Zephyr® Project

Developer Summit 2022

June 8-9, 2022

Mountain View, CA + Virtual

NORDIC®
SEMICONDUCTOR

Zephyr® Project
Developer Summit 2022

June 8-9, 2022
Mountain View, CA + Virtual

NORDIC®
SEMICONDUCTOR

Twister V2: New Framework for Testing Based on Pytest. Overview and Current Status

Maciej Perkowski

Nordic Semiconductor

Outline:

- Why?
- How?
- What?
- Current status
- Demo

Why?

- Twister evolved in a reactive way
- Lacks coherent overall architecture:
  - Low modularity
  - Several places in the code have a similar responsibility
  - Long and hard to follow “if-else” decision trees
  - Workaround fixes and “hacks” applied over the years
- Hard to maintain. Changes require high vigilance on all other parts.
- Developers can be reluctant towards contributing to the source code.
- Low test coverage. Tested mostly in production.
- We need a SOLID framework

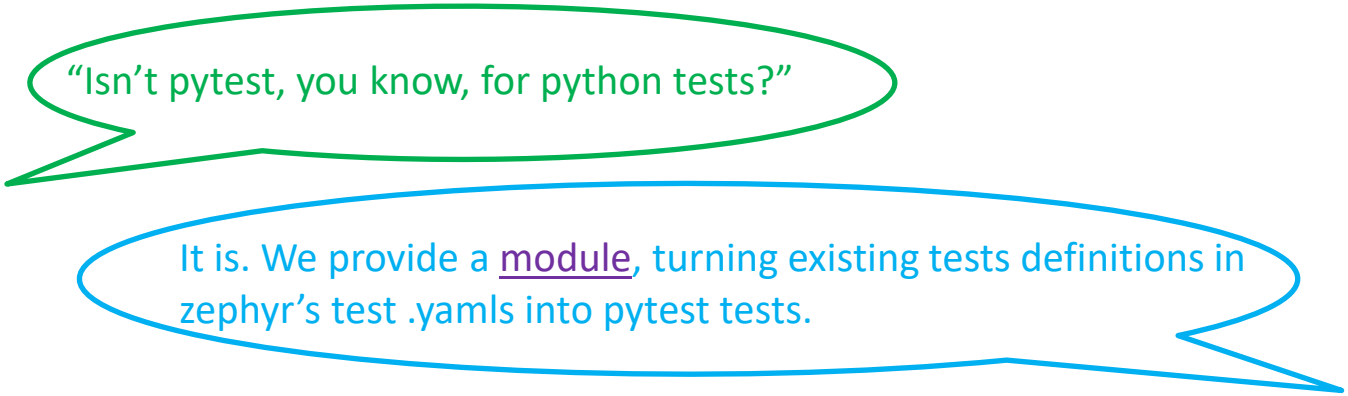
How?

- Using available and popular libraries instead of “reinventing the wheel” wherever it is possible and making sense
- Managed as a python package (like e.g., west)
- Based on SOLID principles
- High test coverage
- Backward compatibility with the existing Zephyr’s tests
- Facilitate new feature addition (e.g., interactive tests and more complex scenarios)
- Engage the community

What?

Based on pytest.

“The pytest framework makes it easy to write small, readable tests, and can scale to support complex functional testing for applications and libraries.”



Pytest benefits:

- Easy to expand and modify thanks to the plugin concept
  - Loosely coupled relations between the host and a plugin.
  - Burden on the host side: hook functions for plugins.
  - *E.g., Useful when adding a new report type, or a way to filter (select and deselect) tests*
- Simple reusability of common functions with @fixture
  - Modular approach (fixtures can use other fixtures)
  - Simplified (and safely managed) teardown logic
  - Easy to trace: explicit declaration in functions, transparent printout of usage
  - *E.g. code blocks for building hexes and DUT management (setup->connect serial->flash->YIELD DUT->disconnect)*

- Being very popular:
  - Don’t have to worry about stuff like test counting and overall runner’s „consistency”
  - A lot of functionality available out-of-the-box (or with some customization):
    - Subsets with https://pypi.org/project/pytest-split/ (even able to optimize load by tests execution time)
    - Parallelization with https://pypi.org/project/pytest-xdist/
    - Coverage with https://coverage.readthedocs.io/en/6.3.2/
    - Different types of reports, e.g interactive html
  - Can be more attractive to new contributors

Current status

Test identification (specification)

DONE:

- Pytest tests automatically generated from existing Zephyr tests
- Pre-filtering based on tests and boards yamls specifications
- Examples of filtering plugins (tags, „slow”)

TODO:

- Add more filters to align with the current twister
- Add test cases („subtests”) identification

Test execution

DONE:

- Building and flashing images based on west interface
- „harness: ztest”
- Abstraction layer (using factory method pattern) for DUT with „native posix” and „HW” classes implemented
- Parallelization with xdist. Works well with „native”
- Single tests within an app recognized as „subtests”

TODO:

- „harness: console”
- Add „qemu” DUT
- Parallelization for HW (DUTs cannot be accessed by multiple workers)
- Improve naming of tests (configurations) and subtests (test cases) in the console output
- Compare performance/benchmarking

Reporting

DONE:

- Generic base class and specialized „writers”
- Implemented custom json and csv
- Other types can be taken from plugins (e.g https://pytest-html.readthedocs.io/en/latest/ or https://github.com/prashanth-sams/pytest-html-reporter)

TODO:

- Improve subtests reporting
- Consider other types
- Customize existing plugins

You are more then welcome to join the project and add your contributions!

- https://github.com/zephyrproject-rtos/twister
- https://github.com/orgs/zephyrproject-rtos/projects/7
  - „good first issue” and „help wanted” labels

```
{
    "environment": {
        "report_time": "17:08:40 01-06-2022",
        "pc_name": "maciej-VirtualBox",
        "duration": 17.77968120574951
    },
    "configuration": {
        "build_only": false,
        "default_platforms": [
            "native_posix"
        ],
        "board_root": null,
        "output_dir": "twister-out"
    },
    "summary": {
        "passed": 5,
        "failed": 0,
        "skipped": 0,
        "xfailed": 0,
        "xpassed": 0,
        "error": 0,
        "total": 5,
        "subtests_total": 225,
        "subtests_passed": 210,
        "subtests_failed": 0,
        "subtests_skipped": 15
    },
    "tests": [
        {
            "suite_name": "zephyrproject2.zephyr.tests.kernel.common.testcase.yaml",
            "test_name": "kernel.common[native_posix]",
            "nodeid": "zephyrproject2/zephyr/tests/kernel/common/testcase.yaml::kernel.common[native_posix]",
            "platform": "native_posix",
            "tags": "base userspace",
            "type": "integration",
            "platform_allow": "",
            "status": "passed",
            "message": "",
            "duration": 0.000534083999809809,
            "subtests": [
                {
                    "name": "bootdelay",
                    "status": "passed",
                    "duration": 0.000534083999809809
                },
                {
                    "name": "irq_offload",
                    "status": "passed",
                    "duration": 0.0002993909874930978
                },
                {
                    "name": "nested_irq_offload",
                    "status": "skipped",
                    "duration": 0.00027339000371284783
                },
```


results.html

Report generated on 29-May-2022 at 22:17:05 by pytest-html v3.1.1

Environment

GIT_BRANCH	updates
Packages	{"pluggy": "1.0.0", "py": "1.11.0", "pytest": "7.1.2"}
Platform	Linux-5.4.0-113-generic-x86_64-with-glibc2.27
Plugins	{"forked": "1.4.0", "html": "3.1.1", "html-reporter": "0.2.9", "metadata": "2.0.1", "subtests": "0.7.0", "twister": "0.0.1", "xdist": "2.5.0"}
Python	3.8.4

Summary

42 tests ran in 735.54 seconds.

(Un)check the boxes to filter the results.

☑ 38 passed, ☐ 28 skipped, ☐ 4 failed, ☐ 22 errors, ☑ 0 expected failures, ☑ 0 unexpected passes

Results

Show all details / Hide all details

Result	Test
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/early_sleep/testcase.yaml::kernel.common.sleep[native_posix]
Passed *(hide details)*	zephyrproject2/zephyr/tests/kernel/events/event_api/testcase.yaml::kernel.events[native_posix]

```
 ----------------------------Captured stderr setup----------------------------
2022-05-29 22:05:36,945:INFO:twister2.builder.west: Building Zephyr application
2022-05-29 22:05:36,945:INFO:twister2.builder.west: Build command: python -m west --zephyr-base /home/maciej/zephyrproject2/zephyr -v build /home/maciej/zep
item kernel.events --build-dir /home/maciej/TwisterV2/TwisterV2/twister-out/native_posix/tests/kernel/events/event_api/kernel.events
2022-05-29 22:05:45,354:INFO:twister2.builder.west: Finished building /home/maciej/zephyrproject2/zephyr/tests/kernel/events/event_api for native_posix
2022-05-29 22:05:45,354:INFO:twister2.device.native_simulator_adapter: Flashing device
2022-05-29 22:05:45,354:INFO:twister2.device.native_simulator_adapter: Flashing command: /home/maciej/TwisterV2/TwisterV2/twister-out/native_posix/tests/ker
2022-05-29 22:05:45,458:INFO:twister2.device.native_simulator_adapter: Finished flashing /home/maciej/TwisterV2/TwisterV2/twister-out/native_posix/tests/ker

 ----------------------------Captured stderr call-----------------------------
2022-05-29 22:05:45,464:INFO:twister2.yaml_test_function: Execution test kernel.events[native_posix] from /home/maciej/zephyrproject2/zephyr/tests/kernel/ev
2022-05-29 22:05:45,465:INFO:twister2.log_parser.harness_log_parser: Found test suite: events_api
2022-05-29 22:05:45,465:INFO:twister2.log_parser.harness_log_parser: Ztest: k_event_init - PASS in 0.0
2022-05-29 22:05:45,470:INFO:twister2.log_parser.harness_log_parser: Ztest: event_deliver - PASS in 0.0
2022-05-29 22:05:45,472:INFO:twister2.log_parser.harness_log_parser: Ztest: event_receive - PASS in 0.45
2022-05-29 22:05:45,473:INFO:twister2.log_parser.harness_log_parser: PROJECT EXECUTION SUCCESSFUL
```

Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/events/sys_event/testcase.yaml::kernel.events.usage[native_posix]
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/fifo/fifo_api/testcase.yaml::kernel.fifo[native_posix]
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/fifo/fifo_timeout/testcase.yaml::kernel.fifo.timeout[native_posix]
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/fifo/fifo_usage/testcase.yaml::kernel.fifo.usage[native_posix]
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/lifo/lifo_api/testcase.yaml::kernel.lifo[native_posix]
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/lifo/lifo_usage/testcase.yaml::kernel.lifo.usage[native_posix]
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/mbox/mbox_api/testcase.yaml::kernel.mailbox.api[native_posix]
Passed *(show details)*	zephyrproject2/zephyr/tests/kernel/mbox/mbox_usage/testcase.yaml::kernel.mailbox.usage[native_posix]

References:

https://en.wikipedia.org/wiki/SOLID

https://docs.pytest.org/en/7.1.x/

https://pluggy.readthedocs.io/en/stable/

https://pypi.org/project/pytest-xdist/

https://pytest-html.readthedocs.io/en/latest/ https://github.com/prashanth-sams/pytest-html-reporter

https://pypi.org/project/pytest-split/

https://coverage.readthedocs.io/en/6.3.2/

Thank you!

Benefits of being an independent package:

- Versioning of the tool itself (won't be tightly coupled with Zephyr’s revision)
- Independent CI/CD
- Simplifies staging branch for CI (testing newer twister version before it enters the main)
- Easier configuration (e.g., combining with other pytest plugins)
- More attractive to external contributors
- Easier to develop new plugins which would use Twister's functionality (but not require Zephyr)