



Contribution ID: 115

Type: **not specified**

## Testing Drivers with KUnit (Does hardware have to be hard?)

*Monday, 13 November 2023 10:30 (25 minutes)*

Unit testing common library code is (relatively) easy, but drivers often deal with a lot of global state, both in code and in hardware. New features like static stubbing go some way towards making this easier, but a lot of work still goes into making “fake devices”.

There are still many open questions, however:

- Are the existing tools helping? Is there something obviously missing?
- Are UML features like LOGIC\_IOMEM a good path forward?
- How should drivers make a fake ‘struct device’? Via a platform\_device (possibly with devicetree support), root\_device, or a new kunit\_device?
- There are lots of ways of managing resources for tests (kunit\_resource, KUnit actions, devres/devm\_ APIs). What should we use, when?
- How do we deal with callbacks, threads, etc, with KUnit contexts?
- How to support other safety/reliability/testing opportunities like hardware fuzzing and Rust?

**Primary author:** GOW, David (Google)

**Presenter:** GOW, David (Google)

**Session Classification:** Kernel Testing & Dependability MC

**Track Classification:** LPC Microconference: Kernel Testing & Dependability MC