

# FT-UNSHADES Executive Summary

---

During the design process of an ASIC, there's a large number of factors that can lead to improper radiation protection of the circuit. Complex circuits are designed by a team of designers that must share information or the synthesis tool can remove redundancy protection, in many cases there is a high risk that protection is lost in certain parts of the circuit. Resistance to SEUs is, in general, difficult to check because large simulation cycles are needed with a huge testing effort. SEU detection is also stimuli set dependant. FT-UNSHADES is a tool that was, in its initial objectives, focused on SEU testing using a hardware platform that can greatly accelerate the simulation cycles.

The FT-UNSHADES project was launched in September 2003 as a small contract agreement between ESA, AICIA-GTE and the University of Sevilla, number 17540. Previous to the project, the experience was obtained to solve the problem of hardware debugging system with the UNSHADES technology. These approaches were capable of producing modifications in the state of an already running netlist on a Xilinx FPGA, using the partial configuration scheme.

The project has been divided into four main tasks:

- Board definition, design and assembling.
- Communication (system definition and design) between board and computer.
- Design preparation procedure.
- Testing tools: Test language definition, board services and test commands.

There are two important contributions of FT-UNSHADES system. Firstly, the fault injection strategy is based on direct manipulations of pieces of the bitstream, they are read directly from the already running FPGA. SEU emulation is performed using a read-modify-write of the Flip-flop contents using the configuration circuit of the Xilinx FPGA. This approach allows the design to be treated as an unmodified "black box", no instrumentation of the Flip-flops is required. Secondly a third party (non Xilinx, or even non FPGA if a compatible library is used) synthesis tool can be used to produce a design under test, as the input to the FT-UNSHADES flow is a post synthesis description of the design.

## 1. Board definition

---

- o The Emulation platform is a Xilinx Virtex II FPGA with FF1152 footprint, compatible FPGAs are XC2V8000, XC2V6000 and XC2V4000.
- o 2M x 102 bits of test vectors memory that can be used in configurations of up to 12M x 17 bits.
- o Resistor bridges allows testing of up to 32 bidirectional outputs.

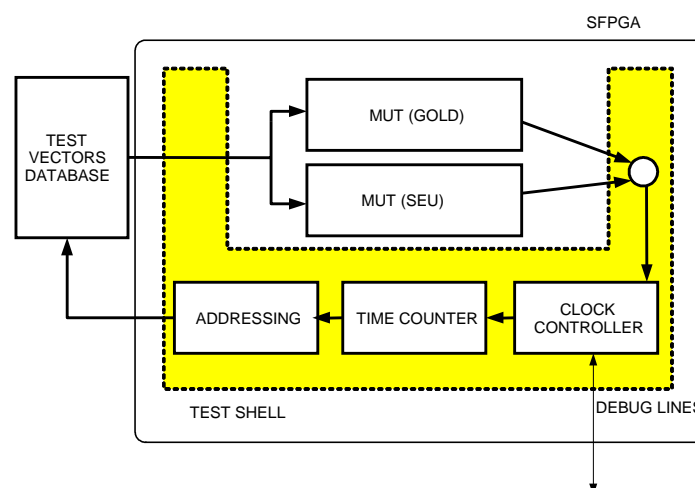


## 2. Communication system

The communication link is a USB 2.0 link or Parallel Port EPP1.9. Frames are read or written through this port using a control FPGA, smaller. This FPGA also generates the clock and supports multi-board communication link.

## 3. Design preparation procedure

The design is prepared from a Test Bench file written in VHDL code. The design description can be a post synthesis version of the design or a synthesisable HDL source code of it.



The design flow starts from a modification of the Test Bench file, where a piece of code is automatically inserted in order to record the inputs of the design simulation and generate a stimuli database that will be downloaded to the on Board SRAM Memories.

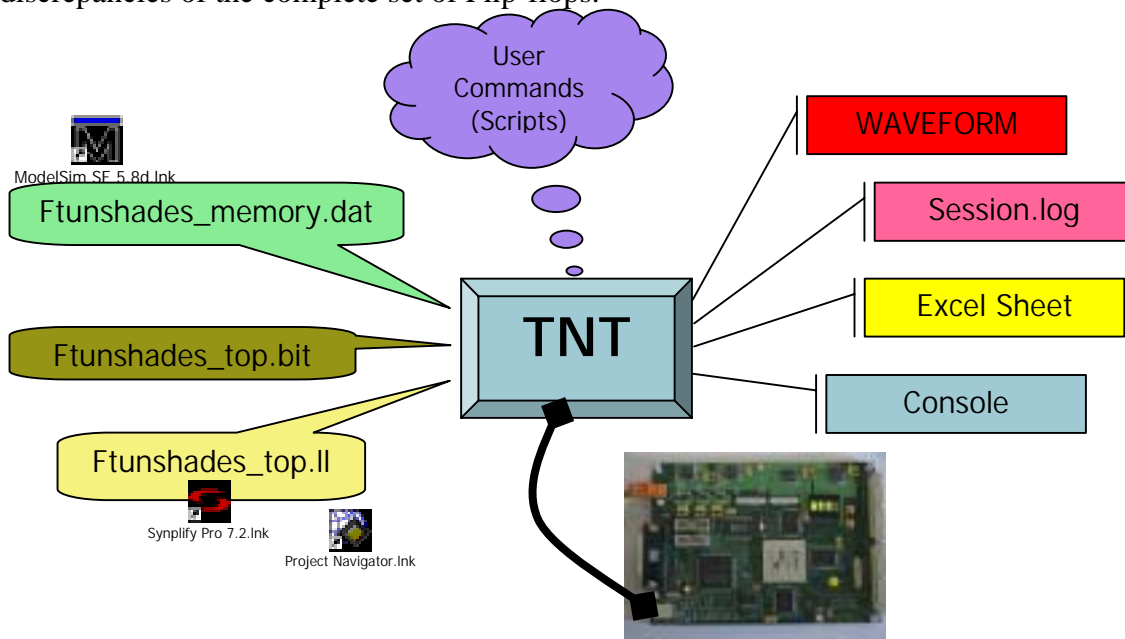
The next step is to produce the design for test emulation (DTE) model that will be inserted into the System FPGA. This is done using the Xilinx design flow. The figure shows a diagram of the test model: Two instances (Called GOLD and SEU) of the Module Under Test (MUT) are placed in a Test Shell, that is a set of pieces of code needed to control the system clock, the *capture and readback* of the state and provide the input test vectors. The design inputs are stimulated from the contents of the SRAMs and outputs of both samples are compared. The main advantage of this scheme is that a comparison can be done Flip-flop to flip-flop. Fault analysis can be performed in a very detailed way, because internal comparisons can be produced.

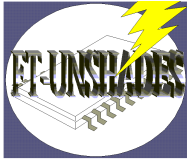
#### 4. Test tools

Test and aNalysis Tools (TNT) are a very important issue in this project, because they represent the main interface between the user and the highly complex test design procedures. Tools are presented through a dedicated command shell that has defined a set of commands that define a test environment for the design. *When* a fault is inserted during the test cycle, *Where* a fault is inserted and *How* the fault is represent an environment definition.

- *When* defines the clock cycle to insert the fault in the test period. Time can be redefined using time windows.
- *Where* defines the subset of the design Flip-flops candidate to receive a bit-flip. User can restrict the bit-flips to a subregion of the design.
- *How* defines the bit-flip model (double, triple, output filtered, time filtered ...) that is to be inserted or detected.

The test model has a definition level. The basic level is *damage*, where a fault is detected when an output discrepancy is found. Second level, called *output damaged* detects which output has discrepancy, and finally a *latent* level detects internal discrepancies of the complete set of Flip-flops.





A fault dictionary is created to rerun and analyze a particular test. Signals can be recorded using single stepping (one clock cycle) runs and offline analysis can be performed.

#### Conclusions

The FT-UNSHADES project was intended to be a test platform to assure that design protections are properly inserted, before place and route. The project goals have been achieved and surpassed; the project has obtained unexpected results: the design flow is completely automatic and the design treatment is as a black box.

The FT-UNSHADES system has been tested using third party benchmarks from ESA and OpenCores IPs. Promising results show that the system provides a powerful testing and analysis platform.

Other purposes of FT-UNSHADES are to produce a selective protected model for a design, to be a testing platform for design run-time debugging. Other applications less restrictive than space designs –aeronautics, automotive, health support- should take advantage from FT-UNSHADES because they can detect the weak parts of the design.

#### Future

Several Challenges of FT-UNSHADES are proposed for its promising future. The system can be reused to insert faults in the configuration memory instead of just Flip-flops. These experiments should evaluate the behaviour of Xilinx FPGAs in radiation environments as well as the scrubbing strategies for configuration error corrections. Other application is to provide information to XTMR tool in order to optimize the redundancy insertion. A damage level test should provide enough information for a Xilinx TMR constraints file. Finally a redesign of the board should provide a good testing platform ready to be inserted in a radiation facility. The testing scheme can be ported to a motherboard-daughterboard scheme for real radiation testing of a device preserving the information level obtained in the actual FT-UNSHADES.

---