Exercise 10: *PyVSC* Coverage for *Saturation Filter* testbench

Objective: *Introduction to the coverage collection in a UVM testbench using the PyVSC library.*

Task: *Implementation of the coverage collector class for the Saturation Filter testbench.*

Develop the coverage collector class for the *Saturation Filter* testbench by creating a new file, named `sat_filter_coverage.py`, inside `<ROOT>/sat_filter/src/tb`. The coverage collector class must be named `sat_filter_coverage` and extended from `uvm_subscriber`.

Following the same steps to implement `uvc_ssdt_coverage.py` in exercise E08, implement a `covergroup` class to cover the `ovf` signal from the output of the DUT.

Notice that the `ovf` signal is not being monitored by any available `monitor` component. Because of this, a coroutine must be implemented.

In the `sat_filter_tb_base_test` create a coroutine that will run concurrently to the test. This coroutine must call the `write` method of the `sat_filter_coverage` and pass the `ovf` value whenever `out_valid == 1`. This coroutine must be launched in the `run_phase`. Remember also to create the `sat_filter_coverage` in the `build_phase` of the `sat_filter_tb_base_test`.

Run the test and look at the coverage results inside `<ROOT>/sat_filter/src/tb/sim_build`. Use the `PyUCIS-viewer` tool to visualize the coverage results.