android Release on those devices.





Et Al.

Maintaining AOSP on the development devices!

There are many projects that extend AOSP in various ways. Here we divide them into:

- Board support: the code required to allow Android to run on a given board or family of boards
- Generic HALs: hardware support that applies to multiple boards
- Android distros: modifications to the Android framework and applications

Board support

The Devboards for Android project (https://devboardsforandroid.linaro.org/en/latest/ adds support to upstream AOSP for several boards including:

- Qualcomm® Robotics RB5 Development Kit
- RB3 Kit for DragonBoard 845c
- Khadas VIM3 / VIM3L
- Snapdragon(tm) 8 Gen 2 Mobile Hardware Development Kit (HDK 8550)
- Hikey 960

BayLibre have support packages for

- MediaTek Genio 350-EVK: https://baylibre.pages.baylibre.com/mediatek/rita/device/mediatek/boards/mtk-android-14/index.html
- Texas Instruments AM62X and AM62PX: https://baylibre.pages.baylibre.com/ti/android/doc/ti-android-15/index.html

Other projects:

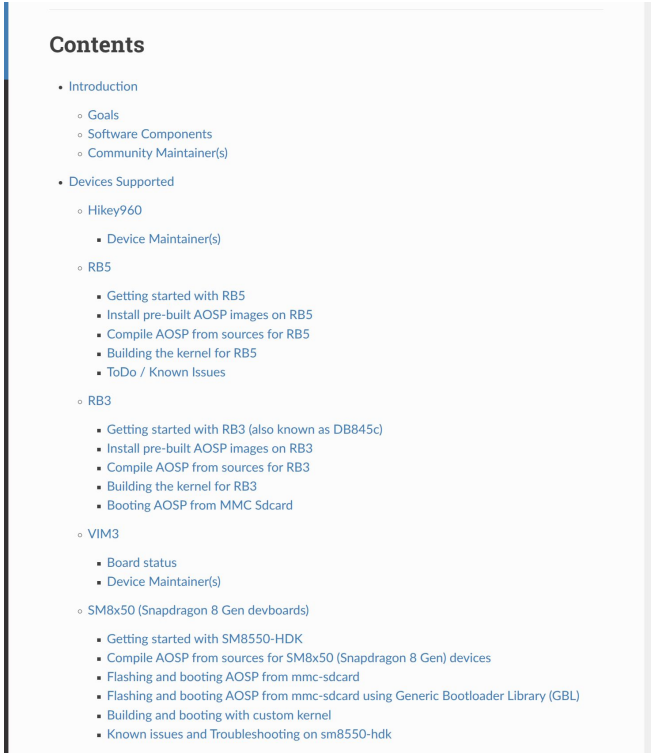
- Generic Qualcomm SDM845 device support: https://github.com/aospm/android_device_generic_sdm845
  - support for several handsets based on the SDM845

Maintaining AOSP on the development devices!

- Choosing Droid of your choice
  - Android release tag (-gsi branches) or aosp/main branch
- Custom manifest files / local_manifests
  - Custom sync rules saves time and download/disk space
- Localized changes not worrying about breaking other AOSP or Google's internal build targets
- Playground for experiments or proof-of-concepts

Maintaining AOSP on the development devices!

- Using project Devboards-For-Android (https://devboardsforandroid.linaro.org/) as the AOSP playground
  - Booting AOSP with vanilla kernel
  - Unified AOSP builds for a family of SoCs from the same vendor
  - Staging area for upstream focussed devices e.g. E850-96
  - Staging area for upstream focussed features or projects e.g. Thermal HAL, GBL integration etc
  - Software rendering builds using Virtual KMS driver



Contents

- Introduction
  - Goals
  - Software Components
  - Community Maintainer(s)
- Devices Supported
  - Hikey960
    - Device Maintainer(s)
  - RB5
    - Getting started with RB5
    - Install pre-built AOSP images on RB5
    - Compile AOSP from sources for RB5
    - Building the kernel for RB5
    - ToDo / Known Issues
  - RB3
    - Getting started with RB3 (also known as DB845c)
    - Install pre-built AOSP images on RB3
    - Compile AOSP from sources for RB3
    - Building the kernel for RB3
    - Booting AOSP from MMC Sdcard
  - VIM3
    - Board status
    - Device Maintainer(s)
  - SM8x50 (Snapdragon 8 Gen devboards)
    - Getting started with SM8550-HDK
    - Compile AOSP from sources for SM8x50 (Snapdragon 8 Gen) devices
    - Flashing and booting AOSP from mmc-sdcard
    - Flashing and booting AOSP from mmc-sdcard using Generic Bootloader Library (GBL)
    - Building and booting with custom kernel
    - Known issues and Troubleshooting on sm8550-hdk

Summary and Call for Participation

- Keeping up with the AOSP is hard. AOSP is constantly evolving, with new features, and integrating them with local or device specific customizations on your own can be challenging at times.
- Let's collaborate to engage on common software solutions.
- Together, we can improve the AOSP ecosystem

linaro™

Arm Solutions at Lightspeed

Questions,Concerns, Feedback?

Amit Pundir <amit.pundir@linaro.org>
Sumit Semwal <sumit.semwal@linaro.org>
IRC: pundir, sumits on #linaro-android, #aosp-developers @OFTC
aosp-devroom, FOSDEM 2025