

Rethinking device support for the long-term

Nícolas F. R. A. Prado

FOSDEM²³

COLLABORA

About me

- Work at Collabora
 - Upstreaming kernel support for Chromebooks
 - Improving KernelCl coverage
- Email: nfraprado@collabora.com

Upstream-focused device support

- Motivations
 - Continuous updates
 - No vendor lock-in
 - Lower maintenance cost
 - Longer lifespan
- Example: Chromebooks

Why do we need CI?

- Long-lived devices require long-term support
- Steady supply of new devices keeps demand high
- To scale and meet demand:
 - Automated regression detection → Continuous Integration (CI)
 - Enable tests early

KernelCI

- CI system for the Linux kernel
- Main instance on linux.kernelci.org
 - Other instances, like chromeos.kernelci.org
- Pipeline:
 - Monitor git branch until new revision is found
 - Build artifacts (kernel, modules, dtbs, rootfs)
 - Queue test to device in LAVA lab
 - Add test result to the dashboard
 - If a regression, report to groups.io/g/kernelci-results



KernelCI configs

- Configuration through YAML files:
 - Build: Trees/branches, config fragments, compiler/version
 - Some maintainers have a 'for-kernelci' branch
 - Lab: definition of labs (currently 11), filters
 - Rootfs: base OS, arch, packages, scripts, fs overlays
 - Test: test plans, rootfs, LAVA job template, parameters, device types (currently 208)
- Simple for anyone to extend the coverage

Tests on KernelCl

- Tests currently available on KernelCI:
 - Baseline (generic and machine specific tests)
 - kselftest (26 tests, including alsa, rtc, etc)
 - Itp (7 tests, including mm, crypto, etc)
 - v4l2-decoder-conformance (fluster)
 - igt (-gpu/-kms)
 - v4l2-compliance
 - lc-compliance (libcamera)
 - cros-ec
 - sleep
 - smc
 - preempt-rt
 - usb

Detecting regressions during upstreaming

- Upstreaming support for Acer Chromebook CB514-2H (MT8192-Asurada-Spherion)
- Test components, locate issues, send fixes
- Mainline is a moving target
 - Need to retest everything on every rebase
- Several issues detected by manually testing
 - Could have been detected by KernelCI!



Regression examples

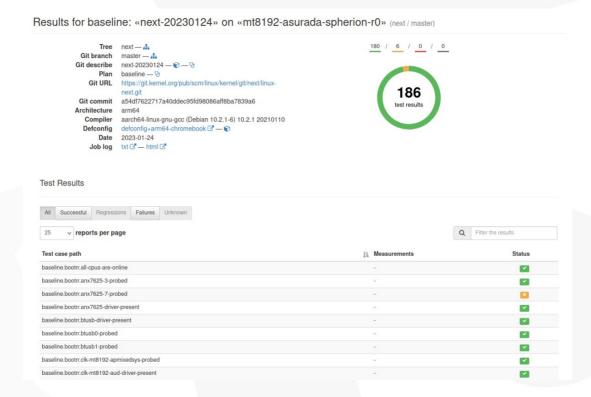
- drm/mediatek: mtk_dsi: Avoid EPROBE_DEFER loop with external bridge (link)
- drm/mediatek: dsi: Add atomic {destroy,duplicate}_state, reset callbacks (link)
 - Display not probing. Detectable by baseline or igt-kms test.
- drm/mediatek: dsi: Move mtk_dsi_stop() call back to mtk_dsi_poweroff() (link)
 - Warnings on suspend. Detectable by sleep test.
- media: mediatek: vcodec: Drop platform_get_resource(IORESOURCE_IRQ) (link)
 - Encoder not probing. Detectable by baseline test.



MT8192-Spherion on KernelCl

- Configs enabled through fragment (kernelci-core#1325, and also being upstreamed)
- Baseline(-nfs) tests enabled (kernelci-core#1324)
- Device probe tests added to baseline (bootrr#20)
- Next to enable:
 - alsa kselftest (needs to apply UCM), cros-ec-test, libcamera, v4l2-compliance, igt-kms
- Tests waiting for patches to land upstream:
 - v4l2-decoder-conformance, sleep, igt-gpu

Baseline results for MT8192-Spherion



Where to grow KernelCl

- More subsystems (eg iio, input)
- More trees from maintainers
- More labs
- More kselftests / ltp
- Support for KUnit

Conclusion

- There's still much to be gained from the Open Source model in linux testing
 - Keep increasing git branch, codebase and device coverage
- KernelCI will enable us to:
 - respond more quickly to regressions
 - give reliable long-term support for devices

Thank you!

We are hiring - col.la/careers

