

Testing and Fault Tolerance
(01RKZOQ / 01RKZOV / 01RKZQW)

Assignments

2021/2022



Goals

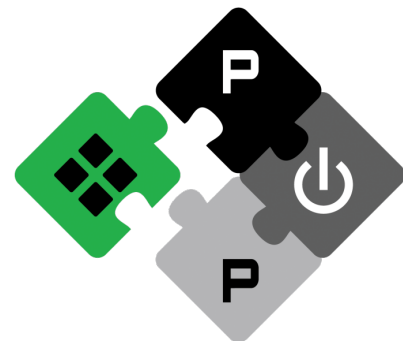
- Assignments are given to students to
 - Improve their technical skills in terms of
 - Usage of tools
 - Adoption of techniques presented in the course
 - Ability to face real problems
 - Promote their ability in managing a project
 - Identifying targets
 - Organizing their work
 - Presenting results
 - Working in a team

Two options

HW) Hardware-oriented assignment

SW) Software-oriented assignment

DUT: RISC V



- The RI5CY CPU is distributed in the frame of the open-source PULP project developed by ETH Zurich and Università di Bologna
- <https://pulp-platform.org/>

HW Assignment

- Develop a synthesizable LBIST for the RI5CY processor
- Evaluate the fault coverage of the LBIST execution and its test application time
- Develop a testbench that invokes the LBIST and then executes a given firmware (when no misbehavior is observed).

HW Assignment: LBIST

- Control/Status signals
 - GO/NOGO, PASS/FAIL, clock...
- You can decide which DfT to insert
 - A basic scan insertion script will be provided
 - You can freely organize scan-chains, controllable/observable points...
- TPG (LFSRs/MISRs or memory mapped)
- Report about the area overhead.

HW Assignment: Fsim

- Produce a VCD of the DUT signals during the LBIST run (including shift and capture cycles)
- Tweak TPG and DfT to increase the (stuck-at + transition) fault coverage as much as possible
- Compare the result with ATPG on the same DUT (with the same DfT inserted).

SW Assignment

- Develop a firmware to execute, in sequence:
 1. A March test algorithm on the RAM
 2. A test scheduler, which invokes a set of SBST routines for (at least) the functional units and the register file in the DUT
 3. A given main procedure.
- Evaluate the fault coverage of the SBST routines and the overall (March + SBST) test application time.

SW Assignment: March

- Write it in assembly to avoid using the RAM (stack...)
- Return 0 in case of success, 1 in case of error
- The caller shall halt the system in case of error

SW Assignment: SBST

- Each SBST routine:
 - shall last no more than 1000 clock cycles (if more, split it...)
 - returns a 32-bit test signature
- The scheduler:
 - Invokes an SBST routine (if enabled)
 - Checks the returned signature and halts the system in case of mismatch, otherwise it proceeds with the next routine

SW Assignment: SBST (2)

- At compile-time, the test engineer provides
 - a vector of test signatures (taken from the fault-free simulation)
 - A vector of flags to enable/disable routines (1=run routine, 0=skip routine)
- Check that signatures are stable after enabling/disabling previous routines
- Try to maximize the fault coverage on the selected modules (values above 80% are typically good).

Organizational issues

- Teachers will provide to students (by Dec.3) the support materials needed for the assignments
- Q&A forums will be created on Exercise to solve possible issues
- After laboratory 7, lab sessions will be devoted to the assignments (Wednesday only).

Procedural issues

- Interested students are requested to
 - Work alone or in pairs
 - Create or join a group using the section “Groups for assignments” of Exercise (by Dec. 12)
 - Complete their work by January 15, 2022 (included), uploading to the section “Assignment” of Exercise a single zip/rar file for each group containing the following data
 - The project folder (with all scripts needed to re-run the flow)
 - A short report
- A meeting will be organized with each group (after Jan. 15) to check the results.

The report

- It must include a description of
 - How the students developed the LBIST architecture (HW) or the test programs (SW)
 - Which results they achieved in terms of fault coverage and test application time.
- Max size: 5 pages

Limitations

- Each group can launch only one synthesis, logic simulation, or fault simulation experiment at a time
- Be careful: endless simulation (e.g., due to exceptions or endless loops) may saturate the available disk space. Please check!
- Violations will be punished by reducing the number of points granted for the assignment

Assignment points validity

- The points from the assignment will be added to the score of the written exam
- The latter (by itself) MUST be ≥ 18
- Assignment scores will NOT be valid after Sept. 2022