From Android to mainline on the Snapdragon 845

Extending the life of Android devices with upstream kernels and postmarketOS

About me

- Caleb
- They/them/he/him
- UK University student.
- Intern at Linaro
- Reaaaaaally wanted to get to see Linux bootsplash on my phone.
- 2 years experience hacking kernels.
- Maintainer of the Snapdragon 845 “close to mainline” kernel used by postmarketOS and mobian.
- postmarketOS developer

@calebccff @calebccff@fosstodon.org

Android, Linux, Halium

- Android devices run heavily modified kernel (Δ5.5 million loc)
- They also depend on proprietary userspace blobs, libraries and daemons to enable basic functionality like graphics, modem, sensors.
- Halium wraps up proprietary android bits in a container and lets you run a regular(ish) Linux system - with caveats. Allows for much easier device enablement at the expensive of blobs and incompatibility with standard distros.

Linaro and Qualcomm

- Linaro have been working on mainline for Qualcomm devices for over a decade.
- Dedicated team (QCLT) for working on Qualcomm hardware.
- Spent several generations (msm8916 -> msm8996 -> msm8998 -> sdm845) trying to get boards into mainline and AOSP.
- Finally succeeded with SDM845 and the DB845c! Only Qualcomm board where support is actively funded.
- This benefits us because lots of awesome phones have SDM845 SoCs in them!
- Chasing regressions is a LOT of work, SDM845 mainline wouldn’t be possible without this support.

What goes into a Qualcomm SoC (SDM845)

* Clock controllers (lots of these)
* GPIO Controllers (pinctrl framework, tlmm)
* Power domains
* Power management interface host bus
* Regulators on the power management IC
* Thermal sensors, some on the CPU, some on various parts of the PMIC
* LCD backlight drivers (also on the PMIC)
* Custom shared memory interfaces
* Remote processor support (ADSP, CDSP, modem)
* The i2c, spi and UART hardware (GENI QUP)
* IOMMUs
* Modem hardware (custom network protocol QRTR which operates over the shared memory regions)
* Interrupt controllers
* Interconnects
* UFS / eMMC controllers
* IPA, custom packet router required for setting up mobile data connections
* Custom sound hardware
* Custom USB controllers
* Custom video encoder/decoder (with it’s own clock controller of course!)
* Custom camera subsystem (ISP)
* Custom GPU and DSI (display interface) controller
* Displayport controller
* Always On Subsystem
* Limits management hardware (configures thermal and power limits for the CPU cores and other hardware blocks)
* Atheros wifi - over QMI (Qualcomm MSM Interface), which operates over QRTR, which operates over shared memory.
* Probably some things I missed

postmarketOS - a 10 year life cycle for phones

- Keeping phones out of landfill since 2017
- Alpine based, extremely lightweight.
- Designed to work around Android device quirks
- Awesome community and wiki with swathes of documentation on mainline kernel porting. Much of which is not documented elsewhere.
- Supports many Snapdragon 410 devices which are cheap, ubiquitous and well supported - easier than ever to get a Linux phone. (See Nikita Travkin's earlier talk on this)
- PureTryOut at FOSDEM in 2020: "you can boot the system, but you probably won't be able to use it as a phone"
- In 2022 let's challenge that!

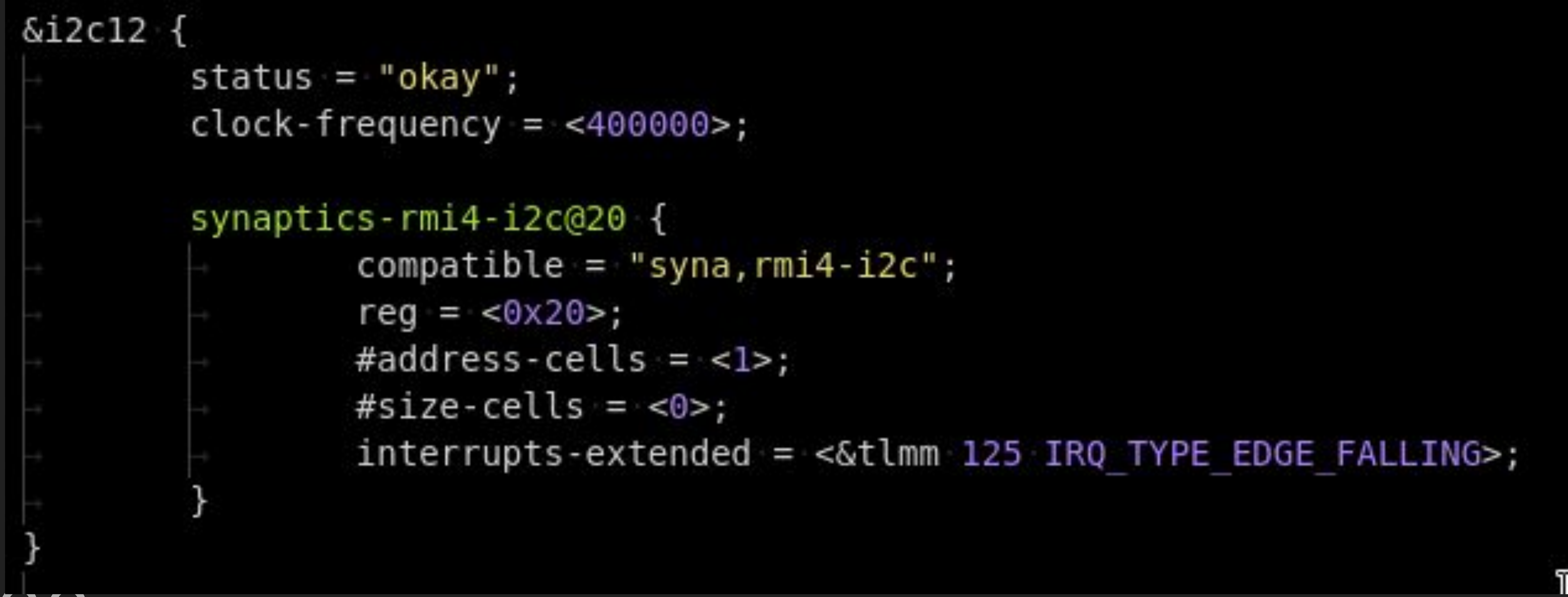
Bootloader quirks

- Bootloader is a black box on most devices, even though the vendor-specific code can be made open source.
- Made of two parts, the XBL (eXtensible BootLoader) and ABL (Android BootLoader (I think?)).
- XBL contains proprietary Qualcomm code for hardware support, ABL is open source but vendors are free to modify and don’t have to open source their changes.
- Some vendors do nightmarish things to make their products ship.
- Pixels refuse to boot unless they find a dtbo match, so we have to craft a dtbo with magic values
- OnePlus 9 is even worse (congrats to robbbbbb for figuring that out)

Bringing up a phone

- OnePlus 6 runs Linux 4.9 downstream (same for most SDM845 devices).
- Attempted postmarketOS port using that kernel, lack of Android userspace blobs meant most things (including display) didn’t work.
- “Hey you should try mainline”
- Copy MTP dev board devicetree, validate regulators and add framebuffer graphics.
- For a detailed guide check out the SDM845 mainlining guide (improvements welcome)

Devicetree



```
&i2c12 {
        status = "okay";
        clock-frequency = <400000>;

        synaptics-rmi4-i2c@20 {
                compatible = "syna,rmi4-i2c";
                reg = <0x20>;
                #address-cells = <1>;
                #size-cells = <0>;
                interrupts-extended = <&tlmm 125 IRQ_TYPE_EDGE_FALLING>;
        }
}
```

- There are many much better talks which explain devicetree.
- in short it’s a syntax for describing hardware, how it relates to other hardware, allows software like a kernel to interact with said hardware.
- Downstream and mainline both use devicetree, one is much nicer to work with.
- Android uses DBTBOs (DeviceTree Blob Overlays), most vendors misuse them, so we have to wipe the dtbo partition on most devices, or even craft our own to deal with…

Mainlining is super addictive

- If you like Linux and haven’t yet experience the kernel booting on a phone, you’re missing out.
- Getting to take a phone with a projected EOL of some time in the near future and extending it almost indefinitely is awesome!
- Making the touch screen work for the first time is a great feeling!
- Spending dozens of hours reading thousands of lines of code and having it pay off when you rewrite it for mainline and it actually works!
- One kernel to support many phones?!

postmarketOS and friends

- postmarketOS aren’t the only kid on the block.
- Mobian are doing fantastic work on supporting Linux mobile, with the aim of upstreaming as much as possible to debian.
- They make releases for the OnePlus 6 and PocoPhone F1!

So what’s still missing for Snapdragon 845?

- Camera (WIP)
- Call audio (WIP)
- Sensors (proprietary Qualcomm IP, custom protobuf)
- USB power delivery, dual role / OTG support.
- GPS (can’t land patches upstream in GPSd because of lacking documentation)
- Displayport over USB-C (not supported by the OnePlus 6 but by some other devices).

https://wiki.postmarketos.org/wiki/OnePlus_6_(oneplus-enchilada)

We aren’t quite there yet

- So awesome, we’re at a point where Linux on phones is totally feasible - we even have companies like Pine64 selling devices with Linux preinstalled out of the box!
- Getting mainline to boot on phones is only half the battle… With some exceptions (like the PinePhone) it’s mostly impossible to actually “own” the hardware.
- Qualcomm devices usually ship with secure boot enabled, making it impossible for users to modify, patch or upgrade the bootloaders.
- Modern Android devices are actually virtual machines with limited access to the physical address space.

The SHIFT we need?

- SHIFT are a sustainability oriented German phone manufacturer.
- Devices are fully repairable by the end user, no glue only screws!
- The SHIFT6mq features a Snapdragon 845 and runs Android, SHIFT want to support their devices for 10+ years
- User-owned, ships with secure boot off - potential to run our own hypervisor.
- SHIFT are supporting the community and have contributed to initial mainline kernel support for their device.

The future of mainline on Qualcomm

- Latest Snapdragon 8 gen 1 SoC (sm8450) got day 1 patches for mainline from Linaro
- Qualcomm increasingly friendly to upstream
- Google have a renewed focus on hypervisors / virtual machines on Android, might actually let end-users get EL2.
- Probably still no true ownership in the near future, until someone figures out how to unfuse SoCs :D

Thanks for watching

- If you enjoyed this talk and want to see this work continue, go get that old phone out of your drawer and check out the postmarketOS wiki
- Join us on matrix @ #main:postmarketos.org
- You can also keep up with my work by following me on twitter or Mastodon

🐦 @calebccff @calebccff@fosstodon.org