🎆 Dasharo Open Source Firmware Validation 🎆

Agenda

- Introduction to Dasharo Open Source Firmware Validation (OSFV)
- Current state
- Recent improvements
- Work in progress - current priorities
- Q&A

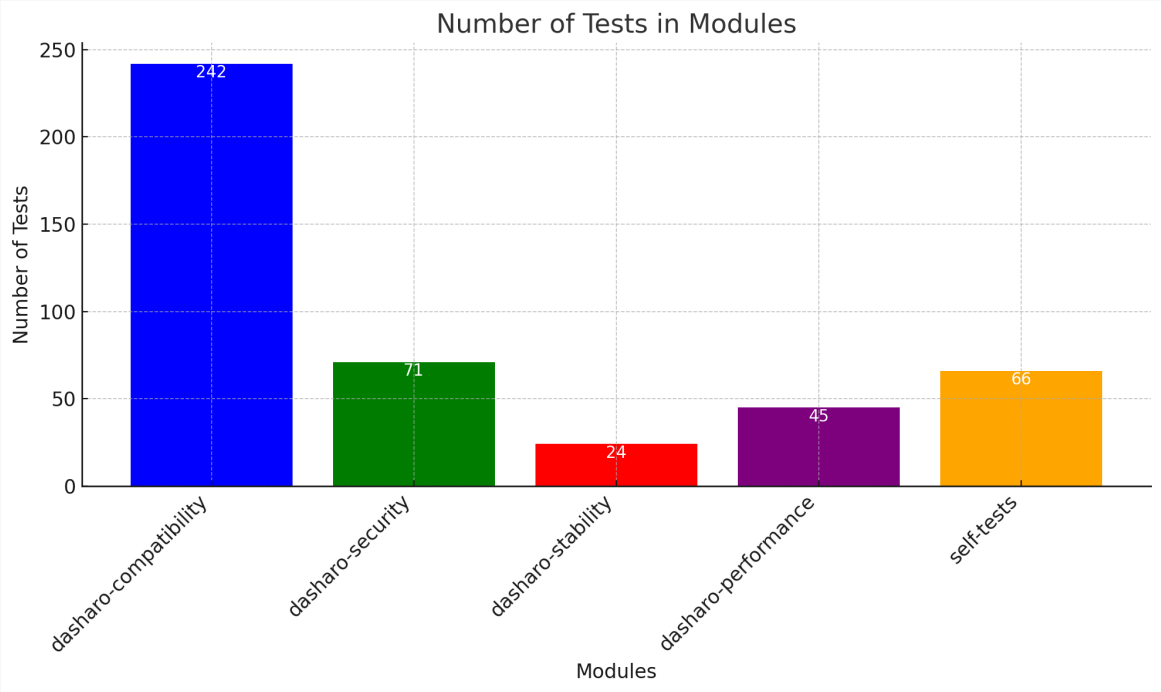
Introduction to Dasharo OSFV

- Main purpose - validation of open-source firmware (mainly Dasharo)
- Using Robot Framework as a base
- Use cases
  - testing Dasharo firmware releases
  - test-driven bug fixing (and adding new features)
  - regression testing
    - after introducing new features
    - after major changes (update base from upstream project)
  - validation of Dasharo related tools (DTS, DCU)
    - where possible, in QEMU

Current state

- No release since the last presentation 😞
- Still quite intense development in the `develop` branch
  - all improvements target this branch now
  - if you want to experiment, this should be a starting point
  - we are aiming for something "stable enough" to merge into `main` and release v0.3.0 version
  - hopefully before DUG#7 ☺

Current state



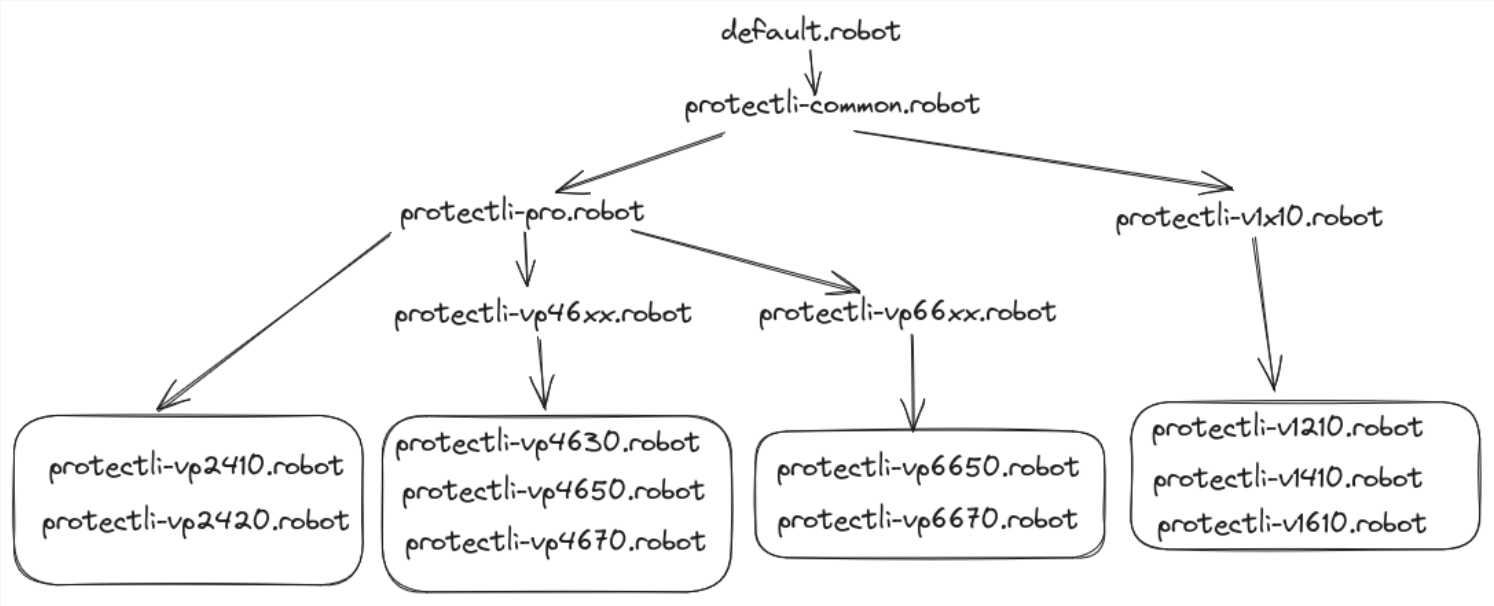
- 382 tests
- 66 self-tests (like unit tests for sustaining correctness of keywords operation)

Recent improvements

- "Recent" - since the last presentation (~6m)
- Notable changes
  - platform-configs rework
  - Documentation generation
  - Switch Sonoff API
  - New tests
  - New platforms
- Continuation of refactoring activities
- Multiple fixes and minor improvements

Notable changes - configs rework

- Reduced code duplication across platforms, added comomon includes



Notable changes - documentation

- Split documentation under docs directory
  - how to add new platform
  - contributing guidelines
  - workflow with QEMU
  - platform-specific quirks

Notable changes - documentation

Adding new platforms

Depending on what type of platform you're adding, the instructions here will vary.

- If no similar board is yet supported, follow the steps in Adding a brand new platform
- If the board is a variant of another, similar, already supported board, follow the steps in Adding new variant of an existing platform

Adding a brand new platform

- Create a new file for your mainboard in `platform-configs/`. For most platforms this file will be called `[platform-vendor]-[platform-model].robot`.
- Copy the contents of `include/default.robot` to your platform config
- Modify the file for your platform:
  - Modify the settings appropriately for your mainboard
  - Remove any unmodified lines - they will be sourced from `default.robot`
  - Add the following at the top of your platform config - this will ensure defaults are used for unspecified options:

```
*** Settings ***
Resource    default.robot
```

Notable changes - switch Sonoff API

- Reflashed firmware on Sonoffs in the lab to a more stable one
  - https://tasmota.github.io/docs/devices/Sonoff-S26-Smart-Socket/

Notable changes - new tests

- Checking for unexpected errors in logs (dmesg)
- Watchdog suite
- APU-specific features
- Suite for DCU tool
- Correctness of CPU / memory information in the main menu

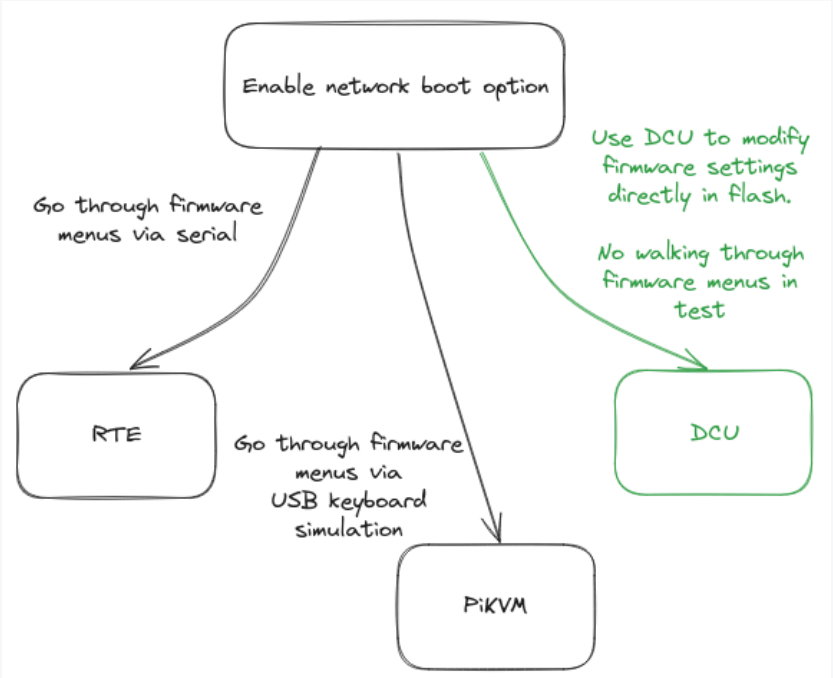
```
================================================================================
                            Setup-Menu-Information
================================================================================
SET001.001 CPU clock speed displayed in setup menu                    | PASS |
--------------------------------------------------------------------------------
SET002.001 RAM speed displayed in setup menu                          | PASS |
--------------------------------------------------------------------------------
SET003.001 RAM size displayed in setup menu                           | PASS |
--------------------------------------------------------------------------------
SET004.001 Expected CPU clock speed displayed in setup menu           | PASS |
--------------------------------------------------------------------------------
SET005.001 Expected RAM speed displayed in setup menu                 | PASS |
--------------------------------------------------------------------------------
SET006.001 Expected RAM size displayed in setup menu                  | PASS |
--------------------------------------------------------------------------------
Setup-Menu-Information                                                | PASS |
                         6 tests, 6 passed, 0 failed
================================================================================
```

Notable changes - new platforms

- New platforms
  - Protectli V1000 series (V1210, V1410, V1610)
  - Protectli VP6000 series (VP6650, VP6670)
  - PC Engines APU2-6
  - Minnowboard Turbot
  - NovaCustom MTL models

DCU integration - alternative interface for changing fw settings

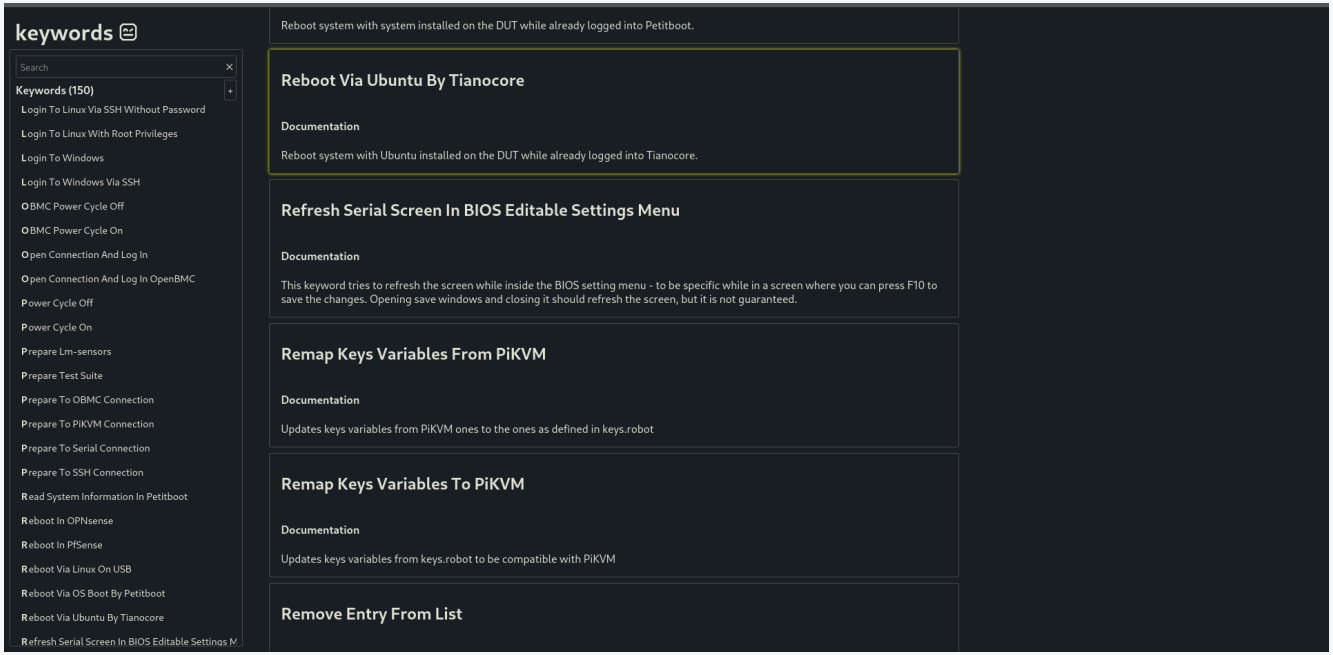
- Instead of manual steps, we can modify SMMSTORE variables directly



In progress - current priorities

Generate documentation from test code

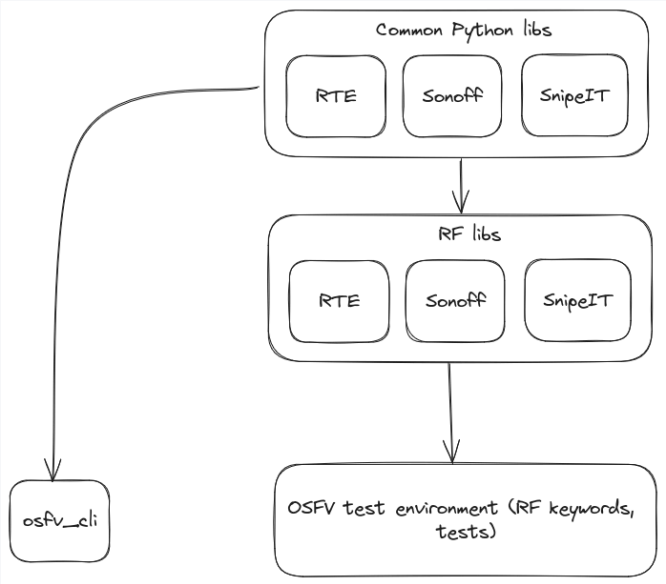
- PR: https://github.com/Dasharo/open-source-firmware-validation/pull/293/files
- To be hosted on github pages for starters (not hosted yet)



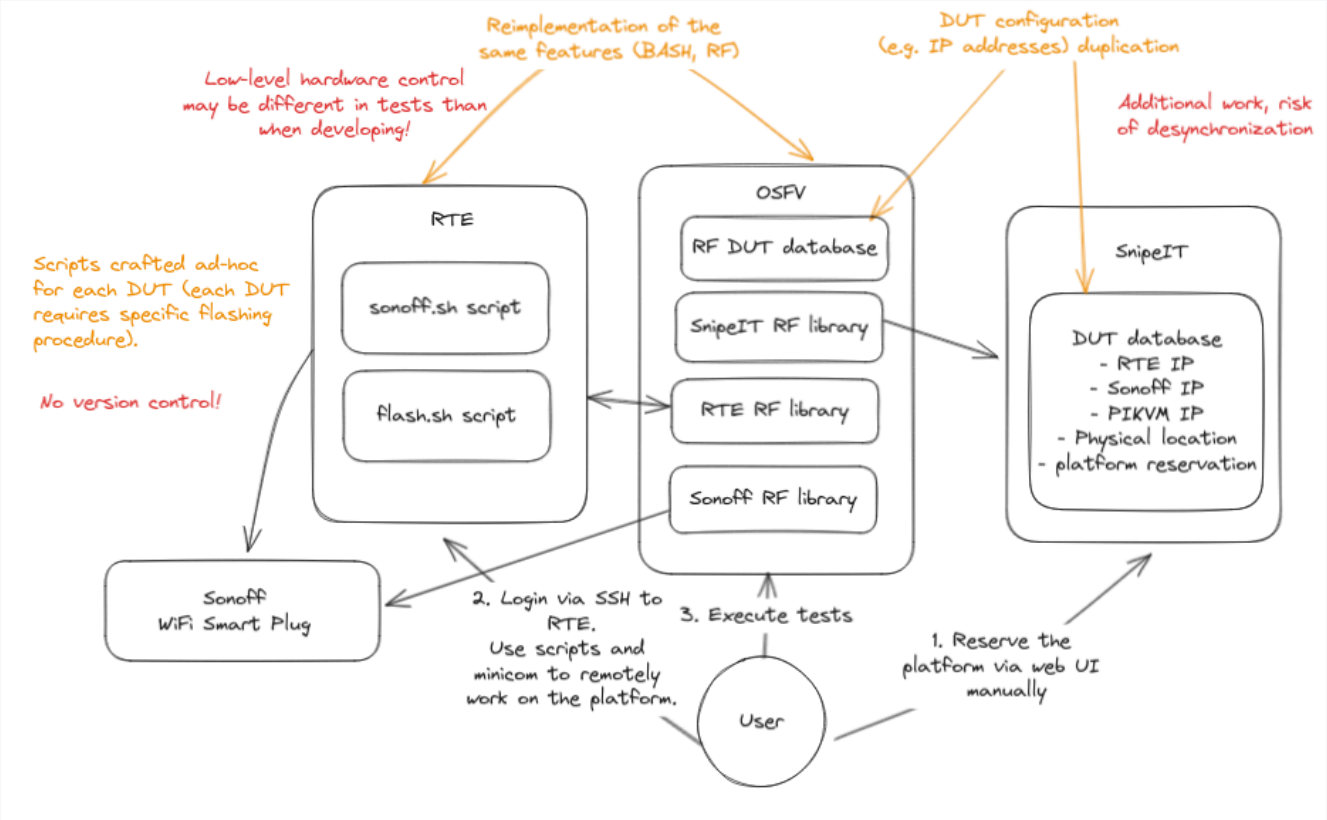
In progress - current priorities

osfv_cli integration

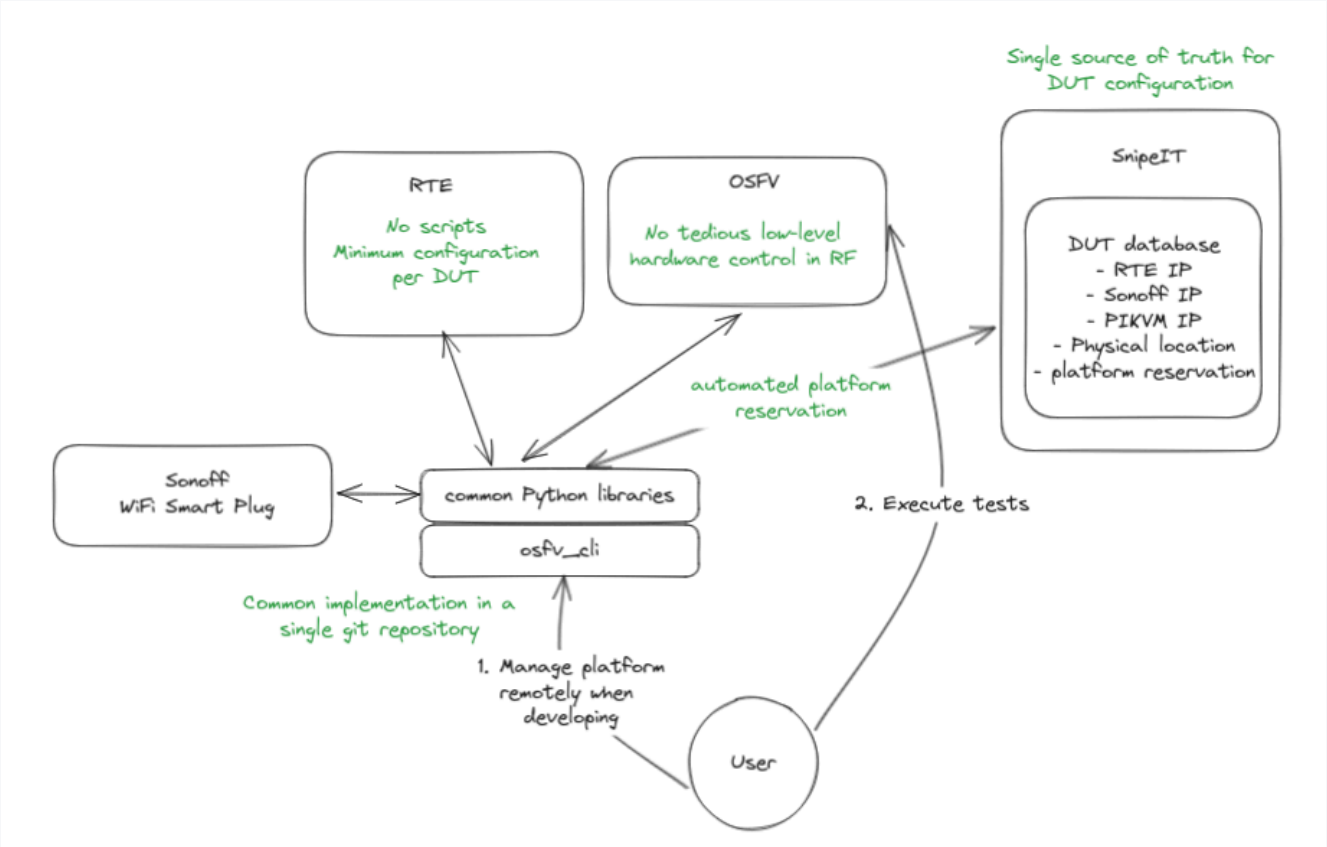
- Integrate low-level hardware operations into Python libraries
- Reuse the same libraries by test framework and CLI tool



osfv_cli - motivation - before



osfv_cli - motivation - after



osfv_cli - developer usage scenario

```
# Reserve platform
osfv_cli snipeit check_out --rte_ip $RTE_IP
# Read backup firmware
osfv_cli rte --rte_ip $RTE_IP flash read --rom backup.rom
# Flash new firmware
osfv_cli rte --rte_ip $RTE_IP flash read --rom backup.rom
# Apply power to platform
osfv_cli rte --rte_ip $RTE_IP rel set high
# Get logs from serial
osfv_cli rte --rte_ip $RTE_IP serial
# Reset platform
osfv_cli rte --rte_ip $RTE_IP pwr reset
```

Next steps

- Stabilize the environment for the v0.3.0 release
  - finalize current priority tasks
  - full regression on selected supported platforms
    - NovaCustom
    - Protectli
    - PC Engines
    - QEMU
    - MSI

How can you help?

- Go through `Getting Started` and `QEMU workflow` sections
  - https://github.com/Dasharo/open-source-firmware-validation/blob/develop/docs/qemu.md
  - try to run it
  - fail miserably
  - report problems in GH repo or Matrix channel
    - https://matrix.to/#/#osfv:matrix.3mdeb.com
  - help us to improve docs/scripts through external validation
- `good first issue` label
  - https://github.com/Dasharo/open-source-firmware-validation/issues?q=is%3Aissue+is%3Aopen+label%3A%22good+first+issue%22
  - take a look at these or any other
  - ask how to proceed if you want to help

Q&A