U-Boot for modern Qualcomm phones

About the lowest common denominator and empowering users

Caleb Connolly <caleb.connolly@linaro.org>
About me

● Caleb (they/them)
● FOSS enjoyer since 2018
● Kernel engineer @ Linaro (Qualcomm Ecosystem Team)
  ○ Happy to hack on pretty much anything (as long as it’s not userspace)
  ○ Especially partial to things that improve UX
● postmarketOS core team member
  ○ Maintain support for Snapdragon 845 devices
  ○ Plotting new ways to keep your embedded devices out of landfill
● Maintainer of Qualcomm platform support in U-Boot
● @cas@social.treehouse.systems
In this talk

1. Demonstrate why running Linux on Android phones suuuucks
2. The magic of UEFI
3. U-Boot as a UEFI bootloader
4. The “how” of U-Boot on Qualcomm devices
5. State of Qualcomm support in U-Boot
6. Demo
7. Upstreaming status
8. Supporting a new SoC
Let’s play... Odd one out

- systemd
- U-Boot
- LinuxBoot
Bootloader

The software(s) responsible for loading the kernel, initramfs, and devicetree, and then jumping to the kernel.
Booting on Android (Qualcomm edition)

ABL
- Load boot.img
- Find matching dtb
- Load dtbo.img
- Apply overlay
- Jump to kernel

"Features"
- No multiboot
- Only custom A/B rollback support
- Heavy integration cost for distros
- No post-boot services
Booting on Android (Qualcomm edition)

**boot.img**
- Android!
- vmlinux
- dtb
- dtb2
- initramfs
- 5 versions so far
- Different untested Caldera for each!

**dtbo.img**
- 1.dtbo
- 2.dtbo
- 3.dtbo
- Doesn't work on
- upstream DT
- Sometimes can be
- bypassed
- Sometimes need a custom empty
- DTBO image
- (Device specific)

**ABL**
- Load boot.img
- Find matching dtb
- Custom non-standard
- DT properties that many
- OEMs don't even follow

- No feedback on failure -
  just drops to fastboot
  unless UART

"Features"
- No multi-boot
- Only custom A/B rollback support
- Heavy integration cost for distros
- No post-boot services

- Will softbrick your
  device without userpace
  intervention
  no way to disable!
What's hiding in the shadows?
Booting with UEFI

- Already universally supported
- The distro has full control over the bootloader!
Booting with UEFI

- Many x86 vendors (and those doing UEFI in the Arm space) still f*ck it up
- Still some limitations (e.g. SetVariable() doesn’t work on devices that can’t dedicate SPI flash to UEFI)
- If done right, an obvious winner
Guess what bootloader Qualcomm ship on their automotive and IoT platforms
U-Boot

- A very cool, and GPL-2 licensed bootloader
- Supports many architectures and devices
- Linux driver compatibility
- Devicetree!!!
- Highly adaptable
- UEFI - passes SystemReady
- Not so great Qualcomm support
- ... Until now!
Booting U-Boot

- We can't replace the Android bootloader :
- ... most of the time anyway
- Code is hashed and the hash is signed by a private key owned by the vendor - the public key is burnt into the SoC at the factory.
- Exploits? Possibly...
- But we can chainload!
  - And glean a bunch of useful info from ABL
  - Build with CONFIG_LINUX_KERNEL_IMAGE_HEADER=y
Qualcomm support in U-Boot

- Provided in arch/arm/mach-snapdragon
  - MSM8916, MSM8996, SDM845, SM6115, QCM2290, SM8250, SC7280, SM8550 WIP
  - And more?
- Almost compatible with upstream DT
  - Deviations are tracked separately in $DEVICE-u-boot.dtsi include files
  - Runtime DT fixup for USB
- No board specific code for platforms added after 2017
  - Read memory map from DT
  - Dynamically allocate load buffers
  - One build target for all supported platforms and devices!
- Support for USB, UFS, and newer MMC is making its way upstream
- And much more!
  - Button support, capsule updates, etc.
IoT dev boards

- **RB1** (low-end - QCM2290)
  - 4 cores, 2GHz
  - 1/2GB RAM, 8/16GB eMMC
  - $199

- **RB2** (low/mid range - SM6115)
  - 8 cores, 2GHz
  - 2GB RAM, 16GB eMMC
  - $249

- **RB3** (mid range - SDM845)
  - 8 cores, up to 2.8GHz
  - 4GB RAM, 128GB UFS
  - $400 (no longer sold)

- **RB5** (high end - SM8250)
  - 8 cores, up to 2.84GHz
  - 8GB RAM
  - $545

- All capable of booting from internal storage or USB
- No secureboot!
- Capable of running U-Boot as the primary bootloader
- PoC highly promising
- More work needed in this area

https://git.codelinaro.org/linaro/qcomlt/u-boot/-/wikis/Building-U-Boot-for-RBx
Initial release for SDM845 phones

https://gitlab.com/sdm845-mainline/u-boot/-/releases/sdm845-phones-v0.1.0

sdm845-phones-v0.1.0

Assets
- Source code (zip)
- Source code (tar.gz)
- Source code (tar.bz2)
- Source code (tar)

Other
- u-boot-fajita.img
- u-boot-axolotl.img
- u-boot-enchilada.img
- u-boot-beryllium-tianma.img
- u-boot-beryllium-ebbq.img

Evidence collection
- sdm845-phones-v0.1.0-evidences-7444788.json

Collected 7 Jan 2024, 22:55

This initial release provides support for booting from internal UFS storage when a valid EFI System Partition is found (containing /EFI/BOOT/bootaa64.efi).

The volume and power buttons can be used to navigate menus (including systemd-boot and GRUB).

The U-Boot Console can be accessed by booting with a USB cable attached, interrupting autoboot with the power buttons, and choosing "Enable serial gadget console" from the menu, this will expose a serial device to your computer which can be accessed with any standard console program (picocom, minicom, etc).

791d4b2b

977b927 sdm845-phones-v0.1.0 Created 7 Jan 2024, 22:55 by
Demo!!

- [https://gitlab.com/postmarketOS/pmaports/-/merge_requests/4599](https://gitlab.com/postmarketOS/pmaports/-/merge_requests/4599)
Upstreaming status

- Initial cleanup mostly done
- Big refactor and migration to upstream DT on the lists
- USB support for SDM845 on the lists
- UFS in progress
- Separate effort by Sumit Garg to make all upstream DT available by importing devicetree-rebasing repo
Integrating U-Boot - future plans

- “Recent Advancements in U-Boot - Simon Glass” [1]
- Most “standard” devices should “just work” with Linux DT (assuming SoC support)
- Enable support for handling DTB variants (e.g. different display panels)
- Let’s make upstream Qualcomm defconfig friendly to phones!
  - Enable a mobile-friendly boot menu based on chassis type
  - Remap volume/power button keycodes to up/down/enter

[1]: https://www.youtube.com/watch?v=YlJBsVZJkDI
Adding a new SoC

- Clock and pinctrl drivers
  - Can be just stubs initially!
- UFS phy configuration data
- Compatible strings for PMIC, SMMU
- Congrats, you can probably boot from UFS now :D
Thank you

slides: calebs.dev/fosdem24.pdf
contact: caleb.connolly@linaro.org
website: connolly.tech/