FT-Unshades2 overview

19th Feb 2019

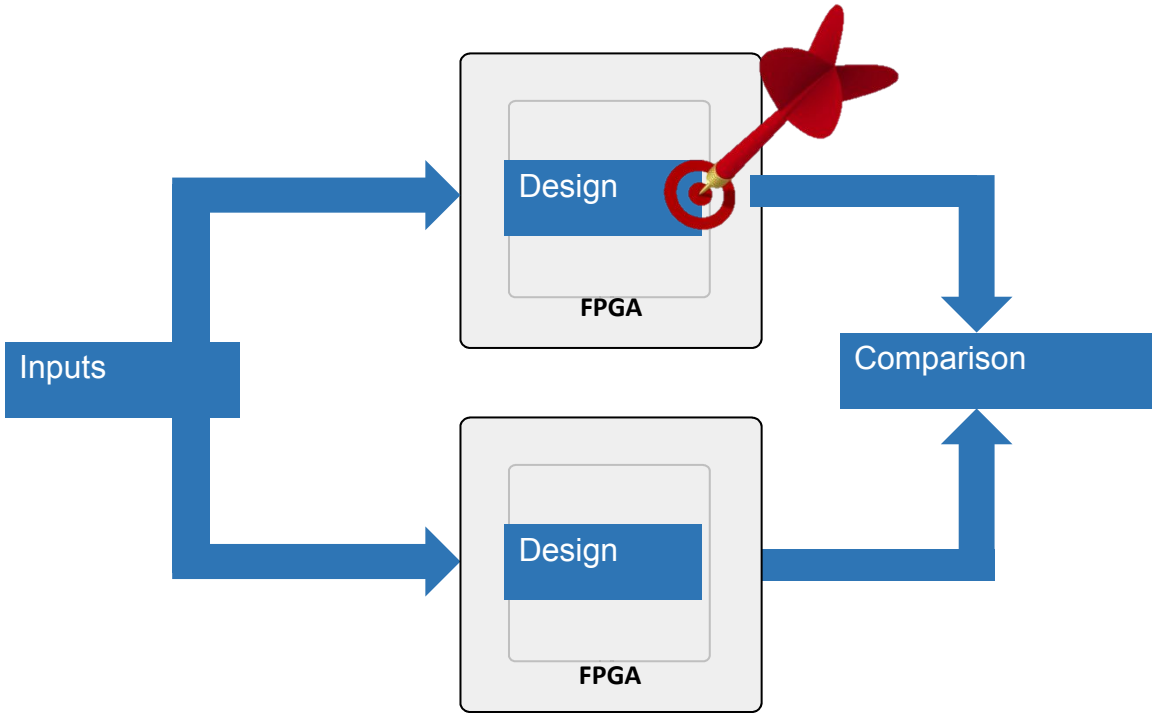
Contents

- Introduction
- Injection coverage
- Supported FPGAs
- User interfaces
- Inputs & outputs
- AFTU: Analog FTU
- Users & collaborations

Introduction

FT-Unshades2: FPGA-based fault injection emulator (SEUs).

Also an analog utility, **AFTU** (SETs).

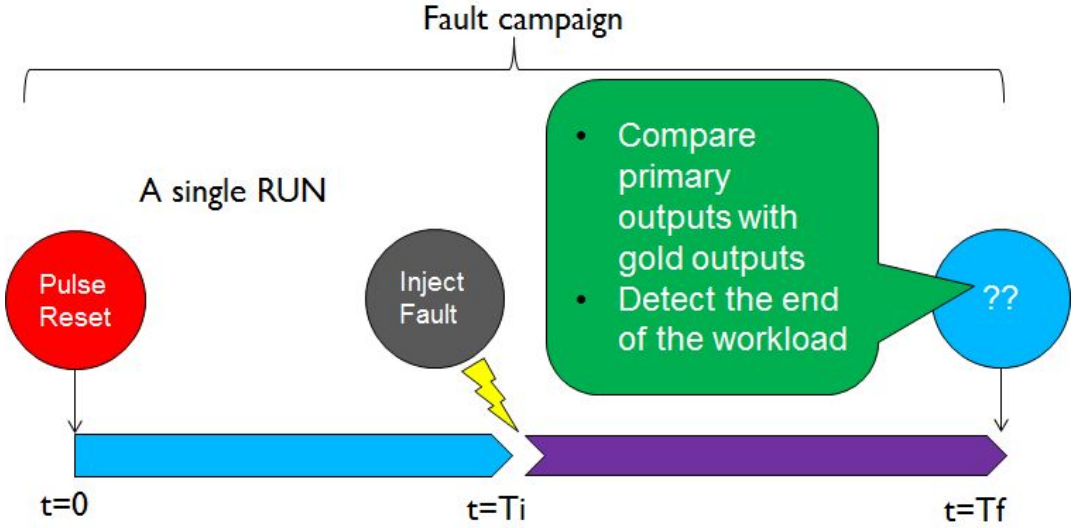


How does it work?

Design is prepared for the target FPGA

A campaign consists of multiple runs

1 Run: complete execution of test vectors + injection(s)

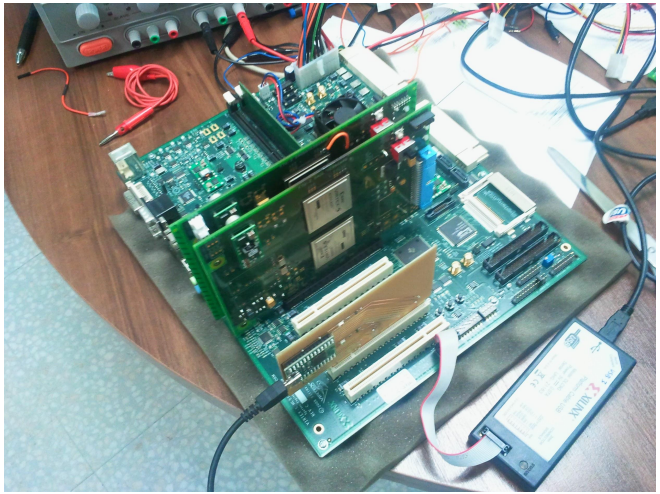


Injection coverage

Injections can be performed in:

- Flip-flops
- Block RAMs
- Distributed RAMs
- Configuration Bits (if in FPGA mode)

Supported FPGAs



Currently:

- Xilinx Virtex-5 (xc5vx70t)

Development:

- NanoXplore NG-MEDIUM

Planned:

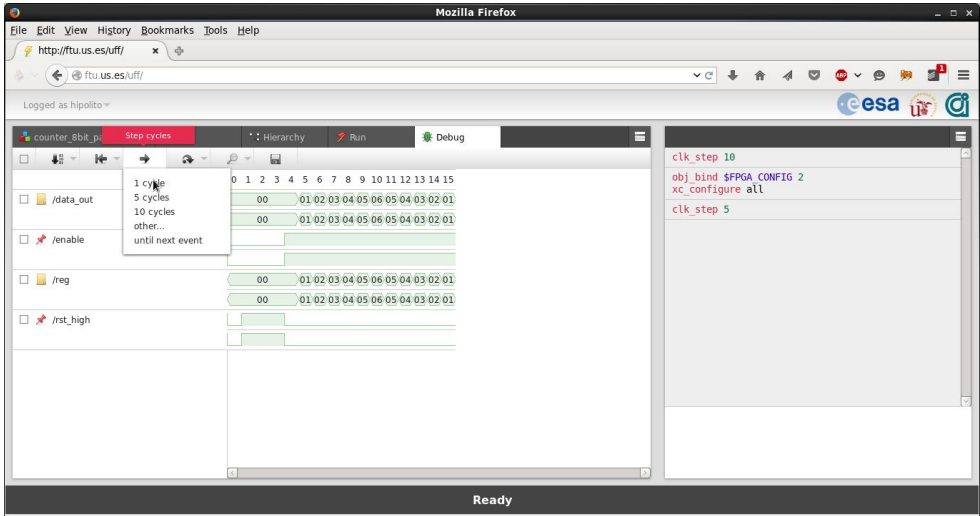
- Xilinx Kintex Ultrascale (KU060)
- Intel Arria 10

User Interfaces

Two user interfaces:

- Shell (tntsh)
- Web interface (UFF)

```
% dbg_ram_set 0x00210100 320 0
% clk_step 10
% dump_logs 0 20
/ram/RAMB36_X2Y25/B:BIT0.g: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
/ram/RAMB36_X2Y25/B:BIT0.f: 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
/registro/data_out.g: 00 00 00 00 00 00 48 48 48 48 48 48 48 48 48 48 48 48 4F 4
F 4F
/registro/data_out.f: 00 00 00 00 00 00 48 48 48 48 48 48 49 49 49 49 49 49 4F 4
F 4F
% dbg_frame_read 0x00210100
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 07 02 F6 0B 00 00 00 00
00 00 00 00 00 00 00 00 00 02 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 02 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 02 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00
```



Inputs & Outputs

Inputs:

- **Design Under Test** (must be synthesized for Virtex-5)
  - Either HDL, post-synthesis, or the user can generate the bitstream
  - A simple pin definition file (.pin) must be defined to automatically generate the correct (.ucf)
- **Test Vectors** (.vcd) or testbench from which to generate them

Inputs & Outputs

Outputs:

- **Log** of all SEU injections and output damages
- **Architectural Vulnerability Factor**, both total and by hierarchy
- **Faulty output patterns** can be recorded for further studies (feed them into a model of another part of the system)

If the number of possible injections is too high, statistical analysis can be applied.

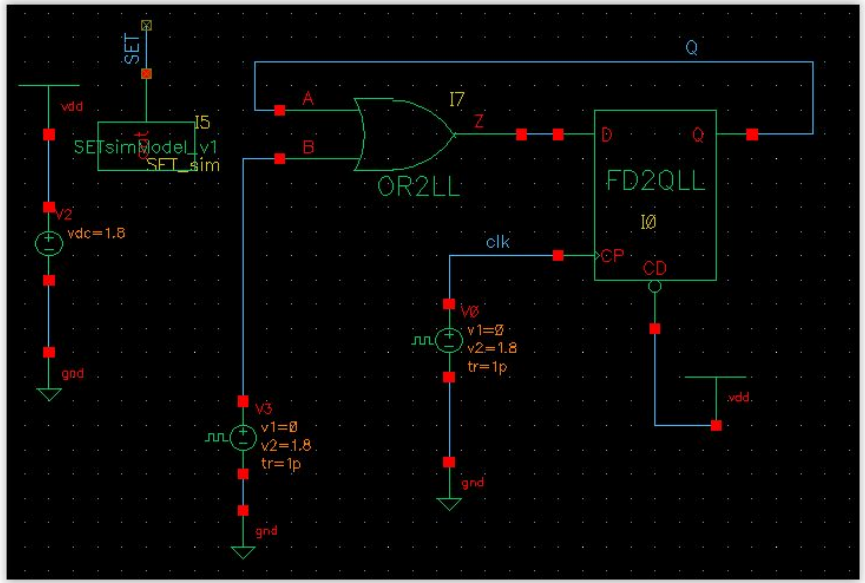
AFTU

What is AFTU?

The Analog FTU Hardware Debugging System is a tool to evaluate the SEE sensitivity of analog/mixed signal circuits at transistor level

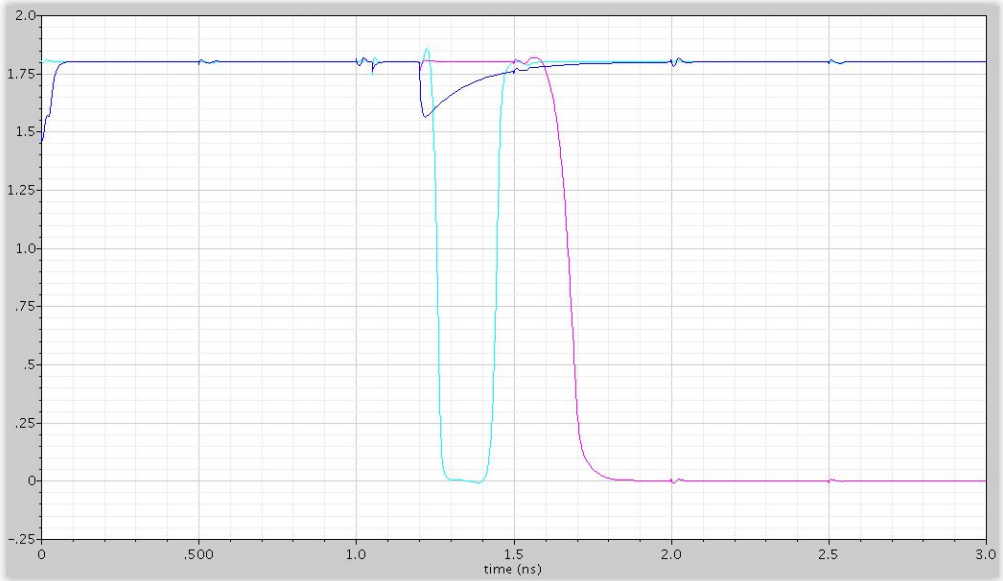
How does it Work?

AFTU takes a Spectre design...



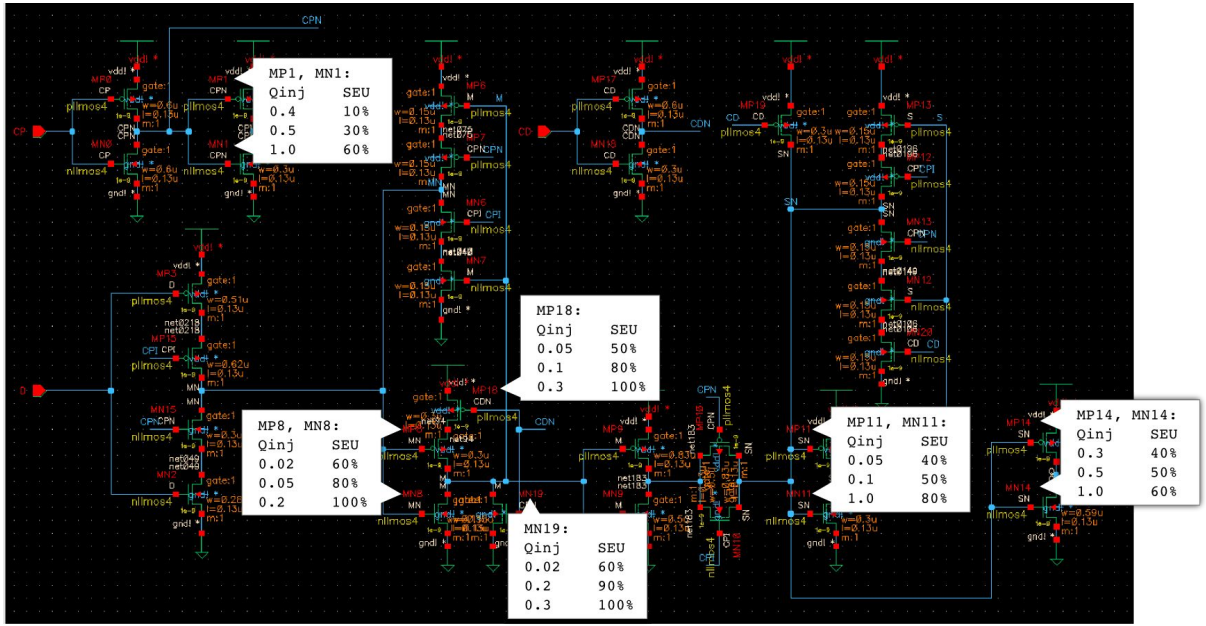
How does it Work?

...emulates radiation conditions...



How does it Work?

... and evaluates vulnerabilities



Analog FT-Unshades

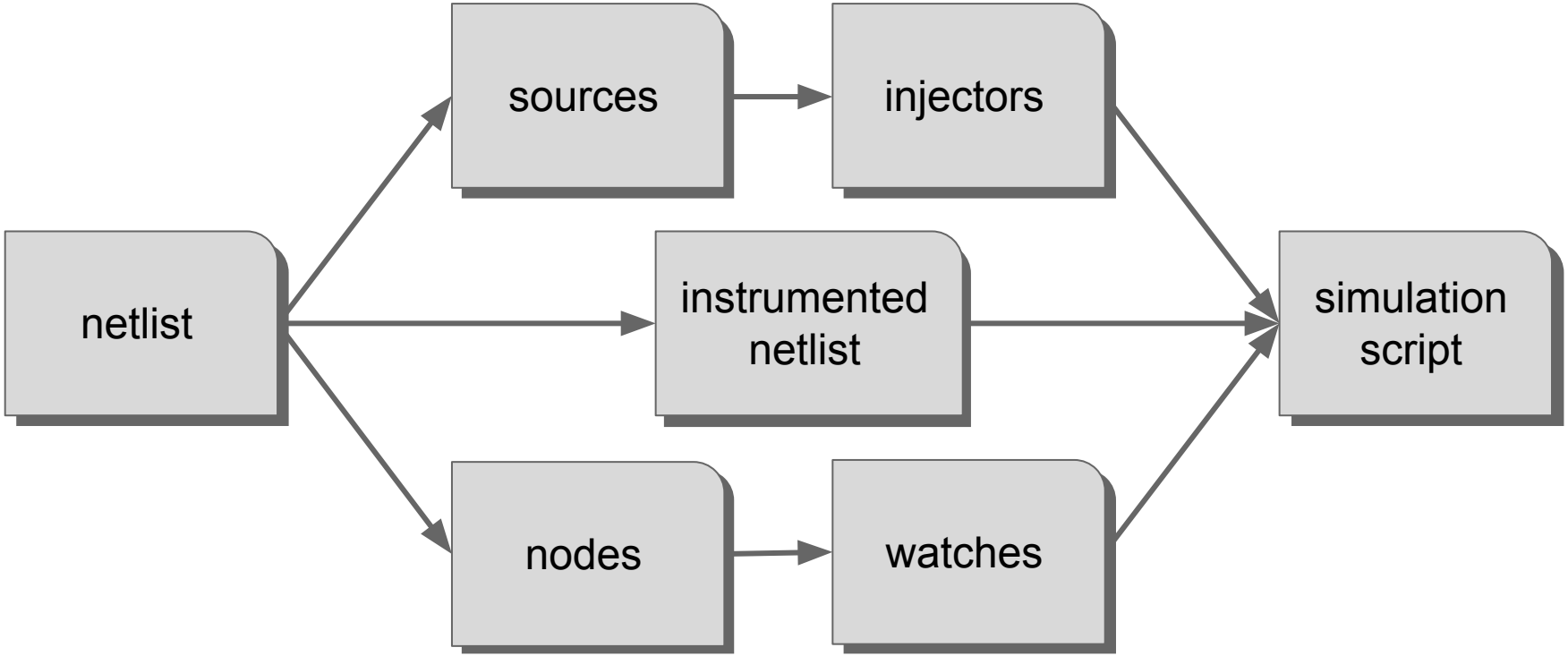
What does it do?

- Analyzes the effects of radiation on analog circuits

How does it do it?

- Instrument a circuit
- Create injectors
- Create watches
- Build the actual simulation

AFTU Workflow



Users & collaborations

- European Space Agency
- CERN (RD-50 collaboration)
- VEGAS project (NanoXplore, ST Microelectronics, Politecnico di Torino, Thales Alenia Space, CRISA, Airbus D&S, Universidad de Sevilla)
- TU Braunschweig (analysis of the NoC of Solar Orbiter's SoPHI instrument)
- IHP Microelectronics (evaluate radiation sensitivity of digital cells)
- CNA (National Accelerators Centre, Sevilla, Spain)

Contact

Hipólito Guzmán Miranda: **hguzman@us.es**

Fernando Muñoz Chavero: **fmunoz@us.es**