ASTERICS: a generic and flexible test platform for soft-error fault injection in processor-based architectures

R. Velazco, Greicy Marques Costa, Fabrice Pancher, Wassim Mansour

The constant progress of integrated circuits (ICs) manufacturing technologies and processes, results in an increase of the device's sensitivity to the effects of natural radiation. This problem must nowadays be considered for any application requiring reliability or dependability, devoted to operate in space or in the Earth's atmosphere, even at ground level. Among these effects, the so-called SEU (Single Event Upset) phenomena [3][4] is considered critical as it may lead to the modification, randomly in time and location, of the content of a memory cell.

SEU mitigation techniques [5][6] exist at different levels, such as HW and/or SW redundancy, Time redundancy, error detecting and correcting codes (EDAC), etc. In all the cases the efficiency of the implemented mitigation technique must be evaluated. This can be achieved by fault simulation, fault injection, accelerated radiation testing or experiments performed in the real environment.

In this demonstration is presented a generic and flexible platform, so-called ASTERICS (Advanced System for the Test under Radiation of Integrated Circuits and Systems), allowing to perform SEU-like fault injection experiments in an application executed by a processor either in a HW version or implemented in the FGPA of the test platform.

During the demonstration will be described the Code Emulated Upset (CEU) approach, used to perform fault-injection experiments, which reproduces in a quite realistic way the random occurrence in time and location, of SEUs in sensitive targets of the considered circuit. This approach, devoted to processors (real or implemented in FPGAs), is based in the activation, at random clock cycles, of asynchronous interruption signals. The execution of the associated interruption code allows modifying the content of memory cell randomly selected among those accessible through the instruction set of the studied processor. Typically targeted resources are general purpose registers, special function registers, internal SRAM, memory caches, etc. It is important to notice that to emulate in a suitable way the consequences of SEUs, the fault injection approach must be not intrusive, i.e. only the targeted resource must be corrupted by the fault injection step. In the CEU approach in all the cases this condition is respected considering that, for most processors, after an interrupt signal assertion the context is saved and restored after the execution of the associated interrupt program. The only condition is that registers used within the interrupt program are preserved either by the processor itself or by the executed interrupt program.

A Leon 3 processor implemented on an FGPA of a dedicated test platform will be the target of fault injection campaigns performed during this demonstration. The application chosen to be executed by this processor will be a *self stabilization program* which in principle is considered as immune to faults occurring in its variables. The fault injection runs will provide evidences on the sensitivity to soft-errors of this application and the "pathology" of these errors, which should allow enhancing the studied application with suitable fault-tolerance mechanisms, thus increasing its robustness with respect to the consequences of natural radiation present in the natural environment.

The ASTERICS system, whose architecture is built around two FPGAs. The first one, called COM FPGA, contains a Leon2 processor. It handles the communication between the user's computer and the resources available on the ASTERICS motherboard. and includes a programmable watchdog to check errors provoking *sequence-loss*. It also monitors the current consumption of the device under test (DUT) in order to protect it against destructive faults such as SEL (Single Event Latchups) which may occur during radiation ground testing experiments. Data transfers are performed over an Ethernet network providing high data rate transfer. The second FPGA, called Chipset FPGA, contains the user design, which can be either the tested design, or used to interface to the ATERICS resources, the hardware part implemented in a daughterboard. In this way are minimized the time, development effort and cost of the hardware platform required to perform radiation ground tests or fault injection experiments on a new circuit.

The consequences of the injected SEU may be classified as follows:

- *Silent fault*: the injected fault does not have any consequence on the program's results. As an example, typical silent faults are those affecting a register or data not used or yet used by the processor. Another possibility is that the algorithm provides a correct result despite the fault.
- *Result error*: the outputs of the program are not the expected ones.
- *Timeout*: the program execution duration exceeds a limit defined by the user.

In our experiments, the SEU faults are injected at instants randomly chosen within the nominal duration of the executed program. Fault injection runs with different running limits can be performed and may allow getting evidences of the impact of the running limit on the number and type of errors was one of the expected results of these experiments.