Analysis of SCU and MCU effects in SRAM-based FPGAs

Niccolò Battezzati

Filomena Decuzzi

Luca Sterpone

Massimo Violante

Goal

- To present an automatic approach to identify, without simulations, sensitive bits of the configuration memory of SRAM-based FPGAs
  - Xilinx Virtex II and 4
  - Atmel AT40K and ATF280E

Outline

- Introduction
- Static analysis
- Conclusions

Introduction

- Two major questions are coming from designers:
  - How can I anticipate radiation-effects analysis?
    - Important to start thinking about radiation effects before prototypes are available
  - Millions of bit are inside the bitstream: how many of them are really sensitive?
    - Important to focus fault injection only on meaningful bits to save time, and to de-rate device cross-section to obtain design cross-section
  - The bit XYZ in the bitstream is leading the circuit to fail: which part of the design it refers to?
    - Important to debug the design quickly and accurately

Our algorithm

1. Read the place & routed design and build the netlist/bitstream association
2. For each bit of the bitstream:
   A. Flip the bit and update accordingly the netlist
   B. Is the original netlist corrupted (the error arrives to outputs)?
      I. Yes → the bit is sensitive
      II. No → the bit is not sensitive

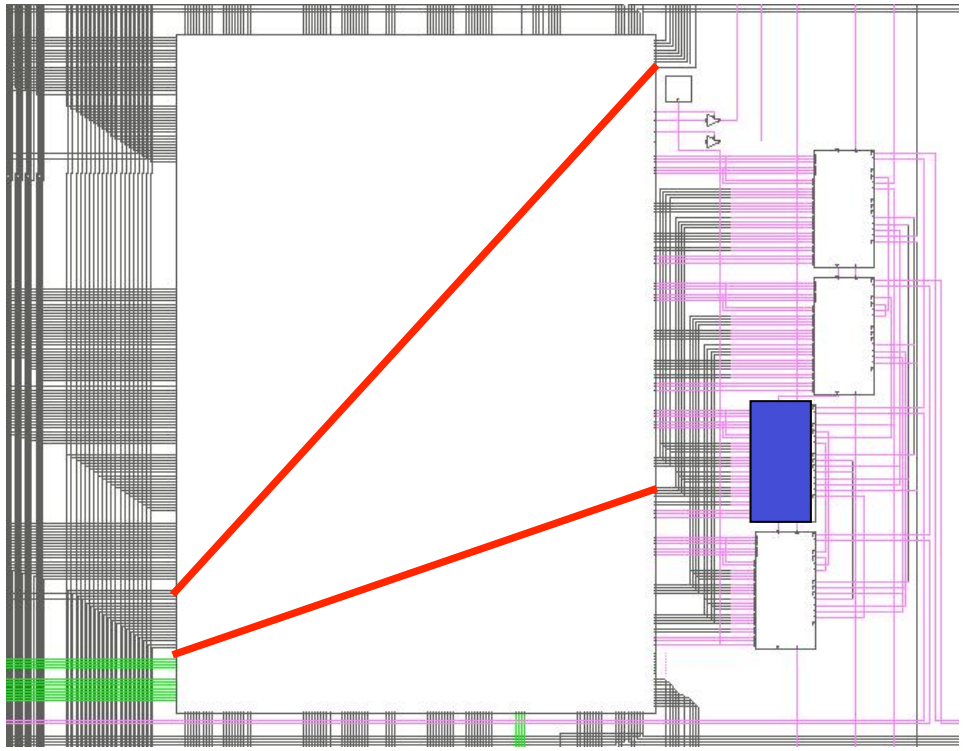
- Analysis is done looking at the error propagation path, and it does not consider workload

An example: original circuit

- The bitstream

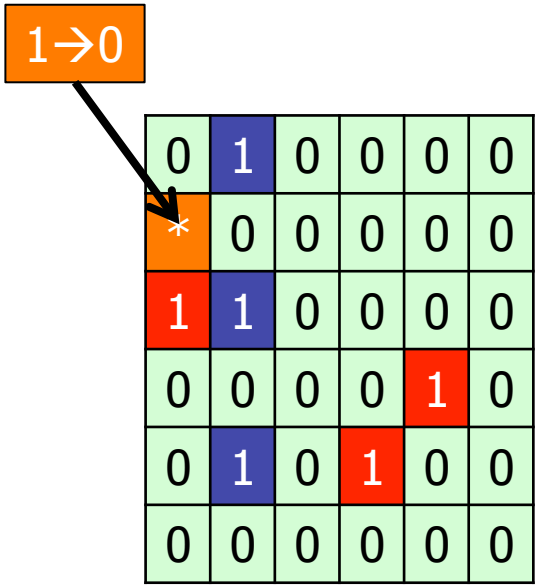
0	1	0	0	0	0
1	0	0	0	0	0
1	1	0	0	0	0
0	0	0	0	1	0
0	1	0	1	0	0
0	0	0	0	0	0

- The original netlist

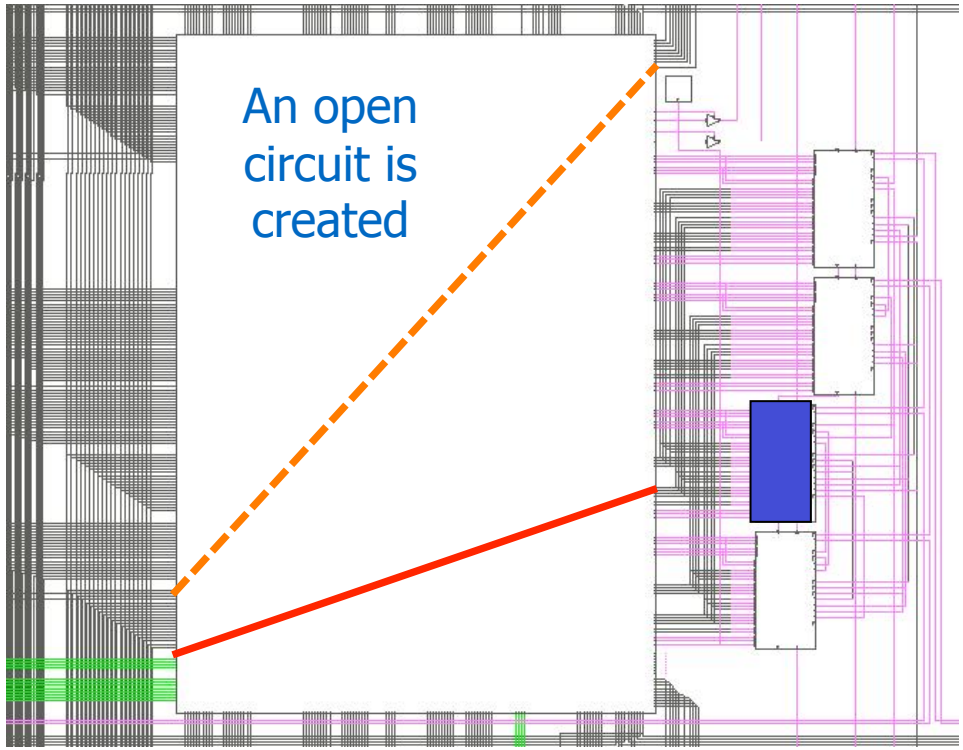


An example: sensitive bit

- The bitstream

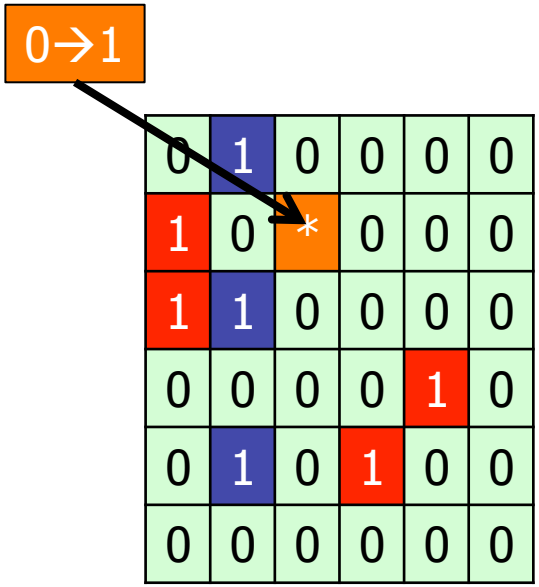


- The corrupted netlist

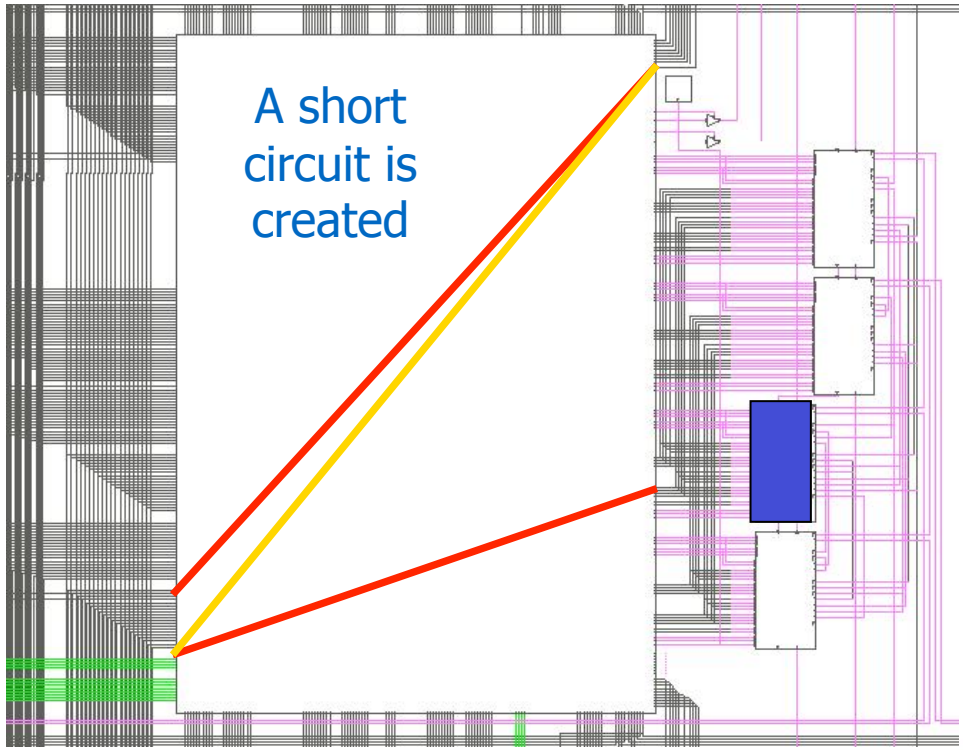


An example: sensitive bit

- The bitstream



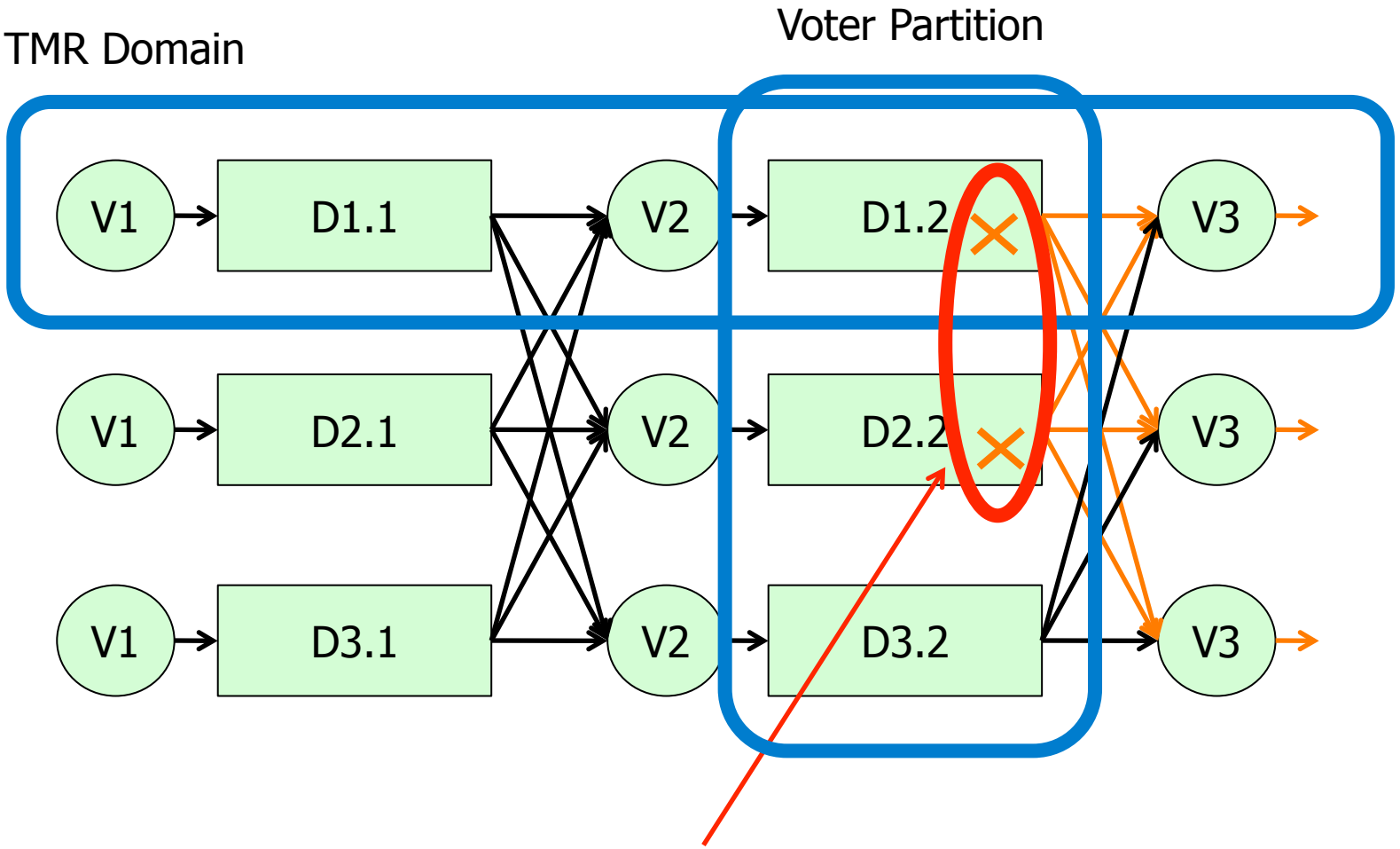
- The corrupted netlist



Operational modes

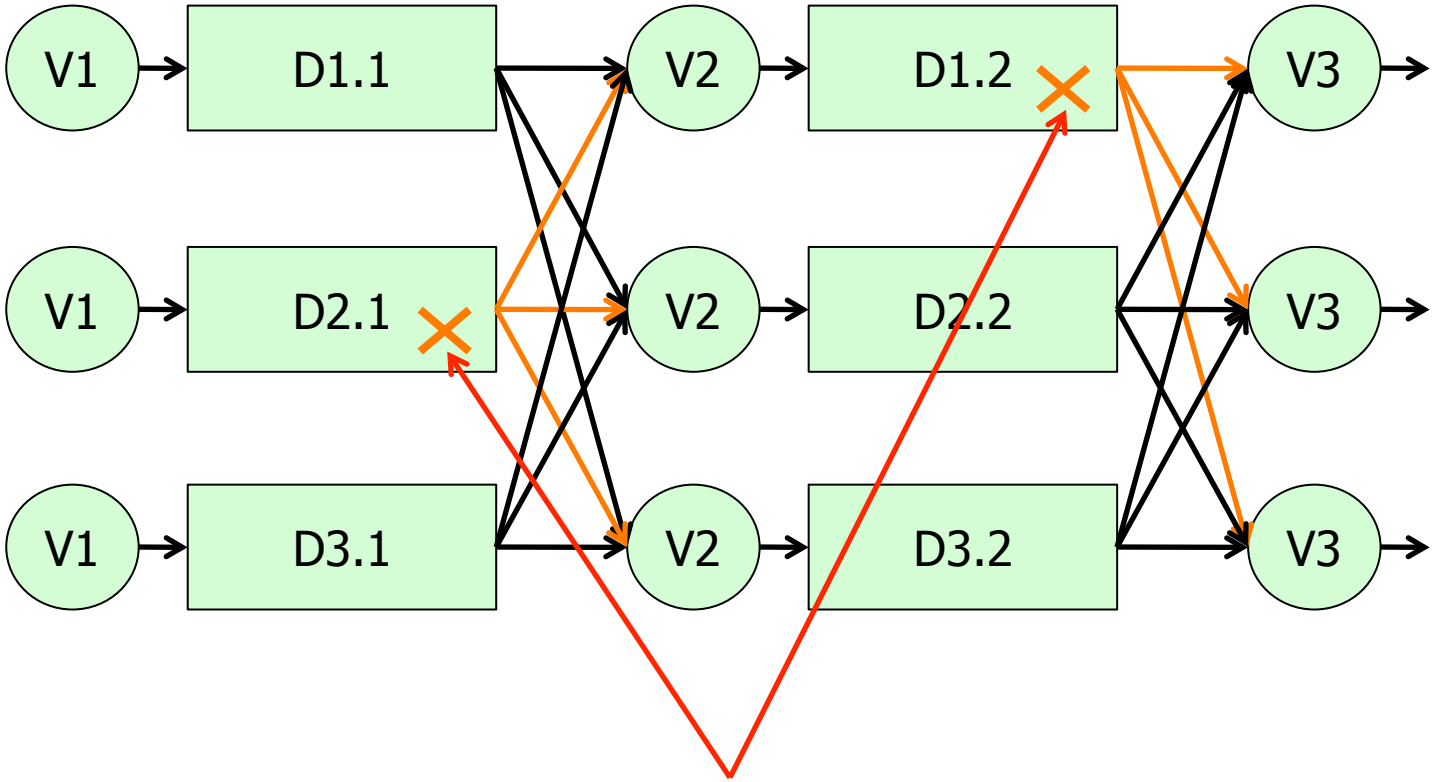
- Discovery mode: it analyzes the bitstream while neglecting mitigation schemes
  - Lists sensitive bits
- TMR mode: it analyzes the bitstream while automatically recognizing (X)TMR mitigation scheme
  - Lists bits that violate (X)TMR scheme (domain crossing events)
  - List bits that produce warnings (may lead to domain crossing events in case of accumulation)

Domain crossing events



One Single Cell Upset (SCU) in the configuration memory provokes two circuit modifications in two TMR domains in the same TMR partition →
The faults propagate beyond the voter boundary

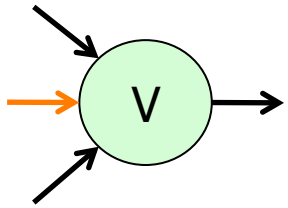
Warnings



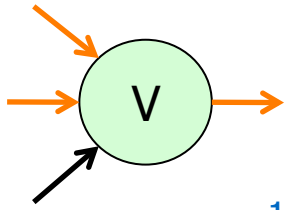
One SCU in the configuration memory provokes two circuit modifications in two voter partitions → The faults stop at the voter boundary

TMR-mode algorithm

- The algorithm recognizes automatically TMR domains, voters, and voter partitions
- Forward error propagation:
  1. Find all the paths from the fault site to the circuit outputs, or memory elements
  2. Is the fault propagating to only one of the voter inputs?
     A. Yes → the bit is not sensitive

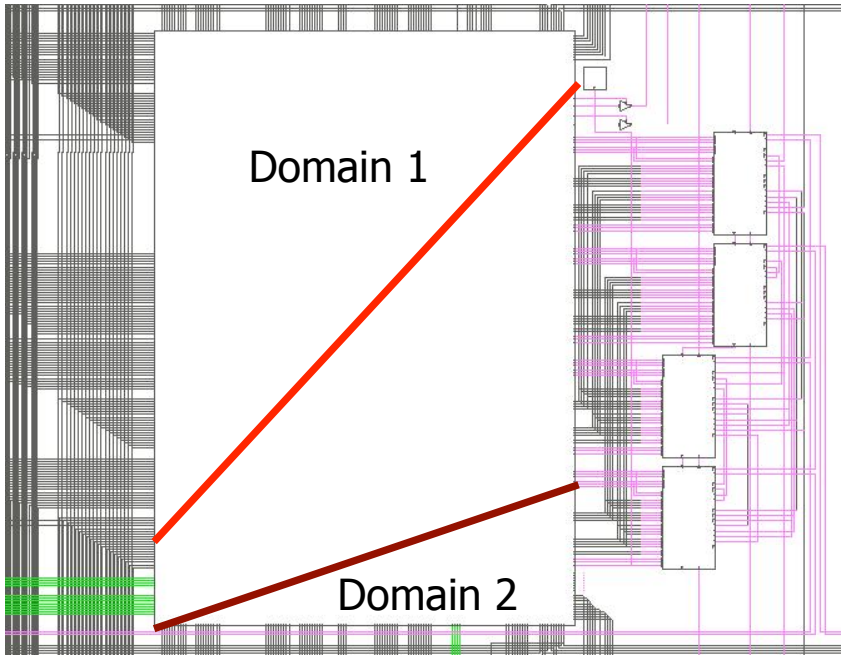


     B. No → the fault propagates to at least two inputs of a voter in the same partition → the bit is sensitive

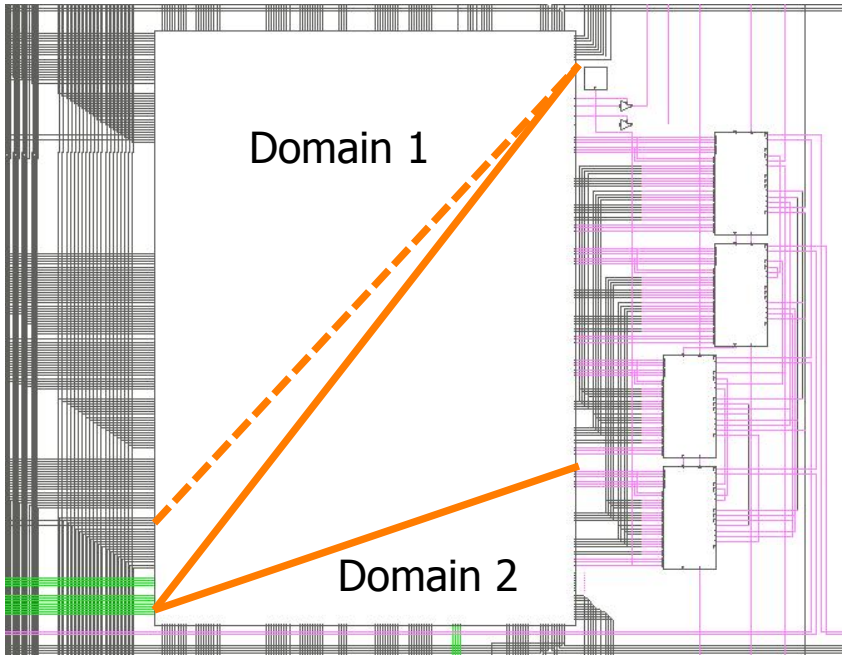


Why (X)TMR may fail?

Original netlist

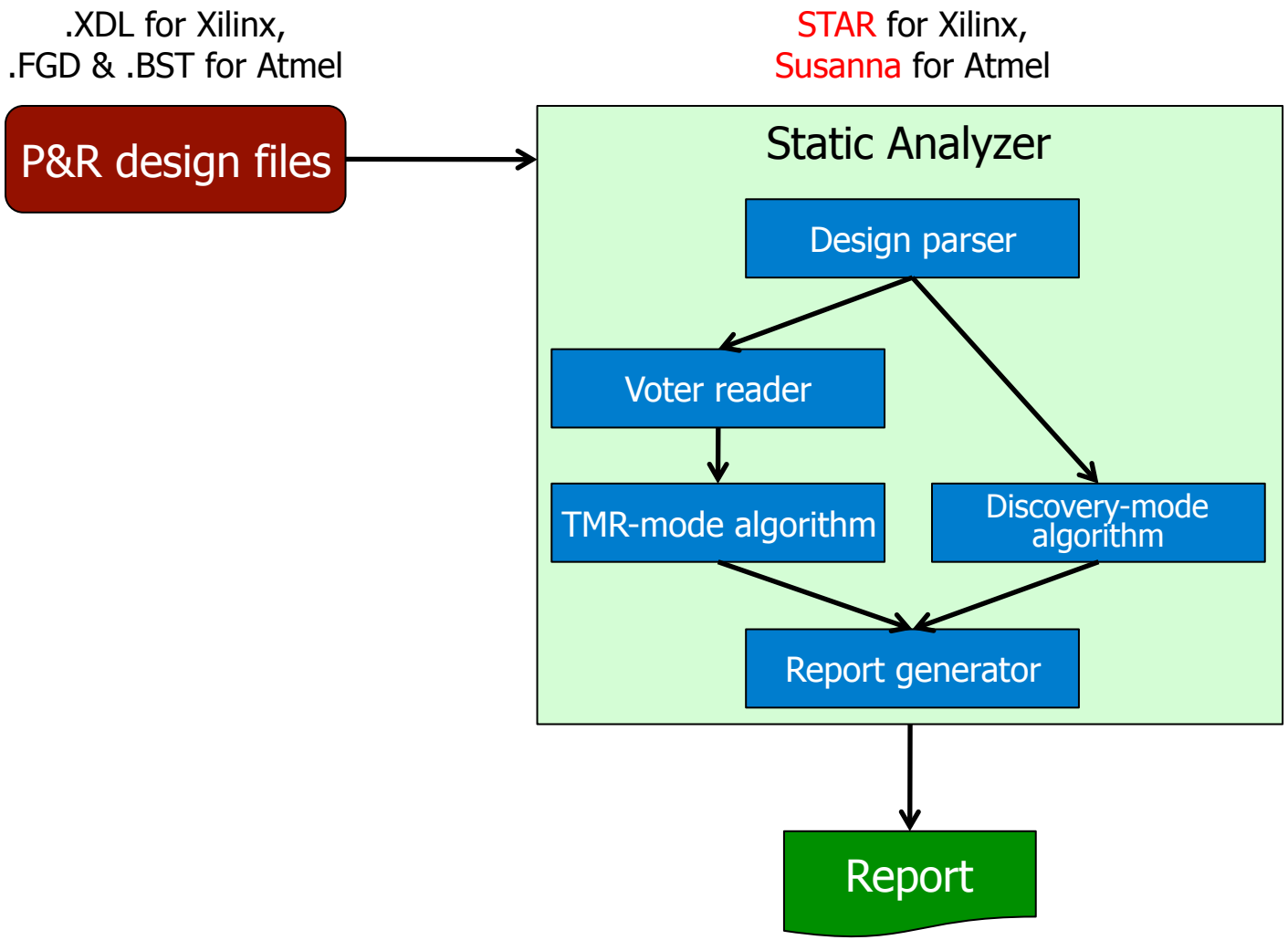


SCU-corrupted netlist



- Dependability-oriented placement constraints and routing rules should be enforced to improve (X)TMR robustness

The Flow


.XDL for Xilinx,
.FGD & .BST for Atmel
STAR for Xilinx,
Susanna for Atmel
P&R design files
Static Analyzer
Design parser
Voter reader
TMR-mode algorithm
Discovery-mode algorithm
Report generator
Report

The report

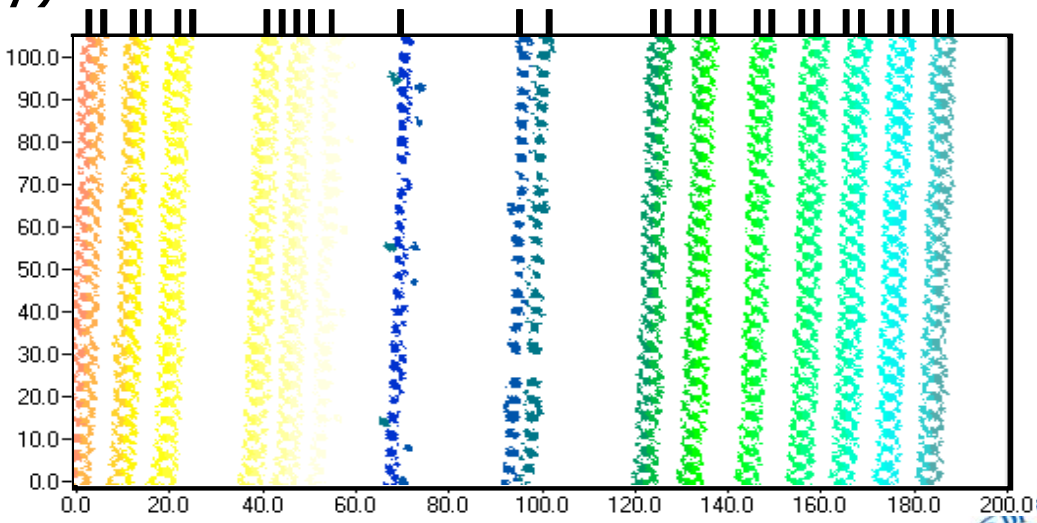
- Detailed report is produced for Xilinx devices

```
Resource: PIP Block Adr 0 Maj Add 6 Min Add 14 Bit 156
Involved PIP : Y1 -- S2BEG2
FAR: 0x000c1c00 Bit: 156
Net = data_bus_IBUF_TR
```

- Less detailed report is produced for Atmel devices (work in progress to improve it)
  - Addresses of sensitive bits
  - Sensitive resources
  - Name of the corresponding design module

Supported fault models

- Single Cell Upset (SCU)
- Multiple Cell Upset (MCU)
  - Growing phenomena at each new generation of devices
  - STAR includes layout information about the analyzed device (Virtex II, only)
- Accumulated SCU



Possible use cases

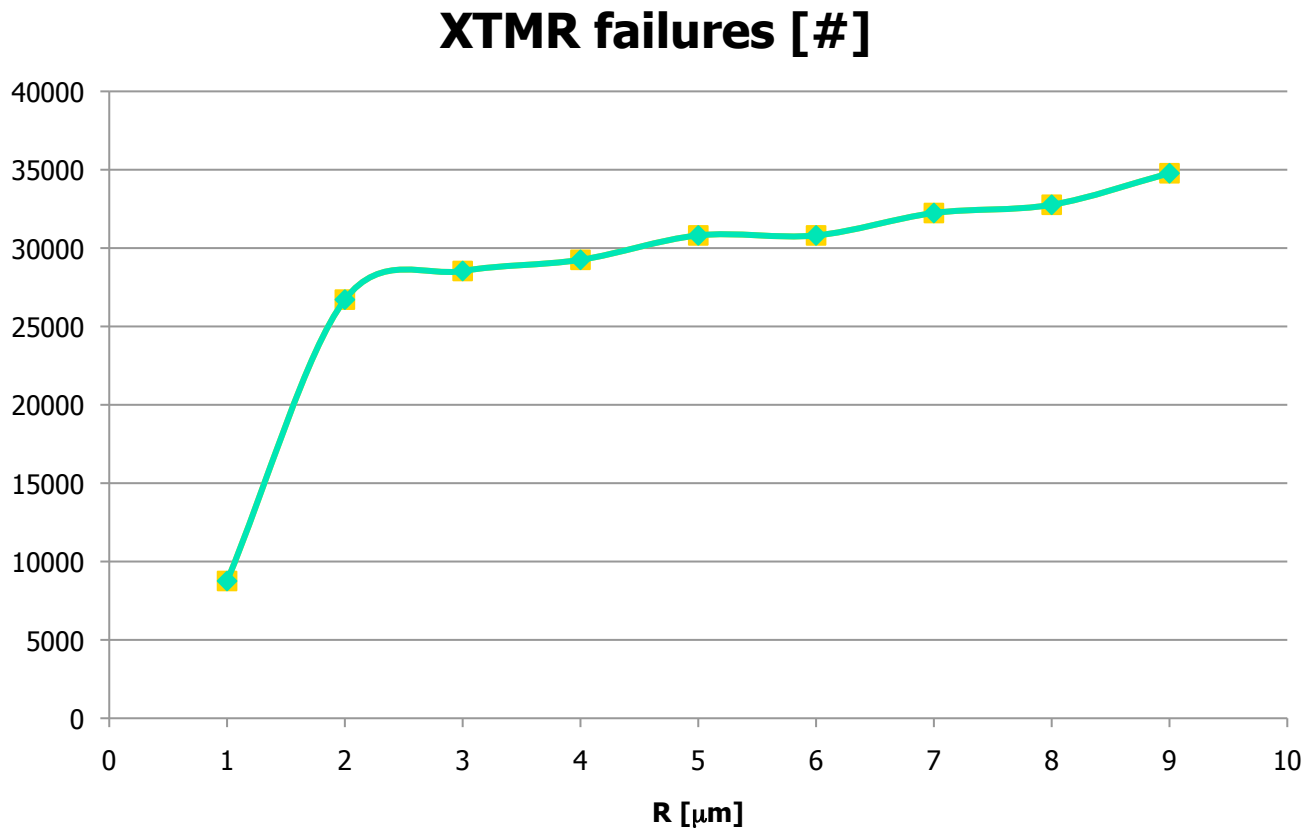
- Circuit debug: the name of the EDIF resources corresponding to sensitive bits can be exploited for this purpose
- Fault injection: the addresses of sensitive bits can be exploited for this purpose
- Radiation testing: the above data can be used during post-processing of radiation testing results

A few figures

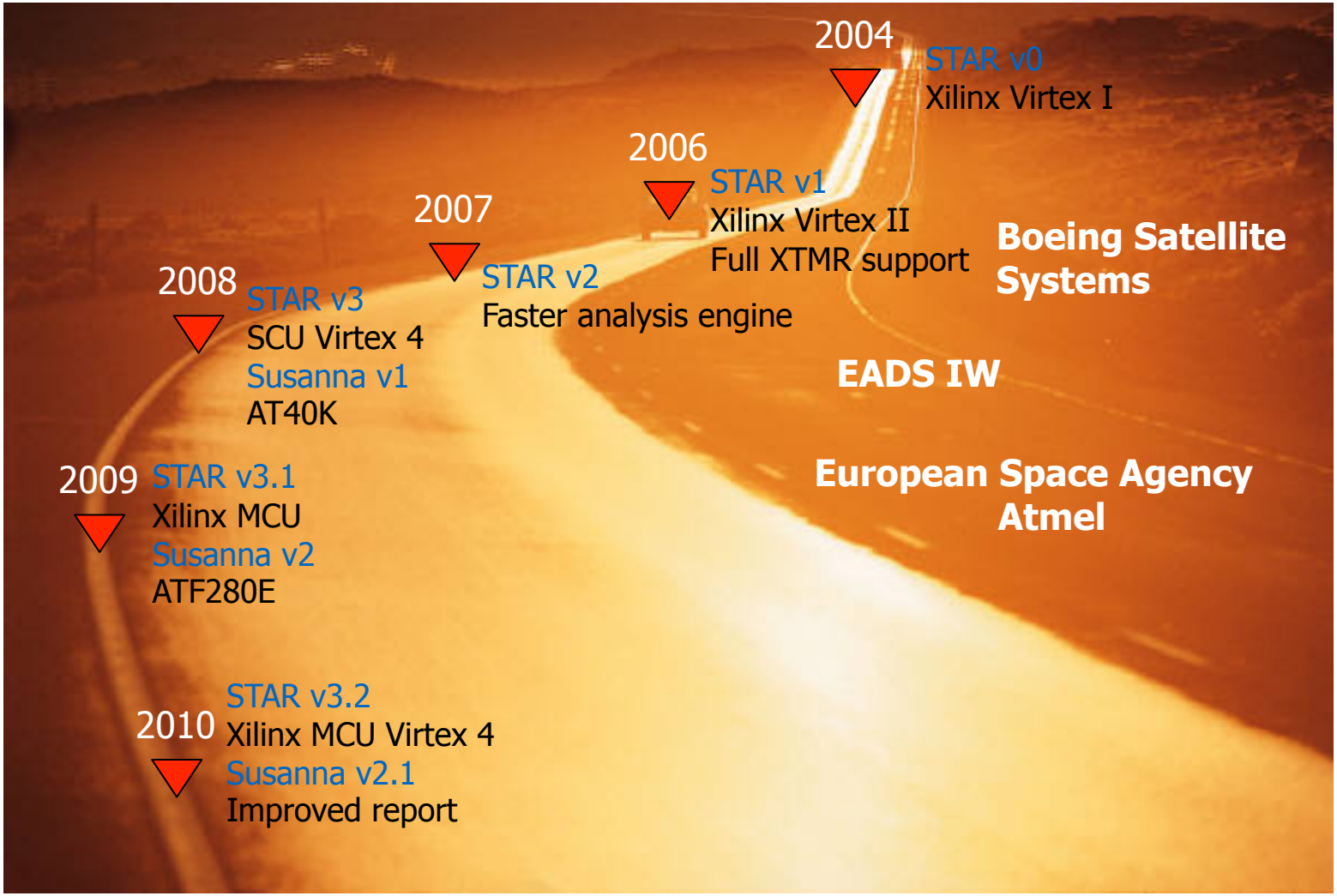
- STAR discovery mode, SCU analysis
  - Design: leon3mp (2,933 PIPs; 10,470 CLBs)
  - Device: xc2v6000
  - Machine: HP hw8400 (Xeon 3 GHz, 8GB RAM)
- Summary of results:
  - 2,107,285 sensitive bits (621,818 programmed; 1,485,467 not programmed)
    - PIP-related bits: 1,672,504
    - CLB-related bits: 434,781
- CPU time: 530 sec

A few figures

- MCU analysis for a benchmark circuit



Roadmap


2004
STAR v0
Xilinx Virtex I
2006
STAR v1
Xilinx Virtex II
Full XTMR support
2007
STAR v2
Faster analysis engine
2008
STAR v3
SCU Virtex 4
Susanna v1
AT40K
2009
STAR v3.1
Xilinx MCU
Susanna v2
ATF280E
2010
STAR v3.2
Xilinx MCU Virtex 4
Susanna v2.1
Improved report
Boeing Satellite Systems
EADS IW
European Space Agency
Atmel

Factsheet

- STAR/Susanna coded in C/C++, run on Windows/ Unix/Linux
- STAR
  - Supports Virtex II/IIpro/4
  - Seamless integration with fault injection tool (FLIPPER and FT-UNSHADES)
- Susanna
  - Supports AT40K/ATF280E

Conclusions

- Automatic solutions are available for investigating upsets in the configuration memory of SRAM-based FPGAs
- Support available for Xilinx Virtex II/4 devices, and Atmel AT40K/ATF2780E devices
- Seamless integration with fault injection is available
- Seamless integration with automatic tools for design mitigation
  - VPLACE/RoRA → see next presentation

Contacts

Massimo Violante

massimo.violante@polito.it

Politecnico di Torino

Dipartimento Automatica e Informatica

C.so Duca degli Abruzzi 24

10129 Torino, ITALY

www.cad.polito.it