Reconfigurable FPGAs in radioactive environment challenges and possible solutions

Massimo Violante

Luca Sterpone

Politecnico di Torino

Dip. Automatica e Informatica

Torino, Italy

Politecnico di Torino

- Leading Engineering School in Italy, founded in 1859
- Department of Automation and Computer Engineering
  - 60 faculties, 80 research assistants, 20 staff
  - 100 PhD students
  - Annual budget (~5 M€)



Electronic CAD & Reliability Group

- Its mission is to support, through techniques, tools, and services, the designer of electronic circuits and systems
  - 7 faculties, 3 research assistants, 6 PhD students
- Strong cooperation with major industries, agencies, and research centers world-wide
- Reliable COTS-based embedded systems
  - Prof. Massimo Violante, and Dr. Luca Sterpone
  - 1 research assistant, 2 PhD students

www.cad.polito.it

Goal

- To illustrate the challenge in designing with reconfigurable FPGAs in radioactive environment
- To illustrate possible solutions (design flow)

Outline

- Introduction
- Xilinx Virtex II & 4
- Actel ProASIC
- Conclusions

Outline

- Introduction
- Xilinx Virtex II & 4
- Actel ProASIC
- Conclusions

Introduction

- A number of reconfigurable FPGAs today available
  - SRAM-based FPGAs: Xilinx, Altera, SiliconBlue, Atmel, …
  - Flash-based FPGAs: Actel
- Plenty of experimental data available showing sensitivity to radiations, but
- Limited support to help designers
  - Some design tools
  - Some application notes
  - Lack of established design flows including: architecture, implementation and validation

Activities at Polito

- Driver: space-related agencies/companies
- SRAM-based FPGAs → Xilinx Virtex II & 4
  - Development of design methods and tools
  - Fault injection: in cooperation with INAF (Milano, Italy), and Univ. of Sevilla (Sevilla, Spain)
  - Radiation testing: in cooperation with Univ. of Padova (Padova, Italy)
- Flash-based FPGAs → Actel ProASIC
  - Development of design methods and tools
  - Fault injection
  - Radiation testing: in cooperation with ESA (Noordwijk, NL)

Outline

- Introduction
- Xilinx Virtex II & 4
- Actel ProASIC
- Conclusions

Why Virtex-II & 4?

- Package qualified successfully for space use
- Extensive database concerning radiation effects/ device reliability
- Availability of design tools
- Newer devices like Virtex-5QV rad-hard are under scrutiny but too new to be adopted without further investigations
  - Package is the biggest issue at the moment
  - Limited availability of radiation/reliability data
  - ITAR

Why Virtex-II & 4?

- Virtex-II & 4 appealing when…
  - Need for large device with plenty of resources
  - Need for high performance
  - Need for in-flight reconfiguration capabilities
- SEL, TID
  - SEL free ($LET_{TH(H.I.)}$=100 MeV-cm^2/mg)
  - TID > 250 krad(Si) ~50.0 rad(Si)/sec (Virtex-4QV)

However

- SEEs are an issue and must be mitigated
- Design validation is crucial

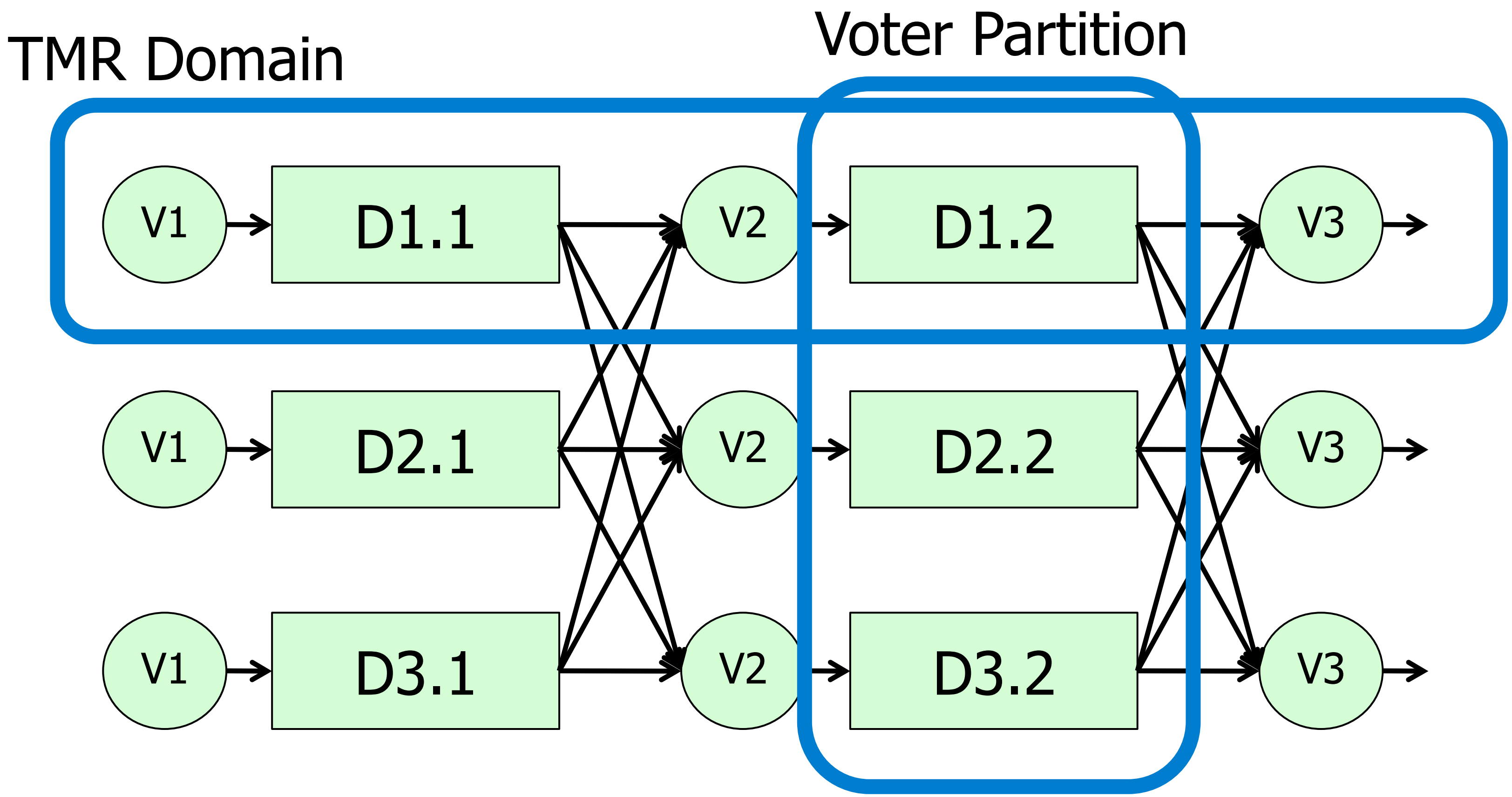
SEE of concern

- SEU/MCU in the device configuration memory
- SEU/MCU in the user memory
- SEU/MCU in hard-IPs
- SEFI
- SET (although difficult to observe)

How to approach the design

- Proper system architecture is needed
  - Payload FPGA, Configuration memory scrubber, SEFI monitor
- SEE-aware design goes in the payload FPGA
  - TMR
- Some vendors provide support for TMR
  - TMRtool from Xilinx, Precision RT from Mentor Graphics, Symplify Pro from Synopsys
- They allow
  - TMR insertion, safe-FSM encoding

Typical architecture



Observation

- All design techniques are based on the single-fault assumption (1 SEE = 1 fault in the design)

But

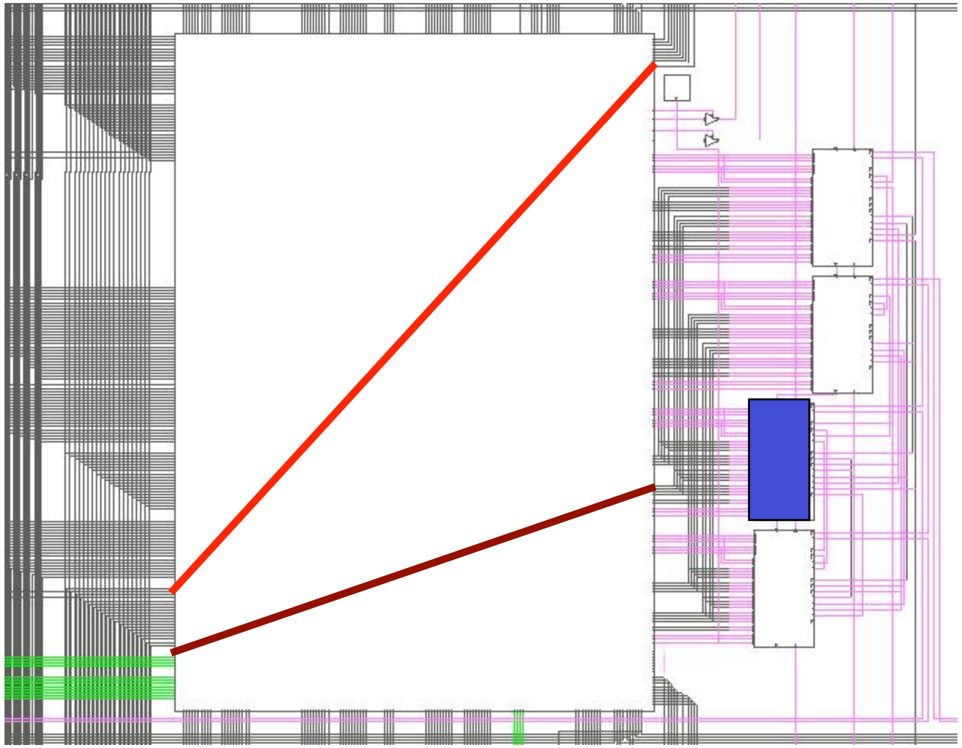
- SEE in the configuration memory may produce multiple faults

An example: original circuit

- The bitstream

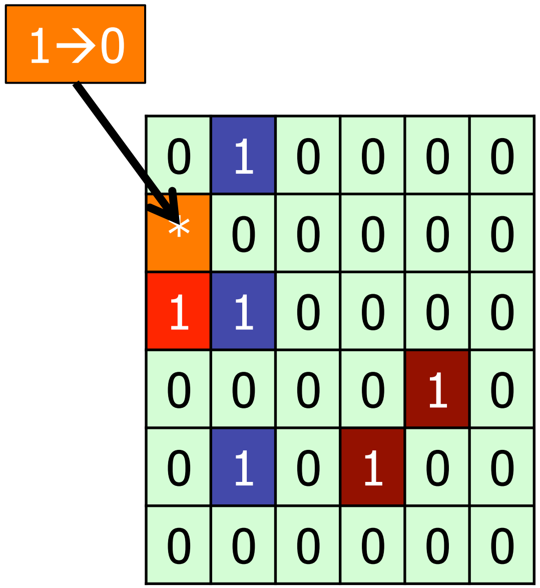
0	1	0	0	0	0
1	0	0	0	0	0
1	1	0	0	0	0
0	0	0	0	1	0
0	1	0	1	0	0
0	0	0	0	0	0

- The original netlist

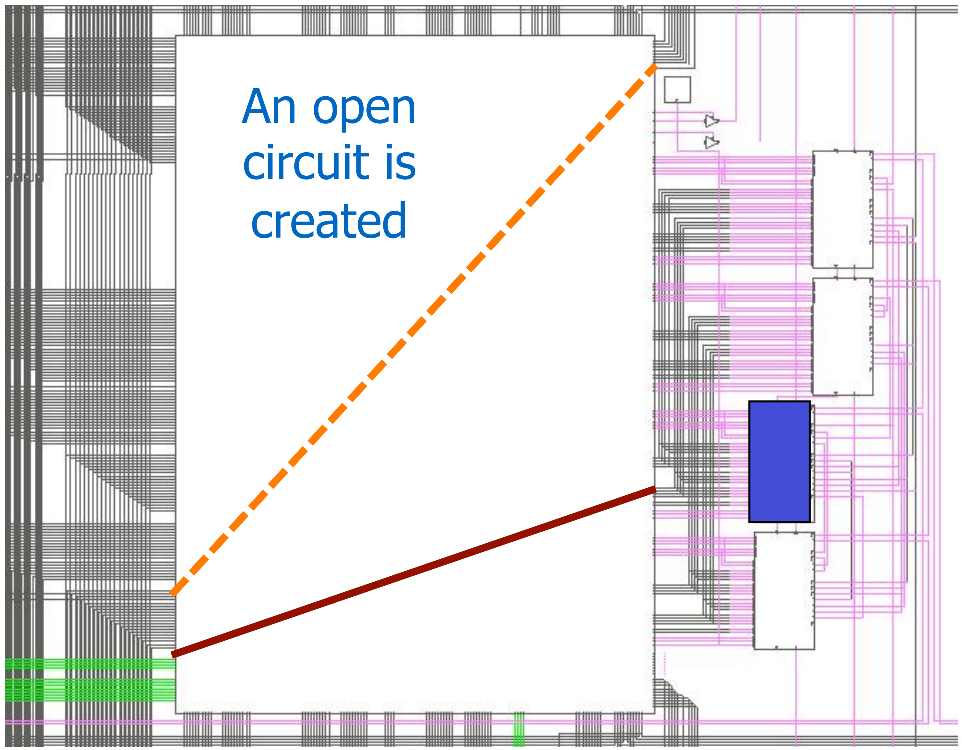


An example: single effect

- The bitstream

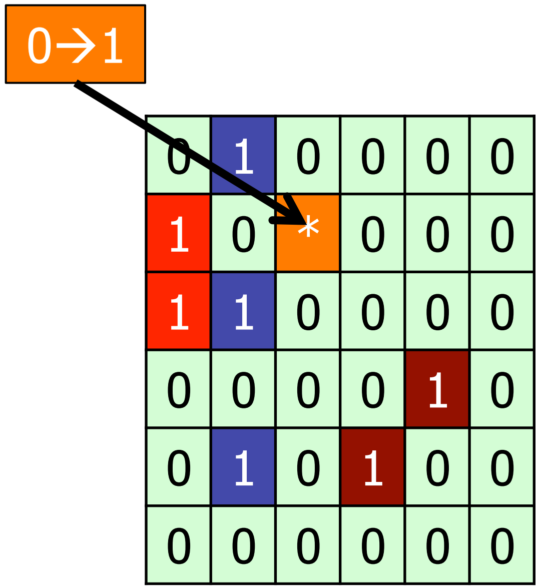


- The corrupted netlist

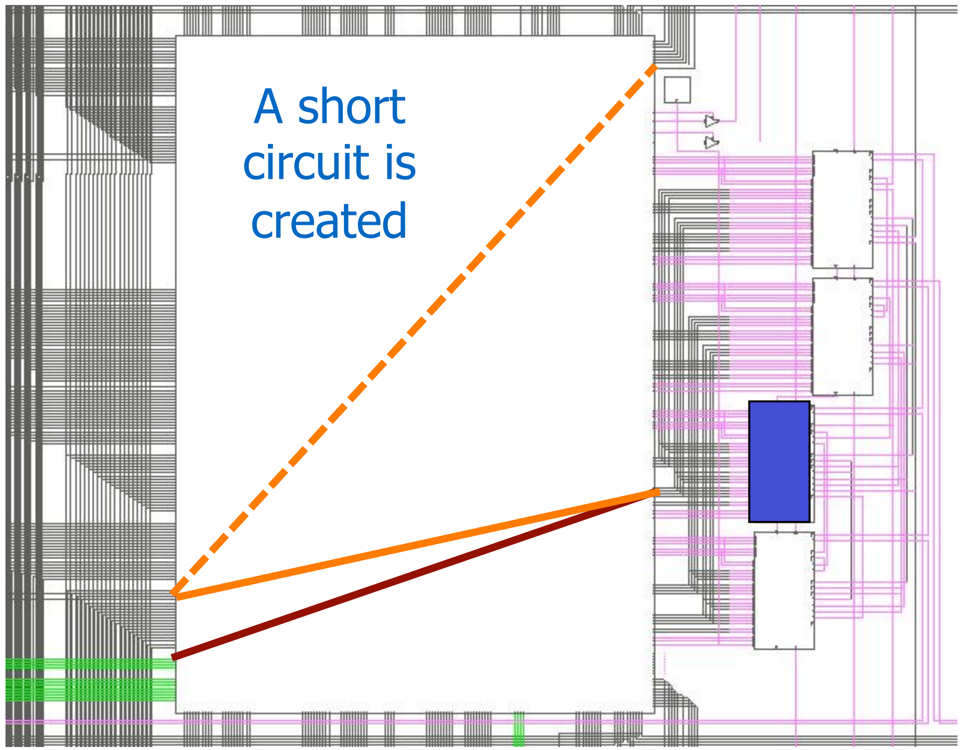


An example: multiple effects

- The bitstream

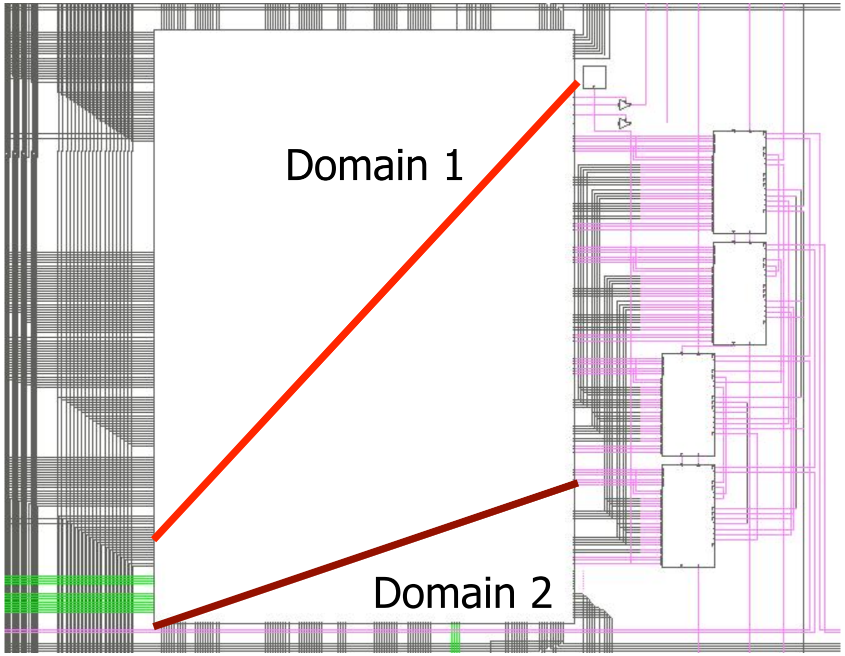


- The corrupted netlist

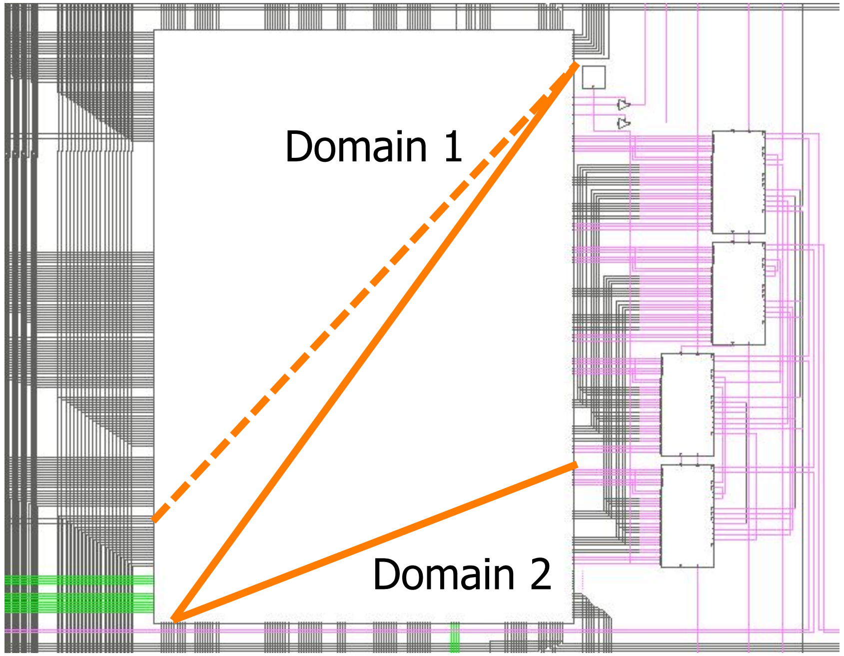


Why TMR may fail?

Original netlist



SEE-corrupted netlist



- The SEE modifies the same signal in two domains → SEE is producing multiple effects not masked by voters

An example

- Design: X-TMR design
  - In theory any SEE should be mitigated
- Fault injection in the device configuration memory

Resource	Failure
LUT	26
Global routing	1,497
CLB Local routing	45
CLB configuration	0
Total	1,568

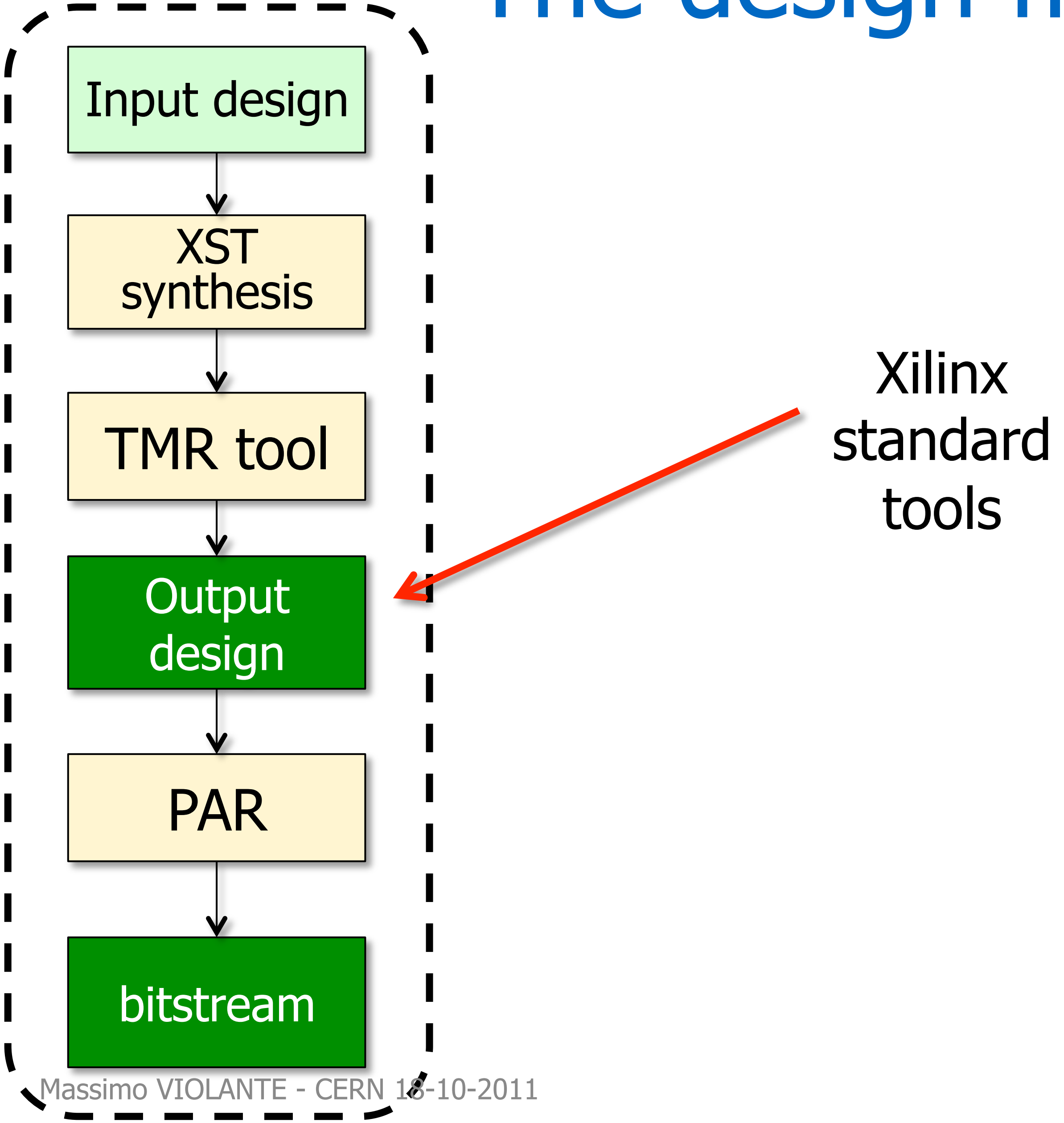
Where is the bug?

- Design tools (TMRtool, Precision RT,…) work at HDL-level only
- Design implementation is done using standard place & route that tend to pack designs tightly
  - Minimize device occupation
  - Minimize delay
- Slices and switch matrices are shared by different TMR domains → SEE may induce multiple errors affecting different TMR domains at the same time!
- A SEE-aware implementation flow is needed

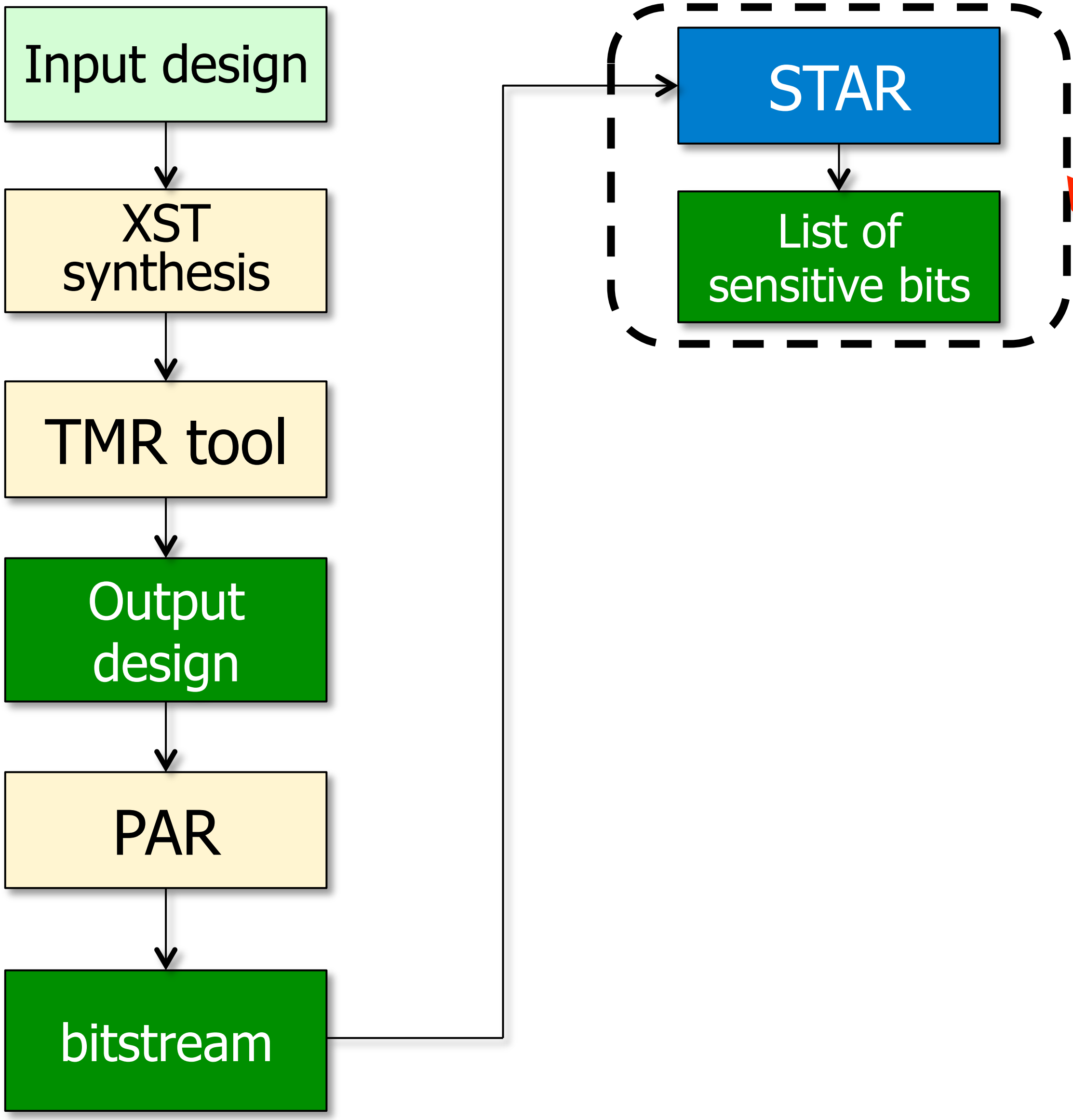
Open questions

- How can I predict radiation-effects analysis?
  - Anticipate analysis before going to beam testing
- Millions of bit are inside the configuration memory: how many of them are really sensitive?
  - Device cross-section vs design cross-section
- The bit 0x000c1c00, offset 156 is leading the circuit to fail: which part of the design it refers to?
  - Debug the design quickly and accurately
- How can I implement my design safely?
  - Avoid SEE effects escaping my architecture

The design flow

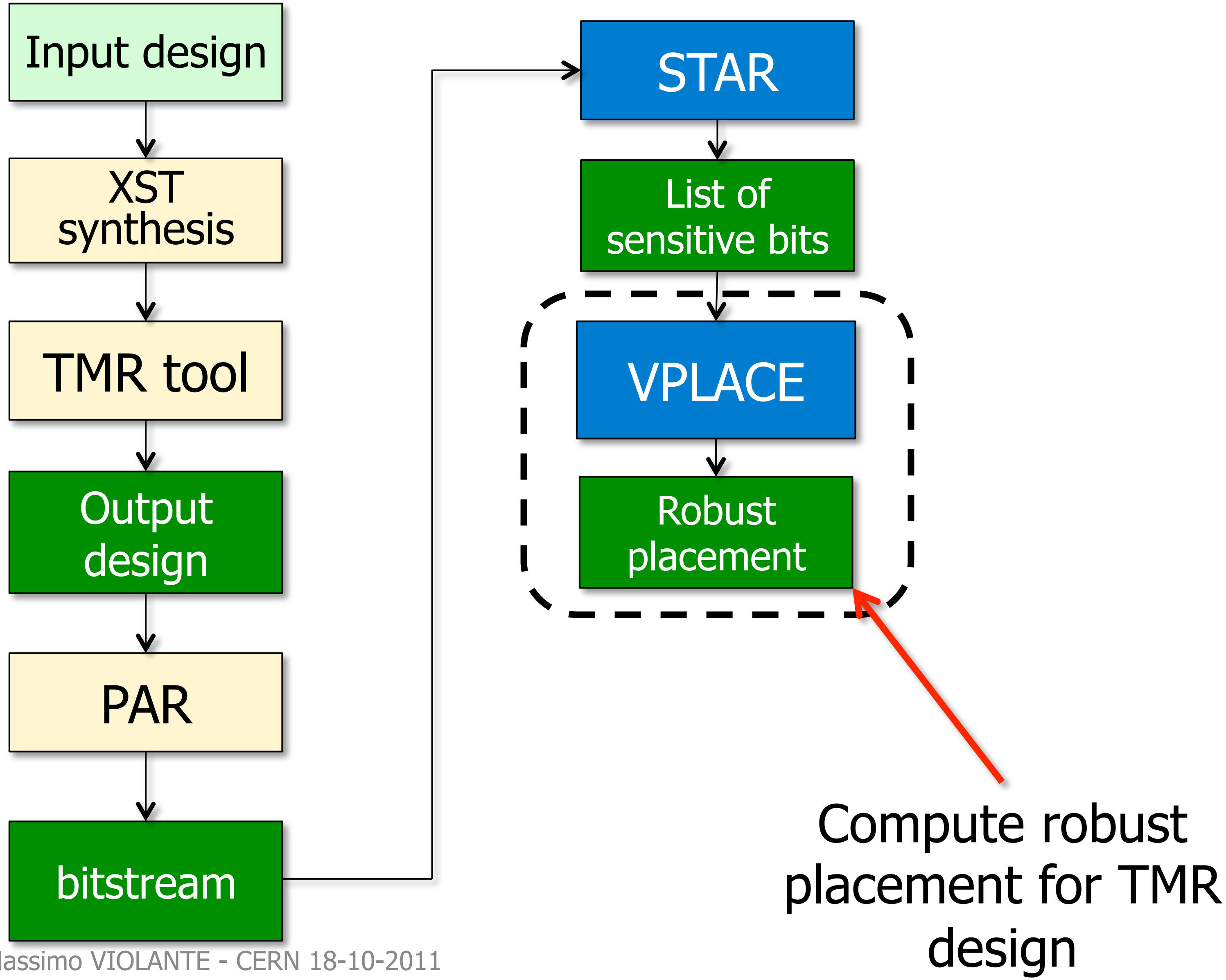


The design flow

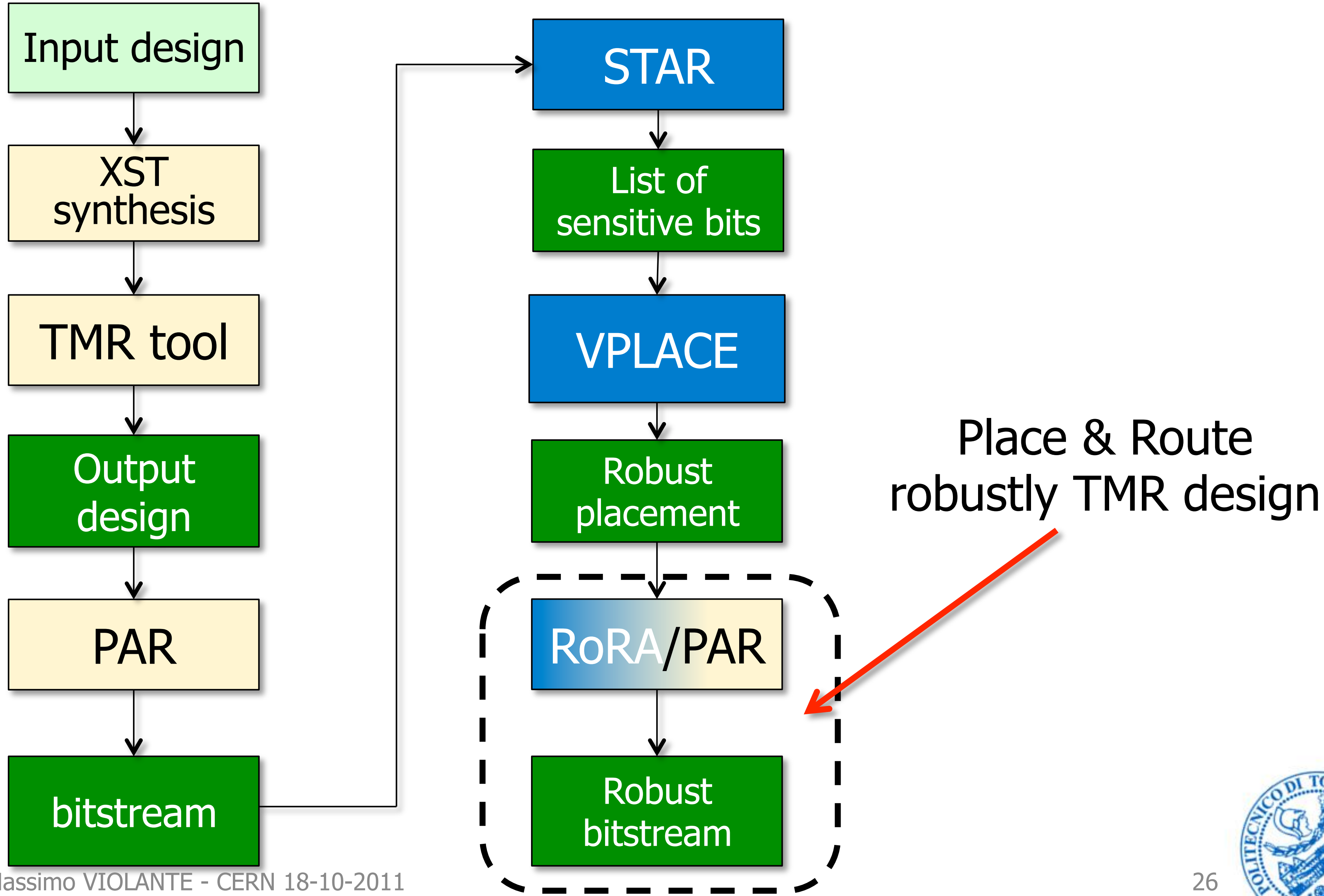


Analyze the impact of SEE in FPGA configuration memory without running simulations

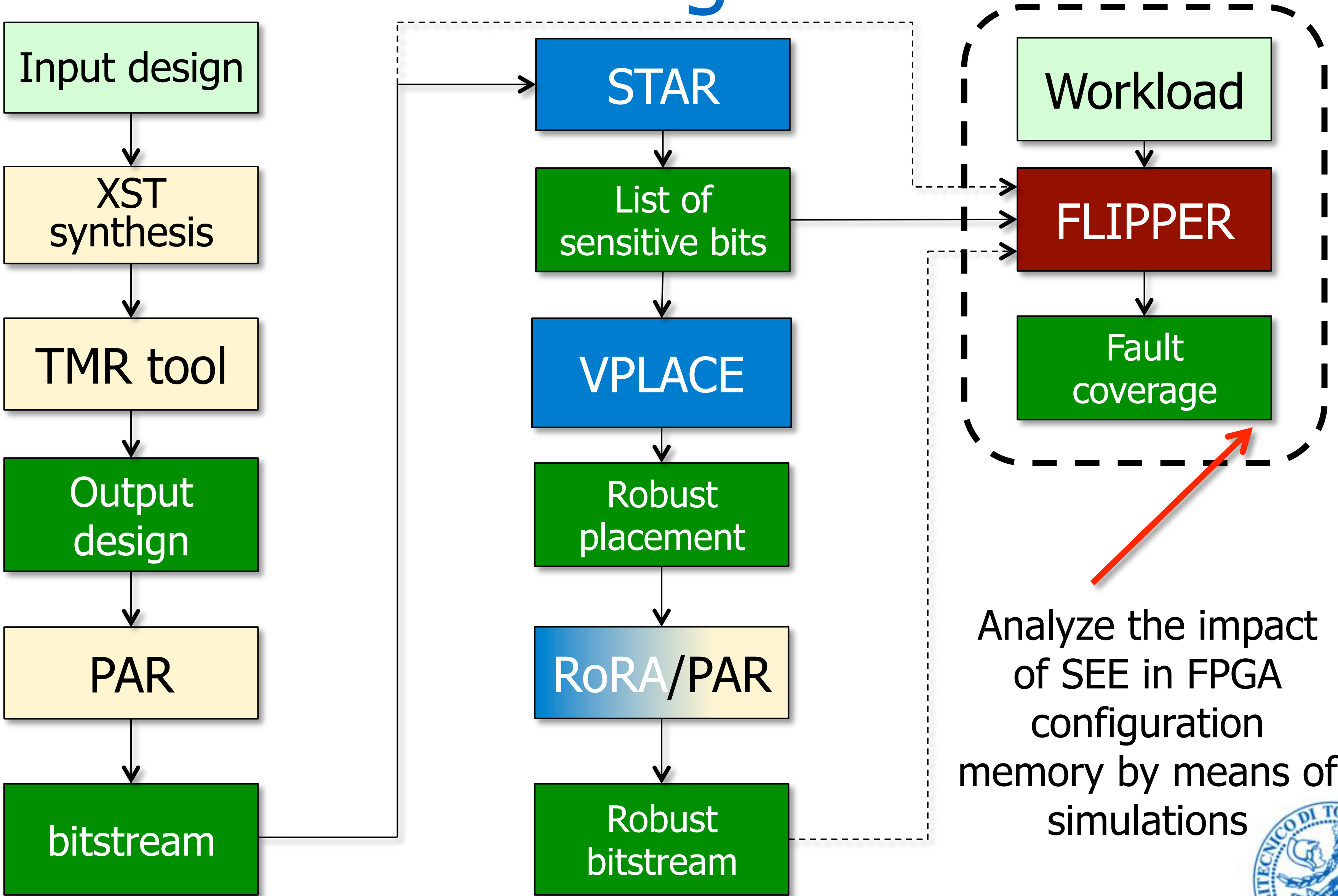
The design flow



The design flow



The design flow



STAR

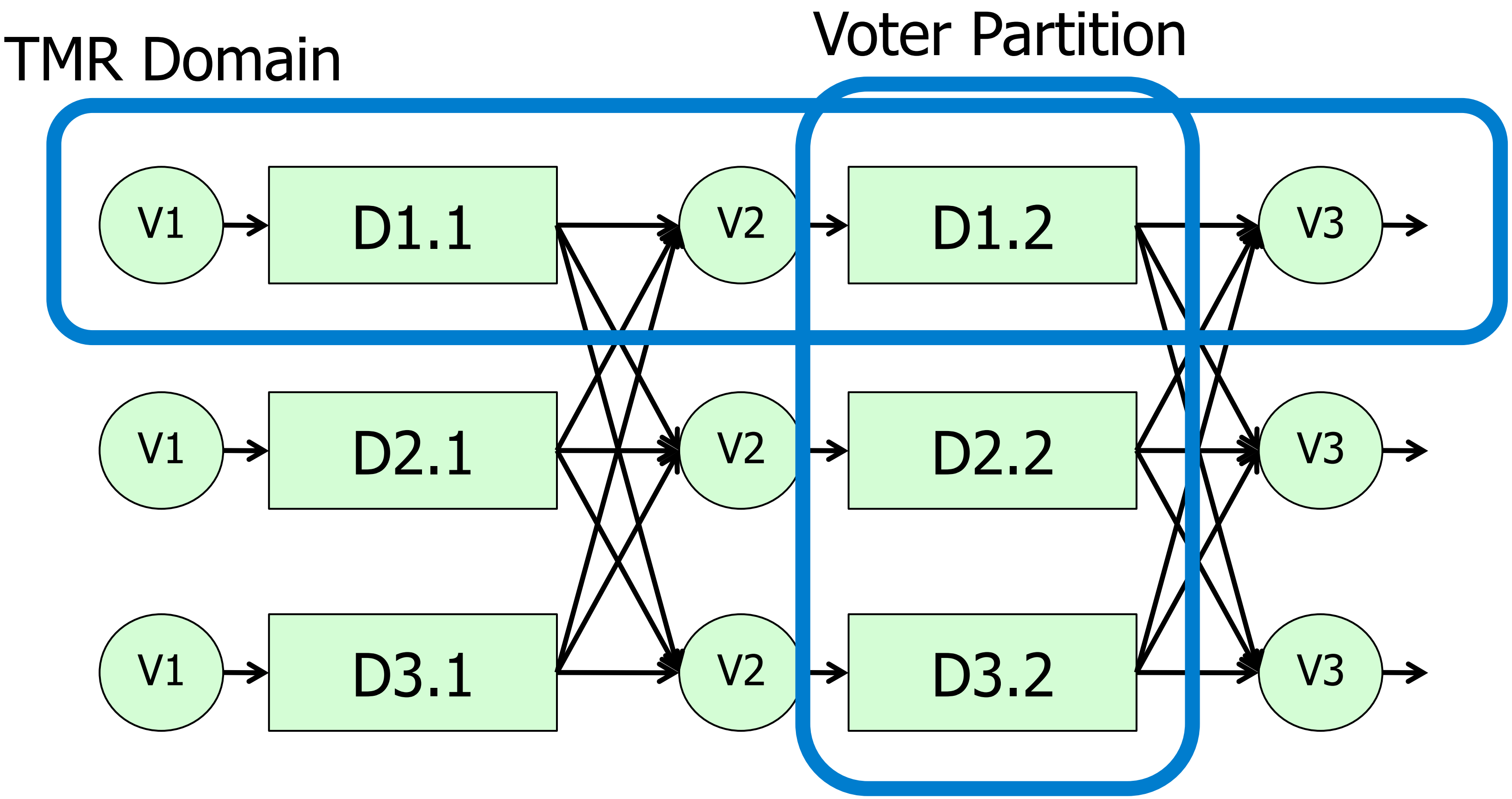
1. Read the place & routed design and build the netlist/bitstream association
2. For each bit of the bitstream:
   A. Flip the bit and update accordingly the netlist
   B. Is the original netlist corrupted (does the error arrive to outputs)?
      I. Yes → the bit is sensitive
      II. No → the bit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

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

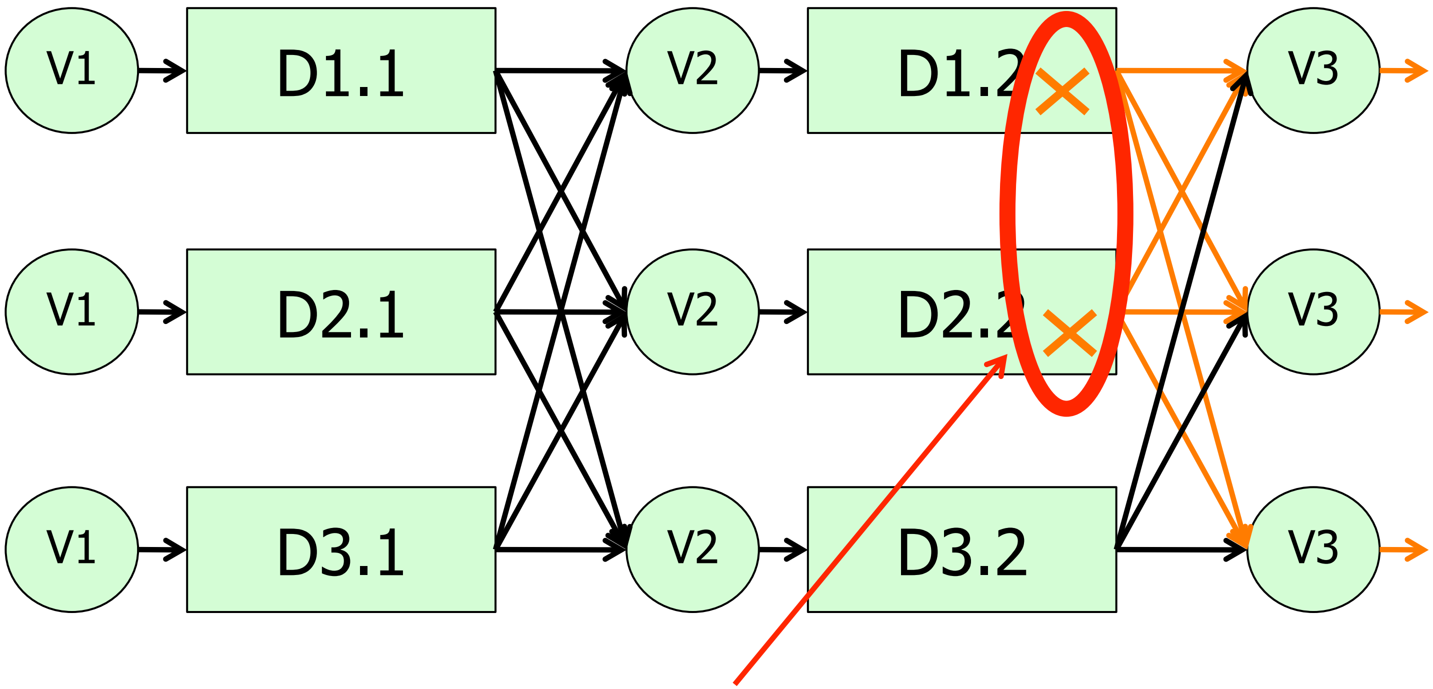
STAR operational modes

- 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

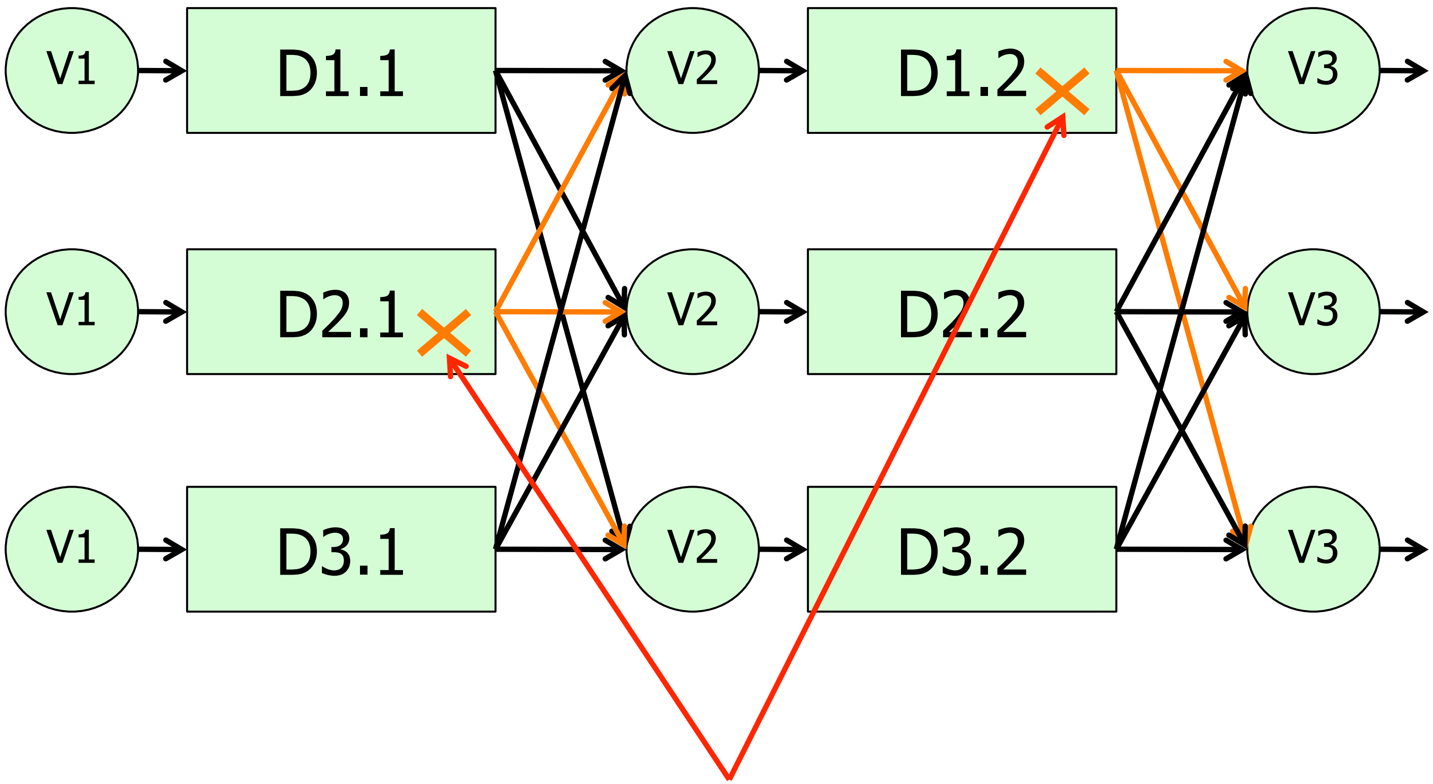


Domain crossing events



One Single Event Upset (SEU) in the configuration memory provokes two circuit modifications in two TMR domains in the same TMR partition → The fault propagates beyond the voter boundary

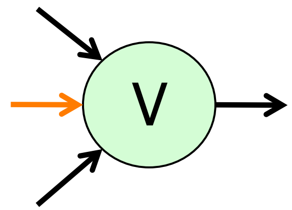
Warnings



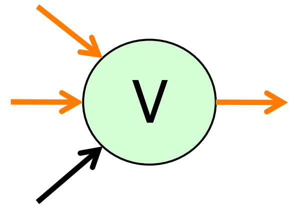
One SEE in the configuration memory provokes two circuit modifications in two voter partitions → The fault stops at the voter boundary

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



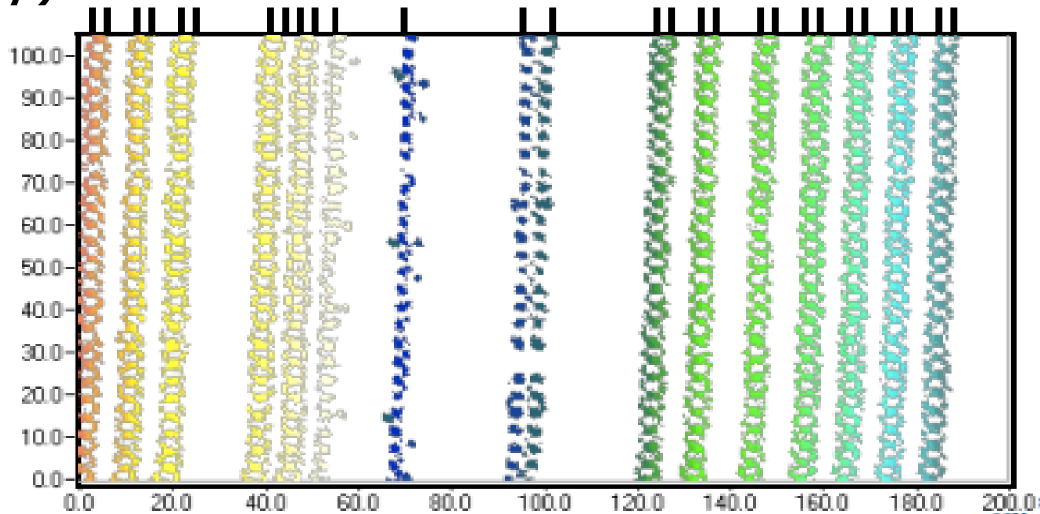
The report

- 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
```

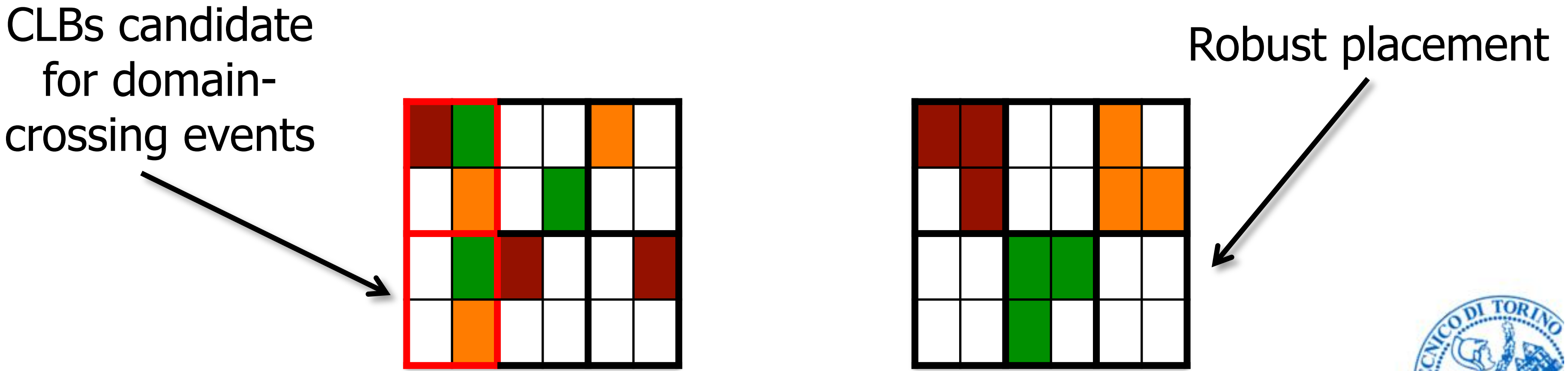
Supported fault models

- 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



VPLACE

- Domain-crossing events observed when different TMR domains are packed in on CLB
- VPLACE avoids them by:
  - Identifying the logic belonging to TMR domains
  - Defining placement constraints (UCF file) to force each domain in dedicated FPGA CLBs



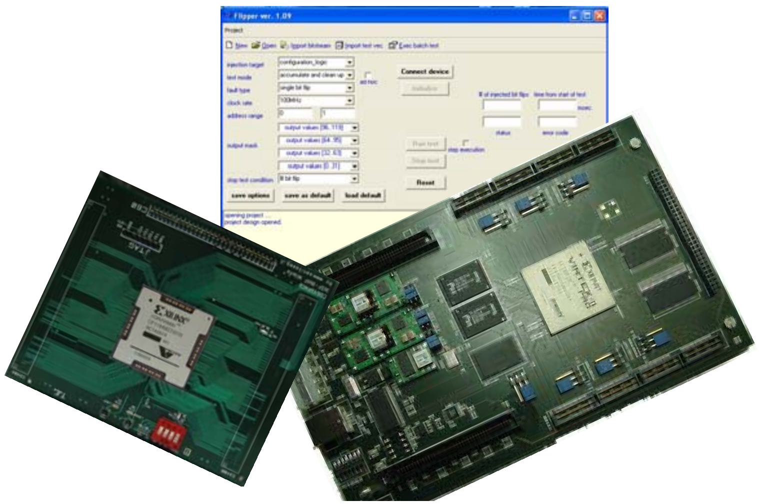
RoRA

- Domain-crossing events observed when net of different TMR domains are routed in adjacent CLB
- RoRA avoids them by:
  - Identifying critical nets
  - Re-routing critical nets
- To minimize run-time:
  - Xilinx PAR is used to provide an initial solution
  - RoRA reworks the initial solution by attacking critical nets only

FLIPPER

- STAR sensitive bits can become a failure or not depending on the workload
  - Pattern-dependent fault masking
- FLIPPER is an emulation platform specifically designed for fault injection of SEU in Xilinx FPGA configuration memory

ISTITUTO NAZIONALE DI ASTROFISICA
NATIONAL INSTITUTE FOR ASTROPHYSICS



Real design

- Problem: optimize SEE mitigation of a design
- Device: Virtex-4 xc4vlx160
- Design: XTMR circuit from Thales Alenia Space
- Sensitive bits according to STAR: 38,392
- Sensitive bits after VPLACE/RoRA: 17,385
  - Improved the robustness w.r.t. SEE by 2x

Outline

- Introduction
- Xilinx Virtex II & 4
- Actel ProASIC
- Conclusions

Why ProASIC?

- Package qualified successfully for space use
- Extensive database concerning radiation effects/ device reliability
- Availability of design tools

Why ProASIC?

- ProASIC appealing when…
  - Need for low-power design
  - Need for simple design
  - Need for limited in-flight reconfiguration capabilities
- SEL, TID
  - SEL free ($LET_{TH(H.I.)}$=68 MeV-cm^2/mg)
  - TID > 30 krad(Si) ~1.0 rad(Si)/minute (no refresh)
  - TID > 90 krad(Si) ~1.0 rad(Si)/minute (refresh)

However

- SEEs are an issue and must be mitigated
- Design validation is crucial

SEE of concern

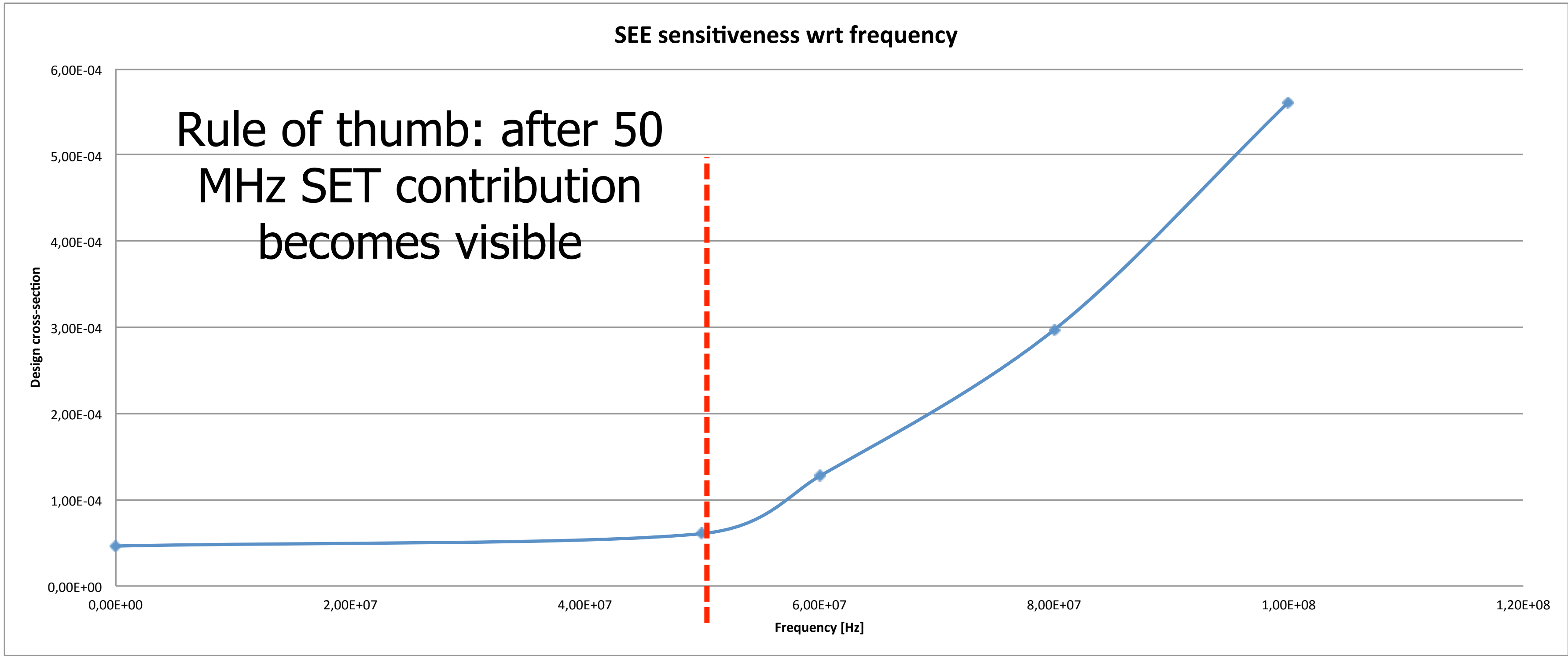
- No SEU/MCU in the device configuration memory
- SEU/MCU in the user memory
- SET

How to approach the design

- Simpler architecture than for SRAM-based FPGAs
  - Although, if refresh is needed it becomes similar
- SEE-aware design goes in the payload FPGA
  - TMR/EDAC for user memory
  - SET (possibly)
- Some vendors provides support for TMR
  - Symplify from Synopsys, Libero from Actel
- They allow
  - TMR insertion

How to approach the design

- Current approaches are suitable against SEU, however SETs may happen

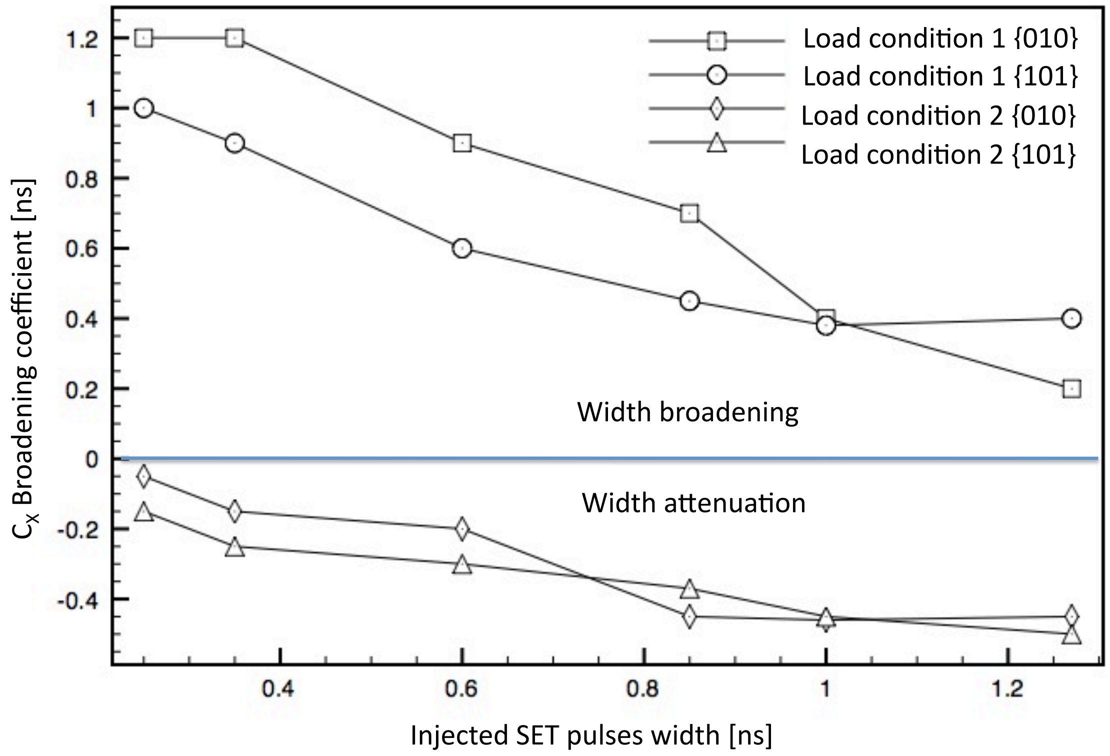


How to mitigate SET?

- Available options:
  - Global TMR
  - Insertion of SET filters (guard gates)
- No widely-accepted solution
- SET-aware design implementation is a possible solution
  - Clever placement of inverting gates allows for SET pulse attenuation

Our approach

- Broadening/attenuation for inverting/not-inverting gates implemented by Actel VersaTile vs fan-out

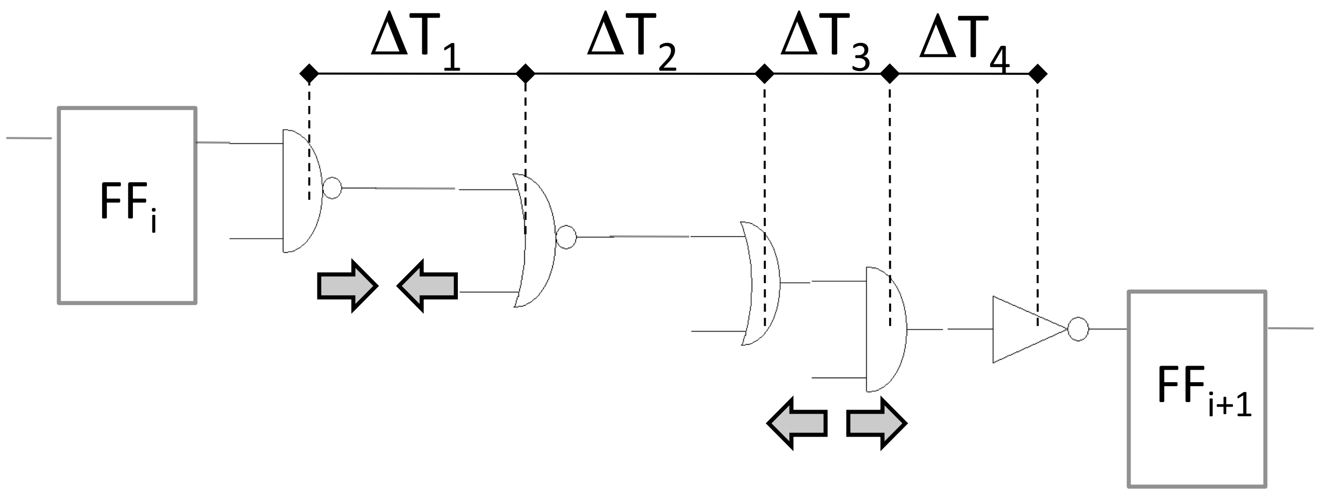


150 routing segments as fan-out

5 routing segments as fan-out

Our approach

- Empirical approach to attenuate SETs
  - Place inverting gates as close as possible
  - Place non-inverting gate as far as possible



Preliminary results

- SET pulse of 1.8V with width of 250 ps, 400 ps, and 600 ps

Circuit ID	SETs [#]	WA Original [#]	WA Hardened [#]
B10	15,000	1,435	429
B13	15,000	1,521	458
B07	15,000	1,232	371
B05	15,000	1,680	498
B12	15,000	1,702	521
B14	15,000	1,858	562

Outline

- Introduction
- Xilinx Virtex II & 4
- Actel ProASIC
- Conclusions

Conclusions

- Designing critical applications using reconfigurable FPGA is possible, but it requires
  - Understanding of SEE effects
  - Suitable mitigation techniques
  - Suitable validation strategy
- In addition to that:
  - Memory scrubbing scheme is needed (SRAM-based)
  - SEFI mitigation scheme is needed (SRAM-based)

Cookbook

- Different "recipes" are possible depending on your "taste"
- High dose → SRAM-based FPGAs
  - Low SEE rate → scrubbing only ☺
  - High SEE rate → scrubbing + TMR ☺
  - Very high SEE rate → scrubbing + TMR + SEFI ☺
- Low dose → Flash-based FPGAs
  - Low frequency → TMR on FFs ☺
  - High frequency → TMR on FFs + SET filtering ☹
- The proper cooking instruments and the chef skills are needed!

Acknowledgment

esa space for europe

EADS

ATMEL®

BOEING

Actel®

XILINX®