



# Corso di formazione INFN: Introduzione alle Tecniche di Trigger e Data Acquisition in Esperimenti di Fisica



Napoli, Oct. 9-12, 2023



## Introduction to Radiation Effects in FPGAs and Mitigation Techniques

Raffaele Giordano

University of Naples “Federico II” and INFN Sezione di Napoli, I-80126, Italy

Email: [rgiordano@na.infn.it](mailto:rgiordano@na.infn.it)

# Outline

- Basics of radiation effects in microelectronics
- Introduction to radiation hardening by design in digital circuits
- Configuration scrubbing in reconfigurable logic
  - Focus on SRAM-based FPGAs



# Corso di formazione INFN: Introduzione alle Tecniche di Trigger e Data Acquisition in Esperimenti di Fisica

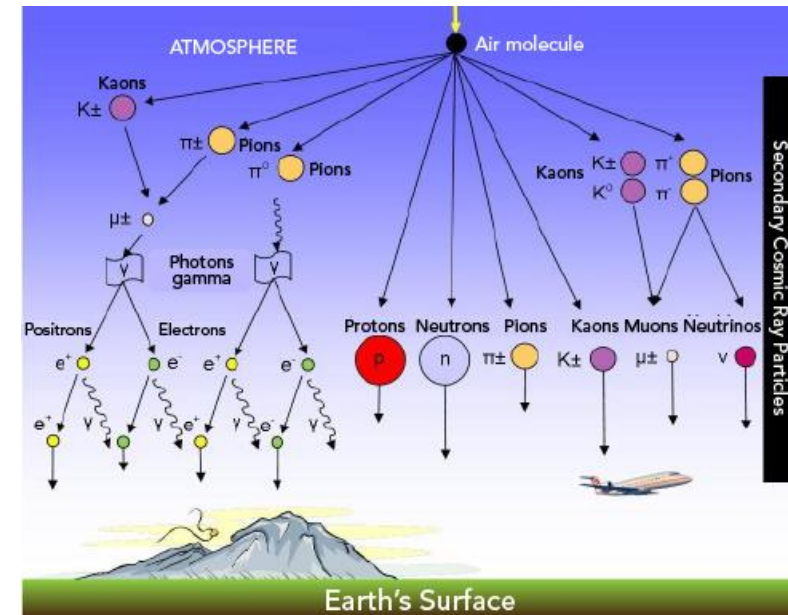
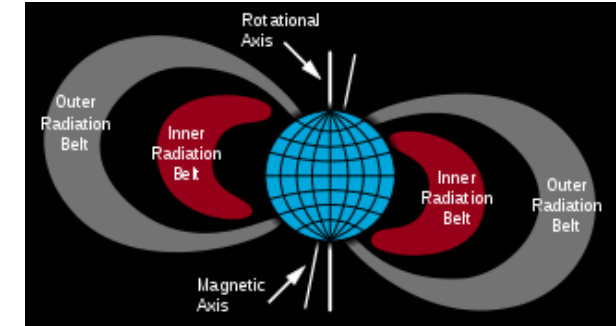
Napoli, Oct. 9-12, 2023



## **BASICS OF RADIATION EFFECTS IN MICROELECTRONICS**

# Radiation

- Radiation is the transfer of energy by means of particles (including photons)
  - For photons usually people consider  $E > \text{UV rays (tens of eV)}$
- Radiation effects in electronics can happen in
  - Artificially generated radiation environments [e.g. at particle accelerators (protons, neutrons, ions, electrons, gammas...)]
  - Space (mainly protons, alphas, heavier ions, electrons)
  - Earth Atmosphere (atmospheric neutrons)
- If high-reliability is required, even at sea-level there is a concern for atmospheric neutrons
  - Medical Applications, Automotive, Data Centers...



# Effects of Radiation on Electronics

- Total Dose (TD) effects, i.e. build-up over time
  - Consequences: degradation of device characteristics,  $\mu$ -electronics but also Si-based detectors, optical components
  - Can be related to **ionizing** & **non-ionizing** energy loss
- Single Event Effects (SEEs)
  - Related to the passage of a single particle
  - Can be destructive and non-destructive
  - Consequences: data corruption, system shutdown, recoverable or non-recoverable malfunctioning, device destruction

# Effects of Radiation on Electronics (2)

Total Dose	Single Event Effects
TD is cumulative and may become visible after the device has been exposed to radiation for some time.	SEEs occur stochastically, no memory from the previous effects.
TD is usually related to long term response of devices.	SEEs are concerned with short time response.
TD uniformly affects the whole device, because it results from the effect of several particles randomly hitting the device.	Only a tiny part of the device is affected by each SEE, corresponding to the volume of the particle strike.
TD is characterized by the maximum drift of the main device parameters.	For SEEs the most important figure is the rate of occurrence.



**DISCLAIMER**

This talk only partly discusses TD effects, not critical for SRAM-based FPGAs, due to time-constraints !

# Effects of Radiation on Electronics (3)

- Energy deposit by ionization may cause
  - charge build-up in insulating layers (cumulative effect)
  - charge injection into sensitive nodes (single ionizing event effects)
- Energy deposit by atomic displacement from lattice sites may cause
  - accumulation of damage to lattice/bulk (cumulative effects)
    - crystal structure damage
    - introduction of trap states and mid-band states



# Radiation Dose

$$D = \frac{E}{m}$$

- Dose is the energy deposited per unit mass
  - Important for cumulative effects
  - Measurement units
    - Gy = 1 J/kg and rad. 1 Gy=100 rad, 1 rad = 1 cGy
  - Sometimes the material which the dose refers to is specified [e.g. Gy(Si) or Gy(H<sub>2</sub>O)]
- Two main types of dose:
  - Total Ionizing Energy Loss Dose (TID)
  - Non-Ionizing Energy Loss (NIEL) Dose also called Displacement Damage Dose (DDD)



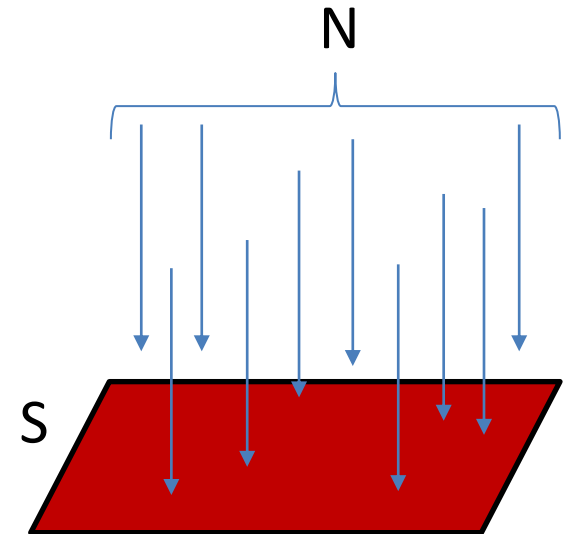
# Fluence and Flux

- Fluence (uppercase  $\Phi$ ) is number of impinging particles per unit area

$$\Phi = \frac{N}{S} \quad [ \text{particles/cm}^2 \text{ or cm}^{-2} ]$$

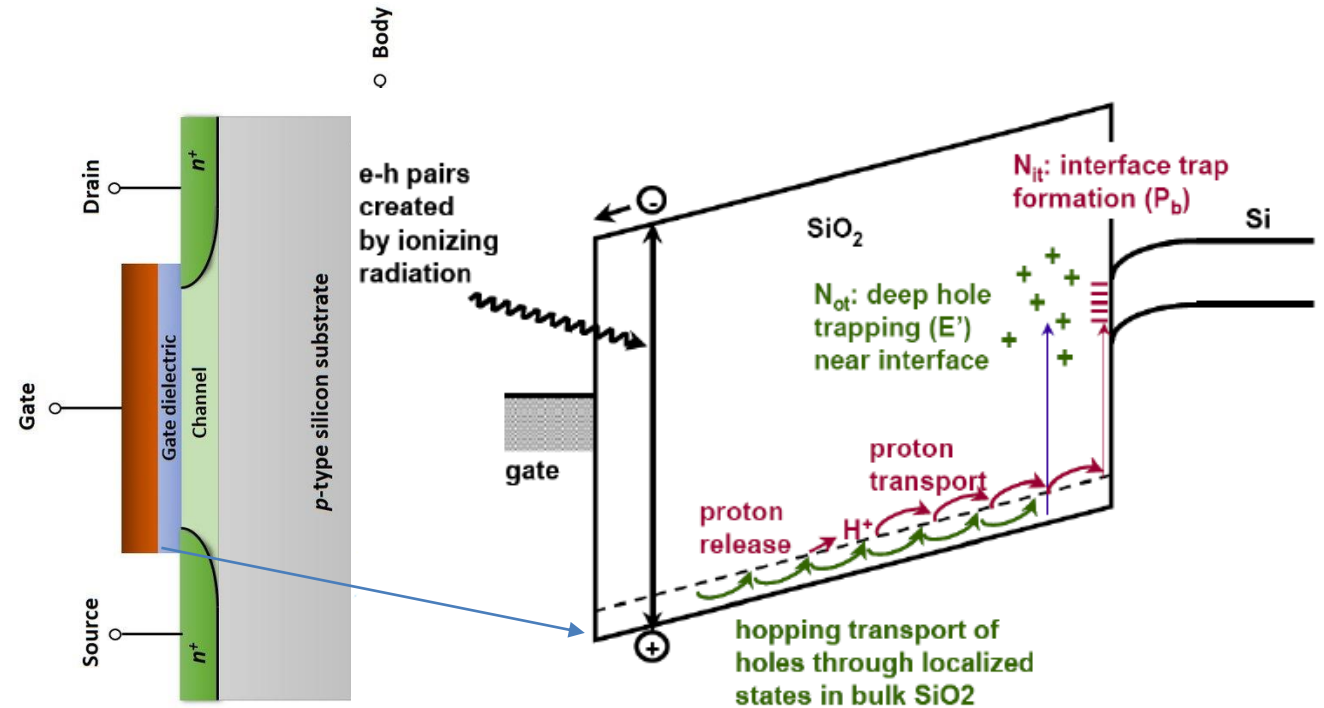
- Flux (lowercase  $\phi$ ) is the fluence per unit time, i.e. the fluence rate

$$\phi = \frac{\Phi}{\Delta t} = \frac{F}{\Delta t} \quad [ \text{particles / (cm}^2 \cdot \text{s) or cm}^{-2} \cdot \text{s}^{-1} ]$$



# TID Effects: $V_t$ shift in MOSFETs

- Charge can get trapped in
  - the oxide (oxide trapped charge) and at the Si/SiO<sub>2</sub> interface (interface trapped charge)
- Charge in the oxide is always positive
- Sign of charge at the interface depend on MOSFET type P- or N-MOS
  - Positive charge in PMOS, negative in NMOS
- Trapped charge work as a series voltage source and impact threshold voltage of the MOS



After, J. R. Schwank *et al.*, doi: 10.1109/TNS.2008.2001040

$$\Delta V_{OT} = -\frac{Q_{OT}}{C_{ox}}$$

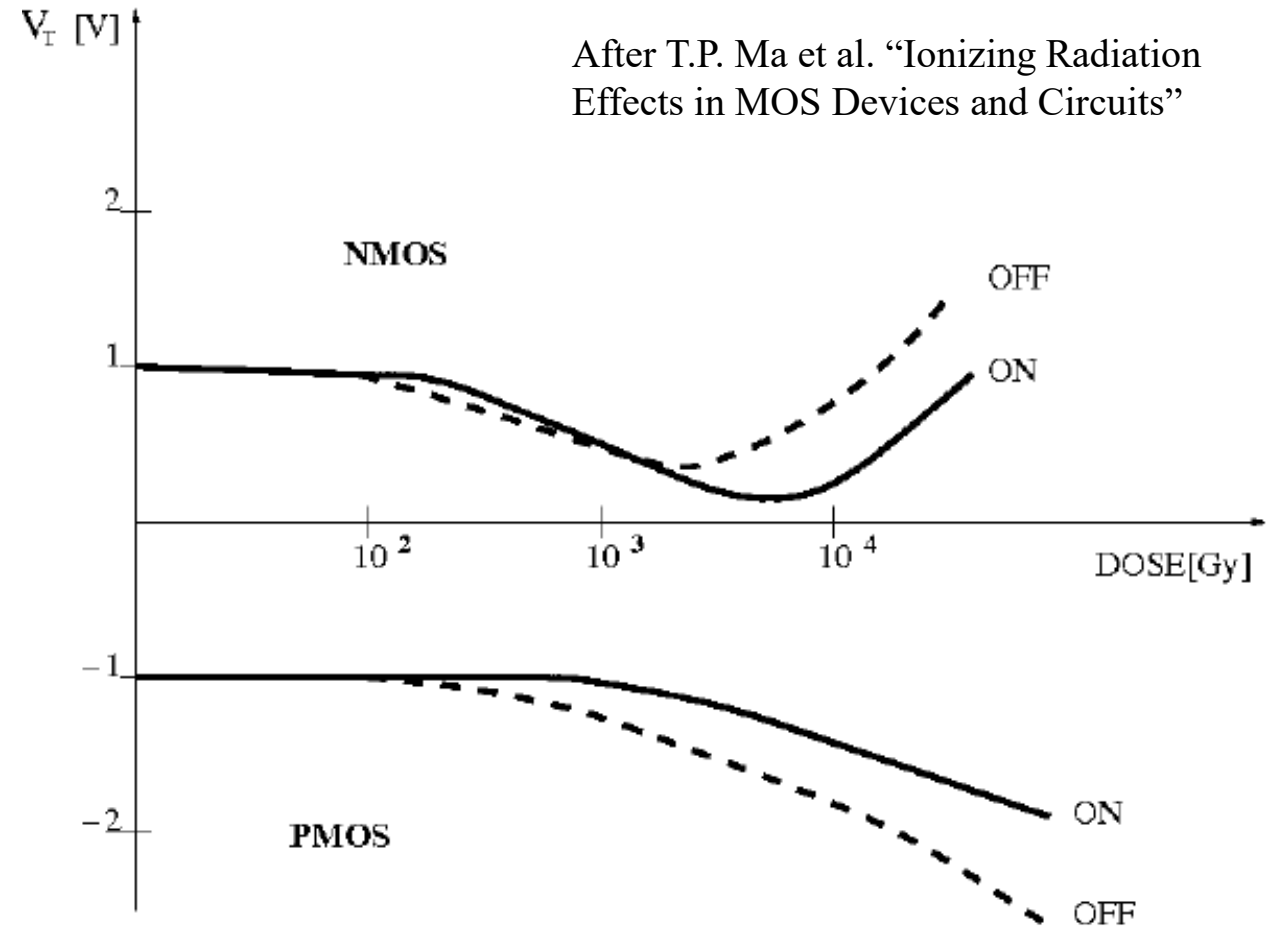
$$\Delta V_{IT} = -\frac{Q_{IT}}{C_{ox}}$$

- $C_{ox}$  = oxide capacitance per unit area
- $Q_{OT}$  = charge trapped in the oxide per unit area
- $Q_{IT}$  = charge trapped at the SiO<sub>2</sub>/Si interface per unit area

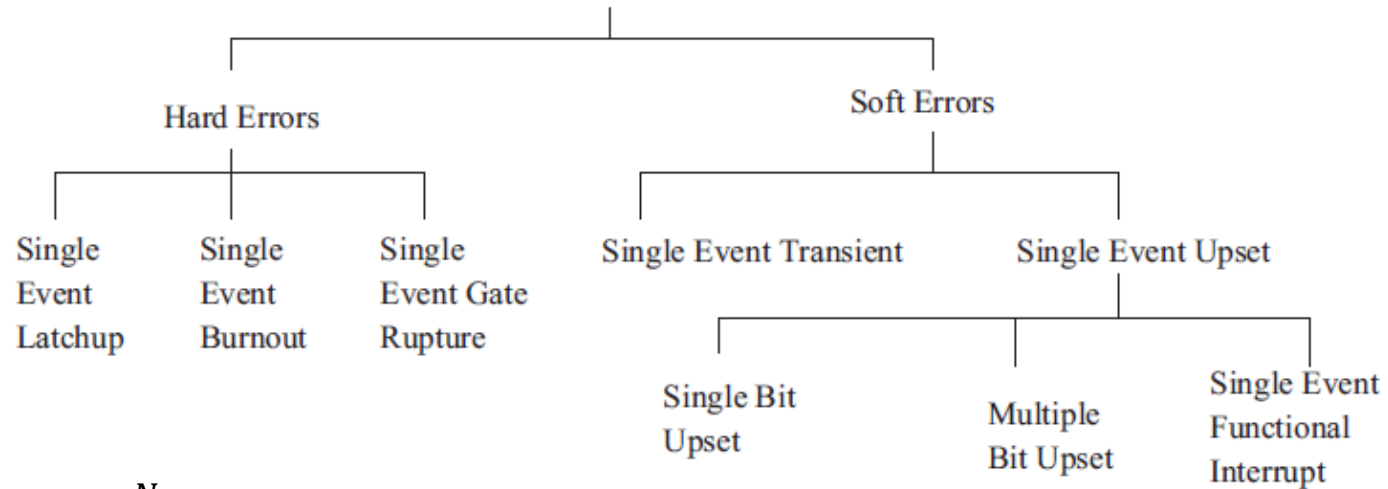
# TID Effects: $V_t$ shift in MOSFETs

$$\begin{aligned}\Delta V_t &= \Delta V_{IT} + \Delta V_{OT} = \\ &= -\frac{1}{C_{ox}} (Q_{IT} + Q_{OT}) = -\frac{t_{ox}}{\epsilon_{ox}} (Q_{IT} + Q_{OT})\end{aligned}$$

- $\epsilon_{ox}$  = dielectric constant
- $t_{ox}$  = thickness of oxide
- In the simplest model  $Q_{OT} \propto t_{ox}^2$  so  $\Delta V_{OT} \propto t_{ox}^2$
- At low doses  $Q_{OT}$  dominates, at higher doses  $Q_{IT}$  dominates
- NMOS  $V_t$  shifts not monotonically, PMOS  $V_t$  shifts always down
- Holes trapped in  $\text{SiO}_2$  are not stable, they disappear (anneal) over times from milliseconds to years



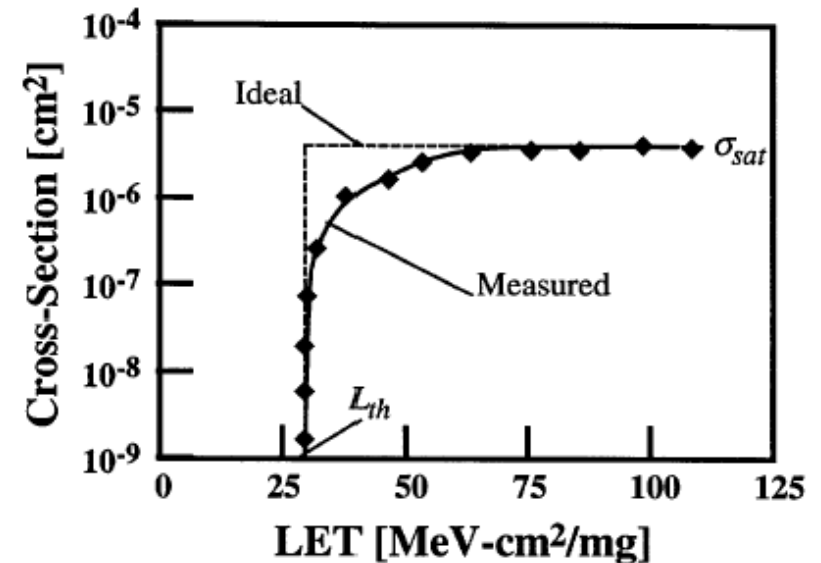
# Single Event Effects



- Cross section ( $\sigma$ ) for SEEs is  $\sigma = \frac{N_{SEE}}{\Phi}$
- Can be plotted against LET (e.g. for ions) or vs energy (e.g. for protons or neutrons)
- A fit with a Weibull function can be performed

$$\sigma(LET) = \sigma_{sat} \left[ 1 - e^{-\left(\frac{LET - LET_{th}}{W}\right)^S} \right]$$

- $W$  and  $S$  are fit parameters
- $LET_{th}$  threshold LET for the SEE
- $\sigma_{sat}$  saturation cross section



# Single Event Effects (2)

- Taking into account energies in space and terrestrial radiation environments
- SEE in CMOS devices are mainly induced
  - Directly by heavy ions ( $Z > 1$ )
  - Indirectly by protons and neutrons on Si or dopants ( $n + {}^{10}\text{B} \rightarrow {}^4\text{He} + {}^7\text{Li}$ ), Oxygen, other atoms in the device (secondaries produce ionization)
- Neutrons cannot induce SEEs directly since they are neutral
- Protons *generally* cannot induce SEEs directly
  - $\text{LET} < .1 \text{ MeV}\cdot\text{cm}^2 / \text{mg}$  (too low)
- For  $E > 50 \text{ MeV}$ , secondaries from p and n produce almost identical effects
- Electrons and muons *generally* cannot induce SEEs directly (LET too low) and neither indirectly (cannot induce nuclear reactions)

# Proton Secondaries in Silicon

- Secondaries contribution to overall LET depends very much on primary proton energy

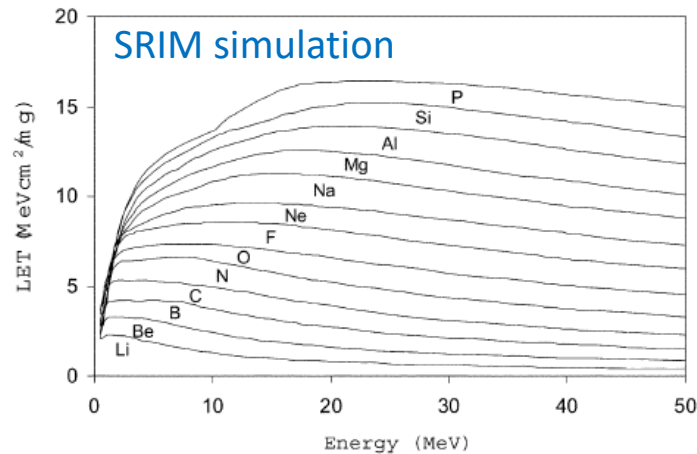


Fig. 2. LET versus energy of nuclei produced by protons in silicon.

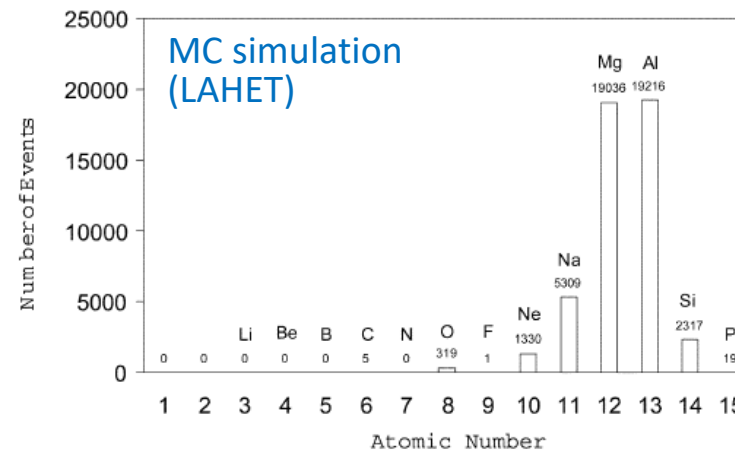


Fig. 8. Contribution to LET spectrum as a function of atomic number for 50 MeV protons.

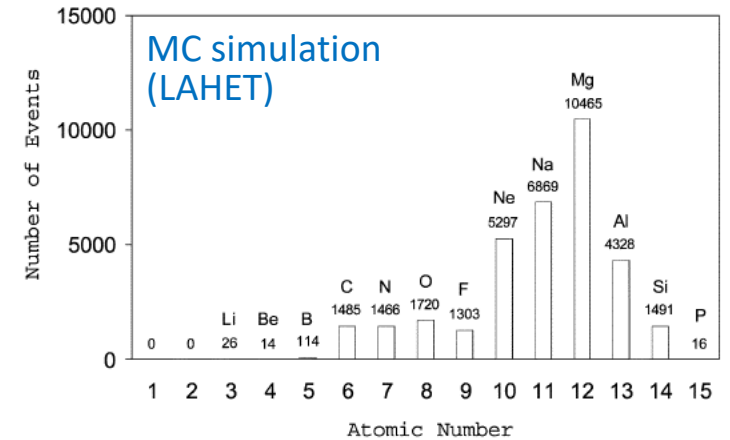


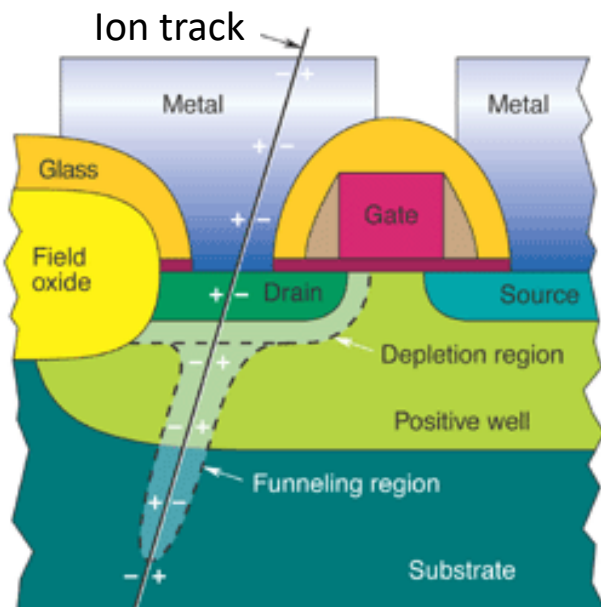
Fig. 10. Contribution to LET spectrum as a function of atomic number for 200 MeV protons.

LET < 1.5 MeV·cm<sup>2</sup> / mg not simulated, therefore Z=1,2 nuclei not shown

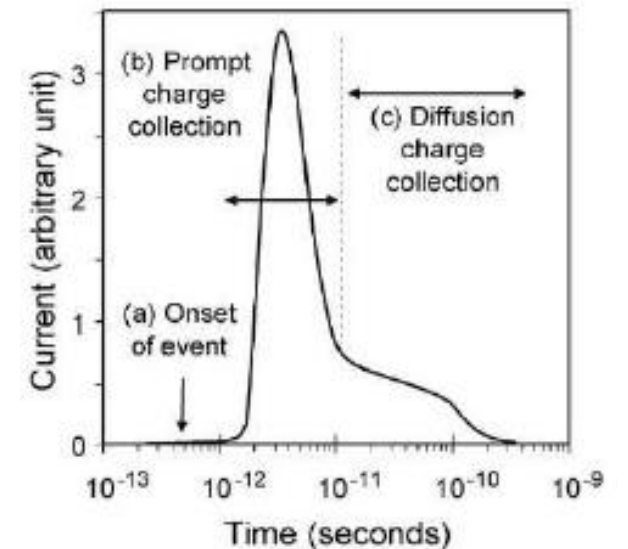
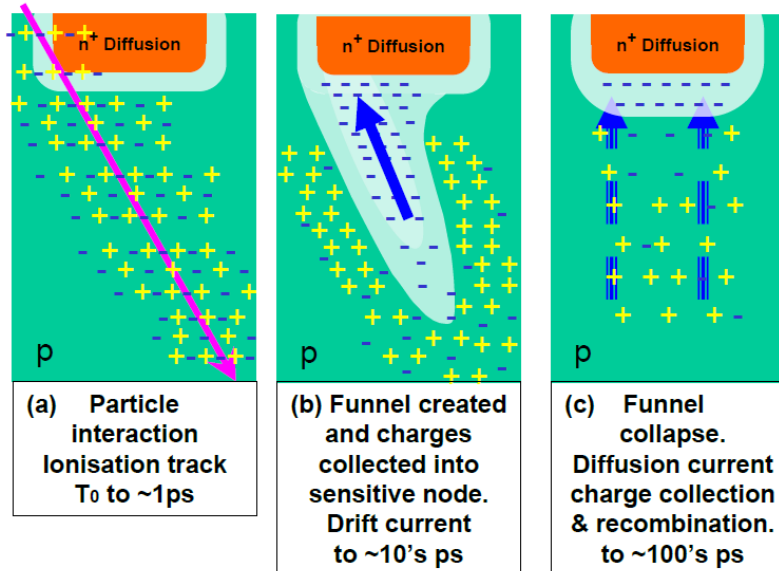
After, D. M. Hiemstra and E. W. Blackmore, doi: 10.1109/TNS.2003.821811

# Single Event Transient

- In MOS transistors in OFF state, drain and substrate are reverse-biased PN junctions
- Charges diffuse and drift in the semiconductor due to electric field
  - This motion results in voltage transients between drain and source
- NMOS more sensitive than PMOS (e mobility higher than h)



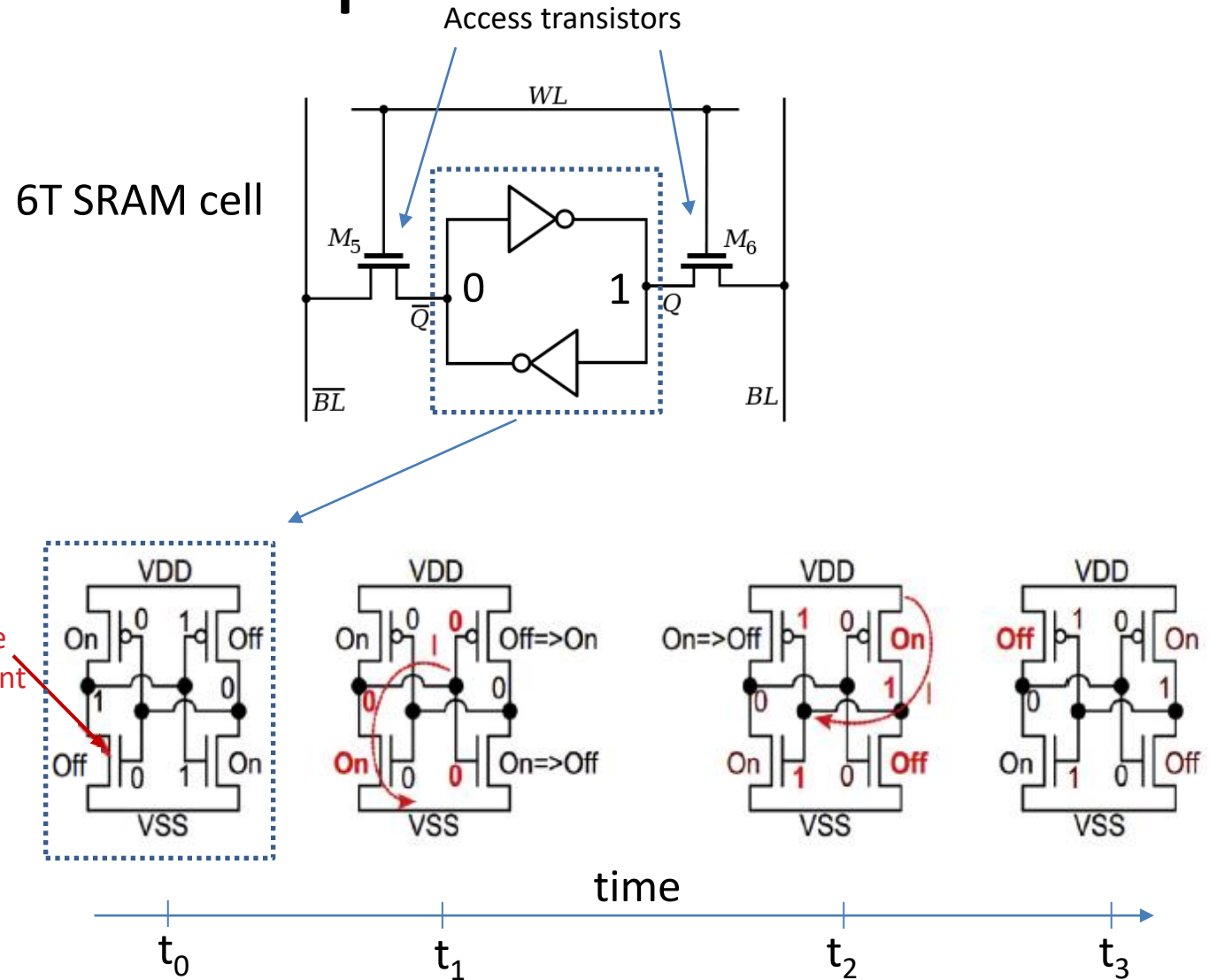
After A. Chugg, «SET Generation & Definition –Overview»





# Single Event Upset

- SETs may happen in memory elements (e.g. latches, flip-flop, SRAM cell...)
- In this case they might be captured by logic feedback of the element and become persistent errors
- They can be cleared by refreshing the correct memory content
- There is a minimum critical charge ( $Q_c$ ) to be collected for a SEU to happen

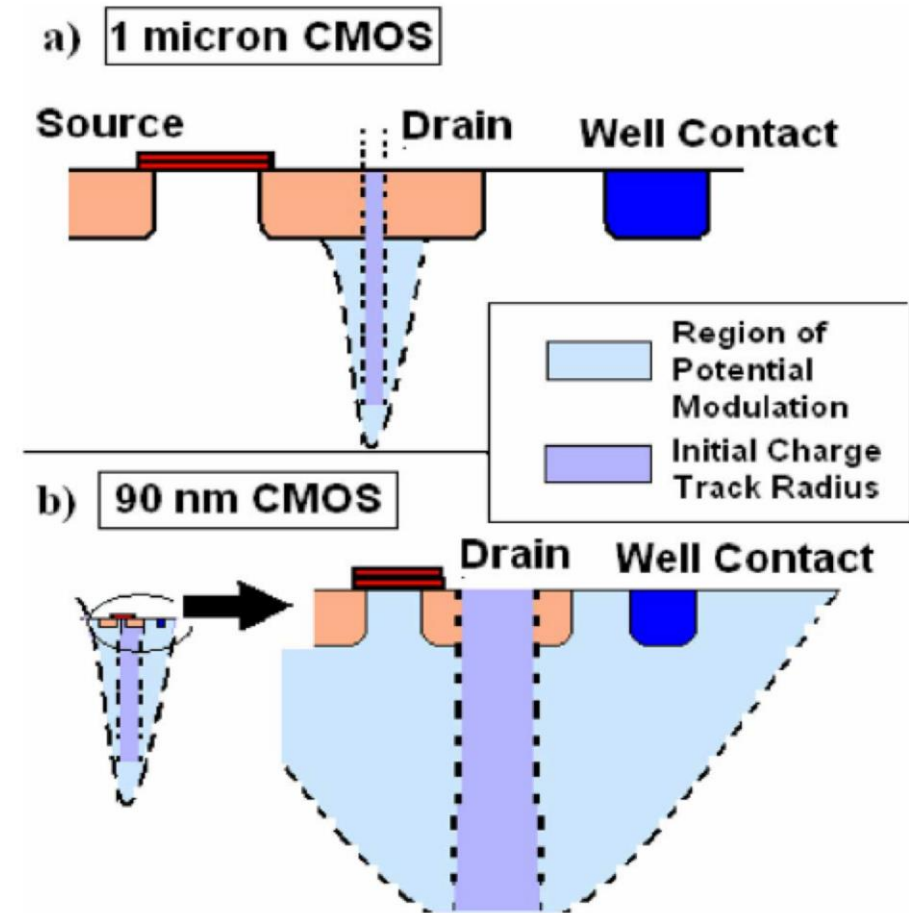


$$Q_c = (L_{th} t \rho q) / E_{pair}$$

$$E_{pair} = 3.6 \text{ eV in Si}$$

# Impact of Scaling on Radiation Effects

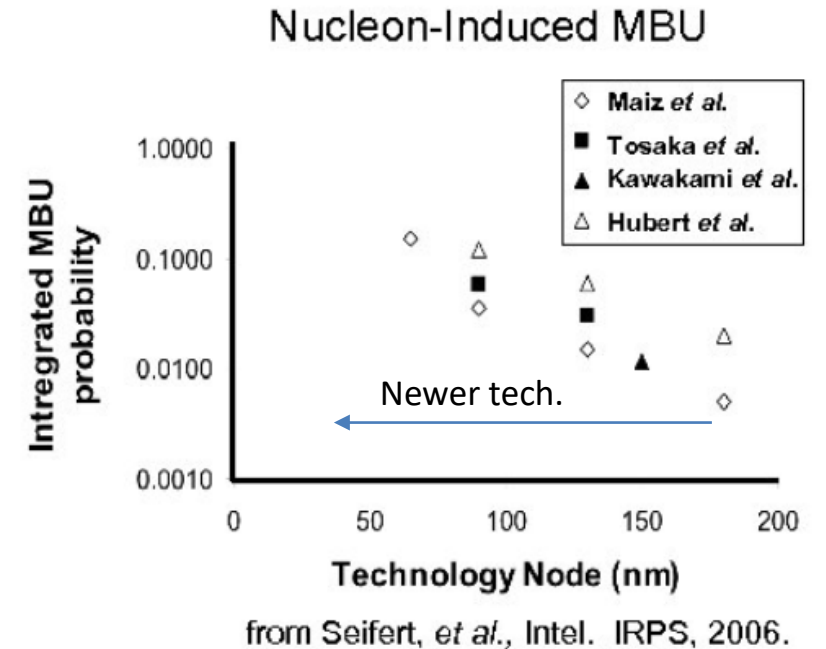
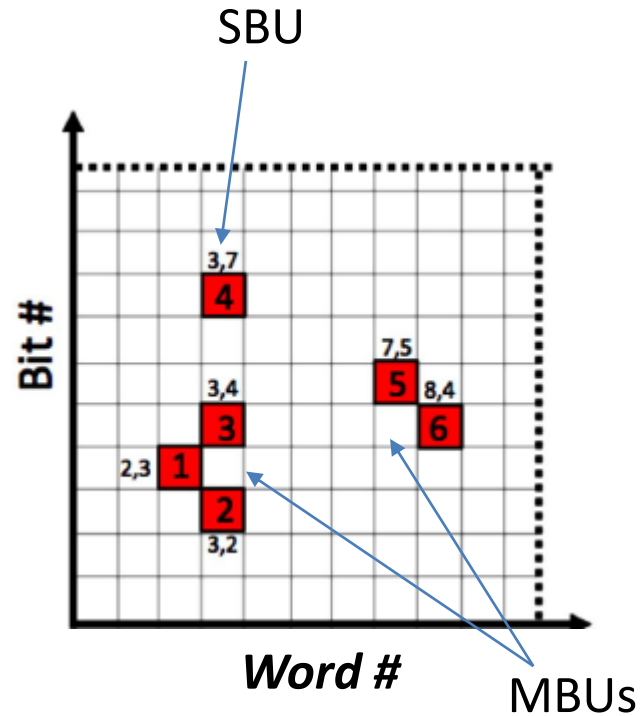
- Generally newer technologies scale to smaller feature sizes
- Scaling of MOSFETs make some of the LET-related effects more important
- A single ion might impact drain, channel, source or even more than one transistor
- Critical charge for SEUs becomes lower => higher cross section, more MBUs
- Oxides become thinner =>  $V_t$  shift milder



S. DasGupta et al. , "Effects of Well and Substrate Potential Modulation on Single Event Pulse Shape in Deep Submicron CMOS", IEEE Trans. Nucl. Sci, vol.54, n. 6 Dec. 2007.

# Multiple Bit Upsets

- In modern CMOS technologies (65nm and beyond) sizes of ion tracks are comparable to the feature sizes of transistors
- A track can trigger upsets in several devices and generate upsets in multiple memory elements
- This increases as the feature scales in planar technologies
  - e.g. in 28nm CMOS SRAMs up to 25% SEUs are MBUs
- Different behaviour can be for newer non-planar devices, such as FinFETs

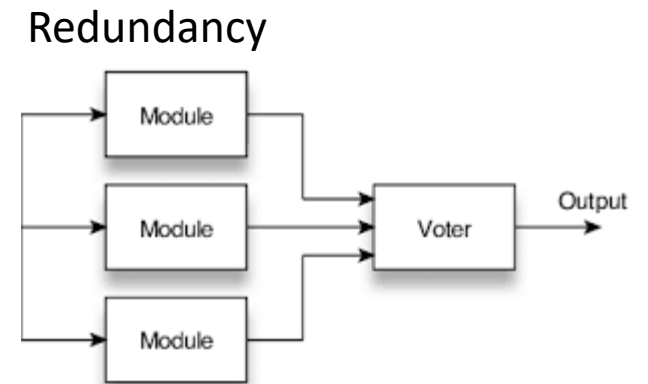
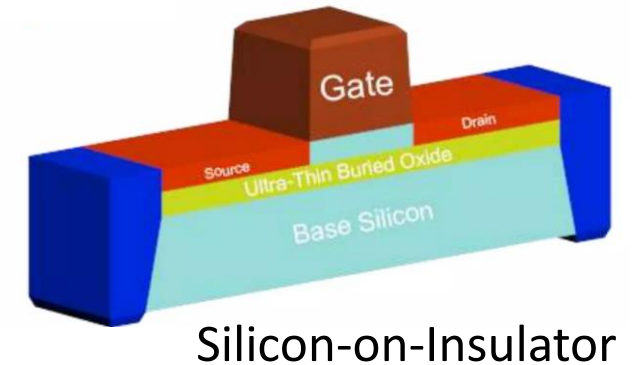
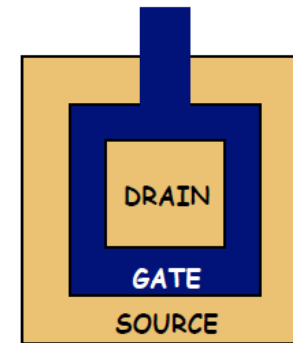


# Radiation Hardening Techniques

- In order to make a device sufficiently radiation hard for an application, it is possible to pursue one or more of the following approaches
  - Hardening by process
    - modifying one or more steps of the fabrication process (e.g. SOI, removal of B, limit O in Si)
  - Hardening by layout
    - modifying the device geometry (e.g. enclosed layout transistors)
  - Hardening by design
    - modifying the overall circuit or system (usually by means of some form of redundancy)

Examples:

Transistor with Enclosed layout





# Corso di formazione INFN: Introduzione alle Tecniche di Trigger e Data Acquisition in Esperimenti di Fisica

Napoli, Oct. 9-12, 2023

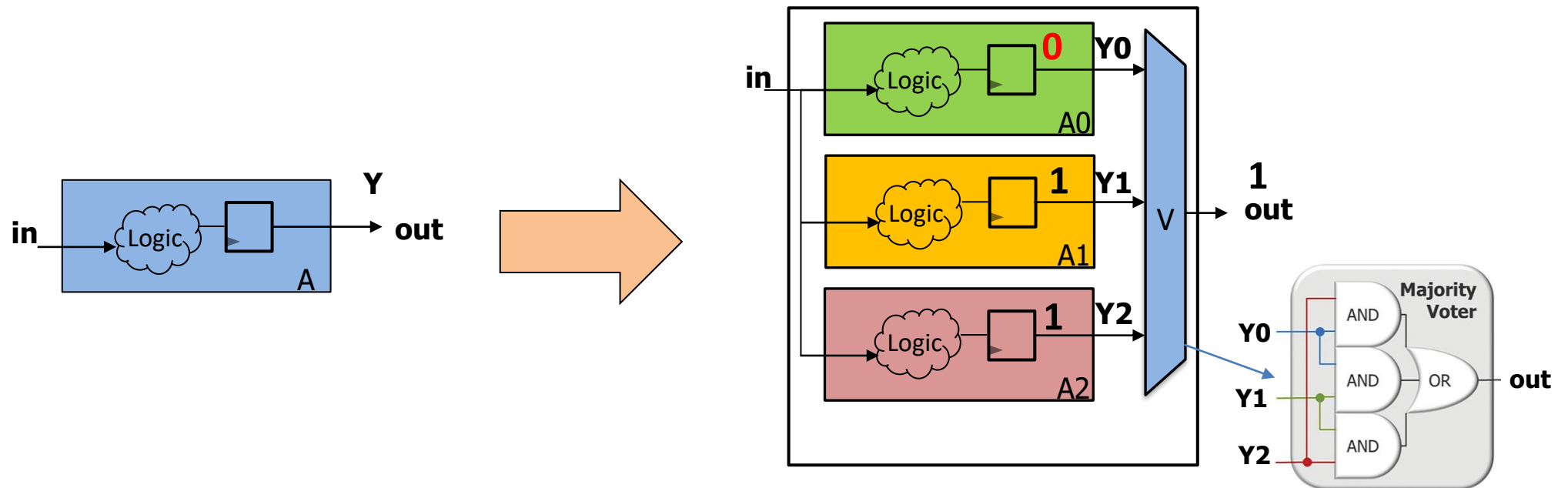


## **RADIATION HARDENING BY DESIGN IN DIGITAL CIRCUITS**

# RHBD Techniques in Digital ICs

- Triple Modular Redundancy
  - Several flavours
- “Safe” Finite State Machines
- Error Correcting Codes
  - Storage or transmission

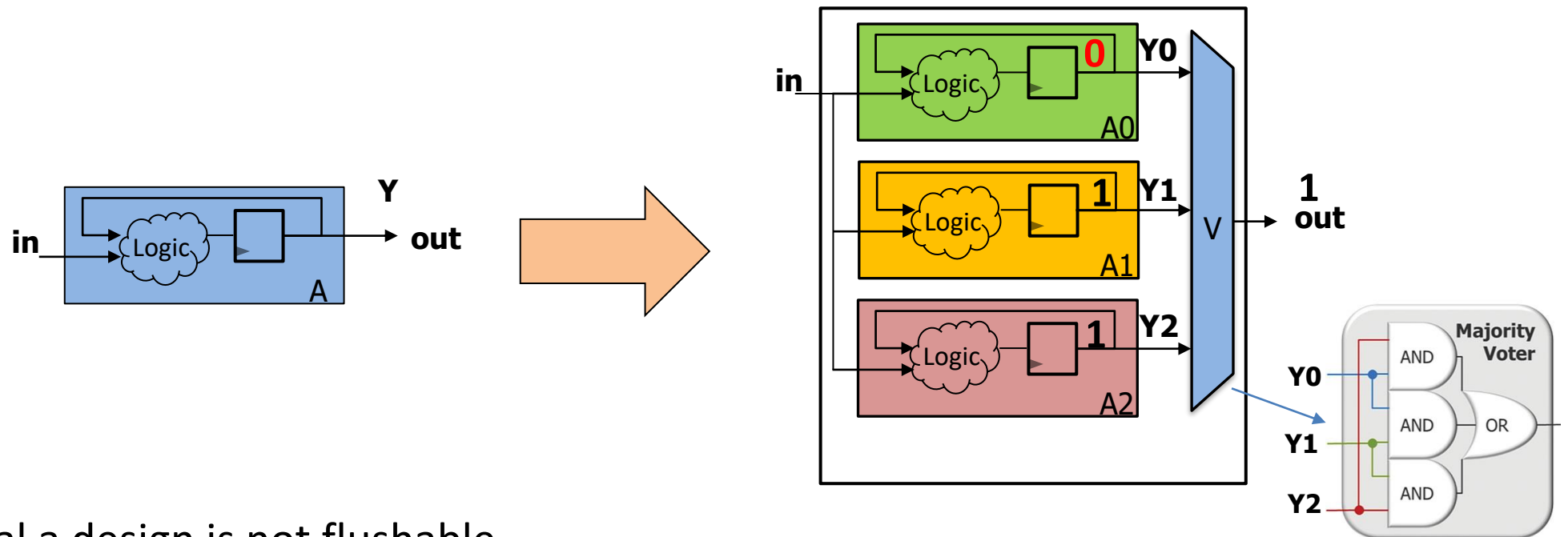
# Triple Modular Redundancy (TMR)



- **Key concept: minimize the # of parts of the system which can cause an overall failure [e.g. single point of failures (SPOFs)]**
- Generate three replicas of the same circuit (module) and majority vote their outputs
- Can mask an error in a single replica (A0, A1 or A2)
- >3x area/power overhead
- This is a flushable design, faults (e.g. SETs and SEUs) will go through the logic and the voter will mask them

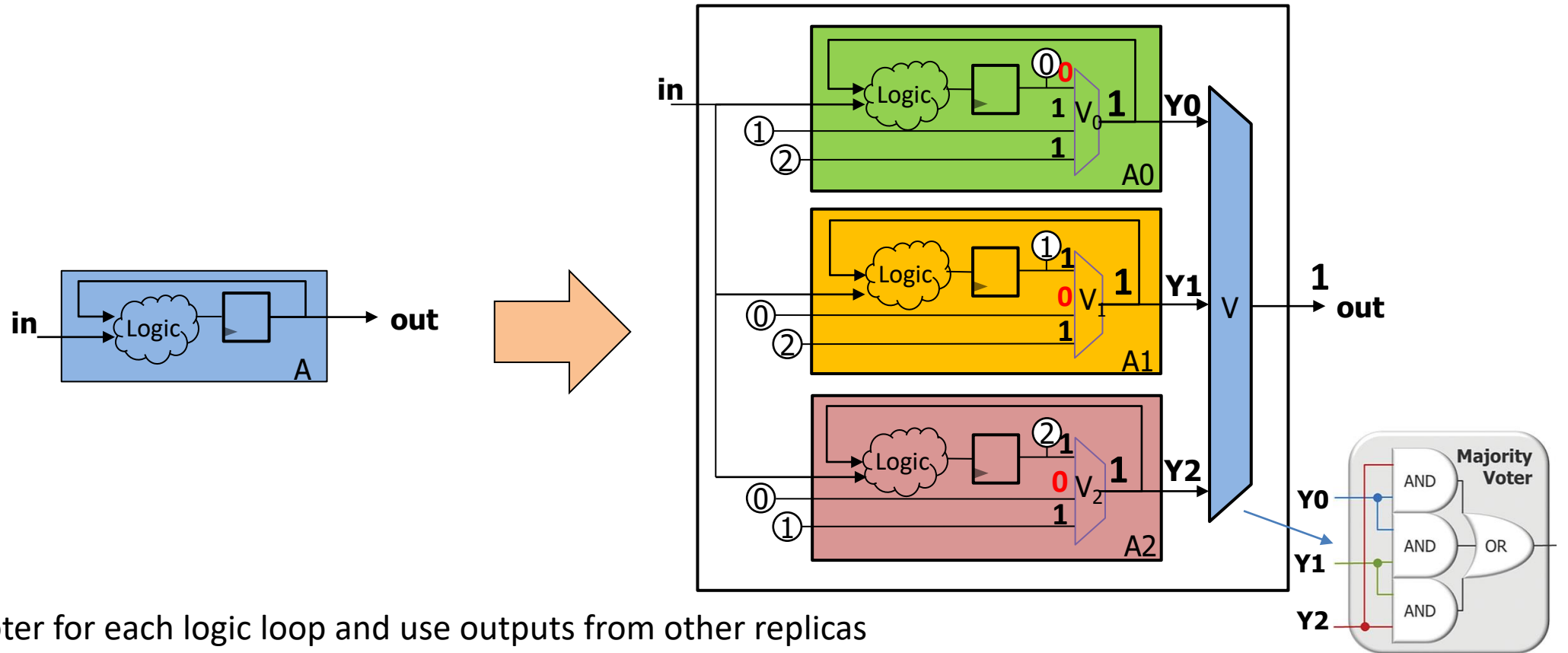


# TMR in Non-flushable Designs



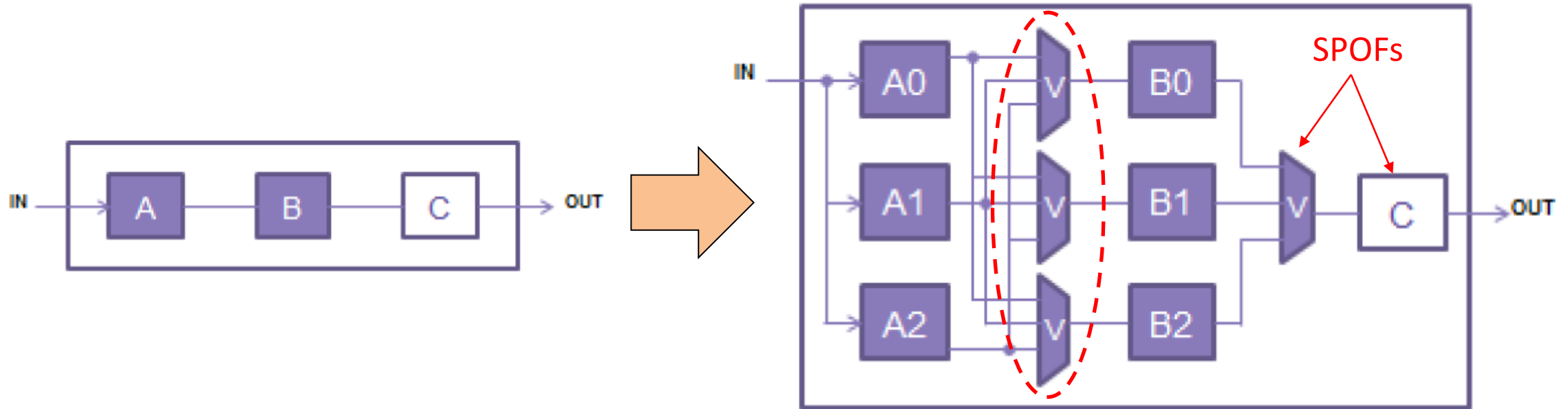
- In general a design is not flushable
- A feedback loop may keep errors in the module
- Voting according to this strategy masks a single module but it does not correct errors

# Logic Loop Voting

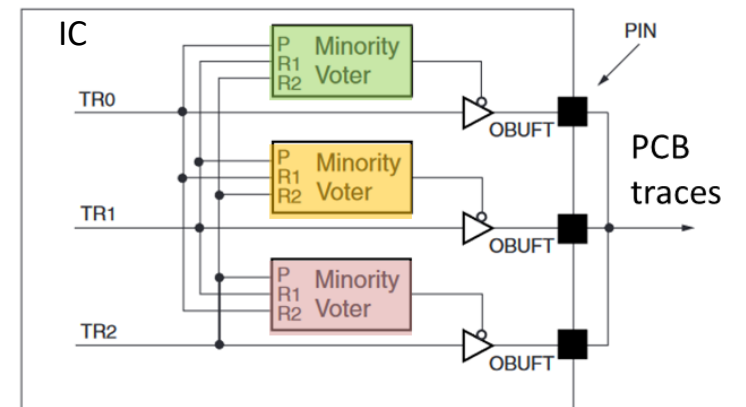


- Insert a voter for each logic loop and use outputs from other replicas
  - aka Distributed TMR in Synplify synthesis tools
- Majority vote the replica outputs
- Errors in each module are not only masked, but also removed by voters in each module
- Higher area penalty w.r.t. to plain TMR, but more reliable

# Connections Between Tripled Modules

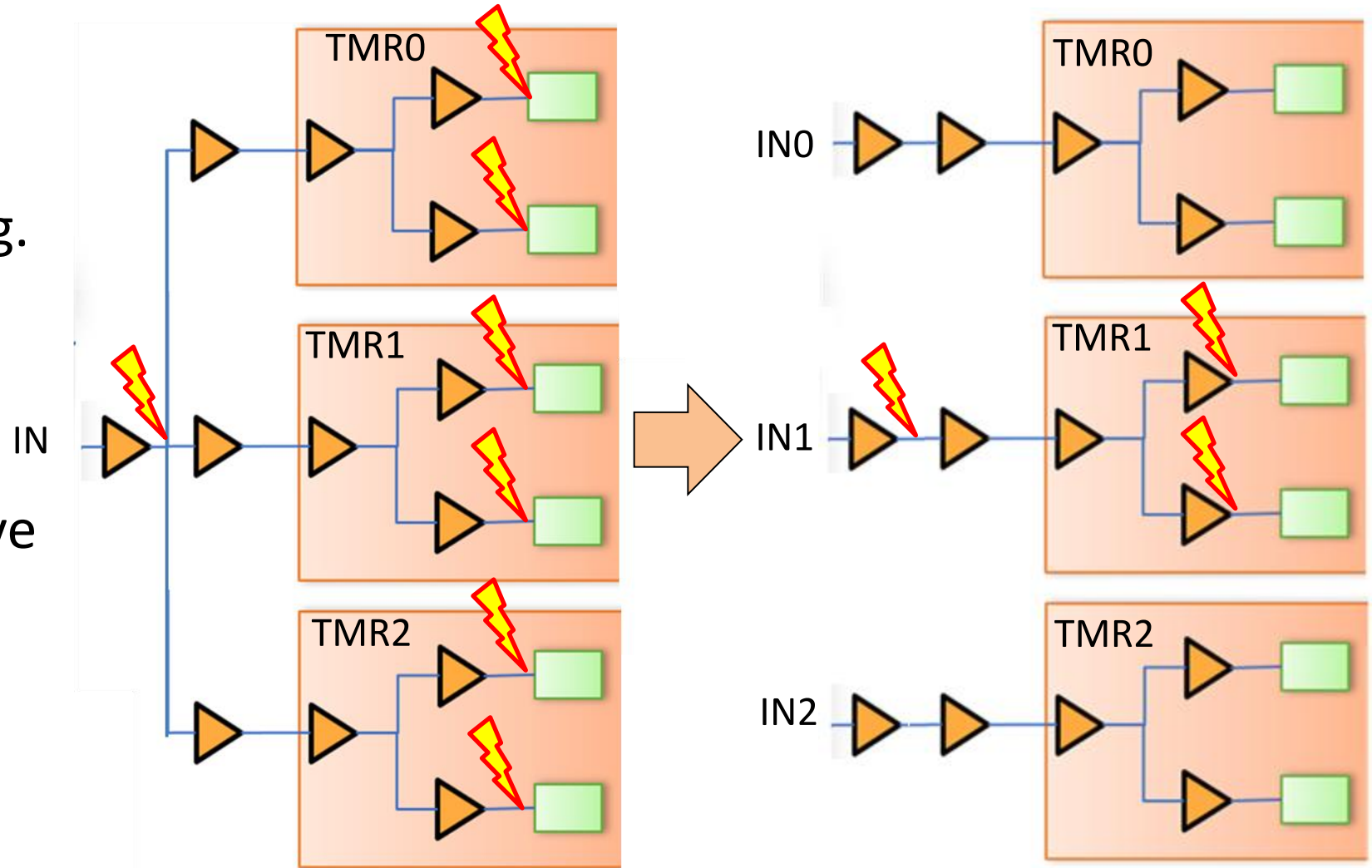


- When two tripled modules are interconnected also the voters providing input to the next module should be tripled (or you generate a SPOF)
- When a tripled module drives a non-tripled module a single voter must be used (that is a SPOF!)
- Ideally tripling should continue up to the output pins of the integrated circuit



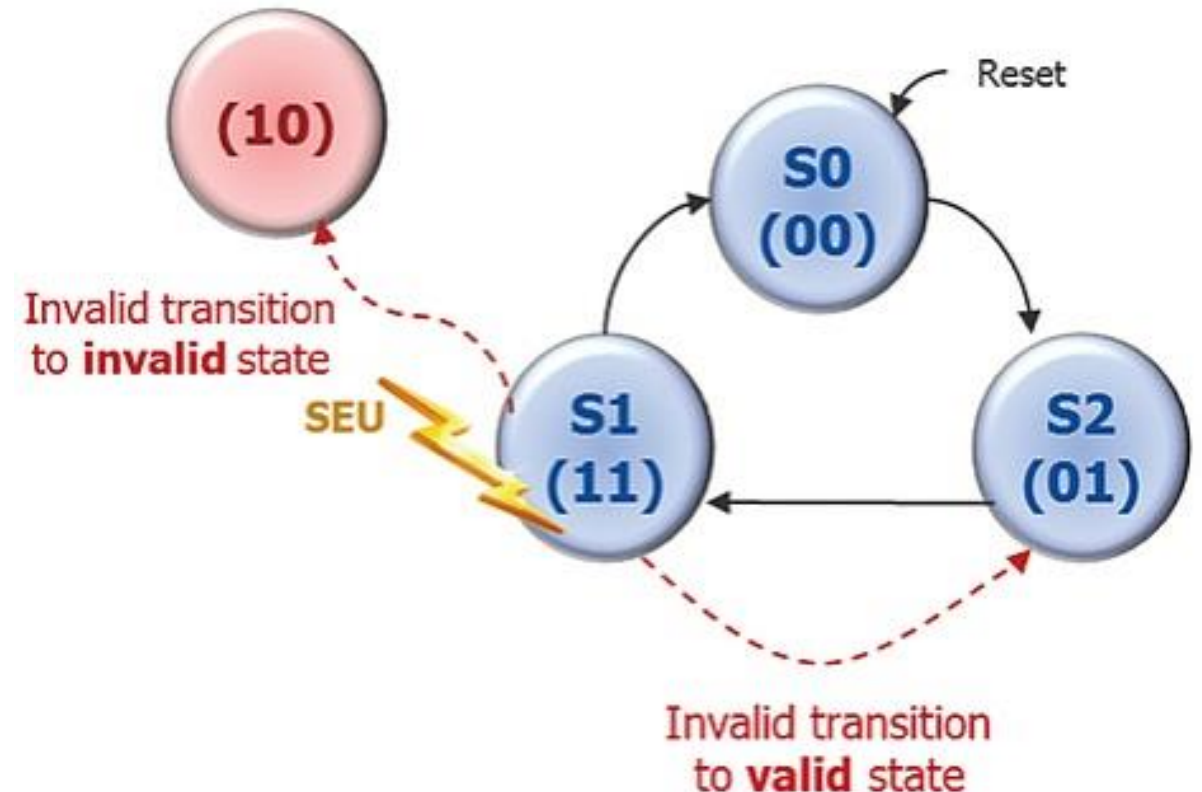
# High-fan-out Networks

- A permanent or transient fault in a high-fan-out network (HFN) (e.g. clock, reset) may affect several elements
- TMR schemes might be ineffective against it
- To mitigate this issue the entire HFN can be replicated



# Safe Finite State Machines

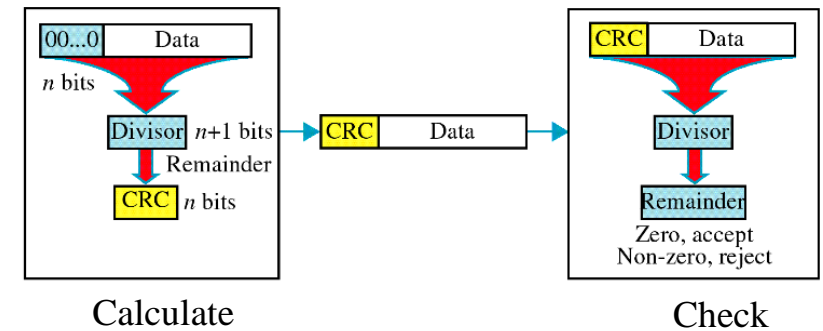
- Error detection and reset of the FSM (or go to a default state)
- “Safe-encoding” FSMs available in standard synthesis tools include additional logic to detect transitions to **invalid states** (effective for one-hot encoded FSMs)
- Dedicated high-reliability synthesis tools can implement safer FSMs capable of detecting also **invalid transitions to valid states** (also when using area-efficient binary and Gray coding)
- Error correcting codes (e.g. Hamming) on state word are normally used to address this issue



# Error Correcting Codes

- Cyclic redundancy checks (only error detecting)
  - Calculate the remainder of polynomial division between data and a predefined polynomial
  - Long sequence of shift and XOR operations, very unlikely to get the correct remainder if data gets altered
- Hamming codes
  - Parity-based code, single error correction and double error detection (SECDED)
  - $m$  data bits, require  $n$  parity bits where  $n$  satisfies  $2^n \geq m+n+1$
  - Logarithmic growing overhead
  - E.g. 11 data bits require 4 parity bits, 1000 data bits require 10 data bits
- Reed-Solomon ( $n, k$ )
  - Considers original data as  $k$  symbols (each symbol may consist of multiple bits) and generates  $n$  symbols ( $n > k$ )
  - Can correct a maximum of  $\lfloor (n - k)/2 \rfloor$  of the  $n$  symbols
  - A single incorrect bit in a symbol makes the whole symbol incorrect

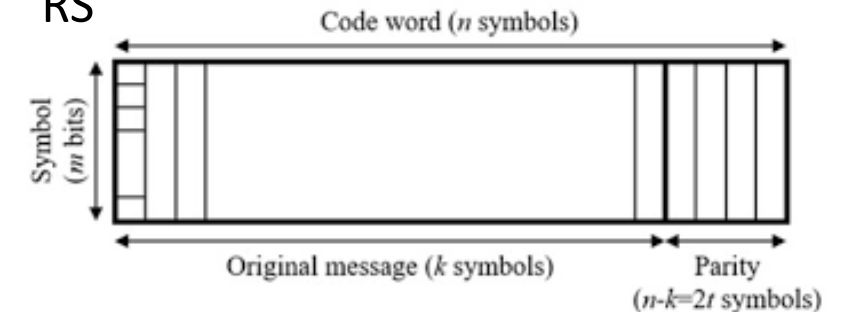
## CRC



## Hamming

Bit position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Encoded data bits	p1	p2	d1	p4	d2	d3	d4	p8	d5	d6	d7	d8	d9	d10	d11
Parity bit coverage	p1	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	p2	x	x		x	x			x	x				x	x
	p4			x	x	x	x					x	x	x	x
	p8							x	x	x	x	x	x	x	x

## RS



# Synthesis Tools for High Reliability

- Are the mentioned mitigation techniques meant to be implemented manually?
- Rarely. Usually, the safest way is to use tools for automated high-reliability synthesis
  - E.g. Some commercially available tools for FPGAs are XTMR (Xilinx only, US only), Synopsys Synplify Premier, Mentor Precision Hi-Rel
- Be careful because sometimes tools have limitations for TMR on IP blocks (e.g. Synplify does not support black boxes)

**Mentor  
Graphics**<sup>®</sup>

**Precision<sup>®</sup> Hi-Rel**

**SYNOPSYS**<sup>®</sup>

**Synplify Pro**<sup>®</sup>



# Remarks About TMR

- In general it
  - Increases area & power
  - Reduces frequency performance (it worsens timing budget)
- In order to make it effective replicas should have an independent failure probability
  - Physical separation of replicas is a good practice
- The designer should carefully assess the benefit in reliability before adopting it
  - Sometimes using it only on critical functions of the circuit could be a fair trade off between reliability, power, area and performance



# Corso di formazione INFN: Introduzione alle Tecniche di Trigger e Data Acquisition in Esperimenti di Fisica

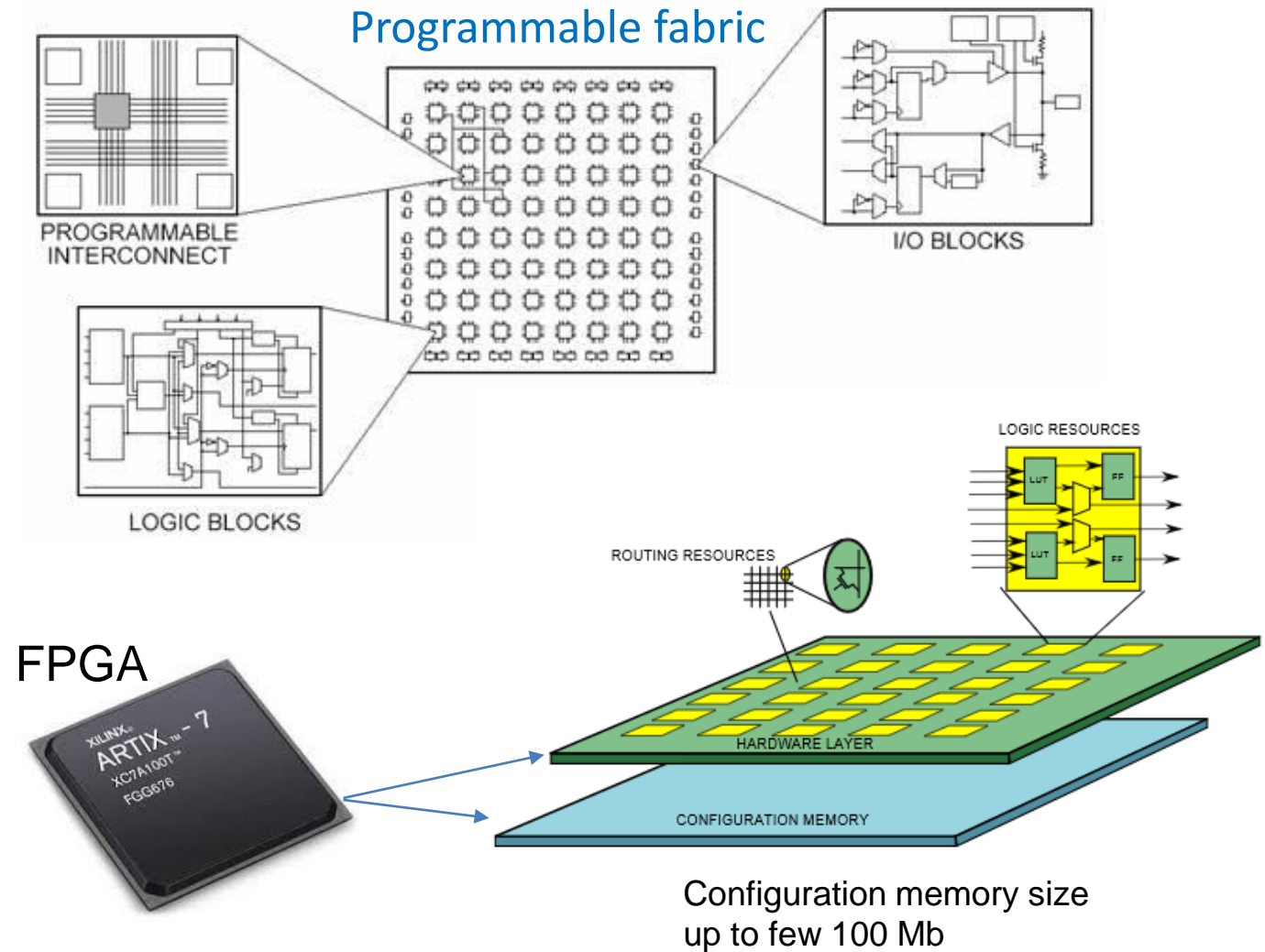
Napoli, Oct. 9-12, 2023



## **CONFIGURATION (SELF-)SCRUBBING IN FPGAS**

# Field Programmable Gate Arrays

- FPGAs are arrays of configurable logic resources
  - Configurable Logic Blocks
  - Programmable Interconnect
  - Input Output Blocks
  - Block RAMs (few Mb), multiply and accumulate for DSP, high-speed transceivers (up to 58 Gbps per differential pair)
- All programmable features are determined by the content of memory cells, collectively known as “configuration memory”
- Widely used in scientific and industrial applications for real-time data processing and transfer

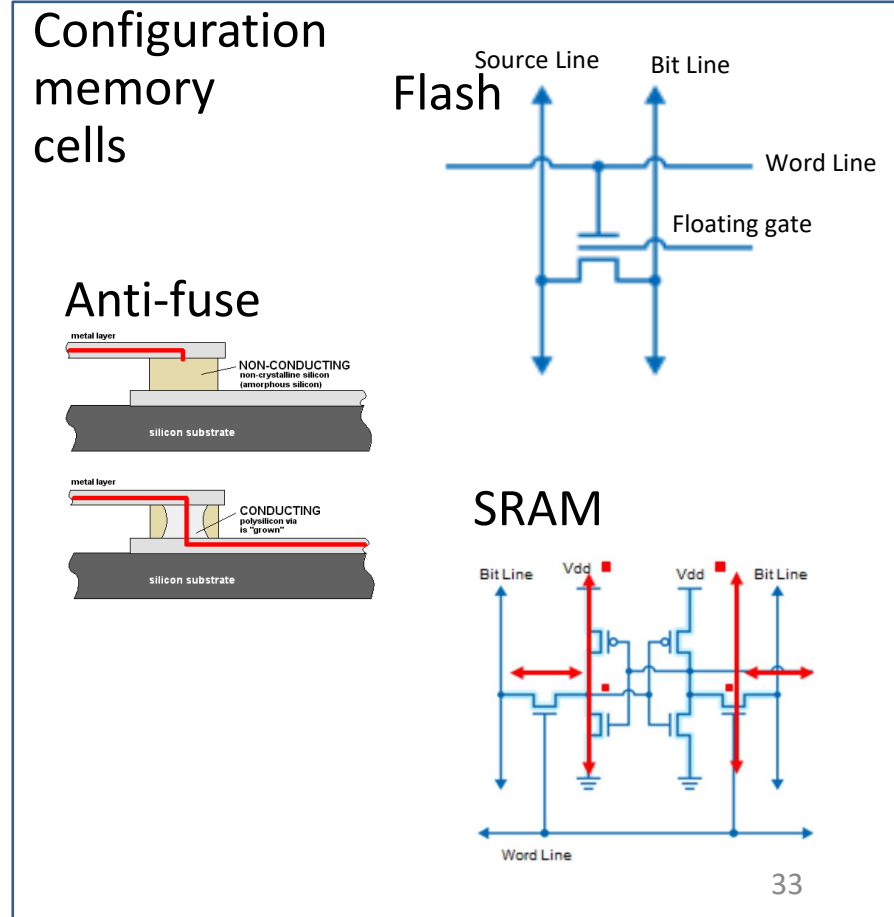


# Configuration Memory Types in FPGAs

- All FPGAs presently on the market are fabricated with CMOS processes
- The technology for the configuration memory can vary

Technology of configuration memory	Vendors	CMOS process
Flash/SONOS	Microchip (formerly Microsemi)	non-standard
Anti-fuse*	Microchip (formerly Microsemi)	non-standard
Static RAM	AMD (formerly Xilinx), Intel (formerly Altera), Lattice	standard

\*One-time programmable

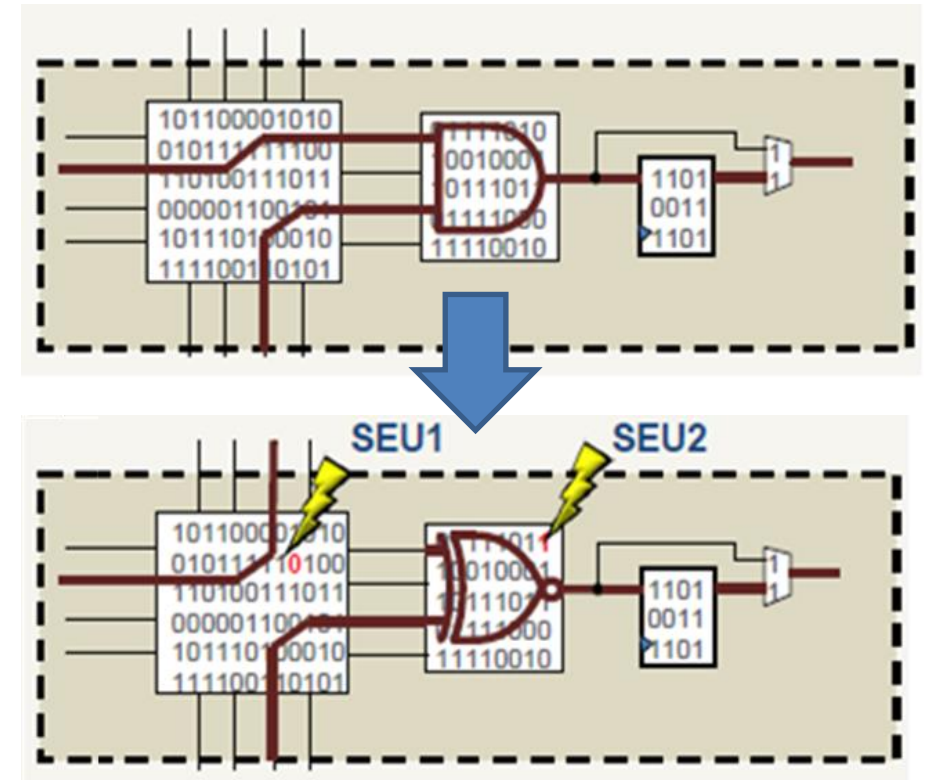


# FPGAs Vs Radiation

Technology of configuration memory	Advantages	Drawbacks
Flash/SONOS	<ul style="list-style-type: none"> <li>• Reconfigurable</li> <li>• No Configuration SEUs</li> <li>• HSIO at few Gbps (e.g. PolarFire) but slower than SRAM-based FPGAs</li> </ul>	<ul style="list-style-type: none"> <li>• TID tolerance &lt; kGy (e.g. Igloo2)</li> <li>• Design tools are not as mature as SRAM-based FPGAs</li> </ul>
Anti-fuse	<ul style="list-style-type: none"> <li>• No Configuration SEUs</li> </ul>	<ul style="list-style-type: none"> <li>• Limited amount of logic resources</li> <li>• One-time programmable (OTP), not reconfigurable</li> <li>• LVDS @ few hundreds Mbps</li> <li>• TID tolerance &lt; kGy</li> </ul>
Static RAM	<ul style="list-style-type: none"> <li>• Reconfigurable</li> <li>• Latest CMOS process (e.g. Ultrascale+ 16nm FinFET), highest performance for logic and HSIO (tens of Gbps)</li> <li>• Best TID tolerance (few kGy at least e.g. Kintex-7)</li> </ul>	<ul style="list-style-type: none"> <li>• SEUs in configuration</li> <li>• Require configuration repair «scrubbing»</li> <li>• Higher power consumption</li> </ul>

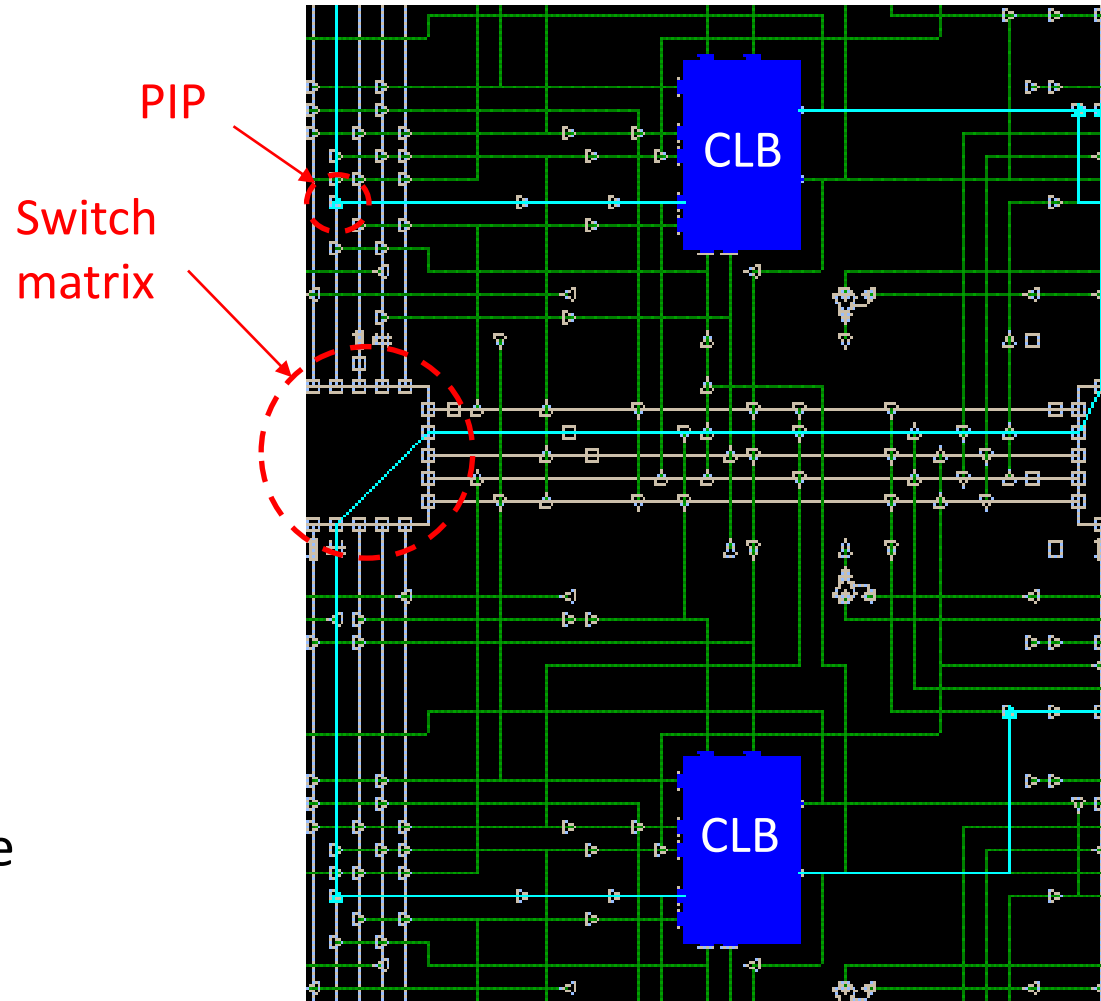
# SRAM-based Field Programmable Gate Arrays

- Many devices are TD-tolerant and latch-up free
  - e.g. Xilinx 7-Series withstand TD > 1 kGy
- Highest performance among FPGAs
  - LVDS up to 1.6 Gbps, SerDes up to 112 Gbps
- Always remember **the configuration is the circuit!**
- **SEUs in configuration RAM might alter functionality** and must be corrected
- Radiation-hardened FPGAs exist but they are aimed at space applications mostly, price can be as high as 50 k\$ per device! (e.g. Virtex-5QV)
- We refer to AMD (Xilinx) devices in this lecture



# Programmable Interconnect

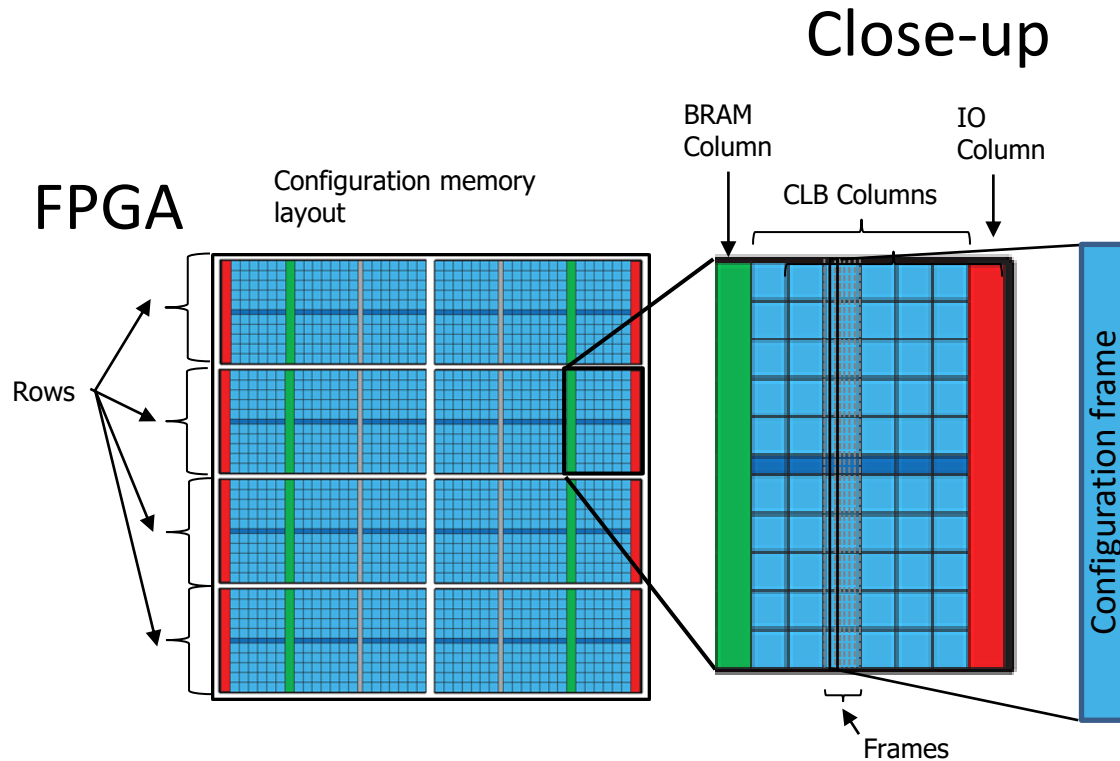
- Programmable routing is a key element of an FPGA
- An interconnection is realized by properly closing or opening switches between predefined metal lines
  - Programmable Interconnection Points (PIPs)
  - Switch matrices (arrays of PIPs)
- Typically ~ 90% of configuration bits determine routing! (excluding BRAMs)



Xilinx  
XC3000



# Configuration Memory in Xilinx FPGAs



- Configuration RAM is divided in rows
  - Horizontal slices along the device
- Each row is divided in columns
  - vertical slices within a row
- Columns configure different resources (CLB, routing, IO, etc.)
- Each column is divided in frames
- The frame is the minimum accessible element of configuration memory
  - In 7-Series a frame is 3232b long

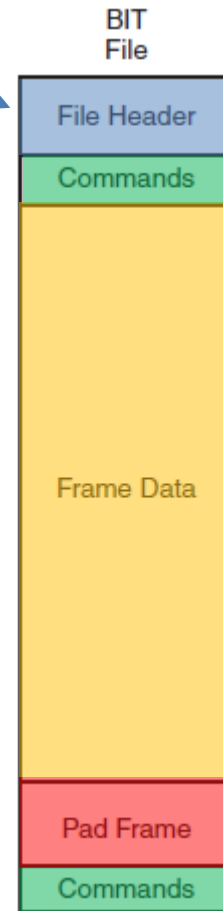
- Number of frames in a column depends on type of resources configured by that column
  - In 7-Series column size few tens of frames (28-128)
- In Xilinx devices one row is high exactly as one clock region



# Bitstream

- The interface to configuration is managed by R/W to dedicated registers
- Before and after proper **frame data** there is the need to supply **commands** for properly initializing these registers according to the needed behavior
  - Read/write configuration frames
  - Activate IO/interconnect
  - FPGA shutdown/start
  - CRC check
  - IDCODE verification
  - Dynamic elements masking in readback
- Collectively these commands and data are called bitstream
- A bitstream is generated by CAD tools and can be loaded through several interfaces
  - JTAG, SelectMAP, ICAP, BPI, SPI
- How to look in a bitstream?
  - <http://torc-isi.sourceforge.net/> supports up to Virtex-6

Design filename, used tool, date, etc.



## E.g. Xilinx 7-Series Commands

32b words	Description
3001C001	Packet Type 1: Write COR1 register, WORD_COUNT=1
00000000	COR1[31:0]=00000000 (hex) (Sets configuration options, e.g., POST_CRC enable, BPI page mode.)
30018001	Packet Type 1: Write IDCODE register, WORD_COUNT=1
03651093	IDCODE[31:0]=03651093 (hex) (If written IDCODE does not match device IDCODE, then error.)
30008001	Packet Type 1: Write CMD register, WORD_COUNT=1
00000009	CMD[4:0]=01001 (binary) = SWITCH (change CCLK frequency)
20000000	NOOP
3000C001	Packet Type 1: Write MASK register, WORD_COUNT=1
00000401	Bit mask for write to CTL0/CTL1 register. MASK[31:0]=00000401 (hex)

## Configuration Registers

Name	Read/Write	Address	Description
CRC	Read/Write	00000	CRC Register
FAR	Read/Write	00001	Frame Address Register
FDRI	Write	00010	Frame Data Register, Input Register (write configuration data)
FDRO	Read	00011	Frame Data Register, Output Register (read configuration data)
CMD	Read/Write	00100	Command Register
CTL0	Read/Write	00101	Control Register 0
MASK	Read/Write	00110	Masking Register for CTL0 and CTL1

# TMR & Configuration Scrubbing

- Reliability  $R(t)$  is the probability of correct operation in a time interval  $[0,t]$  assuming correct operation at  $t=0$ ,  $R(t=0)=1$
- Let us suppose we want to estimate the reliability of a TMR system where the simplex reliability decays exponentially

$$MTTF = \int_0^{\infty} R dt$$

$$R_{\text{simplex}} = e^{-\lambda t}$$

$$MTTF_{\text{simplex}} = \frac{1}{\lambda}$$

TMR system

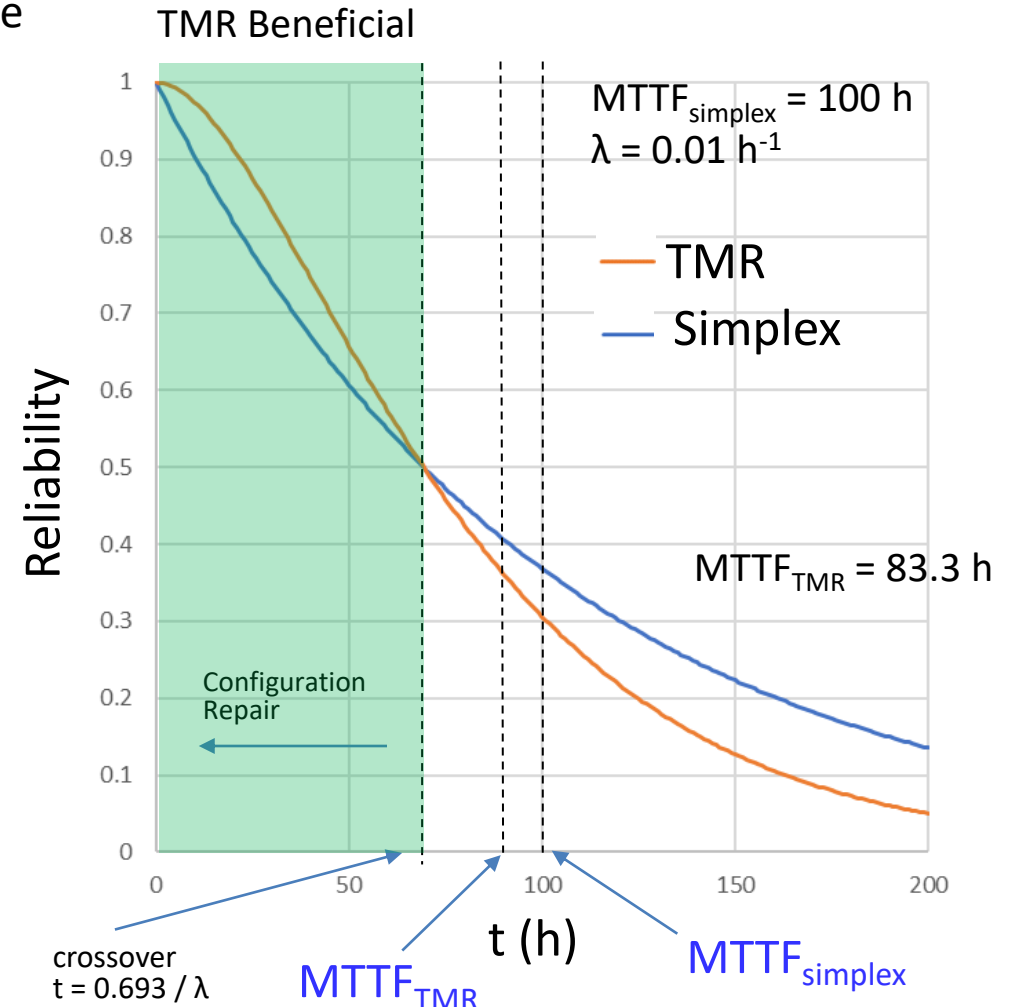
$$R_{\text{TMR}} = 3e^{-2\lambda t} - 2e^{-3\lambda t}$$

$$MTTF_{\text{TMR}} = \frac{3}{2\lambda} - \frac{2}{3\lambda} = \frac{5}{6\lambda} < \frac{1}{\lambda}$$

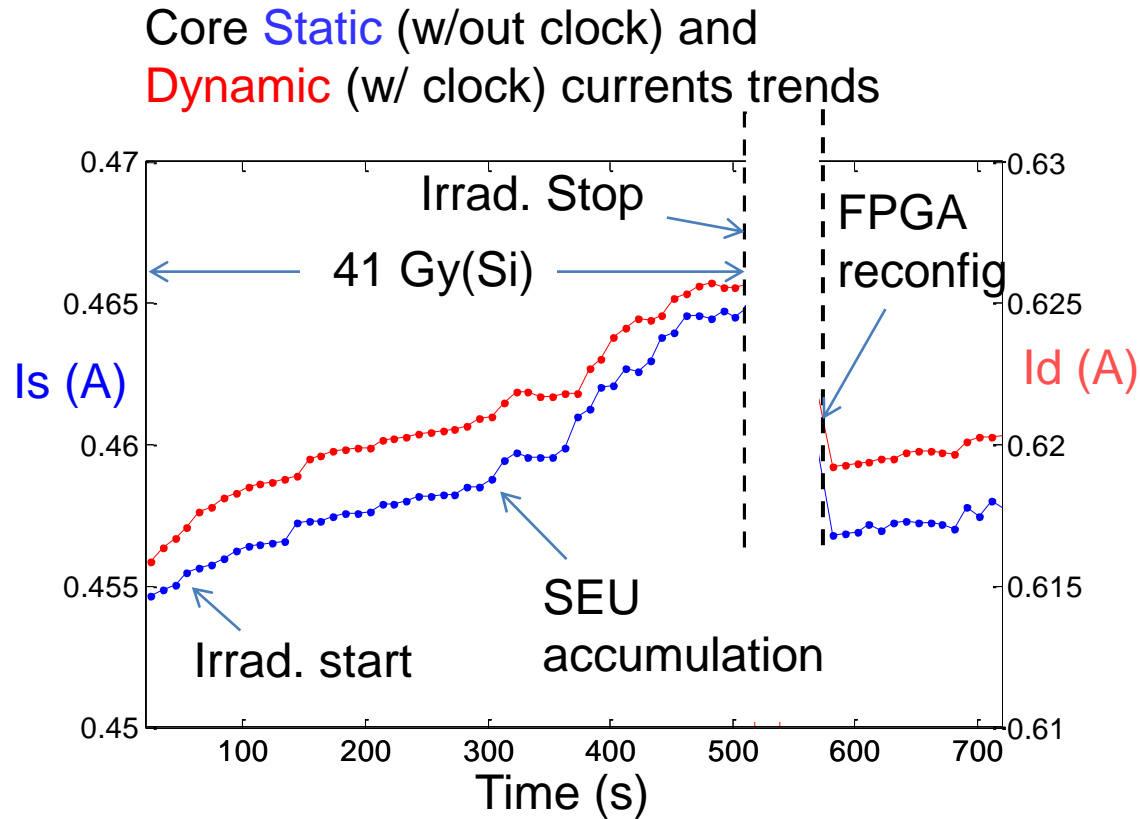
$$MTTF_{\text{TMR}} < MTTF_{\text{simplex}}$$

$$R_{\text{TMR}} = 3R^2 - 2R^3$$

- Mean time to failure in TMR systems ( $MTTF_{\text{TMR}}$ ) is worse than  $MTTF$  for each module ( $MTTF_{\text{simplex}}$ )
- For TMR to be beneficial,  $R_{\text{simplex}}$  must be  $> 0.5$
- Configuration repair (aka "scrubbing") can restore  $R(t)$



# SEU Accumulation Vs Power Consumption

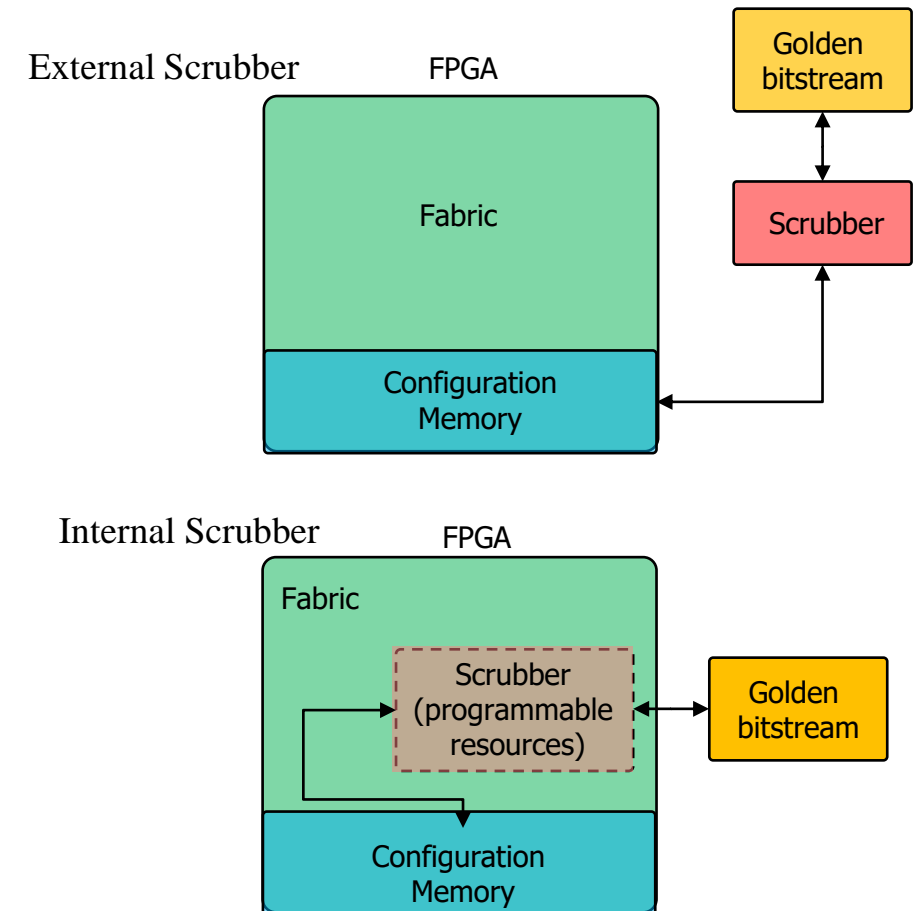


- Irradiation @ 5 Gy/min (Si)
- 5174 SEUs in 484 s (time to failure)
- 641 SEU/min => 128 SEU/Gy (Si)
- Total current increase 15mA => 2.8  $\mu$ A/SEU

- 62-MeV proton irradiation of Virtex5 LX50T FPGA
- Accumulation of SEUs in the configuration leads to current increase
  - Possibly clashes on programmable routing
- Dynamic and static core currents exhibit very similar trends => due to quiescent current trend
- Reconfiguration partially reduces the current increase

# External Vs Internal Scrubbing

- Configuration scrubbing can be performed in several ways
  - **Blind scrubbing**, i.e. overwriting configuration retrieved from “golden” (radiation-hardened) memory (simple and fast, but no information about upsets is gathered)
  - **Readback scrubbing**, configuration is read back and compared to the “golden” image, only in case of differences it is written back
  - More complex readback scrubbing methods based on error correcting codes on frame data, or even tripled configuration
- The **scrubber can be internal or external** to the device
  - If internal it is subject to SEUs in the FPGA, but the system is more compact
  - There are vendor-provided, ready-made solutions
- Scrubbers normally cannot correct errors in BRAMs, distributed RAM, flipflops



# The Soft Error Mitigation Controller

- Xilinx Soft Error Mitigation (SEM) controller
  - Internal scrubber
  - **not radiation-tolerant!**
  - designed for very low upset rate environments (e.g. Earth atmosphere)
  - Accesses configuration via the internal configuration access port (ICAP)
  - Hamming codes at frame level and CRC at device level
  - Log/control via UART
  - Ready-to-use IP, just select options and generate core

Table 2-5: Non-SSI: Max Error Correction Latency (100 MHz) No Throttling on Monitor Interface

Correction Mode	Errors in Frame (Correctability)	Error Correction State at ICAP_F <sub>Max</sub>
Repair	1-bit (Correctable)	610 $\mu$ s
	2-bit (Uncorrectable)	25 $\mu$ s
Enhanced Repair	1-bit (Correctable)	610 $\mu$ s
	2-bit (Correctable)	18,790 $\mu$ s
	2-bit (Uncorrectable)	9,110 $\mu$ s
	BFR <sup>(1)</sup> -only (Uncorrectable)	10 $\mu$ s
Replace	Any (Correctable)	830 $\mu$ s
Any	CRC-only (Uncorrectable)	10 $\mu$ s

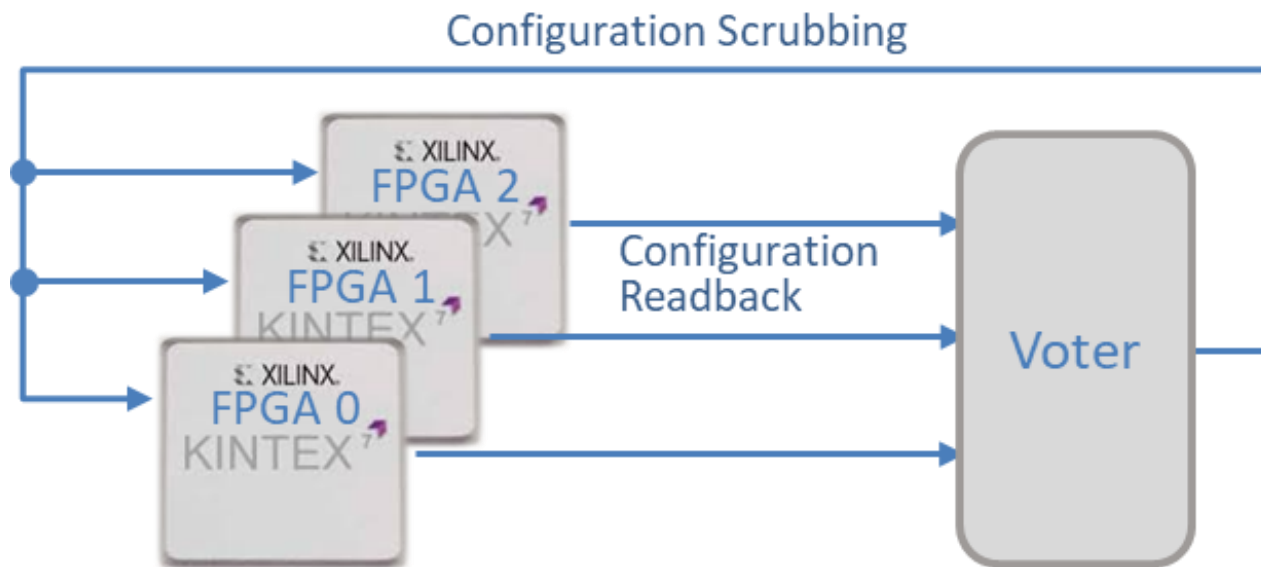
**Notes:**

1. BFR is an error condition due to a multi-bit upset in an enhanced repair checksum stored in block RAM.

## SEM resource occupation

Device	LUTs	FFs	18kb BRAMs
Virtex-7	510	361	3
Kintex-7	511	361	3
Artix-7	512	361	3
Zynq-7000	509	361	3
Spartan-7	498	406	3

# TMR-based External Scrubbing

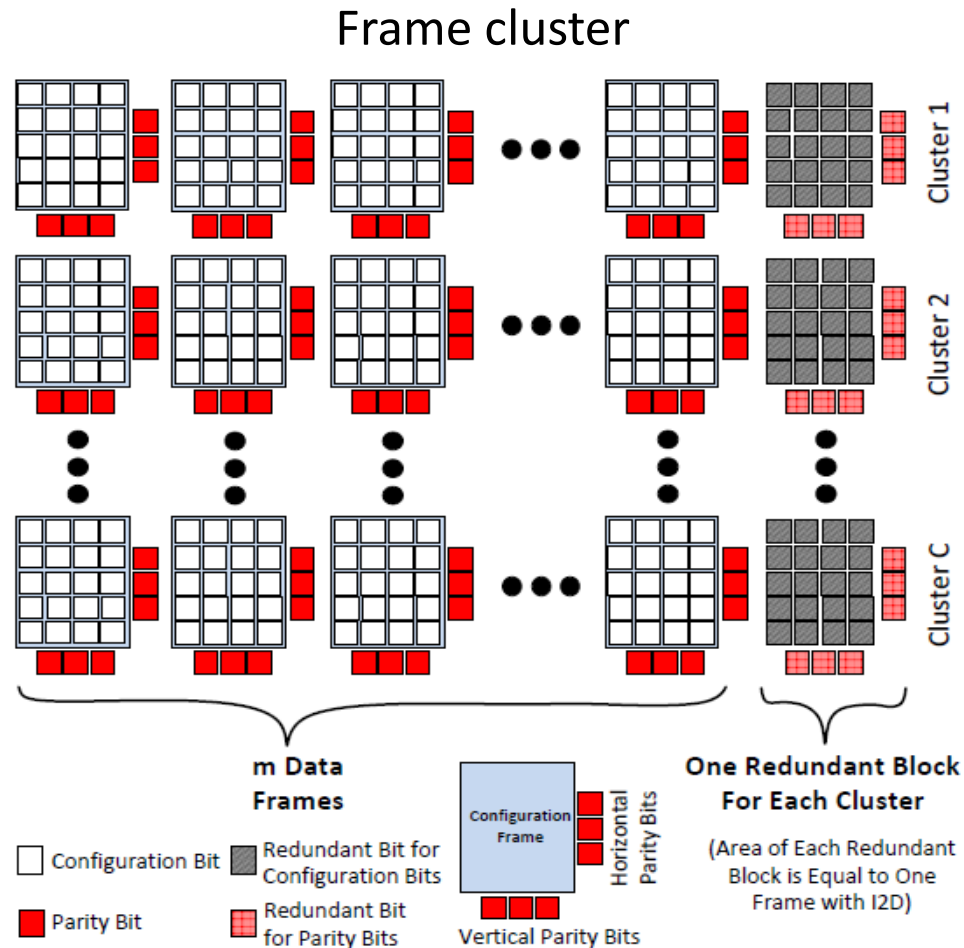


- Redundant FPGAs with identical bitstreams and operating on the same data
- Pros
  - TMR also for functionality
  - Simple to implement
  - Very high correction capability
  - Low repair latency
- Cons
  - TMR to generate redundant configuration
  - 3x cost and power
  - need for a radiation-tolerant voter





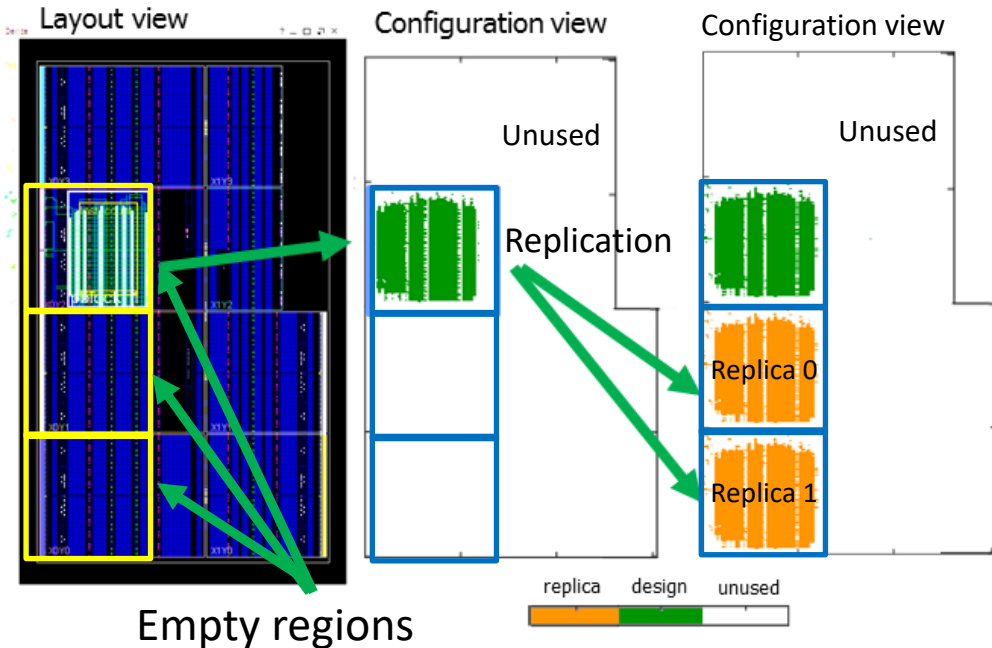
# Example of Custom ECC-based Scrubbing



- Clustering frames and adding parity bits to each frame for error detection
- Erasure code for error correction in clusters
  - Wrong frame gets “erased” (i.e. ignored) and replaced by means of code calculation
- Pros:
  - Minimal power consumption increase
  - Low memory overhead
- Cons:
  - Repair requires readback of whole cluster, latency depends on cluster size
  - Correction capability depends on number of clusters (1 frame per cluster can be corrected)
  - Efficiency of code depends on device



# Redundant-configuration-based Scrubbing



1. Generate FPGA bitstream and identify used configuration frames
2. Replicate used frames and keep track of redundant and remaining empty frames
3. During FPGA operation, **majority vote** redundant frames for error detection and correction and **check** empty frames through ICAP

- Self-contained, no need for external memories
- Robust, can correct all the bits in a frame
  - Assuming no errors in homologous redundant bits
- Configuration replicas do not receive clock => *no additional dynamic power consumption*
- Approach portable on several FPGA families
- Drawbacks:
  - need to analyze bitstream
  - need at least 3x bigger device



R. Giordano, "Method for generating redundant configuration in FPGAs," PCT Application no. PCT/IB2018/060461, 2018, Dec. 20

R. Giordano et al. "Configuration Self-repair in Xilinx FPGAs," in IEEE Trans. on Nucl. Sci., vol. 65, no. 10, pp. 2691-2698, Sept. 2017. [Open Access](#)

<https://ieeexplore.ieee.org/document/8456573>

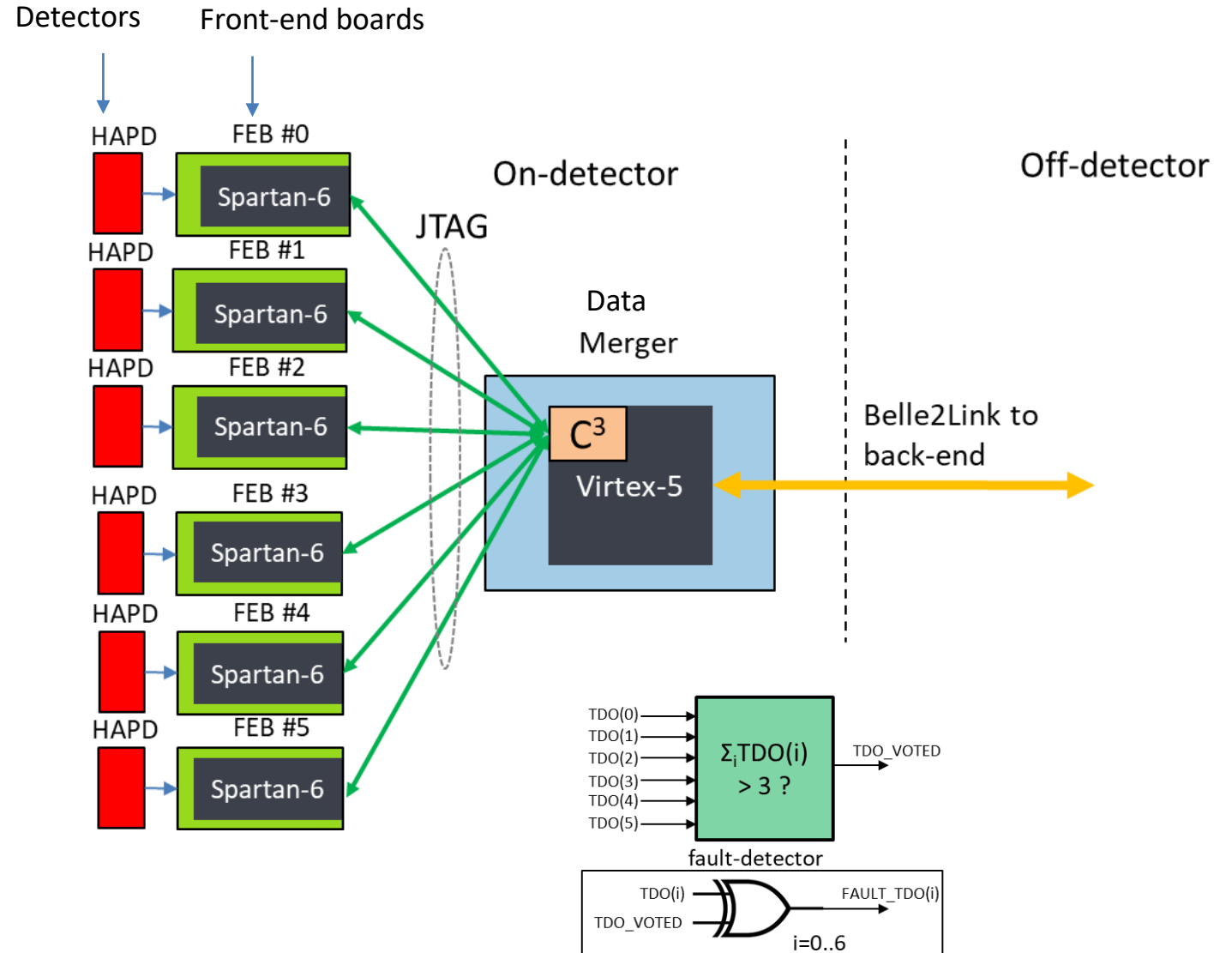
# On-the-fly Self-Scrubbing at System Level

- Example from Belle II Aerogel Imaging Ring Cherenkov detector
- Star read-out topology
- FEB FGAs are programmed with the same bitstream => redundancy at system-level

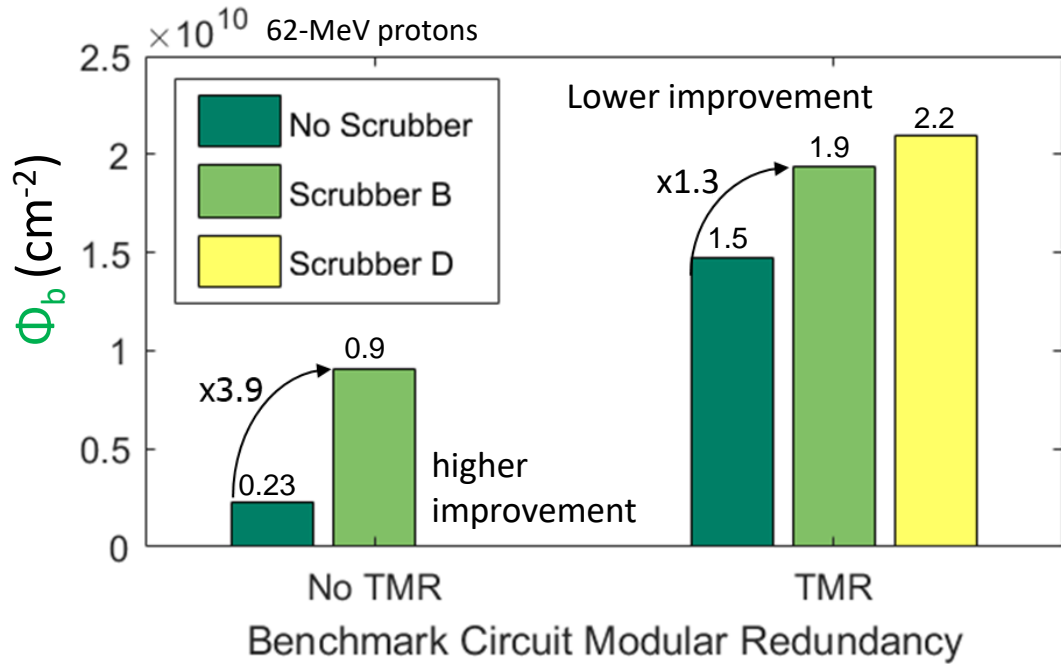


## Idea

- Parallel readback of FEB (Spartan-6) configuration from Merger (Virtex-5)
- Real-time 4-out-of-6 bitwise majority voting on JTAG streams (TDOs) for error detection
- Quick single frame reconfiguration for error correction
- Readout topology is widely used in DAQ systems, easily-exportable solution



# Mean Fluence Before Failure



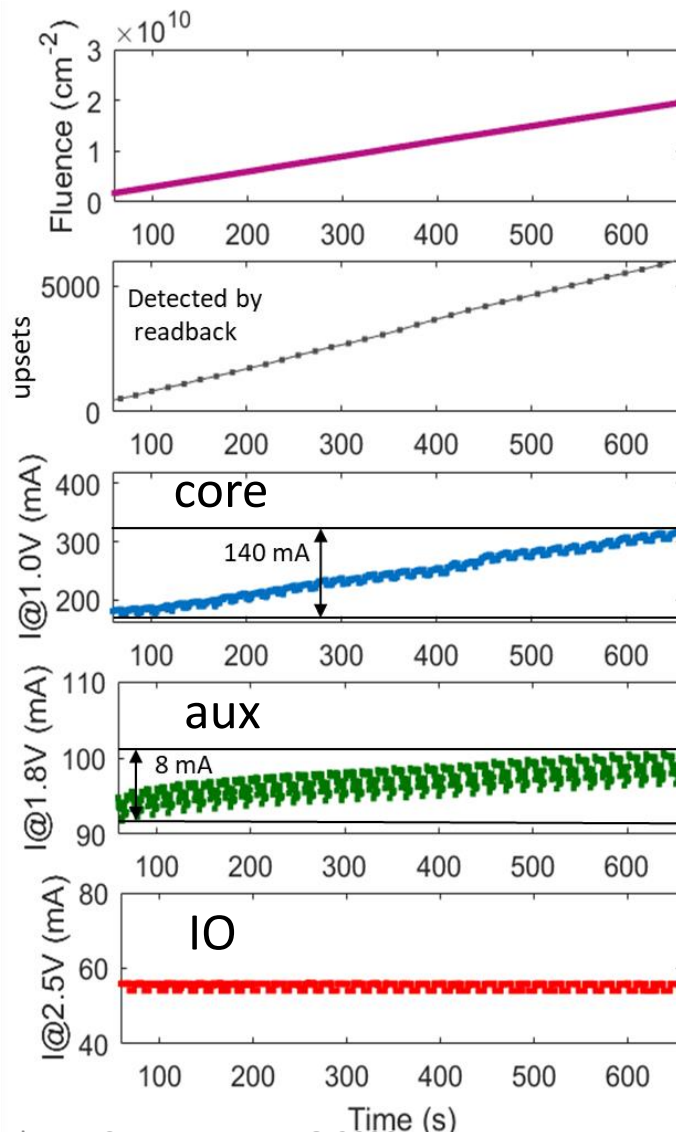
Firmware	# of runs	<Flux> (cm <sup>-2</sup> s <sup>-1</sup> )	Φ <sub>b</sub> (cm <sup>-2</sup> )	Φ <sub>s</sub> (cm <sup>-2</sup> )
No TMR /No Scrub	20	2.2·10 <sup>7</sup>	2.3·10 <sup>9</sup>	-
No TMR /Scrub B	21	3.5·10 <sup>7</sup>	9.0·10 <sup>9</sup>	7.7·10 <sup>9</sup>
TMR /No Scrub	20	3.5·10 <sup>7</sup>	1.5·10 <sup>10</sup>	-
TMR /Scrub B	30	3.2·10 <sup>7</sup>	1.9·10 <sup>10</sup>	1.1·10 <sup>10</sup>
TMR /Scrub D	19	3.4·10 <sup>7</sup>	2.2·10 <sup>10</sup>	1.2·10 <sup>10</sup>

- Φ<sub>b</sub> = proton fluence to failure for **Benchmark**
- TMR on the Benchmark improves Φ<sub>b</sub> by 6.5x (No Scrubber) and 2.4x (Scrubber B)
- Scrubber B increases Φ<sub>b</sub> by 3.9x w/out TMR and by 1.3x w/ TMR
- Scrubber D increases Φ<sub>b</sub> by 1.5x
- Φ<sub>s</sub> = proton fluence to failure for **Scrubber**

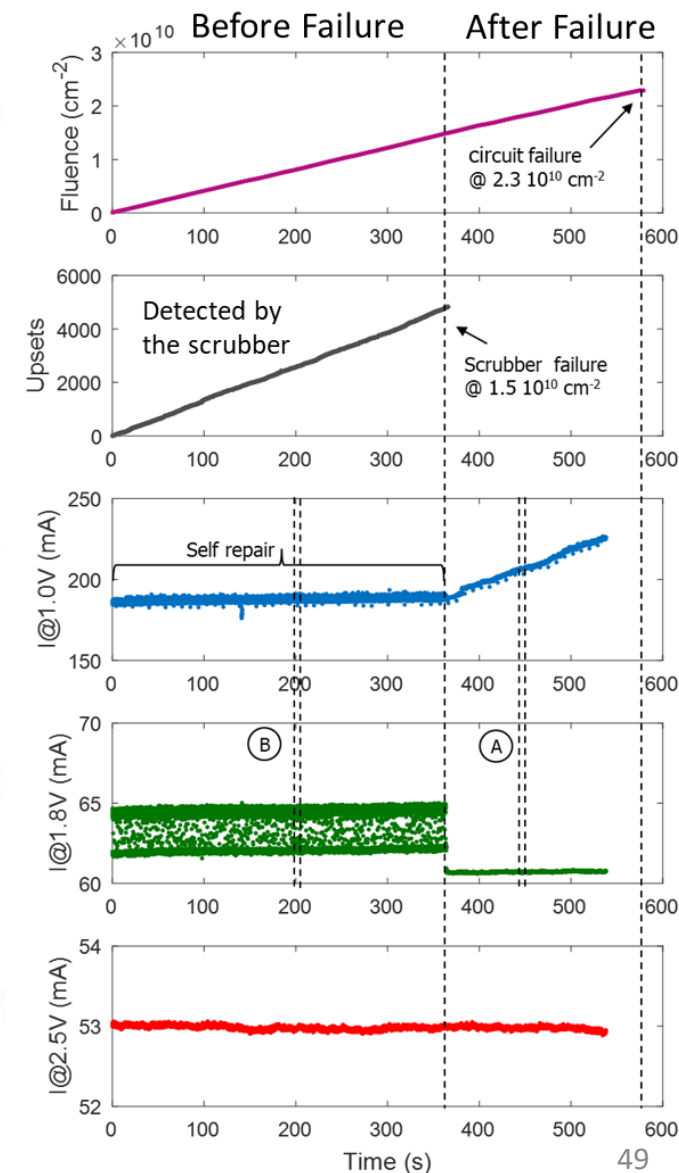
# Power Consumption

- Experimental results during 62-MeV proton irradiation
- When scrubbing is not active, upsets accumulate, and power consumption at core (1.0V) and auxiliary power supplies (1.8V) increases with fluence
- When scrubbing is active, different behaviors before and after the failure of the scrubber
  - Before: upsets are detected and removed, power consumption stable ( $\Delta I < 1\text{mA}$ )
  - After: upsets begin to accumulate, power consumption at core grows, eventually benchmark fails

w/out scrubbing

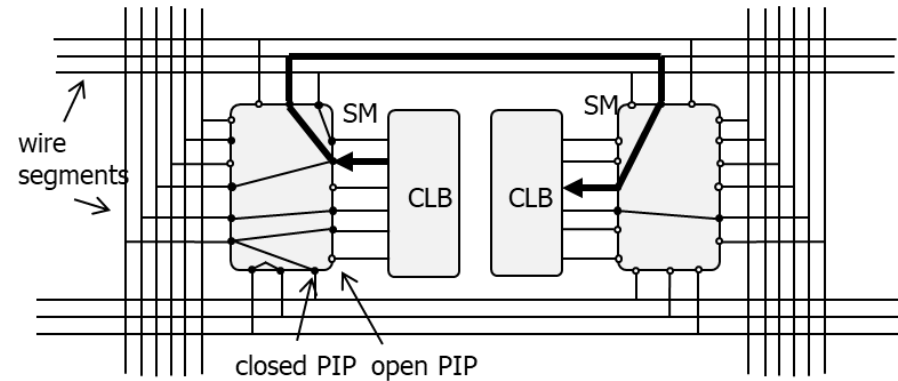


w/ scrubbing

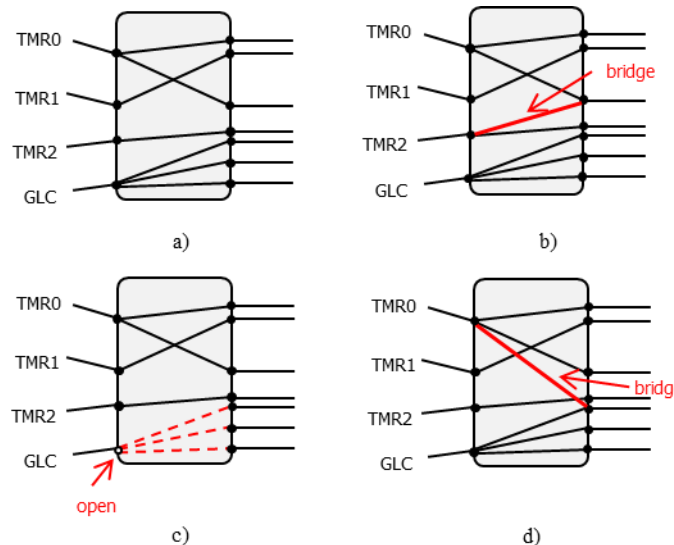
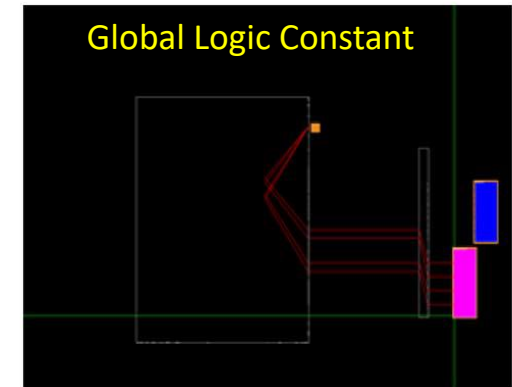


# (TMR) Domain Crossing Errors

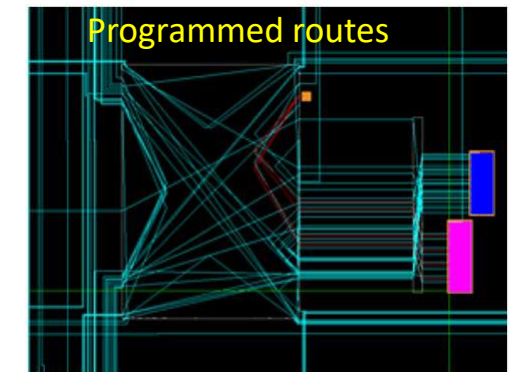
- When TMR domains share resources (interconnect or CLBs) single and multiple bit upsets can affect more than one TMR domain
- For instance, SEUs in SMs may corrupt more than one TMR domain
  - bridging between TMR domains
  - disconnecting global logic constants (GLCs)
- Experimental studies on FPGAs show that TMR can be defeated even by single bit upsets
- Routing is a concern
  - Virtex-II accelerator test showed 48% critical SEUs involved routing



Switch matrix



Switch matrix

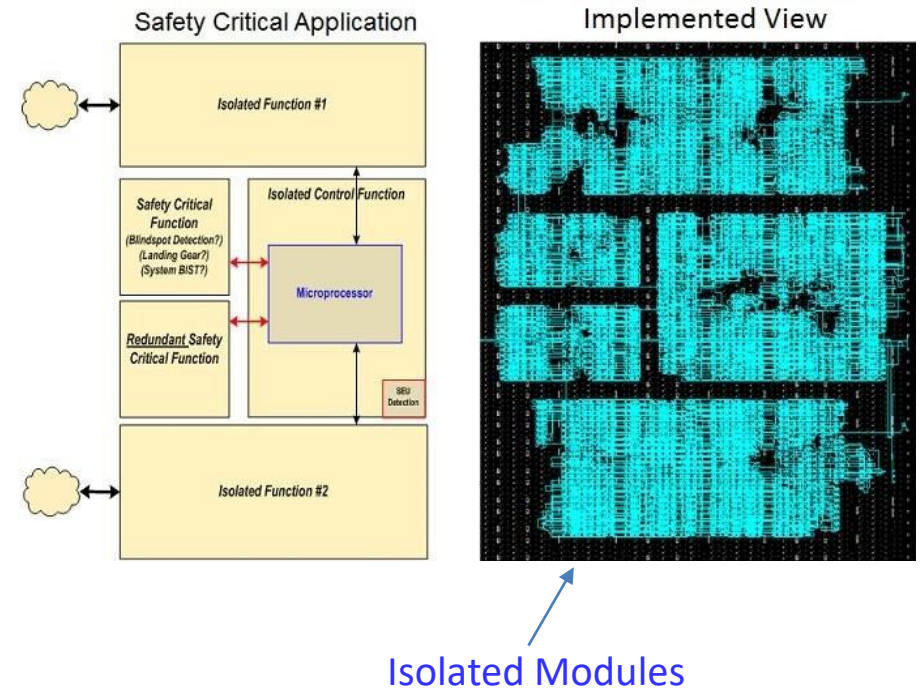




# Placement and Routing Hardening

- Reliability-oriented place and route algorithms minimize SPOFs, optimized to
  - avoid placing logic pertaining to different domains in the same CLB
  - reduce # of PIPs
  - reduce # of switch matrices which mix TMR domains
  - do not route GLCs through switch matrices
  - complete module isolation
- Examples
  - Xilinx Isolation Design Flow (available in Vivado and ISE)
  - RoRA academic tool designed by PoliTo\*

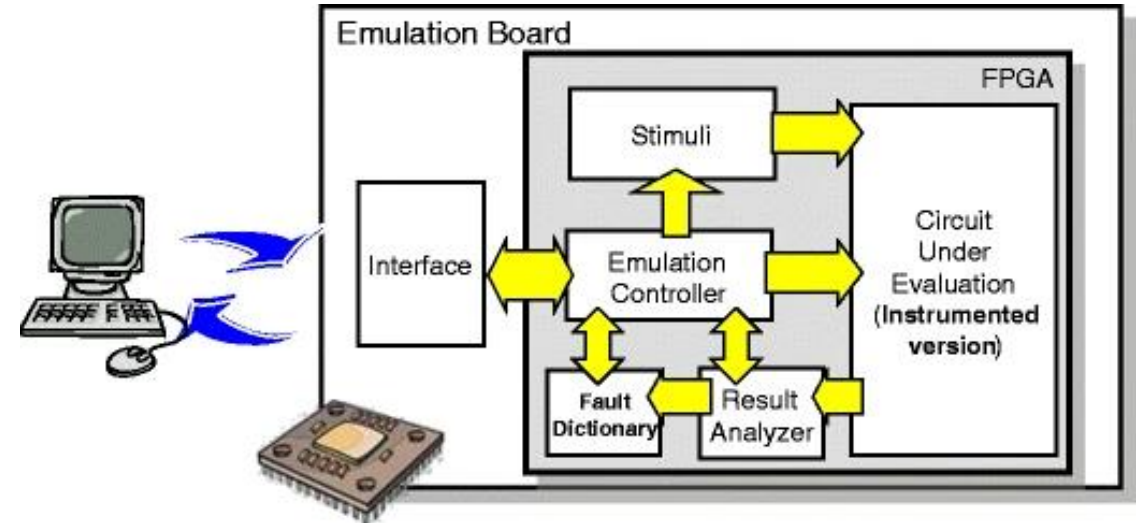
Xilinx Isolation Design Flow



\*L. Sterpone, M. S. Reorda and M. Violante, doi: 10.1109/RME.2005.1543031.

# Fault-injection Testing

- We have seen that configuration access ports (SelectMAP, JTAG, ICAP, etc.) can be used to repair configuration
- They can also be used to intentionally alter the configuration to emulate SEUs and become a tool for testing
- This makes it possible to probe the sensitivity of the circuit to upsets before performing irradiation tests
- A limitation of this approach is that
  - SEUs cannot be injected in the CAP itself (no SEFIs)
  - SETs are not emulated
- Example of fault-injectors: Xilinx SEM\*, FLIPPER\*\*
- It is also possible to do it via JTAG by means of dedicated scripts (configuration user guide)



\*[www.xilinx.com/video/fpga/seu-integration-test-by-error-injection.html](http://www.xilinx.com/video/fpga/seu-integration-test-by-error-injection.html)

\*\*M. Alderighi *et al.*, doi: 10.1109/TNS.2010.2043855.



**Thank you!**



# Questions?

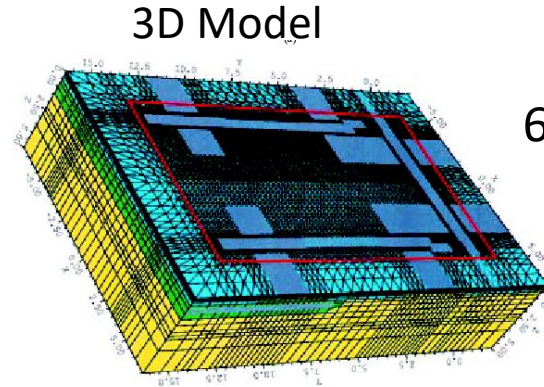
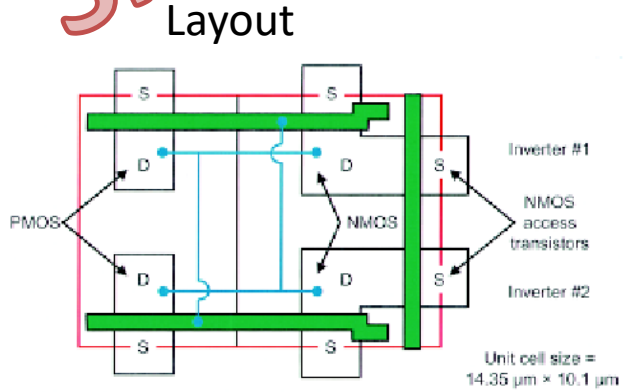


Backup

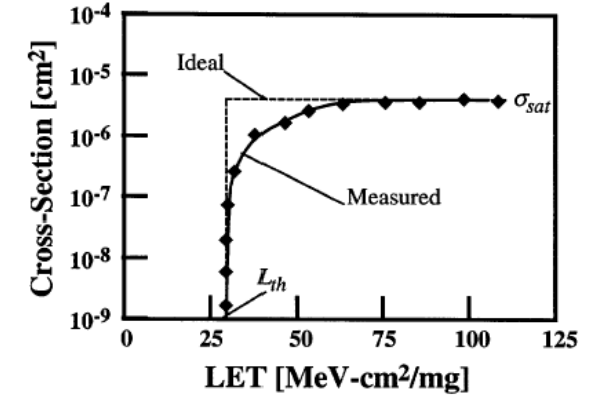
**BACKUP**

Skip

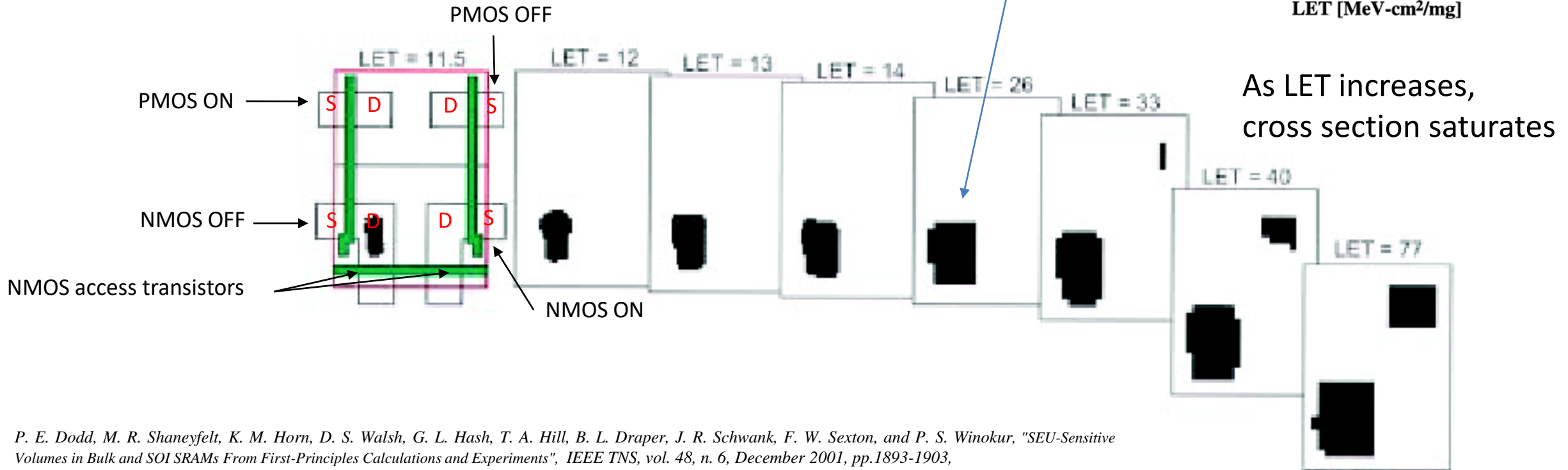
# SEU Sensitive Volume in a SRAM cell



6T SRAM cell



Sensitive area

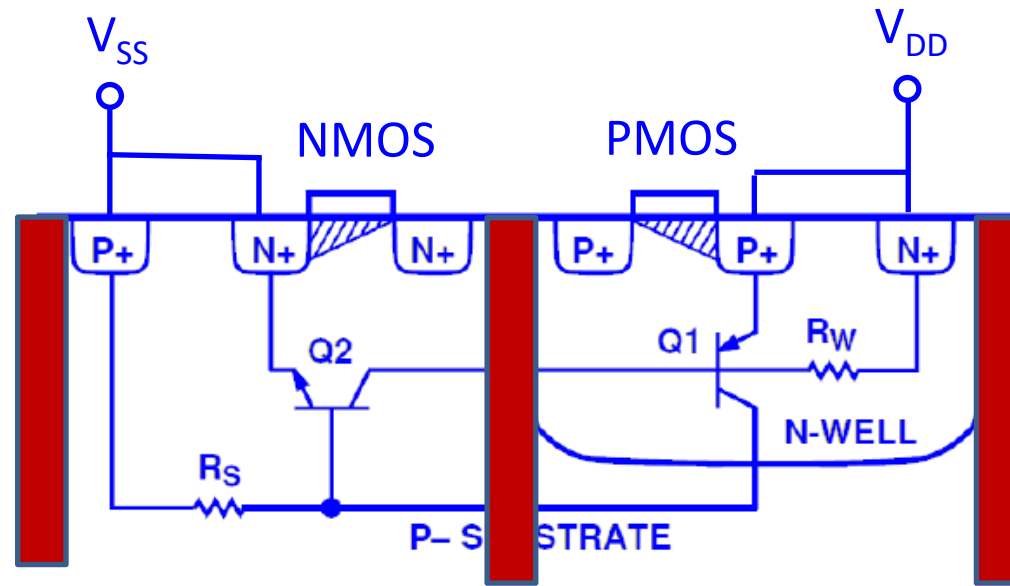


P. E. Dodd, M. R. Shaneyfelt, K. M. Horn, D. S. Walsh, G. L. Hash, T. A. Hill, B. L. Draper, J. R. Schwank, F. W. Sexton, and P. S. Winokur, "SEU-Sensitive Volumes in Bulk and SOI SRAMs From First-Principles Calculations and Experiments", *IEEE TNS*, vol. 48, n. 6, December 2001, pp.1893-1903,

Skip

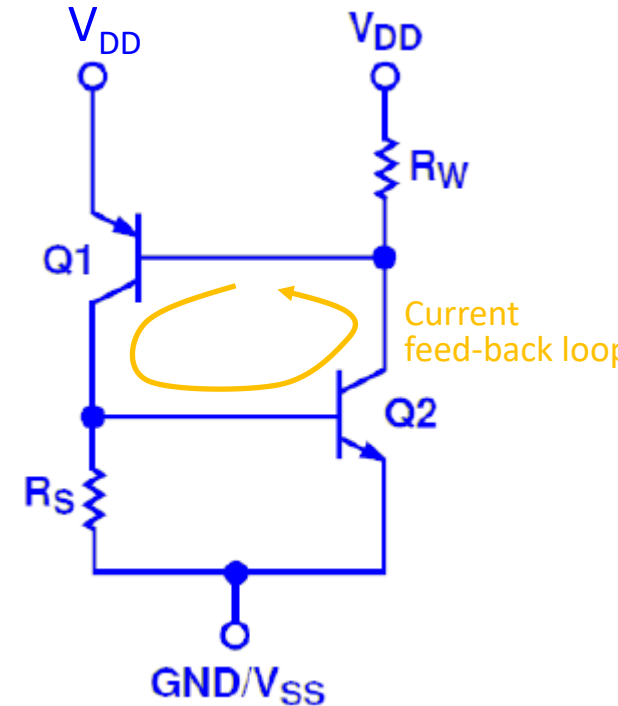
# Single Event Latch-up

- CMOS devices include a parasitic PNP structure
- **Normal operation:** both BJTs do not conduct (Base of Q1 at  $V_{DD}$  and Base of Q2 at  $V_{SS}$ )
- Current loop path between Q1 and Q2 (in BJT  $I_C = \beta I_b$ )
- If  $\beta_1\beta_2 > 1$  **injection of charge** in the loop might trigger a self-amplifying current
- Low-impedance path between  $V_{SS}$  and  $V_{DD}$  may generate a **permanent failure of the component**
- This potentially destructive effect is called single event latch-up (SEL)
- Countermeasures can be taken by reducing  $R_w$  and  $R_s$  or introducing STIs or using SOI technologies



Shallow Trench Isolation

Parasitic pnpn structure

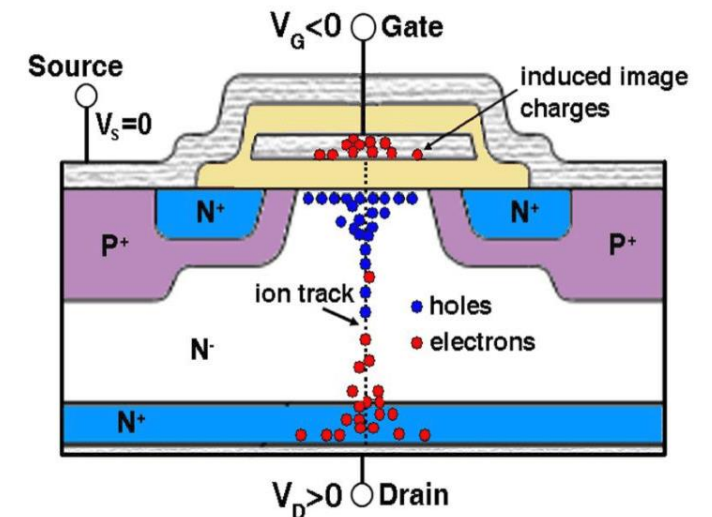


Skip

# Other Single Event Effects

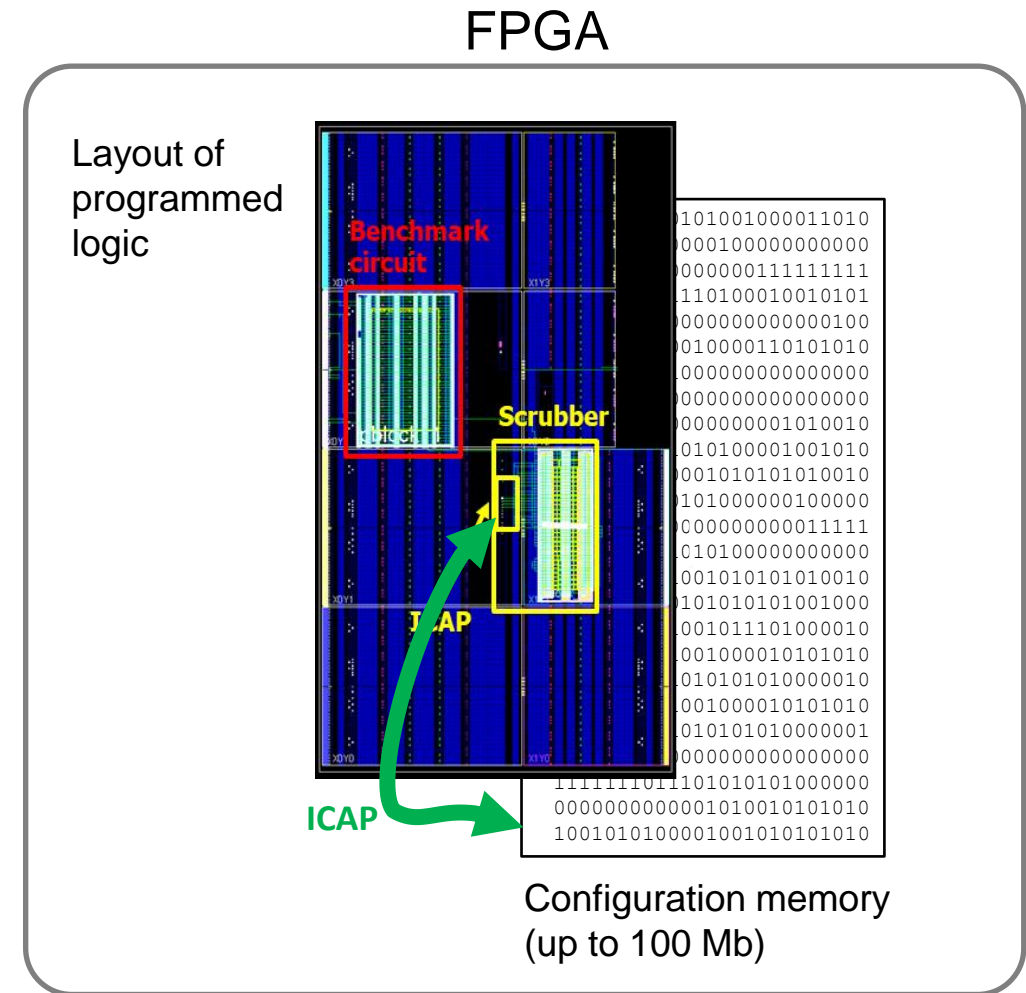
Definition (Acronym)	Description	Affected Devices	Destructive
SEFI - Single Event Functional Interrupt	Loss of normal operation	complex devices with control logic (e.g. FPGAs, microprocessors)	No
SHE - Single Event Hard Error	Permanent failure of a memory element	memories, FFs, latches	Yes
SEB - Single Event Burnout	Destructive Burnout	BJTs	Yes
SEGR – Single Event Gate Rupture	Rupture of gate dielectric	Power Mosfets	Yes

SEGR



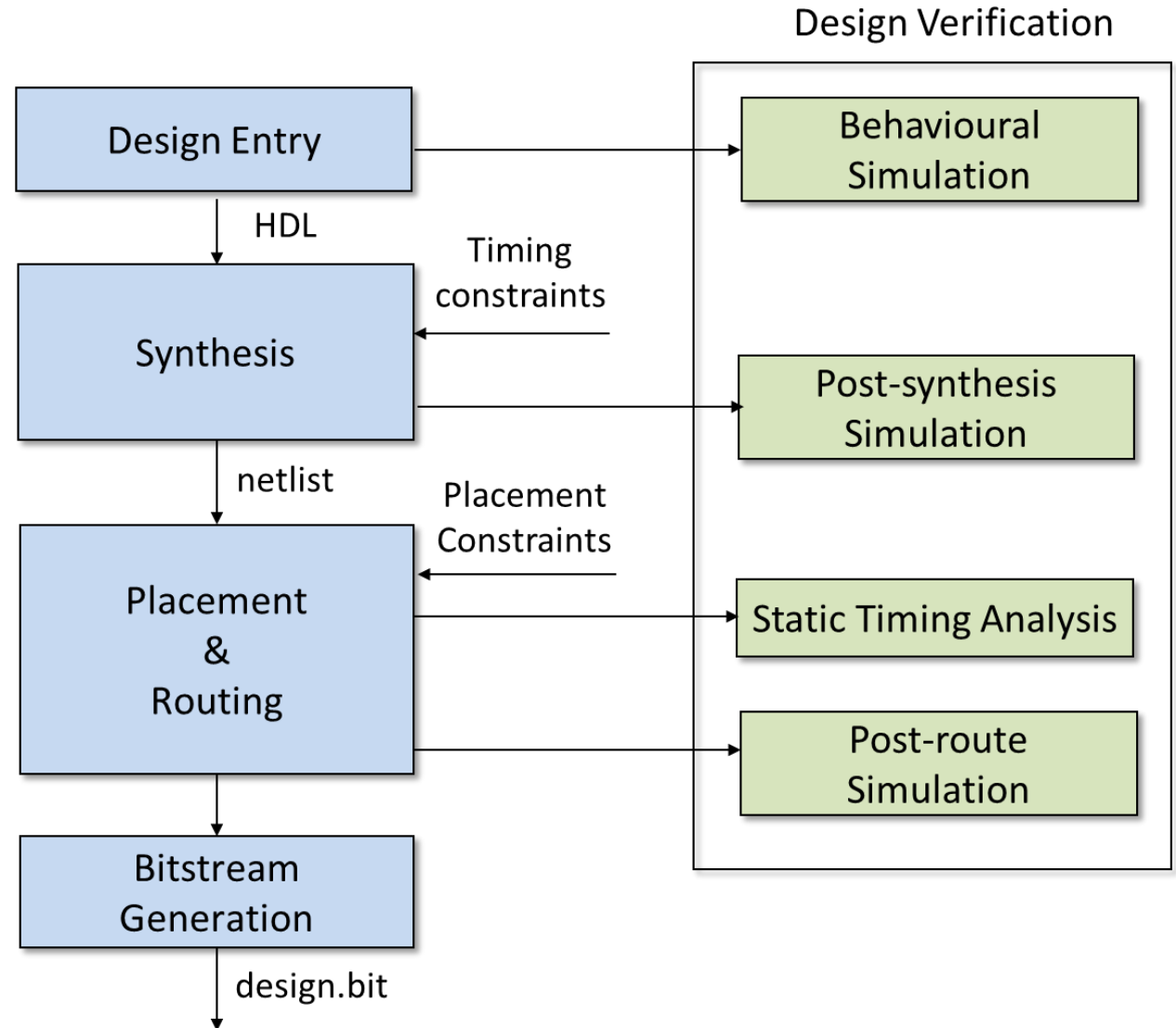
# Configuration-Redundant Self-scrubbing Circuits

- An internal scrubber can access configuration memory by means of internal configuration access port (ICAP in Xilinx devices)
- The scrubber is a critical module
  - If working improperly it can **damage** the circuit instead of fixing it
  - Its MTTF must be sufficiently high in the given radiation environment or its benefits might be marginal



# Design Flow for FPGAs

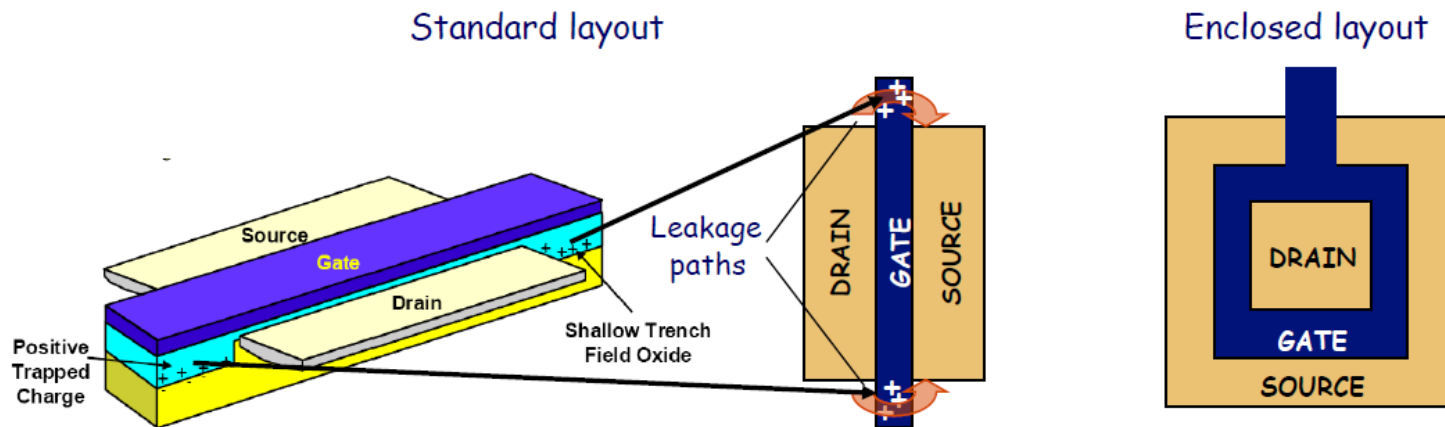
- Description of logic with hardware description languages and requirements with constraints
- Synthesis of logic to device primitives (CLBs, BRAMs, etc.)
- Placement and routing
- Verification of functionality at several intermediated steps
- Static timing analysis
- Generation of the bitstream





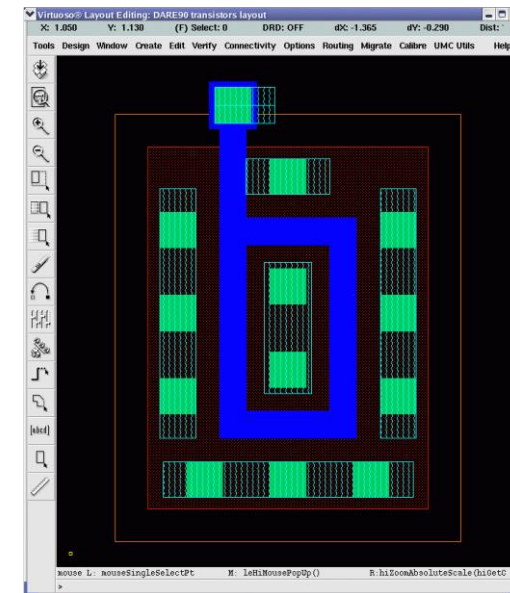
# Layout Hardening

- As mentioned in NMOS devices, positive charge trapped in shallow trench isolation oxides activates parasitic paths at the side of the channel
- Enclosed layout does not have STIs around the channel
- Typical area overhead >2-4x, power overhead > 2x
- Example
  - DARE90U library from IMEC for UMC 90nm CMOS supports up to 3 kGy TID
  - <https://dare.imec-int.com/technologies>



After L. Ratti, Total Dose Effects in Electronic Devices and Circuits – Legnaro, April 13<sup>th</sup> 2011

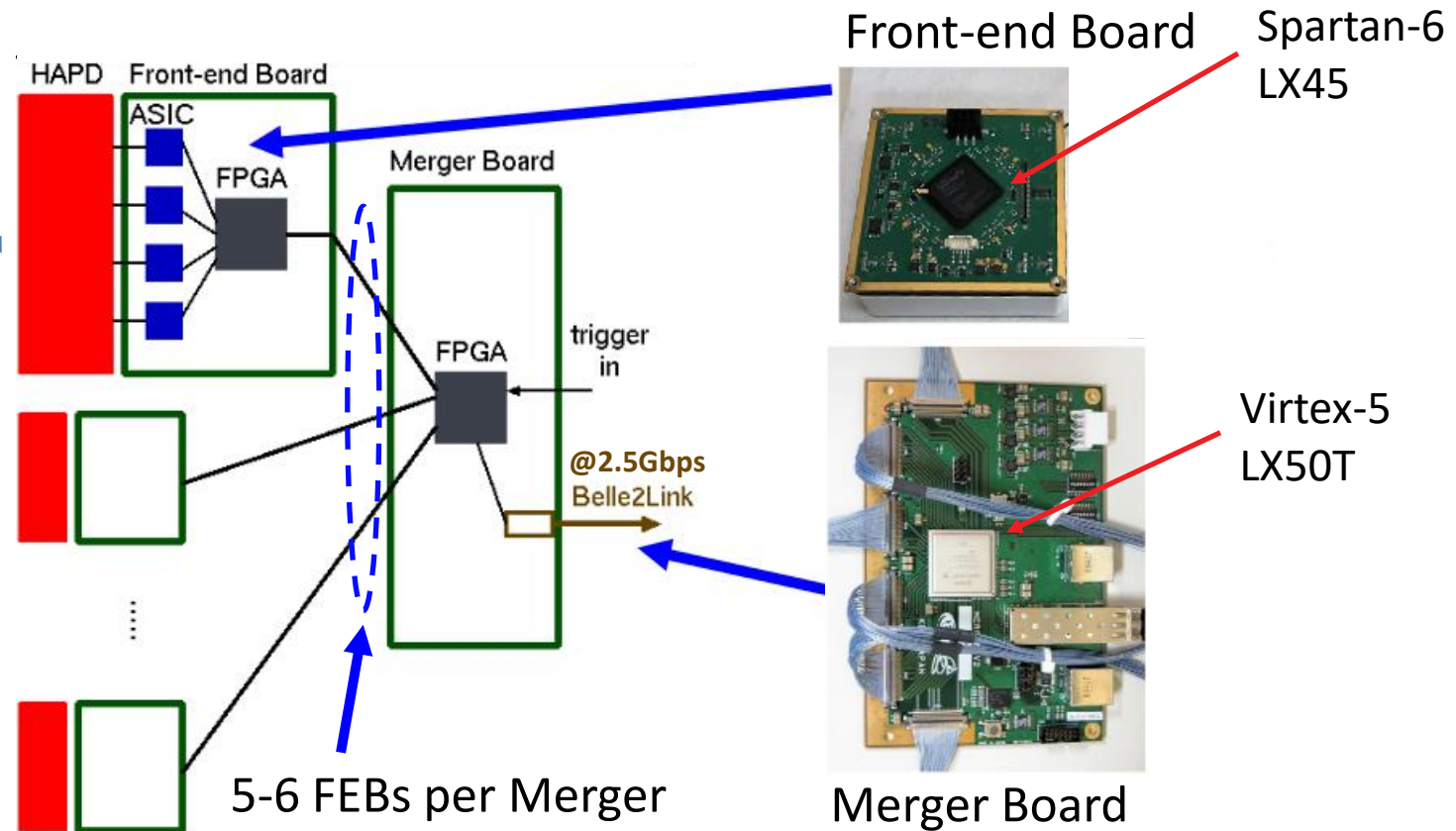
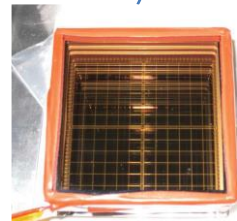
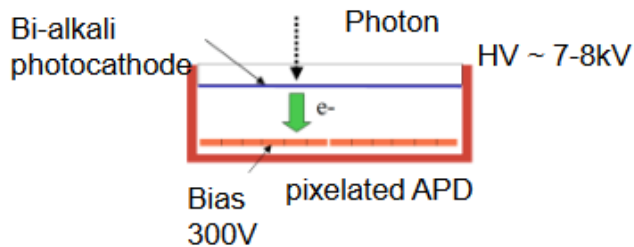
DARE90 Enclosed Layout Transistor  
After G. Thys  
Microelectronics Presentation Days 2010





# Redundant Configuration at System Level

- Case study: Aerogel Imaging Ring Cherenkov of Belle II detector
  - 420 Aerogel tiles read by Hybrid APDs (HAPDs), each HAPD read by one Spartan-6 FPGA
  - Spartan-6 45nm CMOS process Boron for P doping, sensitivity to thermal neutrons (25 meV)
  - 3.3kSEU/h expected in the FEB FPGAs @collider design luminosity



- Merger boards aggregate data from FEBs to on a high-speed link and manage FPGA configuration via JTAG