AOSP Devboards

Linux Plumbers Android Microconf 2017

John Stultz & Dmitry Shmidt
Now there are two: https://source.android.com/source/devices

HiKey

HiKey960
Hardware overview

HiKey
- HiSilicon Kirin 620 SoC
- 64bit ARM A53
- 8 Cores
- 1 or 2 GB RAM
- 4 or 8 GB eMMC
- Mali 450 (Utgard) Graphics
- USB2 Type A & USB-OTG
- $100-120

HiKey960
- HiSilicon Kirin 960 SoC
- 64bit ARM 4xA53 + 4xA73
- 8 core big.LITTLE design
- 3 GB RAM
- 32 GB UFS Storage
- Mali G71 (Bifrost) Graphics
- PCIe
- USB3 Type A & USB-C (USB2)
- 5GHz WiFi & BT 4.1
- $240
Remaining out-of-tree patches (4.14-rc)

**HiKey**
- Minor USB improvements
- ION ABI revert HACK
- Mali gpu

**HiKey960**
- regulators
- ufs
- mailbox
- cpufreq/thermal
- usb
- iommu
- drm driver
- dts
- Mali gpu
- ...

---

**Linaro | 96 Boards**
96Boards.org
Vendor neutral hardware standard for developer boards

Allows ecosystem of peripherals that can be shared
96Boards.org
Vendor neutral hardware standard for developer boards

Allows ecosystem of peripherals that can be shared
Neonkey

Neonkey is certified 96Boards mezzanine: [https://source.android.com/source/devices#neonkey](https://source.android.com/source/devices#neonkey)

- CPU: STM32F11CE
- NOR Flash: 512 KB
- SRAM: 128 KB
- Pressure sensor: BMP280
- ARM Hall sensor: MRMS501A
- Temp/Humidity sensor: SI7034-A10
- ALS/Proximity sensor: RPR-0521RS
- LED driver with 15LEDs: LP3943
- Accel/Gyro+Geomagnetic sensors: BMI160+BMM150
- 4 GPIO-driver LEDs
- I2C expansion
- GPIO (2 lines) expansion
- JTAG connector
- 96Boards LS Expansion connector
New Phones, Old Kernels

- v3.10 Released June 2013
- v3.18 Released Dec 2014
- v4.4 Released Jan 2016
- v4.9 Released Dec 2016

2015 Nexus devices (Oct 2015)

2 years, 4 months
New Phones, Old Kernels

- v3.10 Released June 2013
- v3.18 Released Dec 2014
- v4.4 Released Jan 2016
- v4.9 Released Dec 2016

2015 Nexus devices (Oct 2015)

2016 Pixel Phone (Oct 2016)

2 years, 4 months
1 year, 10 months
New Phones, Old Kernels

v3.10 Released
June 2013

v3.18 Released
Dec 2014

v4.4 Released
Jan 2016

v4.9 Released
Dec 2016

2015 Nexus devices
(Oct 2015)

2016 Pixel Phone
(Oct 2016)

Early 2016 Flagship
(Mar 2016)

Early 2017 Flagship
(April 2017)

New Phones, Old Kernels

2 years, 4 months

1 year, 10 months

1 year, 3 months

1 year, 4 months
A Year Ahead

- v3.10 Released June 2013
- v3.18 Released Dec 2014
- v4.4 Released Jan 2016
- v4.9 Released Dec 2016

- HiKey on 4.4 (Apr 2016)
- 2015 Nexus devices (Oct 2015)
- 2016 Pixel Phone (Oct 2016)
- Early 2016 Flagship (Mar 2016)
- Early 2017 Flagship (April 2017)
A Year Ahead

v3.10 Released
June 2013

v3.18 Released
Dec 2014

2 years, 4 months

v4.4 Released
Jan 2016

1 year, 10 months

Early 2016 Flagship
(Mar 2016)

HiKey on 4.4
(Apr 2016)

1 year, 3 months

2015 Nexus devices
(Oct 2015)

HiKey on 4.9
(Feb 2017)

Early 2017 Flagship
(April 2017)

v4.9 Released
Dec 2016

1 year, 4 months

2016 Pixel Phone
(Oct 2016)
Kernel/Device Release Latencies

- 2015 Nexus devices (Oct 2015): 2 years, 4 months
- 2016 Pixel Phone (Oct 2016): 1 year, 10 months
- Early 2016 Flagship (Mar 2016): 1 year, 3 months
- Early 2017 Flagship (April 2017): 1 year, 4 months

- HiKey on 4.4 (April 2016): 3 months
- HiKey on 4.9 (Feb 2017): 2 months

- v3.10 (Jun 2013)
- v3.18 (Dec 2014)
- v4.4 (Jan 2016)
- v4.9 (Dec 2016)
Kernel Release Latencies

- **2015 Nexus devices (Oct 2015)**: 2 years, 4 months
- **2016 Pixel Phone (Oct 2016)**: 1 year, 10 months
- **Early 2016 Flagship (Mar 2016)**: 1 year, 3 months
- **Early 2017 Flagship (April 2017)**: 1 year, 4 months
  - HiKey on 4.4 (April 2016)
  - HiKey960 on 4.4 (April 2017)
- **HiKey on 4.9 (Feb 2017)**
- **HiKey960 on 4.9 (Aug 2017)**

Kernel versions:
- v3.10 (Jun 2013)
- v3.18 (Dec 2014)
- v4.4 (Jan 2016)
- v4.9 (Dec 2016)
Benefits to Android Devs

- Inexpensive Android development board
- Amount of board-specific patches is small
- Helps development of new Kernel features
  - Android Verity
  - Hibernation
  - Sdcardfs
  - SELinux
- Validation of Kernel patches / -stable updates
  - Wide range of supported kernels
  - Example: 3.18, (4.1), 4.4, 4.9
- Development of Android runtime libraries
  - Wide range of supported kernels
Benefits to Android Devs (cont)

- Development of new Bootloader features
  - UEFI bootloader is provided in source
  - Boot to RAM
- Development and Support of new Hardware modules
  - Display boards
  - Camera modules
  - Sensor modules (e.g., Neonkey sensorhub)
  - Connectivity modules
- Use of overlay manager to support various hardware configurations
- Testing
  - Manual
  - Automation
Upstream Testing

Have a AOSP target that can run mainline kernels.

- LKFT efforts (previously covered)
- Weekly manual testing every mainline -rc release w/ HiKey
- Manually testing mainline ~daily during merge window
  - Small enough patchset that rebase-bisection isn’t bad

Found and reported numerous regressions against upstream (SELinux, drm, binder, ipv6 networking, etc)

Also reported a number of issues that AOSP needs to address to work with upstream kernels (New SELinux policies, dma_buf fences, ION destaging ABI break)
Recent efforts

Bring-up and validation of android-4.9 kernel tree
HiKey960 migration to v4.9
Transition to Generic Linux Bluetooth HAL (hikey & hikey960)
Parameterized EAS PowerHAL to make it shared/reusable
Migration to new mali driver, which works w/ dma-buf fence SYNC_FILE interface
Gralloc binderization & migration to opensource gralloc on hikey960
Migration to Android O
INPROGRESS: libION/Gralloc support for old and new (4.12+) ION ABIs
INPROGRESS: Hibernation proof-of-concept support on HiKey
Direction forward

Utilize HiKey as arm64 target for lots of upstream and LTS testing
HiKey960 upstreaming
UEFI transition(hikey960), updates & improvements (A/B, ?)
drm_hwcomposer
android-4.14 bringup and validation
Proving ground for upstream EAS
Improved treble implementation
Upstream DT fragments/overlay solution
When dev. board becomes more useful for community, more people will use it and more people will fix issues. To do this we need to add more features.

Adding more features will add more problems making the board less popular and therefore less useful.

We want right circle to “spin” quicker than left one.

Dialectic law: transition from quantity to quality.
AOSP pain points

Devboards aren’t really mobile devices (no panel, touch, baseband, etc)
Devboards aren’t “product” level targets internally, so AOSP changes break them fairly often.

Support Effort/Load (bugs/feature requests):
- HiKey: ~240 over 2 years
- HiKey960: ~ 60 over 0.5 years
- ~1 issue every 3 days

Mali blobs make us dependent on ARM for continued support of hardware
No real space for developing shared/common HALs (cross-project dependencies avoided)
Very difficult process to add/update project trees
Community pain points

Support multiple older kernels (v3.18, v4.4, v4.9, v4.14)
Difficulty to develop clean solutions that cross all those points.
Example: drm_hwcomposer HWC2 implementation needs changes in v4.10+
  Still need something for v3.18, v4.4 kernels

Android Sync/dma-buf fence transition, ION ABI changes, etc have been difficult
Would be nice if upstream efforts provided better migration paths
Other Work

Generic Build
- Kconfig based
- Supports multiple devices/architectures from single build system
  (db410c, db820c, nexus7, qemu-x86_64, qemu-aarch64, raspib3, etc)
- Need more community involvement

Work on open-graphics stack
  (mesa3d, gbm_gralloc, drm_hwcomposer, libdrm)
Questions/Discussion

Anyone outside of Linaro/Google finding this helpful/useful?
Price vs Power preference?
Upstreaming board support takes a frustratingly long time!
Multiple boards (and supporting multiple kernels) is hard to scale!

Still not getting to the point where we’re really pushing some of the advanced Android functionality upstream (offloaded sensor collection, wifi/bt scanning, etc).

Standard Kernel interfaces for Android are becoming a smaller issue than standard bootloader interfaces for Android (Reboot reason, dtb handling, fastboot flashing, secure boot, A/B updates, etc). Some work going on in UEFI, but with limited resources.
Thank you!

Questions: John Stultz <john.stultz@linaro.org>