Single Event Effect Mitigation in Digital Integrated Circuits for Space

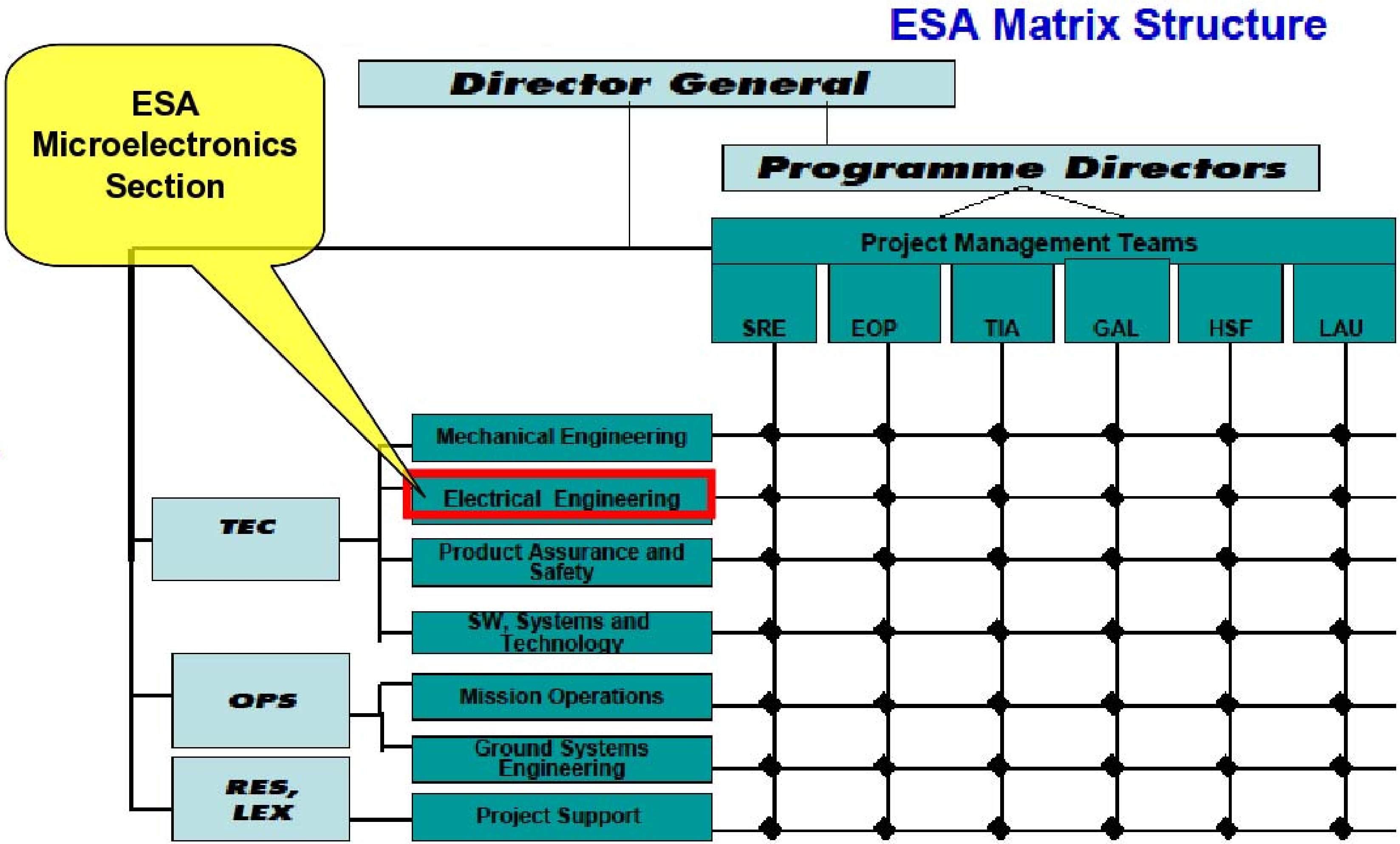
Topical Workshop on Electronics for Particle Physics
21. September 2010
Aachen

Roland Weigand
European Space Agency
Data Systems Division TEC-EDM
Microelectronics Section
Tel. +31-71-565-3298
Fax. +31-71-565-6791
Roland.Weigand[at]esa.int

Summary

- ***The Microelectronics Section at ESA - who we are***
- ***Radiation sources in space environment***
- ***SEU radiation hardening approach***
- ***SEE (SEU/SET) hardening of commercial bulk CMOS***
  - Hardened standard cell library cells
  - Triple Modular Redundancy with clock skew (TMR, STMR)
  - Implications of TMR in the design flow
- ***SEE protection of memory blocks***
  - Memory cell design
  - Error Correcting Codes, parity, TMR, scrubbing
- ***SEE in reprogrammable (SRAM) FPGA***
  - Triple Modular Redundancy(combinatorial and sequential logic)
  - Dedicated rad-hard FPGA design
- ***Validation of SEE hardening***
  - Simulation, emulation, structural/formal verification
  - Ground radiation testing

Who we are... (2)

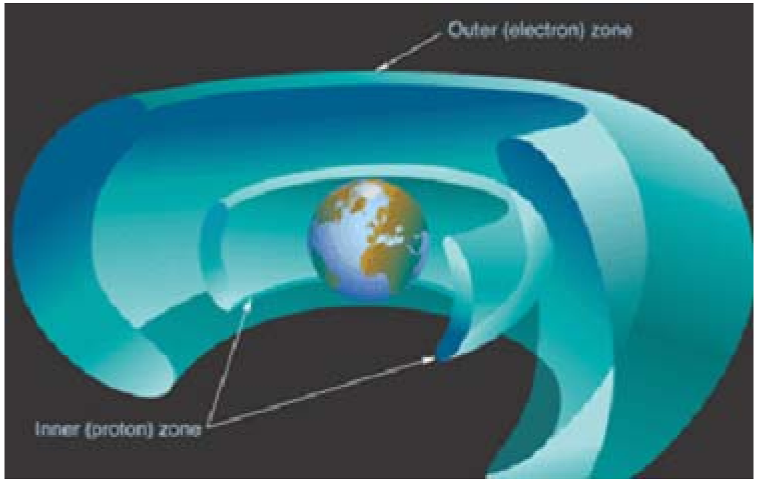

ESA Matrix Structure
ESA Microelectronics Section
Director General
Programme Directors
Project Management Teams
SRE
EOP
TIA
GAL
HSF
LAU
TEC
OPS
RES, LEX
Mechanical Engineering
Electrical Engineering
Product Assurance and Safety
SW, Systems and Technology
Mission Operations
Ground Systems Engineering
Project Support

Who we are... (2)

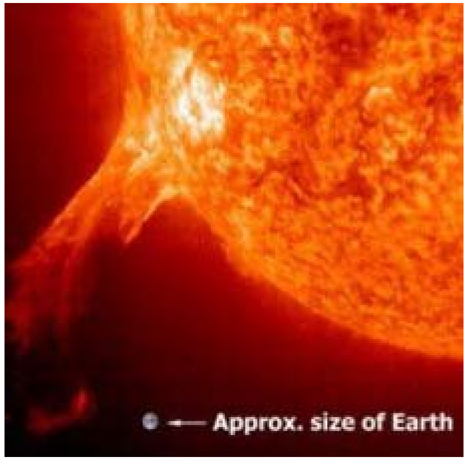
- ***The Technical and Quality Management Directorate (TEC)***
  - http://www.esa.int/SPECIALS/Space_Engineering/SEMB9FVG3HF_0.html
  - Inside TEC, 3 sections work on radiation effects:
- ***The Space Environments and Effects Section (TEC-EES)***
  - Analysis of space environments and their effects on space systems
  - http://space-env.esa.int/index.php/ESA-ESTEC-Space-Environment-TEC-EES.html
- ***The Radiation Effects and Analysis Techniques Section (TEC-QEC)***
  - Analysis at component level and radiation testing
  - https://escies.org/ReadArticle?docId=227
- ***<u>The Microelectronics Section (TEC-EDM)</u>***
  - Availability of appropriate technologies and development methods
  - Availability of space-specific standard components and IP
  - Development support to projects
  - Analysis and mitigation of SEE at design level
  - http://www.esa.int/TEC/Microelectronics/

Radiation Sources in Space

- ***The space radiation environment is <u>dynamic</u> and <u>inhomogenous</u>***
  - Dependency on satellite trajectory/orbit
  - Dependency on mission schedule
- ***Trapped radiation belts (Van Allen belts)***
  - e- and p+ trapped in the earth magnetic field
  - Inhomogenous: e.g. **South Atlantic Anomaly**



- ***Solar Particles***
  - Solar activity cycle 11 years
  - High flux for several days during solar flares
  - Protons and heavy ions with a highly variable energy spectrum
  - Shielding by earth magnetic field



- ***Galactic Cosmic Rays***
  - Anticorrelated with solar activity (high flux during solar low)
  - Particles from protons to heavy ions
  - High energy, up to 10^{20} eV
  - Flux ~ 4 particles / cm^2 / sec

Radiation Hardening

- ***Dedicated processes for space are not affordable any more***
- ***SOI is sometimes used***
  - Low SEU rates, latch-up free, some concerns on TID
  - SOI is less readily available, analog IP need to be re-developed
- ***Total Ionising Dose (TID)***
  - Most space missions are limited to 100 krad dose, and in 180 nm or below, TID protection might be limited to e.g. screening of (commercial) library cells, elimination of certain transistor types
  - Some long duration, deep space missions are in the Mrad domain, requiring mitigation e.g. by special transistor geometries (ELT), guard rings or derating
- ***Single Event Latch-Up (SEL)***
  - Horizontal: mitigation in layout, e.g. guard rings
  - Vertical: thickness of the epitaxial layer, deep n-well
- ***Single Event Effects (SEE) by Transient and Upset (SET, SEU)***
  - Spatial or temporal redundancy
  - Mitigation by design of library cells or in logic design → see below

Single Event Transients (SET)

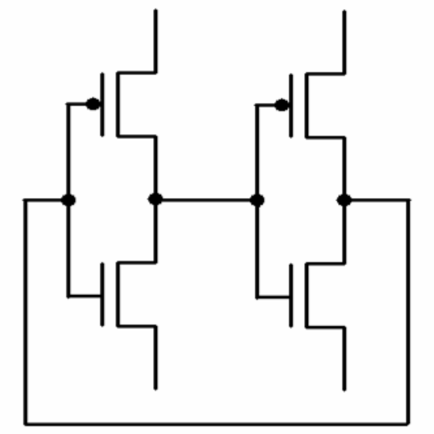
- ***Collision induced carrier generation in PN junctions***
  - Propagate as glitches in combinatorial logic
  - Latched into storage cells when arriving at data input during clock edge
  - ➔ <u>Upset rate increases with the clock frequency</u>
  - Seen already in ERC32 processor (0.5 μm technology)
  - … definitely a concern in 0.18 μm and below
- ***Analysis of SET effects in simulation and radiation tests***
  - SET pulse length and amplitude are most important parameters
  - Specific test structures to catch and characterise the pulse
  - CNES contract with Atmel on SET effects in the 0.18 μm technology
- ***Mitigation of SET effects***
  - Propagation of complementary logic levels ("Dual Stream")
  - Using stronger drivers and higher capacitive loads
  - Delay filtering on all flip-flop inputs (clock, data, reset)
  - STMR: Triple skewed clocks in conjunction with the TMR flip-flop
    - » Triplication of clock-like nets (including asynchronous resets)
    - » see below …

SEU hardening approach

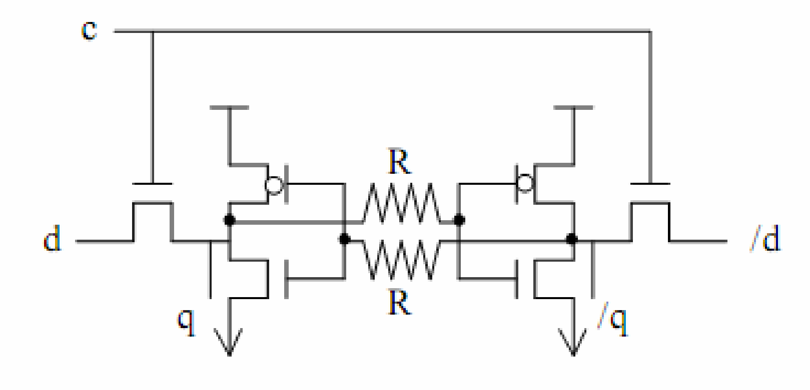
1. ***Determine mission requirements***
   - Fix reliability goal (FIT, # faults tolerated per time unit)
   - Determine radiation profile (orbit, solar cycles)
   - Shielding in the mechanical structure
   - For standard components: take worst possible requirements
2. ***Characterise target silicon technology***
   - Simulation and ground radiation testing in accelerators
   - LET threshold and saturated cross-section
3. ***Calculate error rate per bit (flip-flop, memory) and per chip***
   - CRÈME models for space SEE rates: https://creme-mc.isde.vanderbilt.edu/
   - Bit error rate to be multiplied with the number of bits in a chip

- ***The reality … is sometimes different***
  - Requirements are unclear, radiation analysis is dropped or incomplete
  - Uncertainty leads to overprotection, causing huge design overhead
  - Projects may hit the ceiling of feasibility or affordability
  - Several silicon iterations and radiation validation required

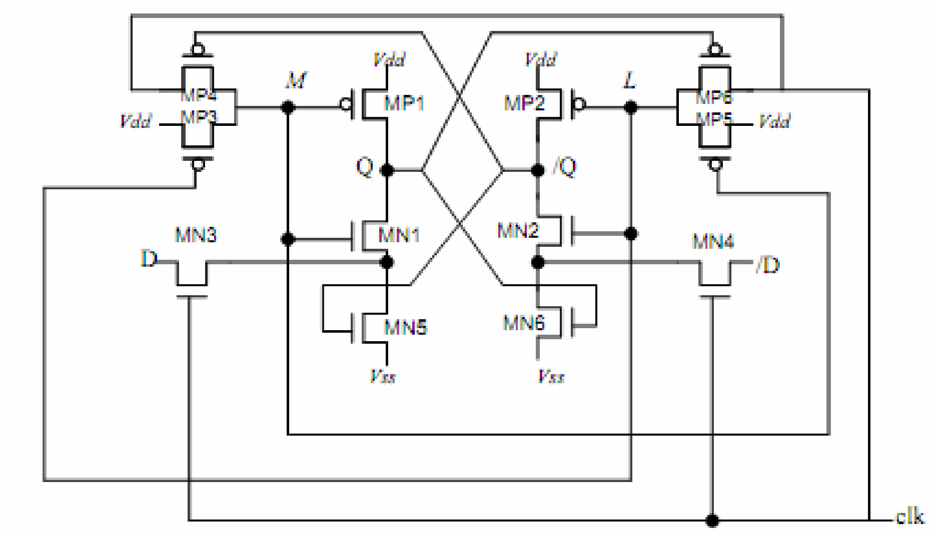
SEU hardening of flip-flops and SRAM



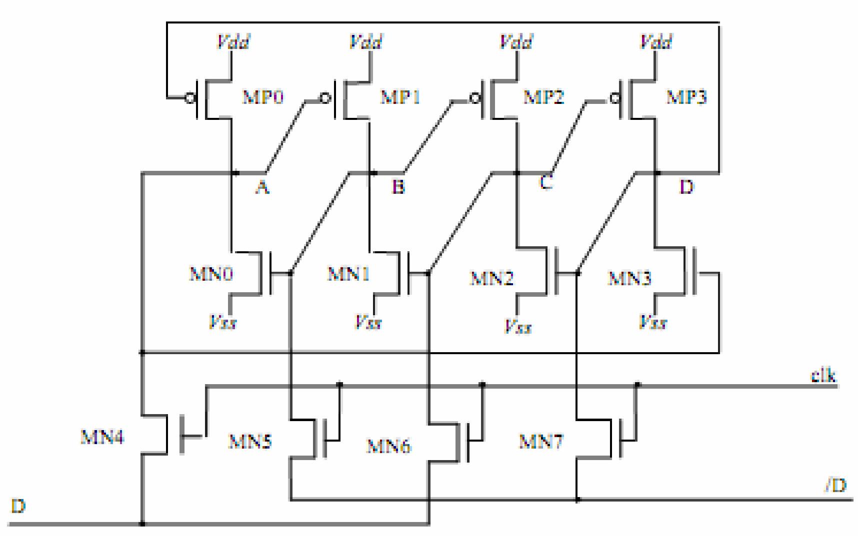
standard latch



Resistor memory cell



HIT memory cell



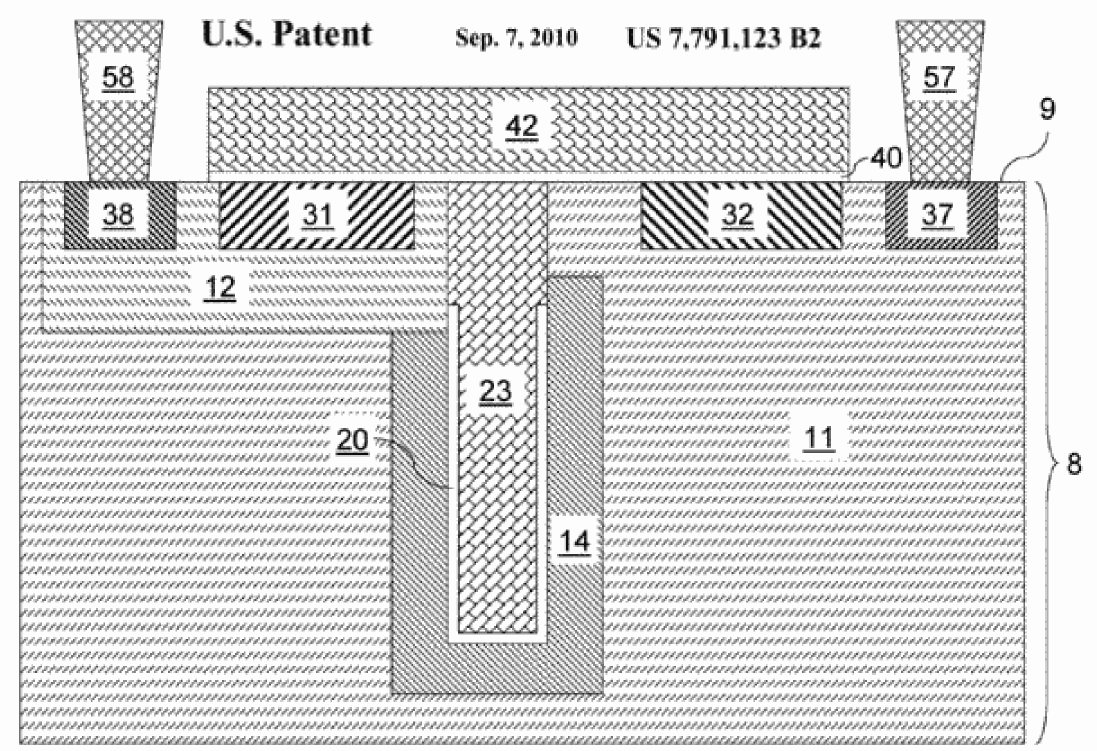
DICE hardened memory cell

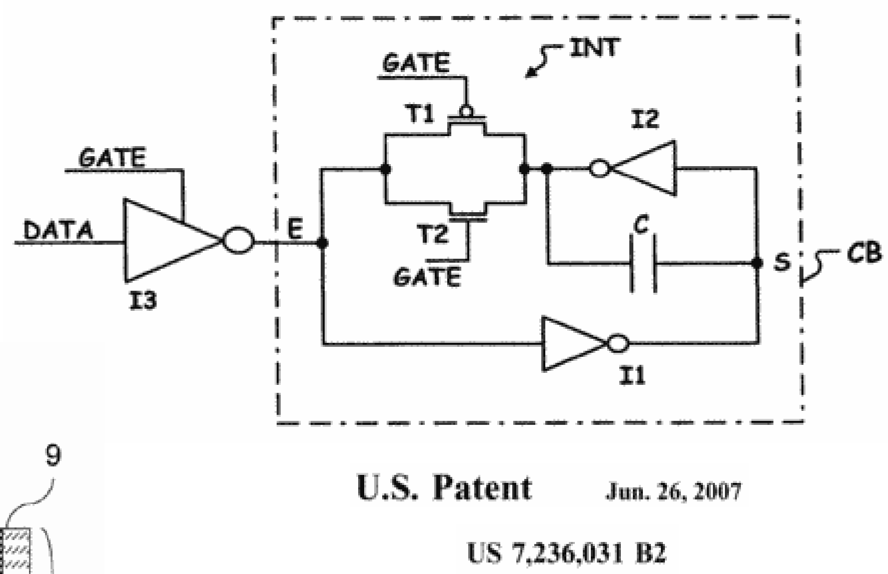
Radiation hardened standard cell libraries

- ***Resistor Memory Cell***
  - H. T. Weaver, C. L. Axness, J. D. McBrayer, J. S. Browning, J. S. Fu, A. Ochoa, R. Koga, "An SEU Tolerant Memory Cell Derived from Fundamental Studies of SEU Mechanisms in SRAM," Nuclear Science, IEEE Transactions on , vol. 34, no. 6, pp. 1281-1286, Dec. 1987
- ***HIT = Heavy Ion Tolerant storage cell***
  - D. Bessot R. Velazco, "Design of SEU-hardened CMOS memory cells: the HIT cell" RADECS, 1993
- ***DICE = Dual Interlocked storage CEll***
  - R. Velazco, D. Bessot, S. Duzellier, R. Ecoffet, R. Koga, "Two CMOS memory cells suitable for the design of SEU-tolerant VLSI circuits," Nuclear Science, IEEE Transactions on , vol. 41, no. 6, pp. 2229-2234, Dec. 1994.
- ***Examples of hardened libraries around the world***
  - **ATMEL MH1RT (350 nm) and ATC18RHA (180 nm) technologies –** http://www.atmel.com
  - **DARE (Design Against Radiation Effects) library for UMC 180 nm and 90 nm (development)**
    - http://microelectronics.esa.int/mpd2010/day1/MPD-IMEC-DARE-30March2010.pdf
  - **ST Microelectronics library for 65 nm under development**
    - http://microelectronics.esa.int/mpd2010/day2/DSM65nm.pdf
  - **Ramon Chips library for 180 nm Tower Semiconductors (130 nm under development)**
    - http://nepp.nasa.gov/mapld_2008/presentations/i/05%20-%20Ginosar_Ran_mapld08_pres_1.pdf
  - **Aeroflex (600, 250, 130, 90 nm) –** http://www.aeroflex.com/RadHardASIC
  - **MRC Microelectronics on TSMC (0.35/0.25), UTMC/AMI, HP, NSC, Peregrine**
    - http://parts.jpl.nasa.gov/mrqw/mrqw_presentations/S4_alexander.ppt
  - **HIREC/JAXXA - Fujitsu 0.18, OKI 0.15 SOI (NSREC2005)**

SEU/SET

- ***Trench capacitors***
  - embedded DRAM cells can be used to minimise the area penalty
  - IBM patent

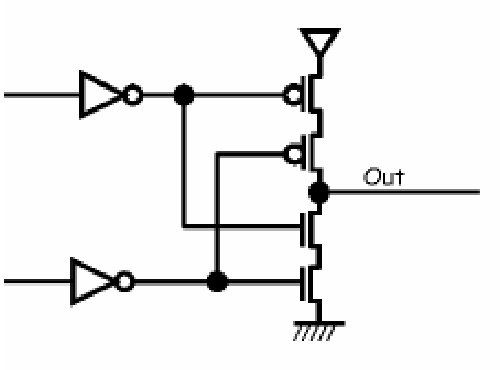


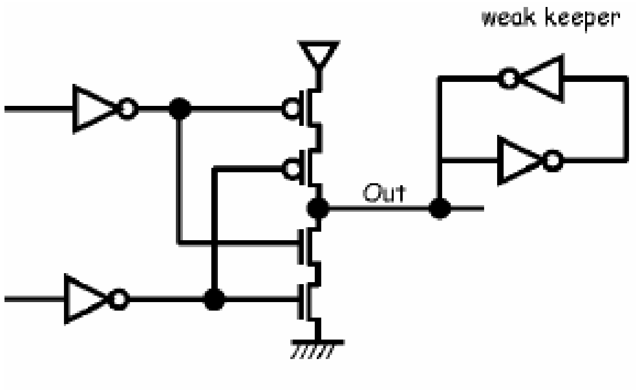


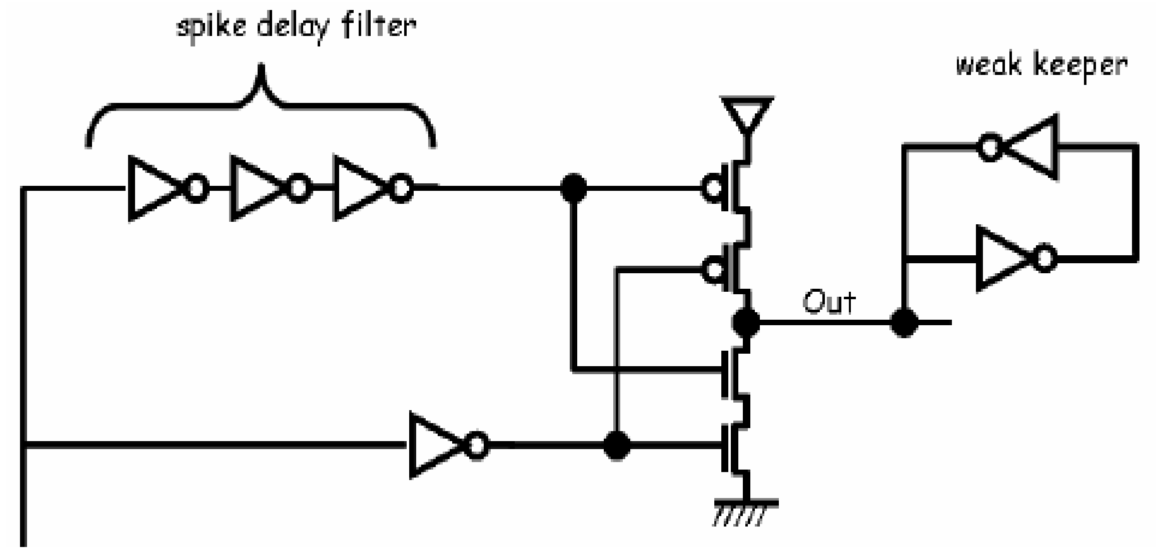
- ***Transmission gates***
  - feedback path is cut off during write cycles to reduce the speed penalty
  - ST patent

Glitch filtering of clock/reset trees

- ***C-element as glitch filter***
  - Enhanced with weak keepers on the output node to prevent floating state
  - Used to recombine a spatial redundant dual logic cone
  - Single logic cone with a spike delay filter (Mongolkachit, RADECS 2003)

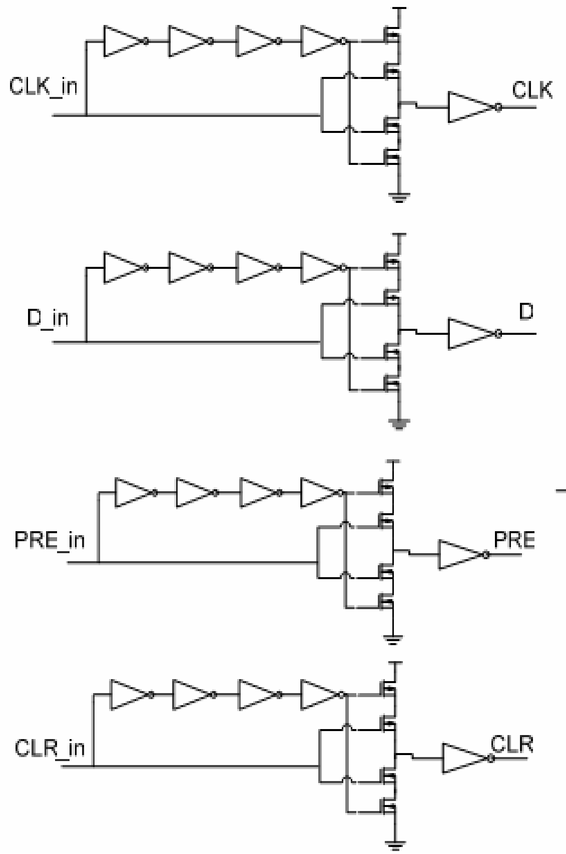


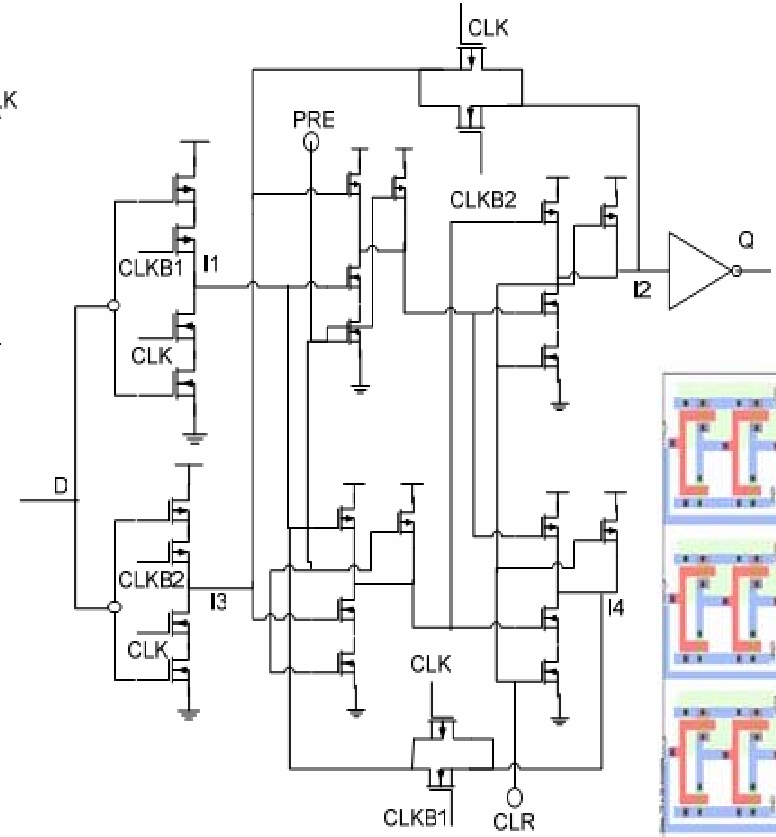


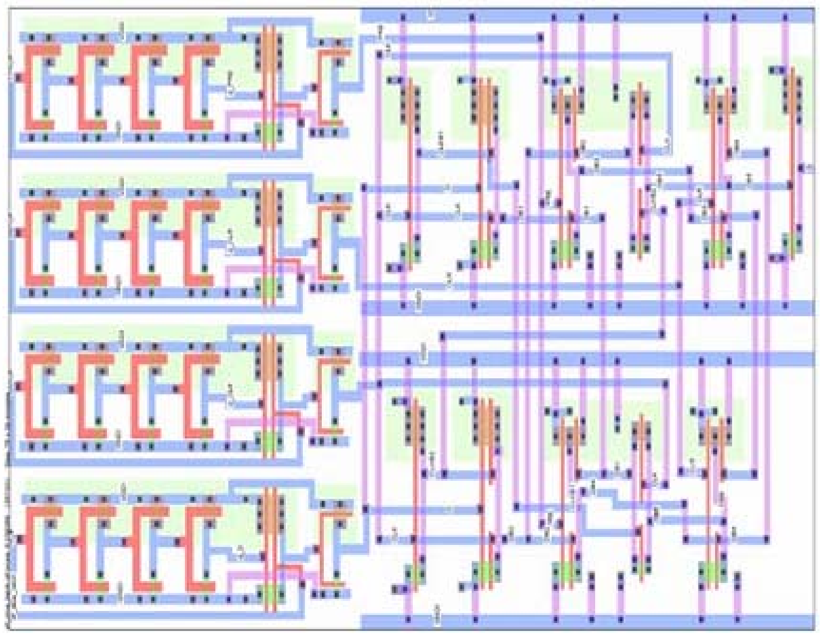


DF-DICE, the SEU and SET hardened FF

- ***http://www.isi.edu/~draper/papers/mwscas05_naseer.pdf***

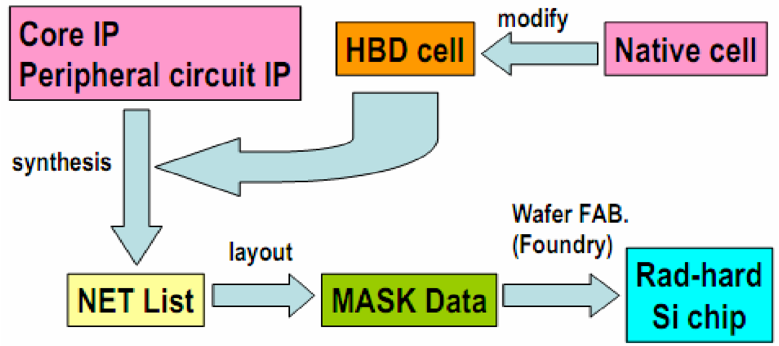




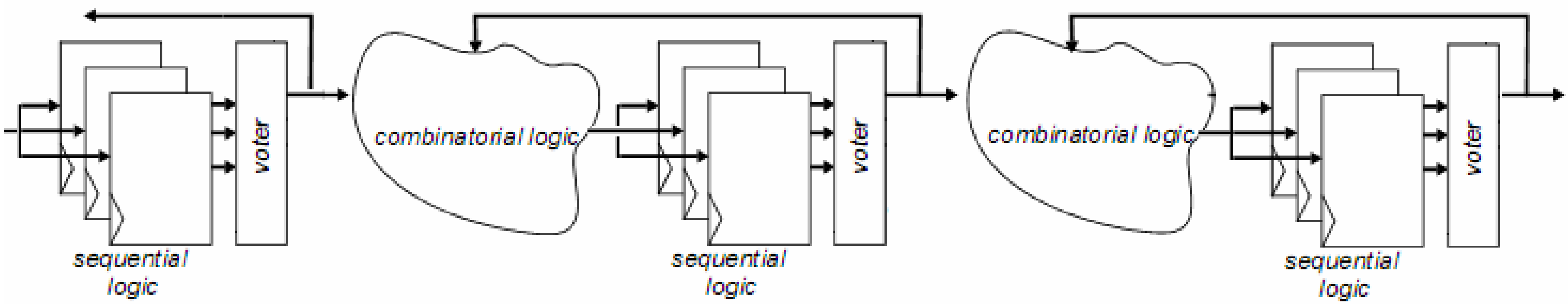


SEU protection by TMR Flip-Flop with voter

- ***Hardened libraries are used at logic synthesis, like native commercial cell libraries***
  - Speed and area (x2) penalty
- ***If no hardened library available …***

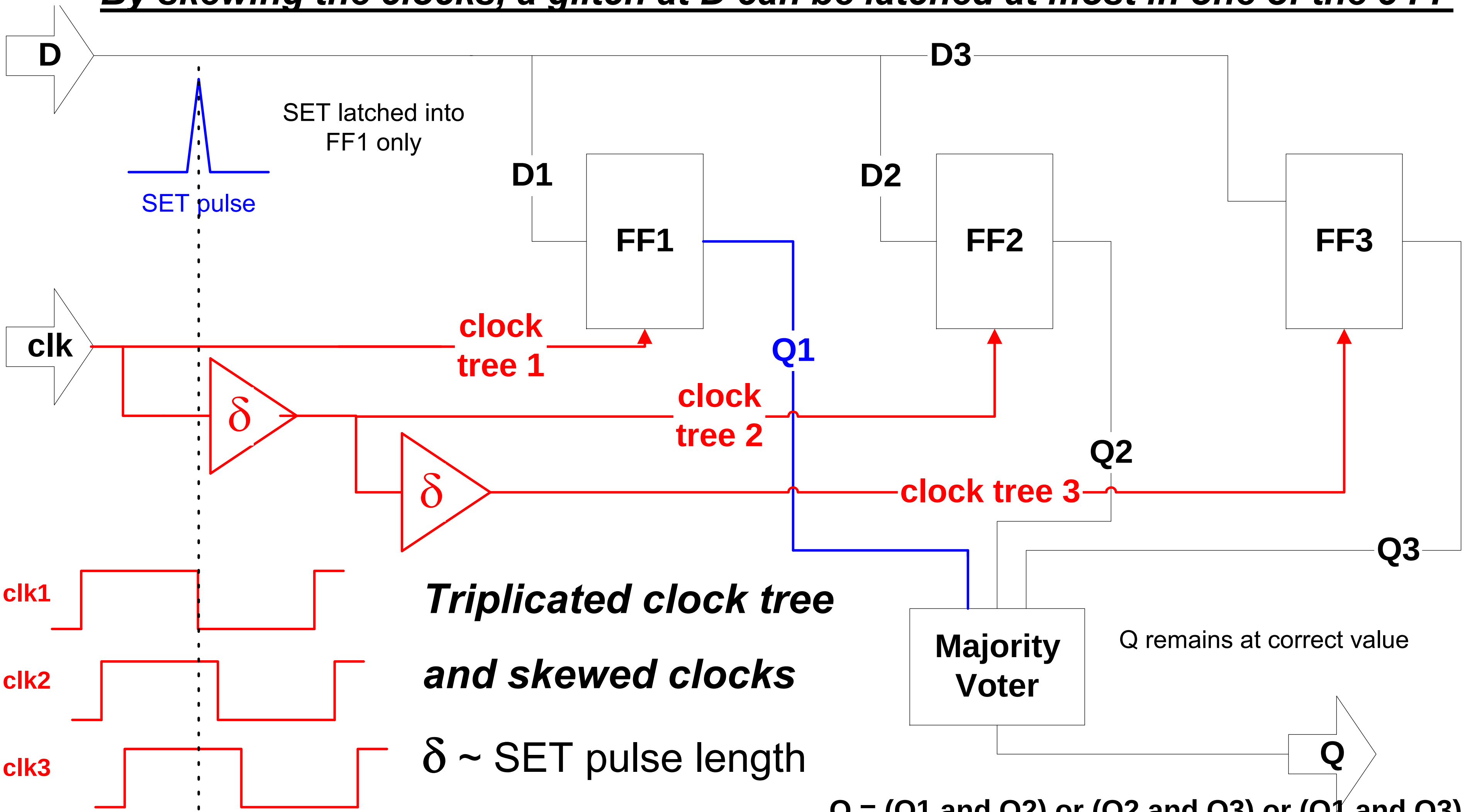


- ***Triple Modular Redundancy (TMR) flip-flops***
  - Using standard flip-flops of the commercial library
  - Data input is fed to three flip-flops at the same time,
    Outputs of the flip-flops are majority voted (combinatorial half-adder)
  - Area overhead on flip-flops is a factor of > 3, but little in combinatorial logic
  - Implemented in the RTL source code, by netlist editing or by synthesis tool



STMR: TMR with triple skewed clock

By skewing the clocks, a glitch at D can be latched at most in one of the 3 FF



Triplicated clock tree

and skewed clocks

δ ~ SET pulse length

Q = (Q1 and Q2) or (Q2 and Q3) or (Q1 and Q3)

STMR in the ASIC design flow

- ***TMR: Increased complexity affects the design flow and –results***
  - Large netlist with higher cell and node count
  - Increased run-time or even crashes of EDA tools
  - Design optimisation is less efficient
- ***Synthesis tools are designed to remove <u>redundancy</u>***
  - Normally, registers are not modified – but be careful …
  - … with sequential optimisation (pipelining, retiming etc.)
- ***Timing issues***
  - TMR voting and clock skewing reduces maximum speed
  - Increased area leads to higher interconnect delay
  - Clock skewing can be removed by hold-time fix
- ***Verification and test issues***
  - TMR and formal verification (1 FF in RTL → 3 FF at gate level)
  - TMR (= redundancy) affects testability in scan testing
  - Implementation of protection has to be verified at netlist level

STMR insertion at RTL or gate level

STMR in VHDL

- Clock nets/ports are a vector of 3 bit
- Use the “two-process” method [6]

```
-- One process per TMR domain:
rx0 : process(clk) begin
    if rising_edge(clk(0)) then r0 <= d;
end if; end process;
rx1 : process(clk) begin
    if rising_edge(clk(1)) then r1 <= d;
end if; end process;
rx2 : process(clk) begin
    if rising_edge(clk(2)) then r2 <= d;
end if; end process;
-- Vote outputs
r <= (r0 and r1) or (r0 and r2) or (r1 and r2);
```

- Synthesis with TMR in one go
- Disallow register merging
- Structural verification required

STMR at gate level

- Used mainly for third party IP
- Library and tool dependent
- Synthesise netlist without TMR
- Create HDL package with TMR equivalent macro-cells
- Edit netlist to triplicate clocks and asynchronous resets

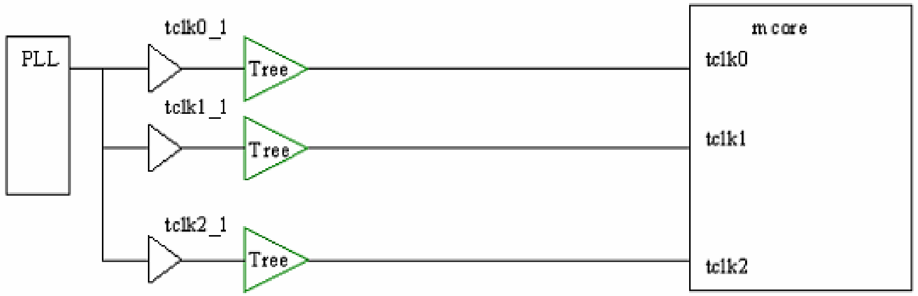
```
sed -e 's/CLK\(.*\) std_logic/CLK\1 std_logic_vector(2 downto 0) /'
```

- Edit netlist replacing every flip-flop by its TMR equivalent

```
sed -e 's/DFF1/DFF1_TMR/'
sed -e 's/DFF2/DFF2_TMR/'
```

- Resynthesise the edited netlist, linking with the TMR macro-cell package
- Disallow register merging
- Structural verification required

Inserting triple skewed clock/reset trees



- **Clock Tree Synthesis (CTS) optimises skew inside a single clock tree → but we need three coherent trees (not supported by CTS tools)**
  - Need to control the insertion delay (X, X+δ, X+2δ)
  - Compromise: insert three distinct trees with well adjusted CTS parameters
- **Delay δ inserted at the origin of the clock trees**
  - Instantiate delay buffers in the VHDL source code for simulation
  - Model δ at synthesis by *set_ideal_latency* and *set_propagated_clock*
  - Initial value for δ is speculative → control/adjustment in backend process
- **Triplicate also asynchronous reset trees**
  - Triplicate any logic in clock and asynchronous reset networks

Coherent clock trees

We need to control the relative clock latency:

X

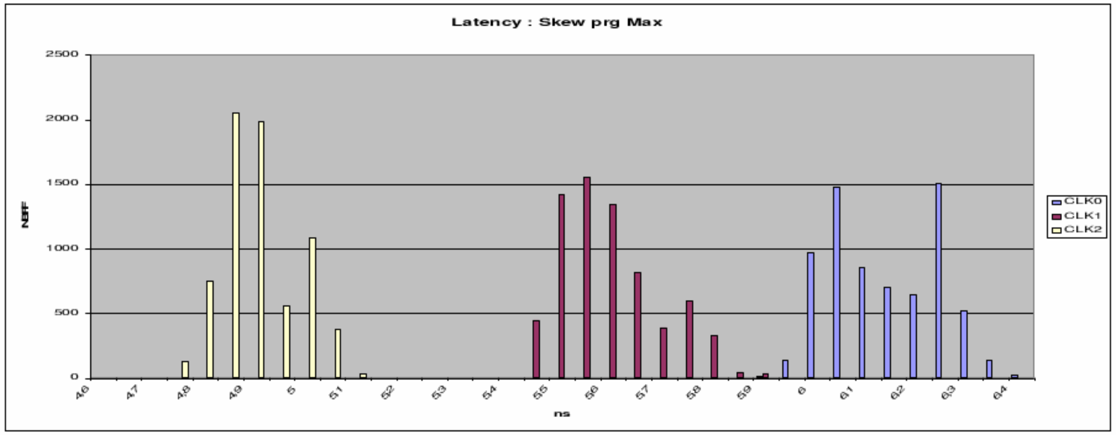
X+δ

X+2*δ

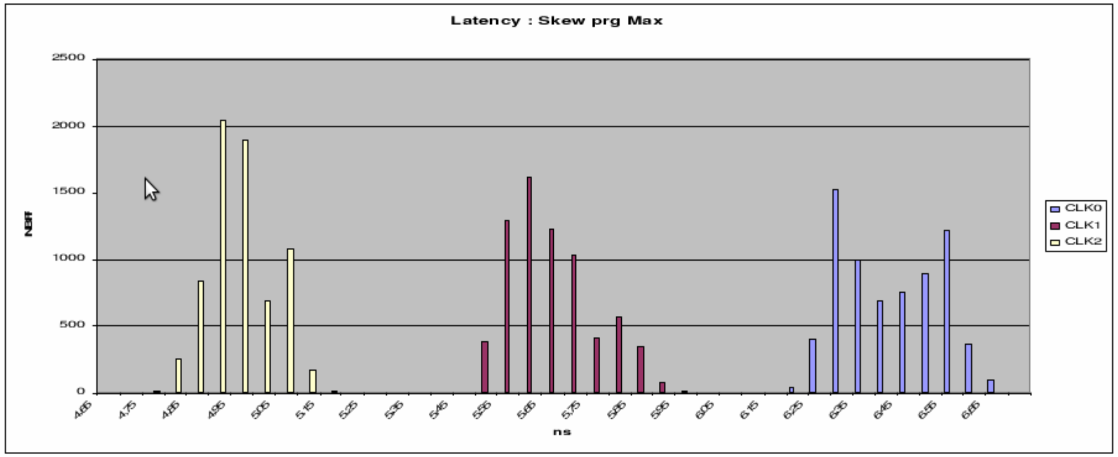
CTS did not achieve goal

→

Manual adjustment of delay elements required

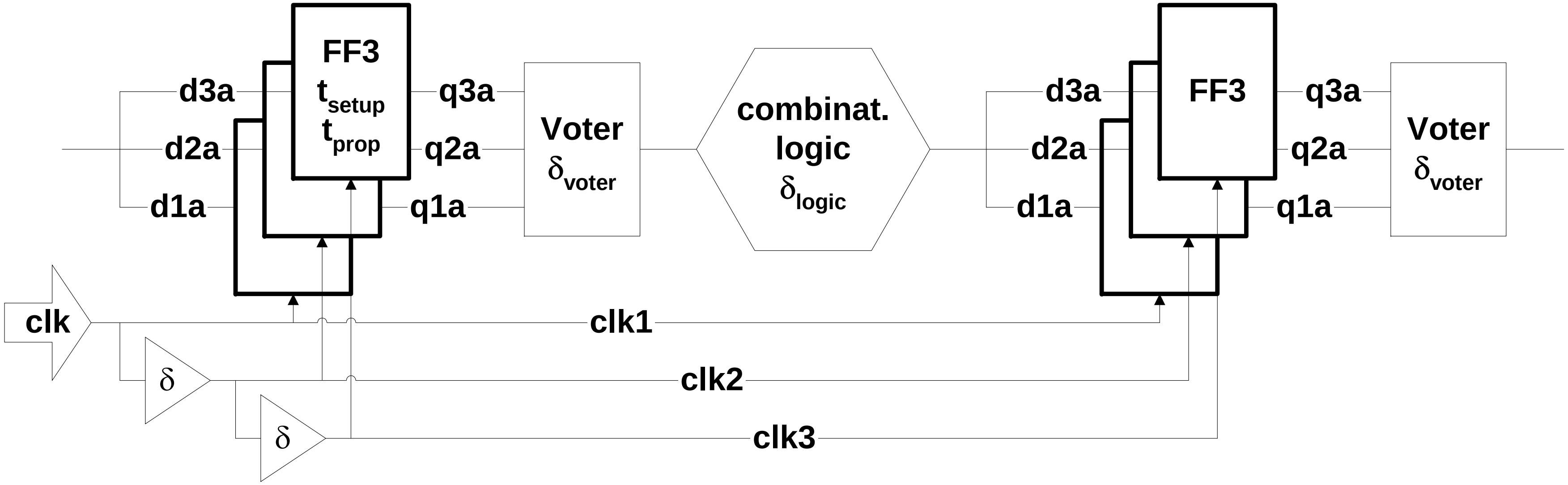


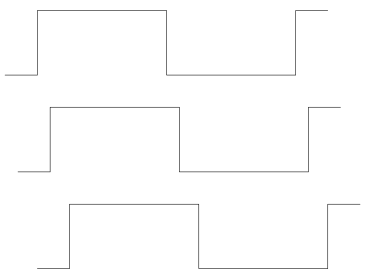
Before manual adjustment



After manual adjustment

TMR Timing Issues





Cycle Time T >= $t_{prop} + \delta_{logic} + t_{setup} + \boxed{\delta_{voter} + 2\delta}$

TMR voters and clock skewing reduce operating frequency

Area and power overheads of hardened FF

- Voted TMR cells
  - Area overhead >~ factor 3
  - Power consumption ~ factor 3
- SEU hardened flip-flops
  - Area overhead factor 2 – 2.5
  - Power consumption factor 2 – 3
- Overhead only on flip-flops
  - Total overhead depends on share of combinatorial and sequential logic
  - A = 3x flip-flops + 1x combinatorial

Share of flip-flops	Area overhead
25%	1.5
50%	2
75%	2.5

Synthesis description of the DARE library

State toggle power increases ~ x3

Standard DFF rise_power(li5X5) {
index_1("0.016, 0.064, 0.128, 0.8, 1.07") ;
index_2("0.03, 0.15, 0.75, 1.5, 3") ;
values("0.260154 0.260608 0.259797 0.262227 0.265544",\
"0.258697 0.259304 0.258465 0.262485 0.264274",\
"0.258899 0.259535 0.258754 0.26171 0.264257",\
"0.259817 0.260501 0.259856 0.262175 0.265157",\
"0.259849 0.260833 0.260201 0.262509 0.265653"); }

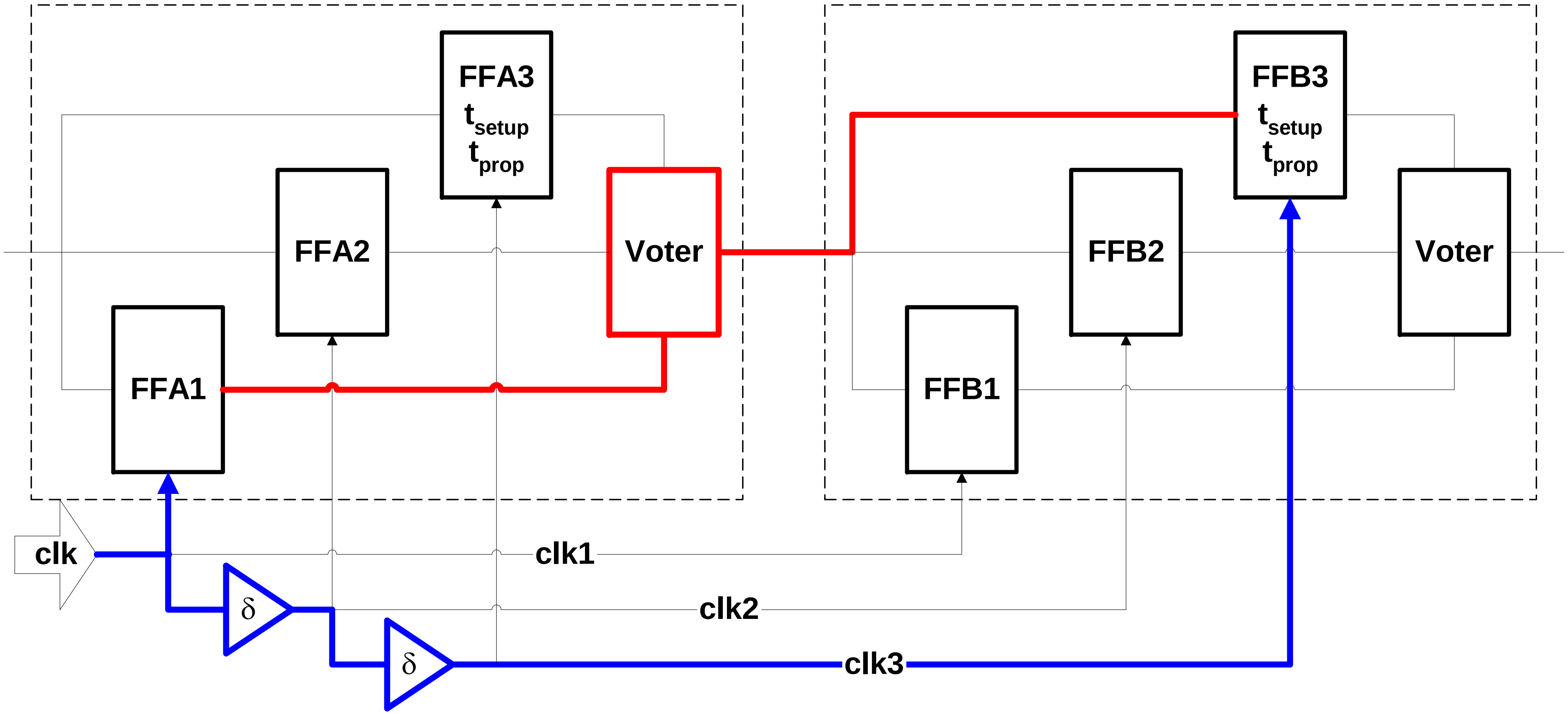
Hardened XDFF rise_power(li5X5) {
index_1("0.016, 0.064, 0.128, 0.8, 1.07") ;
index_2("0.03, 0.15, 0.75, 1.5, 3") ;
values("0.800729 0.800399 0.794199 0.79509 0.799814",\
"0.789216 0.788791 0.7821 0.78533 0.78516",\
"0.781962 0.781545 0.774982 0.777166 0.776802",\
"0.770274 0.769896 0.763804 0.765198 0.769422",\
"0.765816 0.76547 0.759478 0.760922 0.765386"); }

Clock power increases ~ x2

Standard DFF rise_power(i5) {
index_1("0.03, 0.15, 0.75, 1.5, 3") ;
values("0.09928 0.098241 0.111142 0.131959 0.180269"); }

Hardened XDFF rise_power(i5) {
index_1("0.03, 0.15, 0.75, 1.5, 3") ;
values("0.208006 0.207359 0.227548 0.26199 0.344905"); }

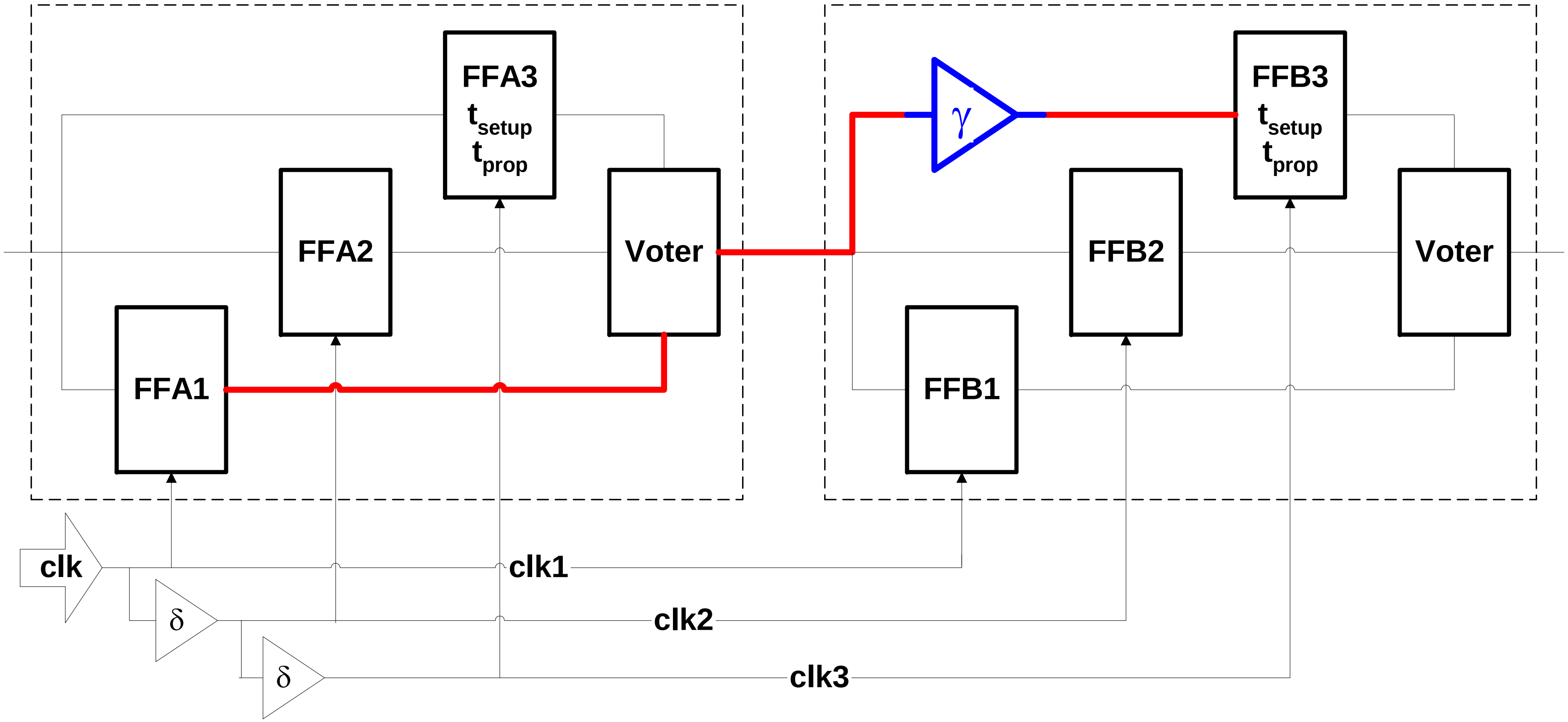
Hold violations with skewed clocks



When propagation delays $(t_{prop}$, voter) < (2 δ) clock skew

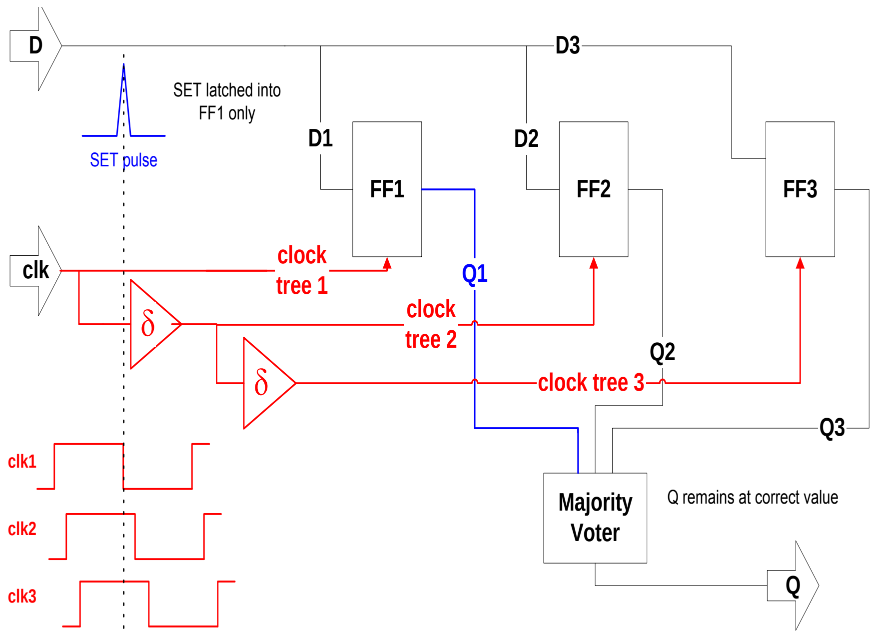
➔ hold violation FFA1 ➔ FFB3

Wrong hold fix by EDA tool



Automatic buffer insertion by fix-hold of synthesis tool compensates clock skew → and spoils SET protection

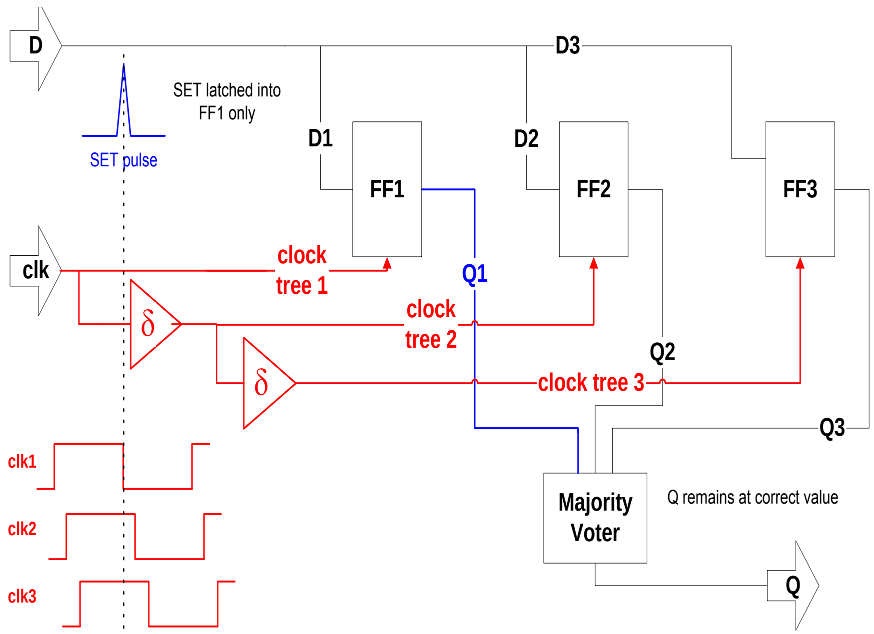
Clock spread dilution by wrong hold fix

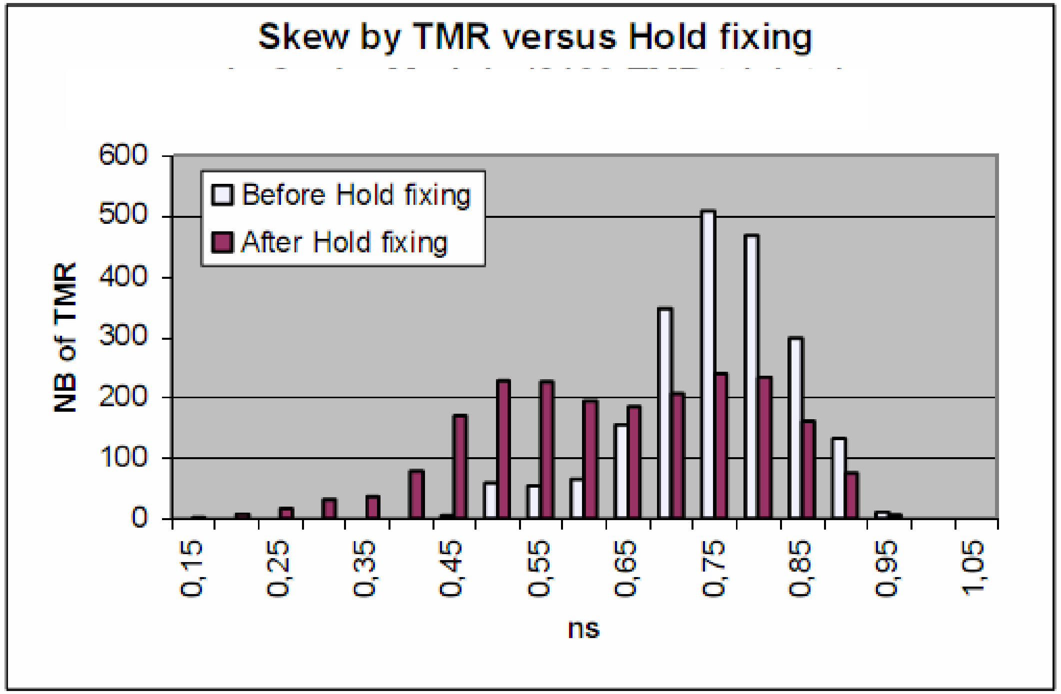


[T(clk2) – T(d2)] – [T(clk1) – T(d1)]

Difference between clock and data arrival in each TMR triplet

Clock spread dilution by wrong hold fix





[T(clk2) – T(d2)] – [T(clk1) – T(d1)]

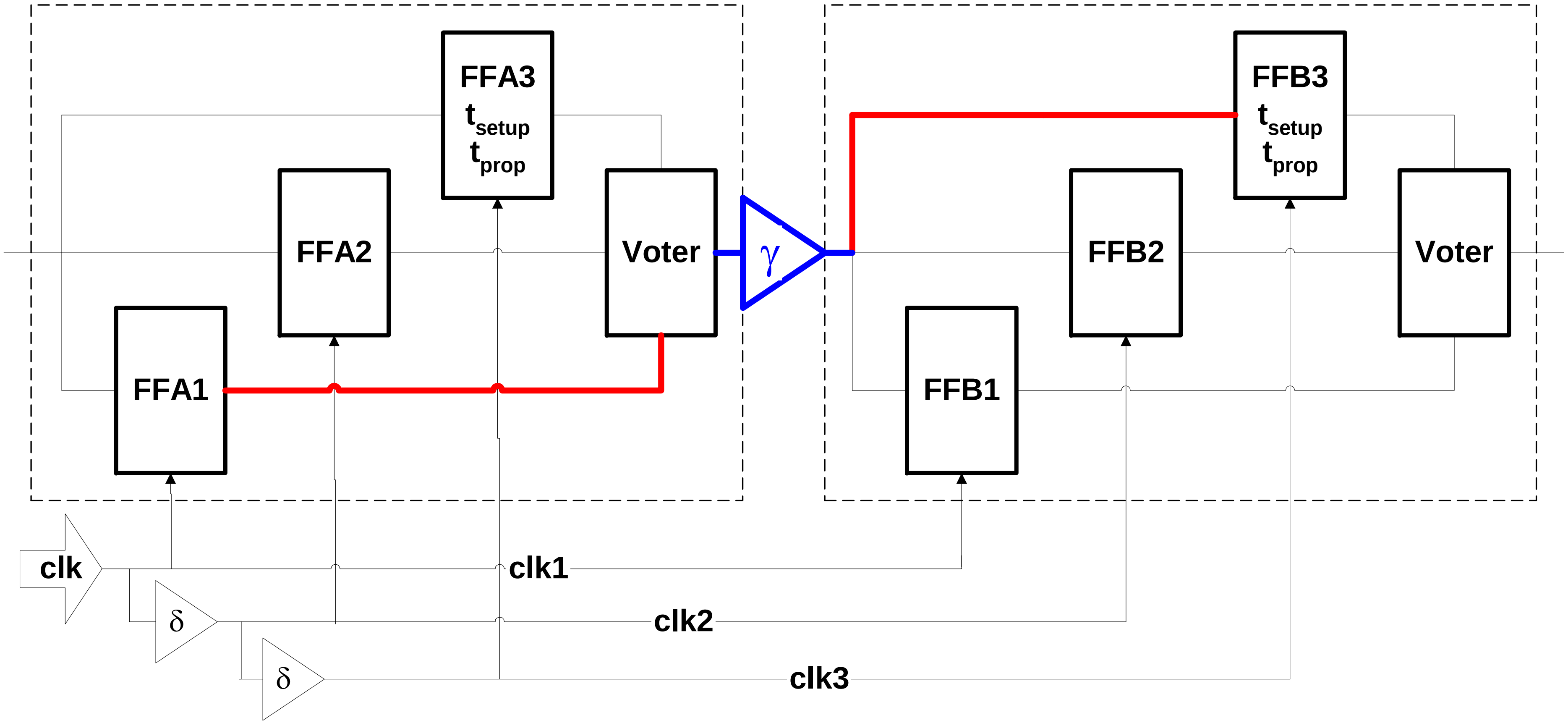
Difference between clock and data arrival in each TMR triplet

Before hold-fix: well pronounced peak $\delta_{eff} = \delta_{nominal}$

Clock skew creates many hold violations

After wrong hold-fix: two maxima (with and without delay insertion)

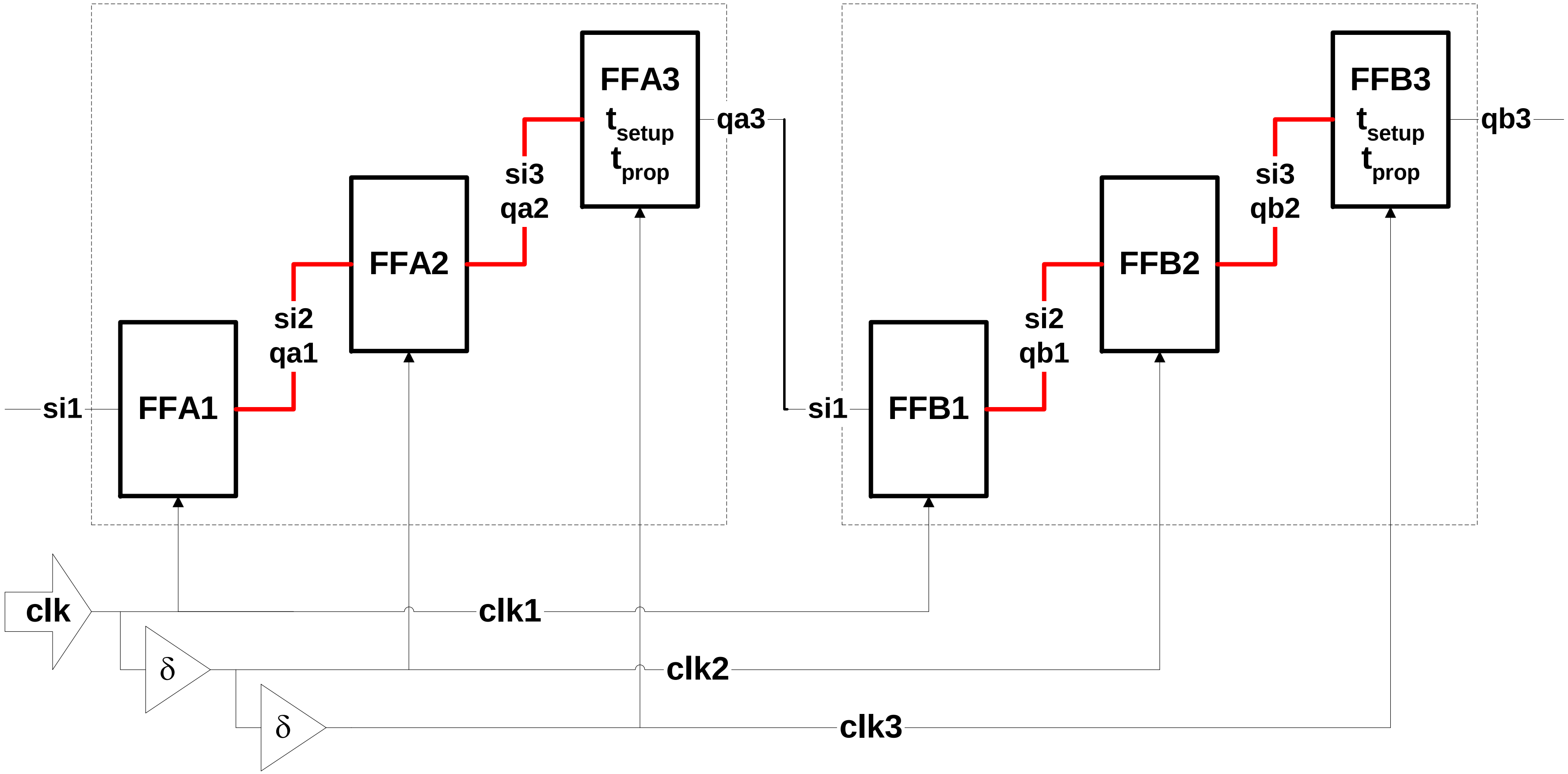
Correct hold fix



Group FF belonging to the same triplet and dont_touch

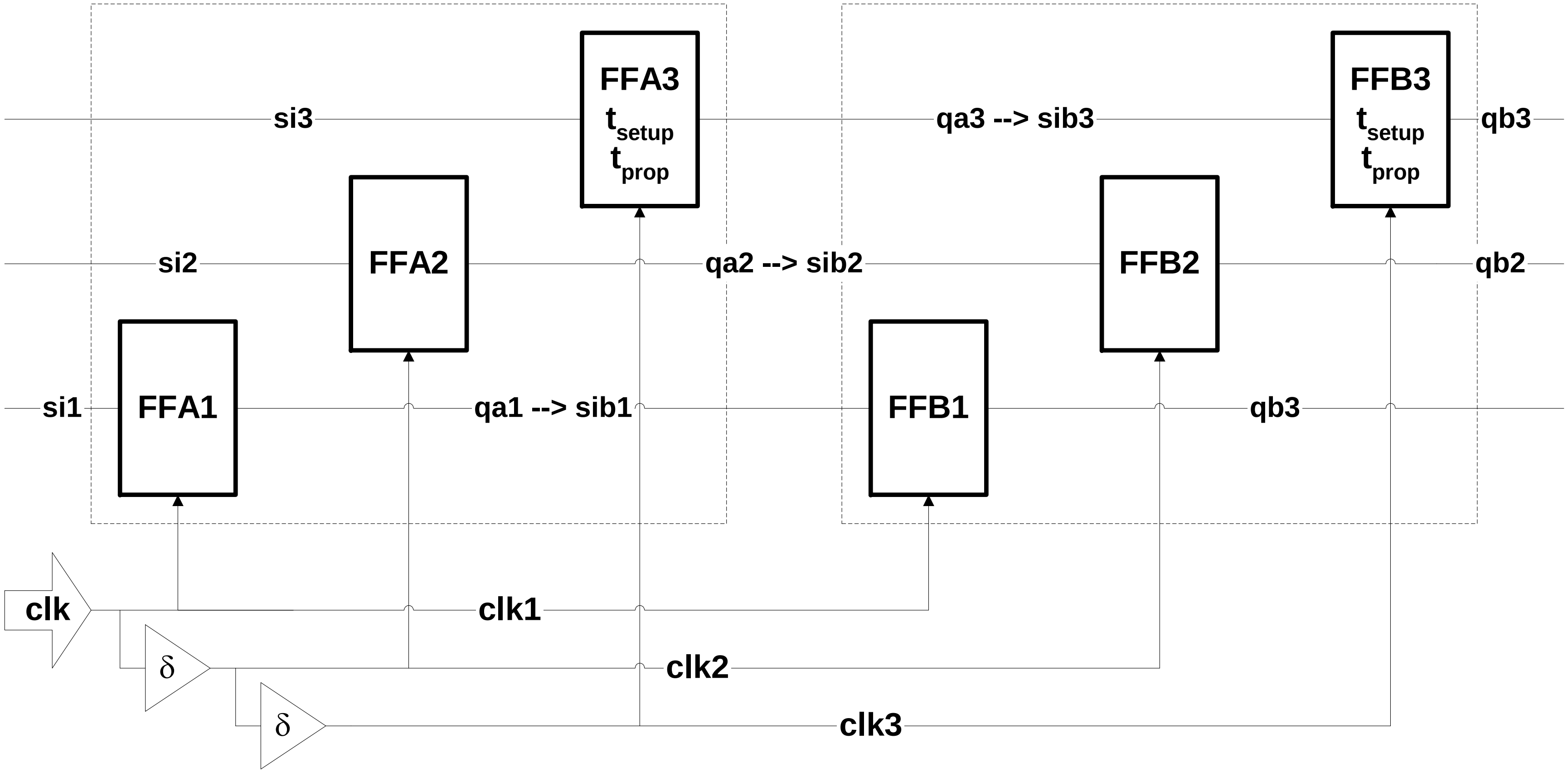
➔ SET protection through clock skew conserved

Scan Path Insertion (wrong)



Scan path routing across sub-clock domains ➔ hold violations

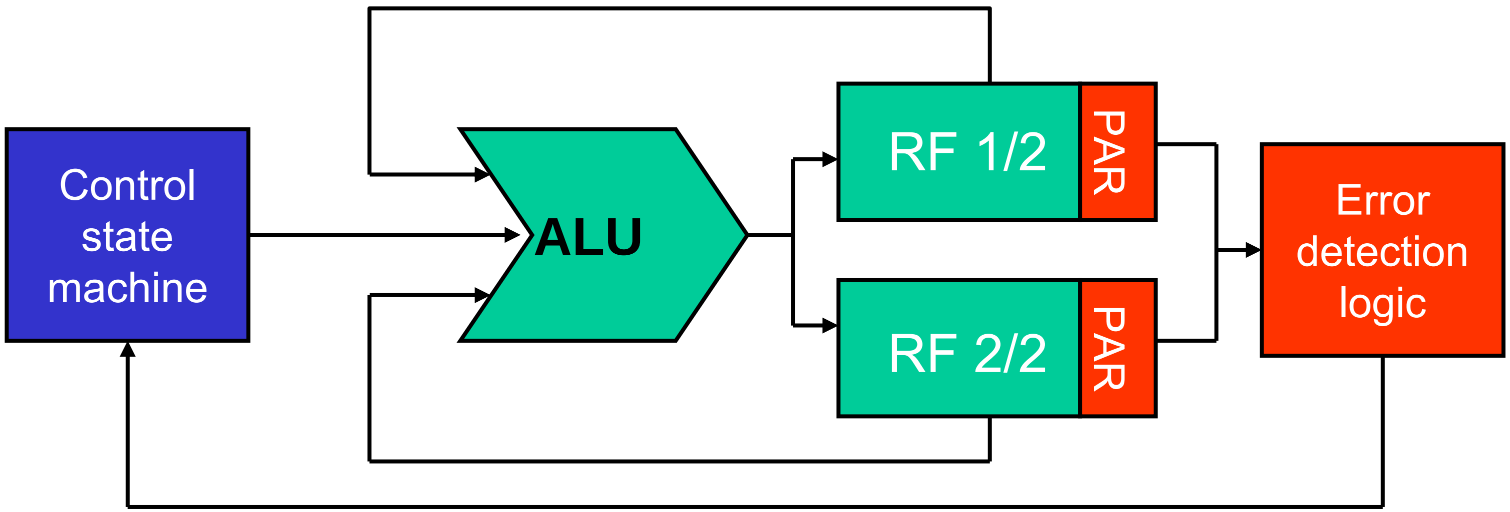
Scan Path Insertion (right)



Better: one scan path per sub-clock domain

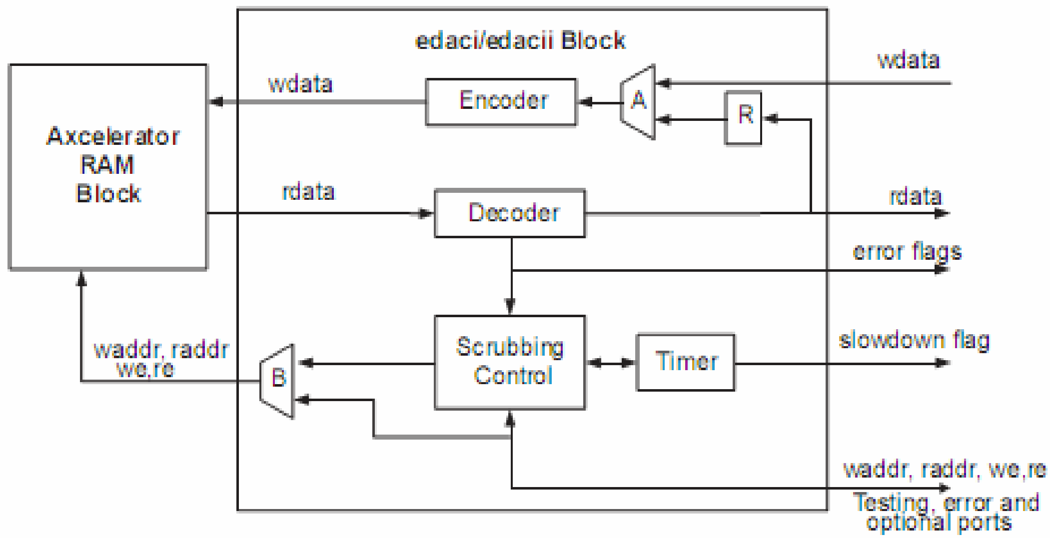
Protection of SRAM blocks (parity)

- ***XOR Parity bits***
  - Employed for a long time, also in ground-based computers
  - Error handling: correction/reload by HW state machine or software (reboot)
  - Loss of data, unless redundant data is available elsewhere in the system…
    - » Cache memories (duplicates in external memory) → cache miss on parity error
    - » Duplicated memories (e.g. a 3-port register file composed of two 2-port memories)
      Error detection while processing possibly corrupt data → normally no timing penalty
      Only in error case: copy correct data from replica and repeat processing



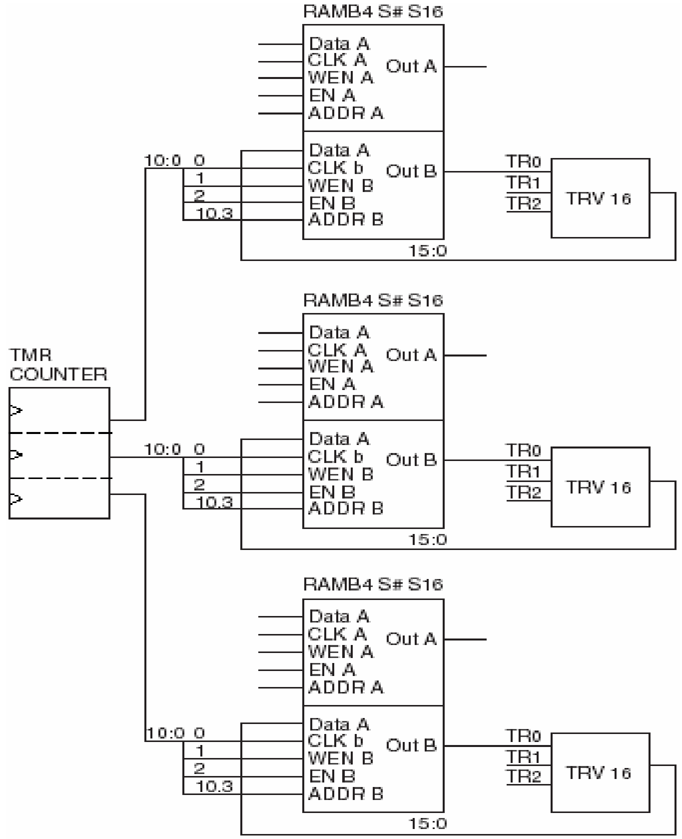
Protection of SRAM blocks (EDAC)

- ***EDAC = Error Detection And Correction***
- ***ECC (Error Correcting Codes)***
  - Hamming codes to correct single bit flips per word
  - EDAC VHDL package from ESA: http://www.esa.int/TEC/Microelectronics/SEMH0X8L6VE_0.html
  - Reed Solomon for multiple bit upsets (MBU) in SDRAM
  - Scrubbing required to prevent error accumulation (scrubbing)
  - Control state machine to rewrite corrected data
  - Timing penalty
    → start processing with uncorrected data and abort processing (rewind pipeline) in case of error
  - Example – ACTEL core: www.actel.com/documents/EDAC_AN.pdf



Protection of SRAM blocks (TMR)

- ***Triplicated memory (Xilinx)***
  - Scrubbing in background using spare port of dual-port memory
  - No
  - Huge area overhead
  - Also efficient against configuration upset

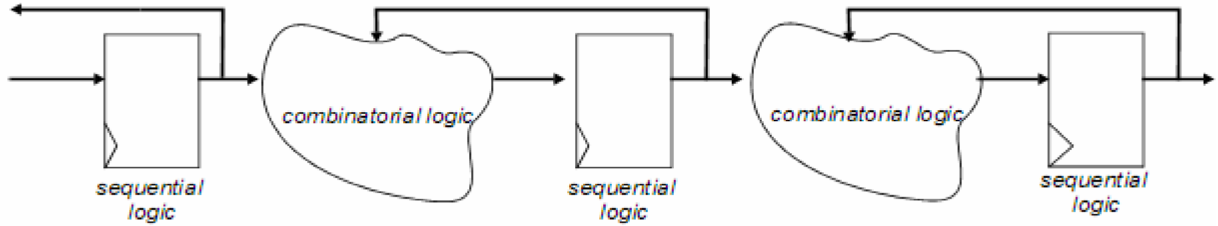


SEU in reprogrammable FPGA (RFPGA)

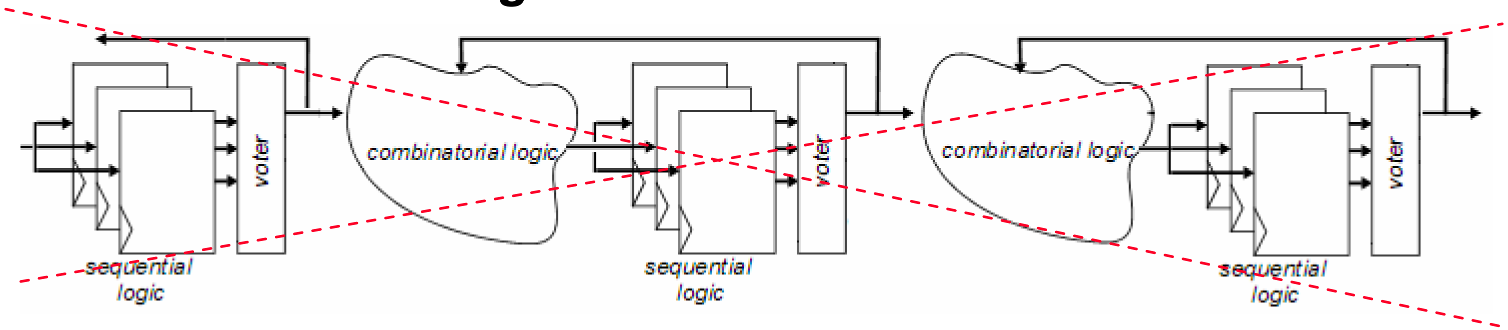
- ***Increasing interest for SRAM based RFPGA***
  - Lower NRE cost than ASIC
  - In-flight reconfiguration capability
  - High performance and complexity allowing System-On-FPGA
- ***SEU in configuration memory***
  - Affect not only user data or state (as in ASIC) …
  - … but alter the functionality of the circuit itself
  - … turn the direction of I/O pins
- ***SEU mitigation for RFPGA***
  - Configuration scrubbing or read-back and partial reconfiguration
  - Triplication of registers <u>and combinatorial logic</u>
  - Voting of logical feedback paths
  - Redundancy for user memory
  - Voting of the outputs
  - Triplication of I/Os

TMR for SRAM FPGA

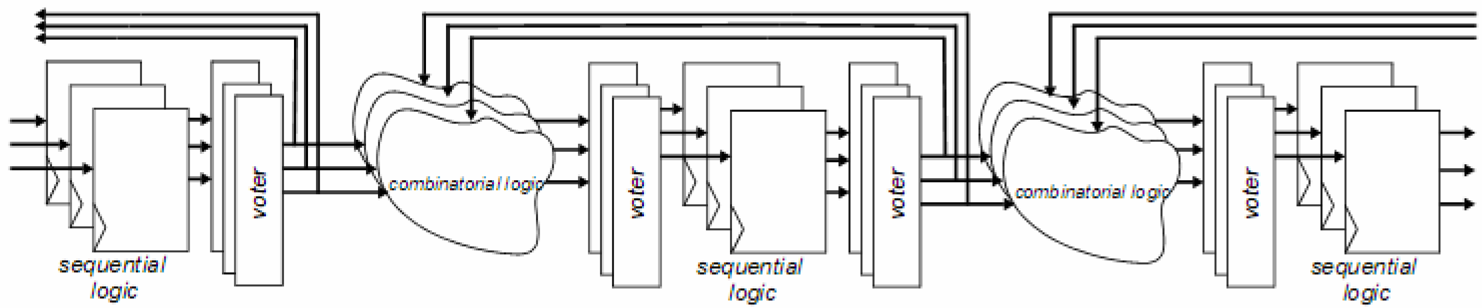
- ***Plain sequential and combinatorial logic***



- ***Standard TMR with single voters** – not for SRAM FPGA*



- ***TMR for sequential and combinatorial logic and voters***



SEU mitigation in reprogrammable FPGA

- ***SEE mitigation by design for commercial RFPGA***
  - Functional Triple Modular Redundancy (FTMR) – combinatorial and sequential triplication and voting in implemented in VHDL source code
    - » http://microelectronics.esa.int/techno/reprofpga.htm
- ***Xilinx TMR tool***
  - Triplication of combinatorial, sequential logic and IO's and feedback voters
  - http://www.xilinx.com/ise/optional_prod/tmrtool.htm
- ***SEE hard reprogrammable FPGA***
  - Atmel AT40KEL and the ATF280 FPGA under CNES contract
    - http://www.atmel.com/dyn/products/devices.asp?family_id=641#1477
  - Xilinx Virtex-5QV (SIRF = SEU Immune Reconfigurable FPGA)
    - http://www.xilinx.com/products/virtex5qv/
  - Actel RT ProASIC3 – flash based FPGA
    - http://www.actel.com/products/milaero/rtpa3/default.aspx
  - JAXXA/CNES/Atmel development, 450kG SRAM based FPGA on 150 nm SOI
    - https://eeepitnl.tksc.jaxa.jp/JP/event/MEWS/22nd/data/16_12_1.pdf

Verification of SEE hardening

- ***TMR or hardened cells are larger and slower than soft FF***
  - » **Redundancy removed by logic optimisation (synthesis and back-end)**
  - » **TMR modified by timing optimisation**
- ***Defects in redundant structures do not appear at simulation***
  - » **TMR simulation "works" even if only two of the three FF are correct**

??????????????????

How do we know if the hardening concept was properly implemented?

Verification of SEE hardening

- ***TMR or hardened cells are larger and slower than soft FF***
  - » **Redundancy removed by logic optimisation (synthesis and back-end)**
  - » **TMR modified by timing optimisation**
- ***Defects in redundant structures do not appear at simulation***
  - » **TMR simulation “works” even if only two of the three FF are correct**

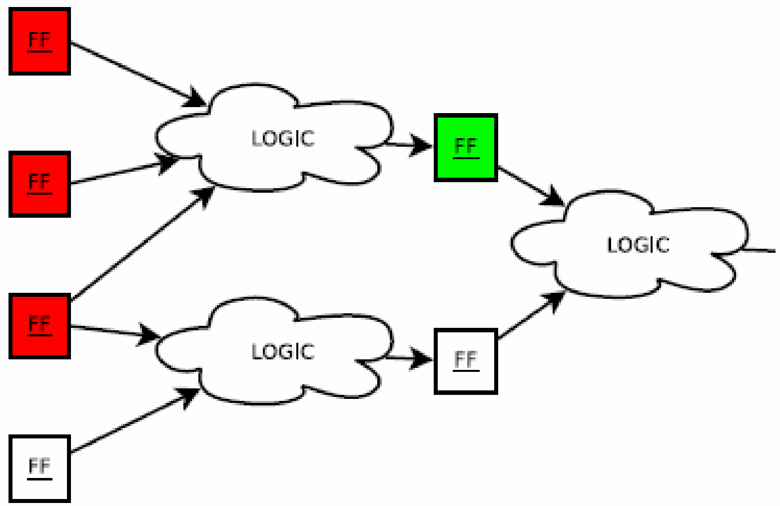
⇒⇒⇒⇒⇒⇒⇒

- ***Structural and formal verification, timing analysis***
  - » **Presence of triple FF, correct wiring of the three clock/reset domains**
  - » **Parsing the netlist with scripts (grep)**
  - » **Increasing complexity requires formal verification tools**
- ***Fault simulation and injection***
  - » **Functional impact of tolerated SEU**
- ***Ground radiation testing***

InFault – Intelligent Fault Analaysis

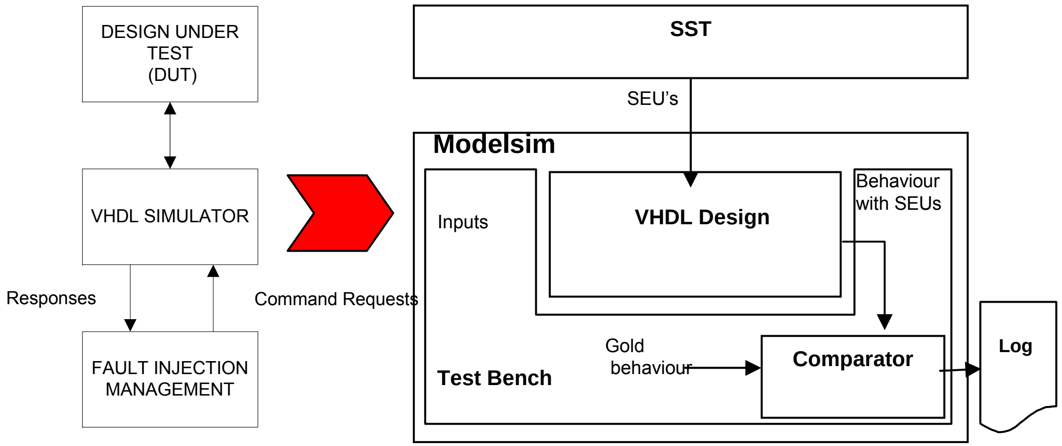
- ***C++ SW to recognise TMR in a netlist and validate its correctness***
  - Simon Schulz, Giovanni Beltrame, David Merodio Codinachs:
    Smart Behavioural Netlist Simulation for SEU Protection Verification
    http://microelectronics.esa.int/papers/SimonSchulzInFault.pdf

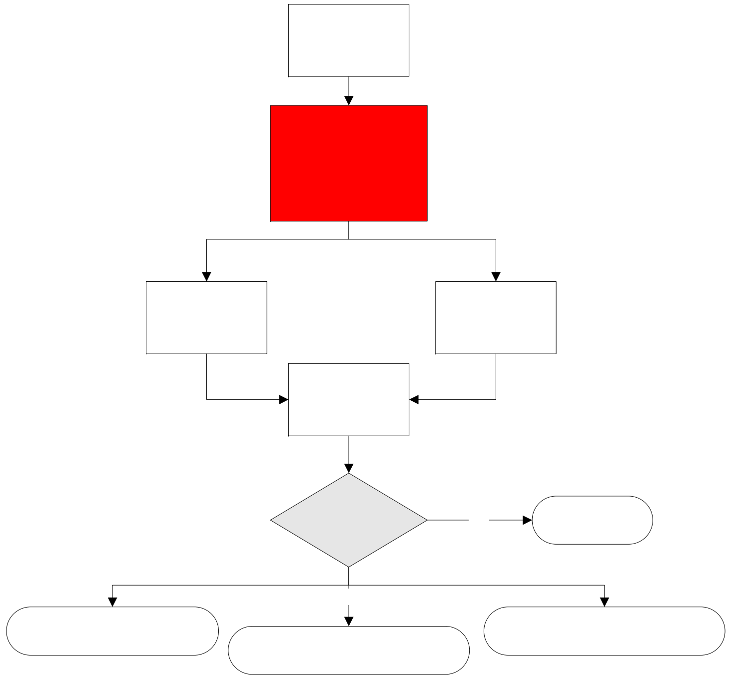
- ***Main algorithm steps***
  - Netlist parser (Verilog, EDIF)
  - Creates an untimed graph representation of the logic
  - Detects TMR triplets and voters
  - Checks TMR and voting logic
  - Checks the (triplicated) clock and reset trees



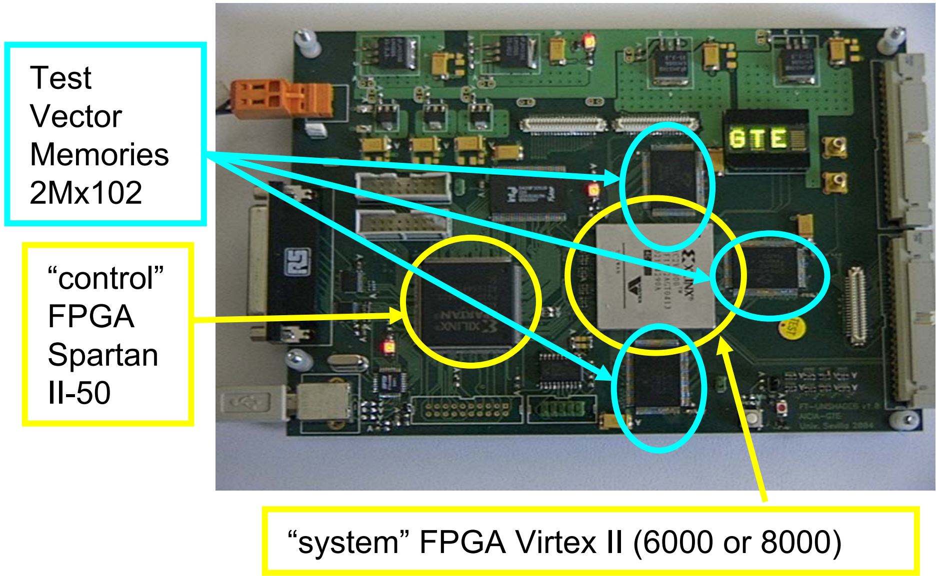
SST: the SEU Simulation Tool

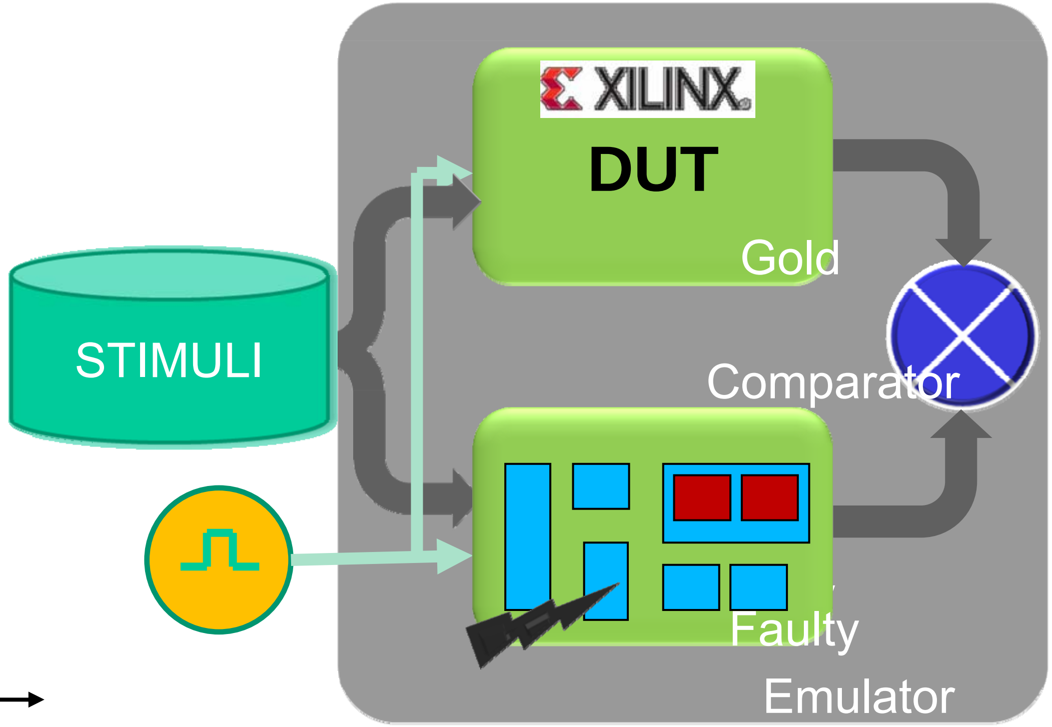
- ***TCL package to inject SEU into flip-flops during Modelsim simulation***
  - http://microelectronics.esa.int/asic/SST-FunctionalDescription1-3.pdf
  - http://www.nebrija.es/~jmaestro/esa/sst.htm

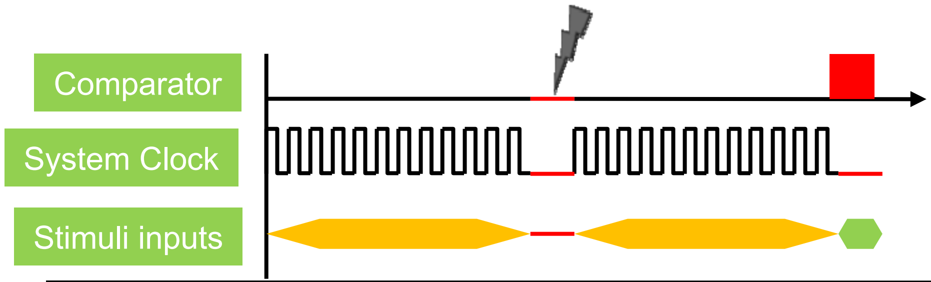




- ***SEU injection into flip-flops based on FPGA partial reconfiguration***
  - http://microelectronics.esa.int/fiws/WFIFT_P9_FT-UNSHADES.ppt (or .ppsx)
  - http://walle.us.es/ftunshades/

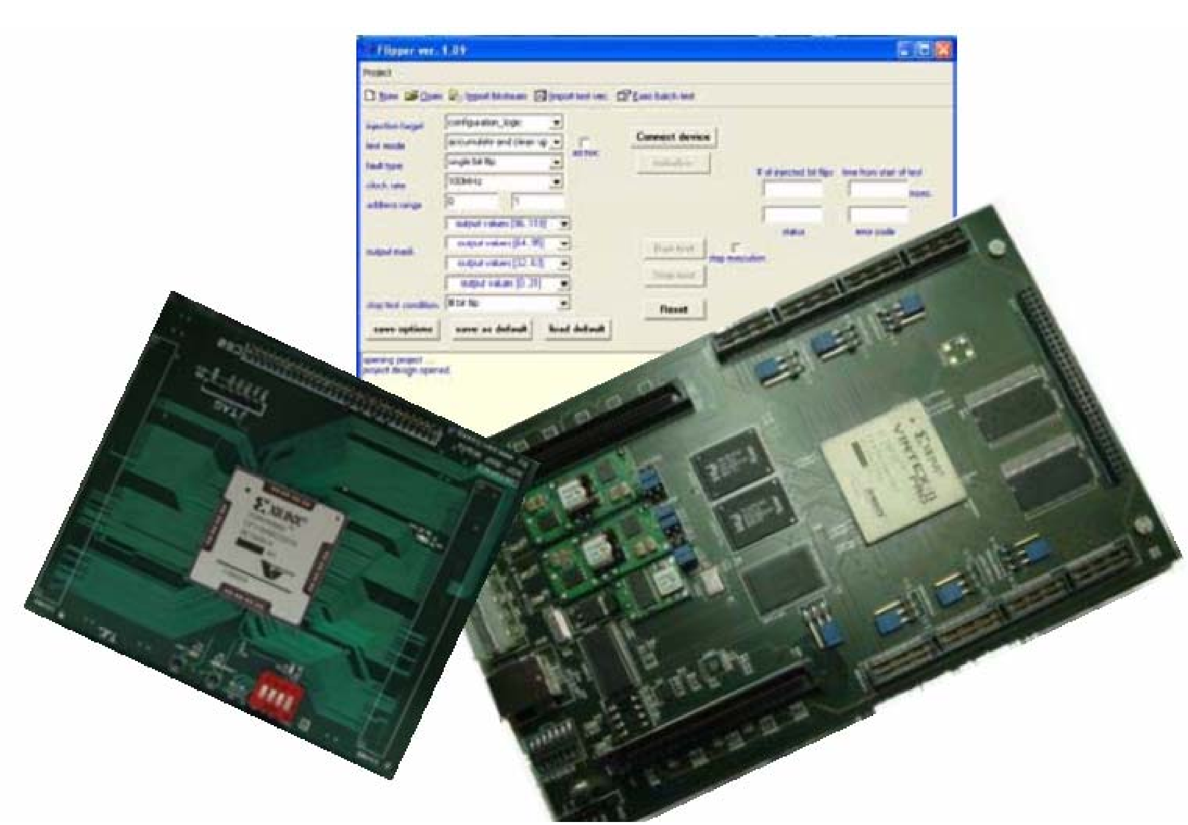


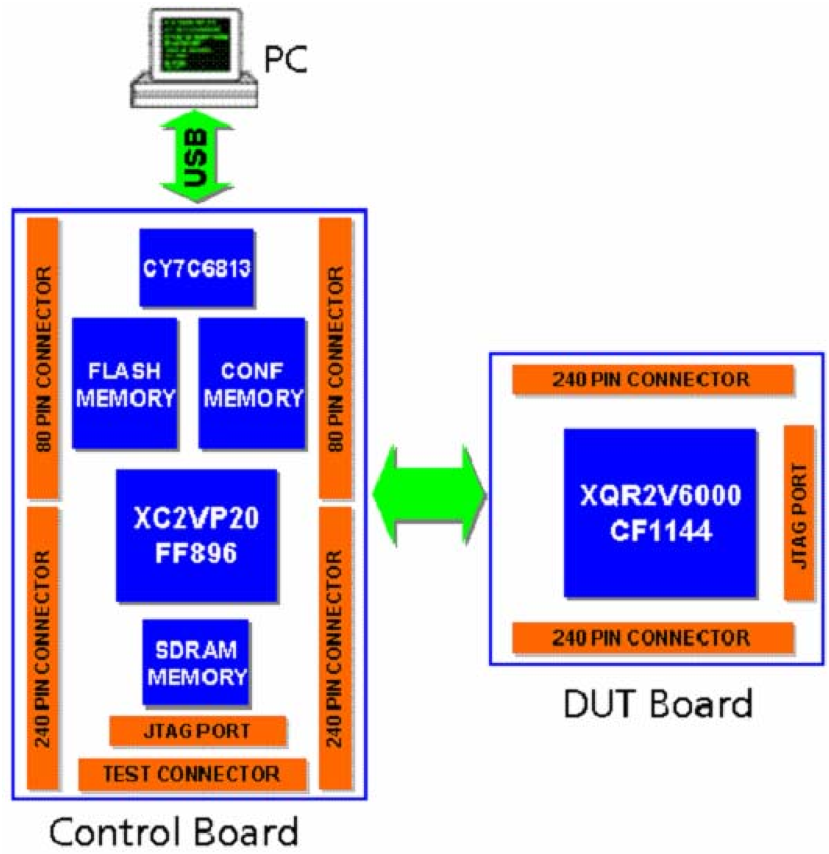




FLIPPER – Injection platform for SRAM based FPGA

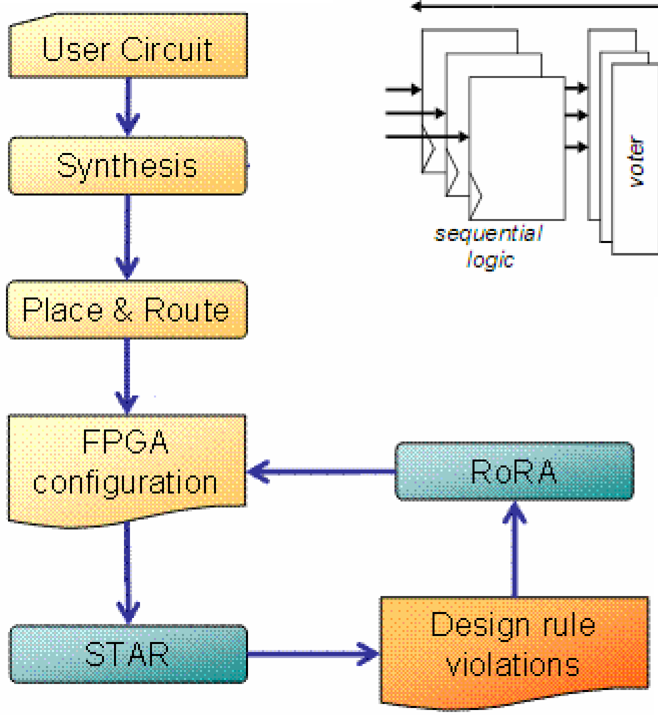
- ***Injection into Xilinx Virtex2 configuration RAM (Virtex4 in preparation)***
  - http://microelectronics.esa.int/fiws/WFIFT_P8_alderighi_FLIPPER.pdf
  - http://cosy.iasf-milano.inaf.it/flipper_index.htm

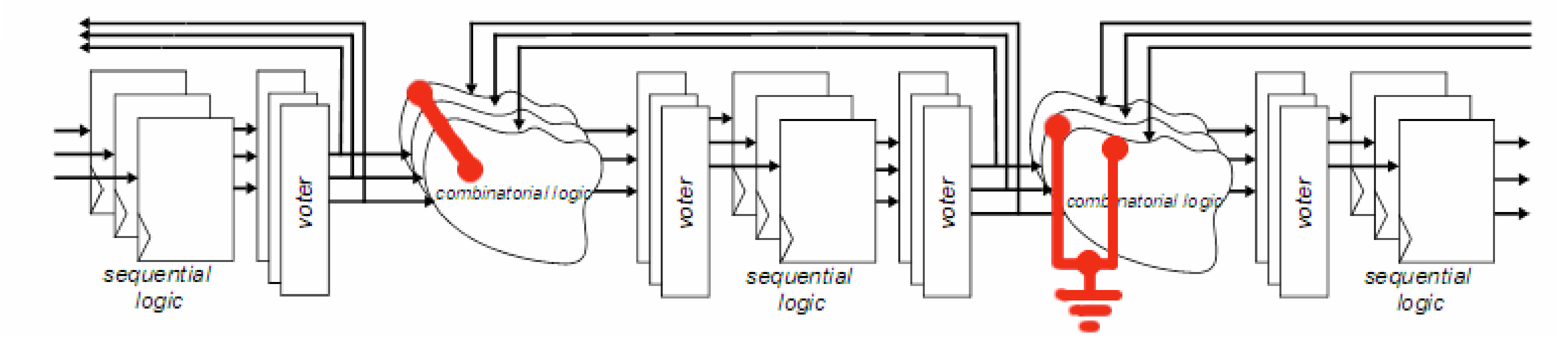




STAR – RORA: SEE protection of SRAM FPGA layout

- ***CAD tools to analyze and improve the layout of SRAM based FPGA***
  - http://www.cad.polito.it/research/FPGA/Field%20Programmable%20Gate%20Array.html
  - http://microelectronics.esa.int/fiws/WFIFT_P6_Analysis_SCU_MBU.pdf
  - http://microelectronics.esa.int/fiws/WFIFT_P7_Mitigation_of_SCU_and_MCU.pdf

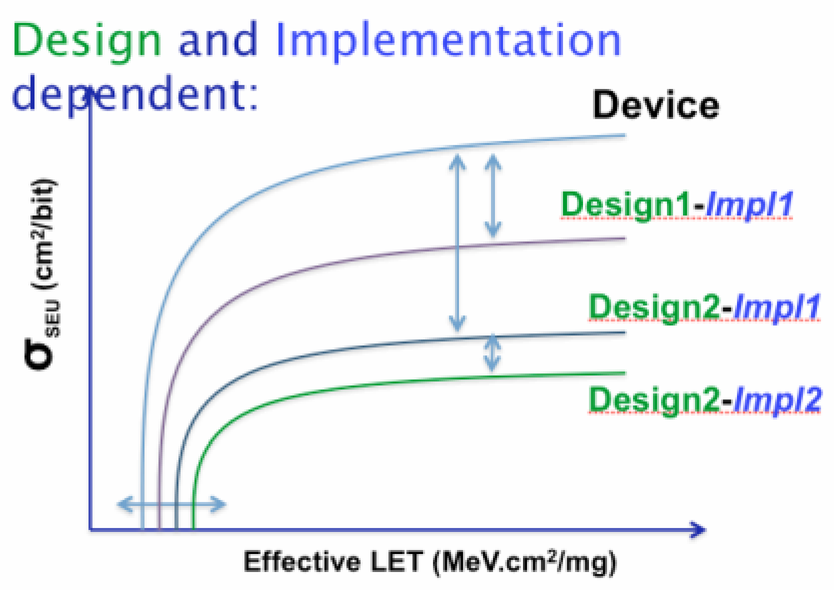


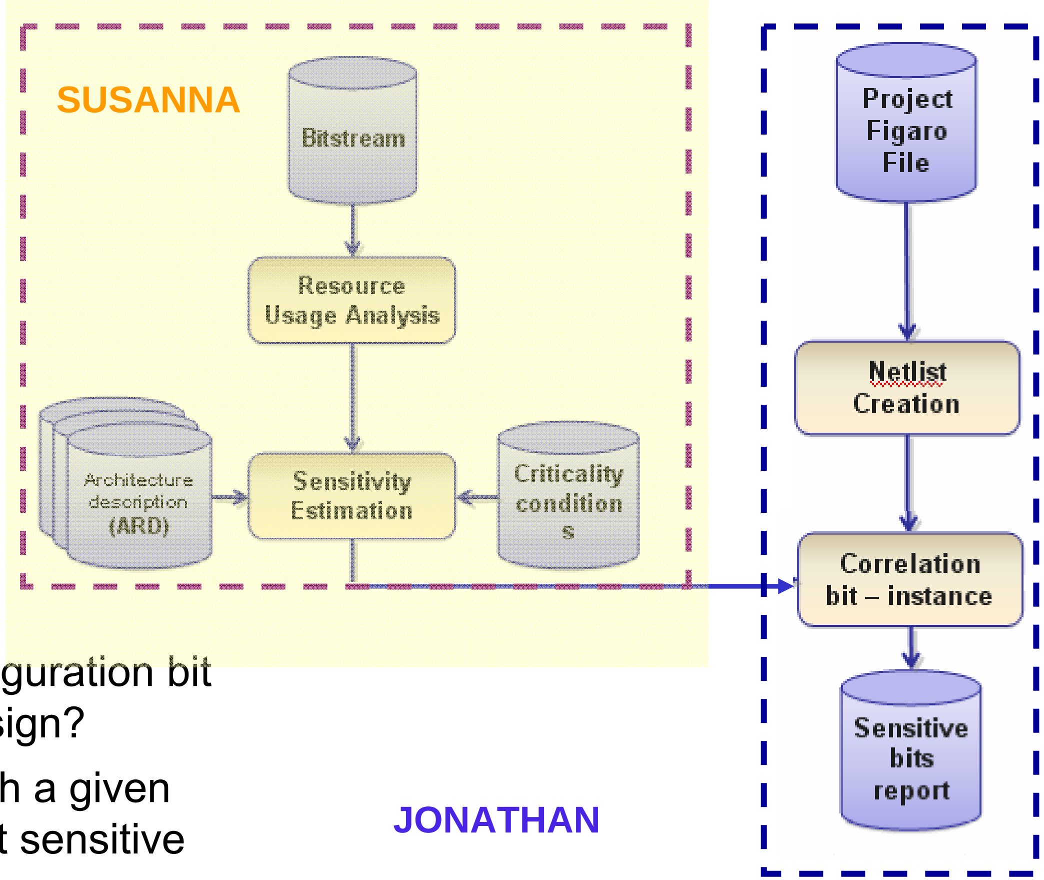


- STAR: STatic AnalyzeR – Identify bits which are sensitive in spite of full TMR, e.g. bits causing faults in two TMR domains
- RORA: Reliability-Oriented Routing Algorithm – Modify the layout (place and route) to fix the sensitive bits identified by STAR
- Supports all Xilinx devices Spartan2 – Virtex4

SUSANNA – JONATHAN – targeting Atmel FPGA

- ***Fault tolerance analysis of designs on Atmel AT40k and ATF280 FPGA***
  - http://microelectronics.esa.int/papers/SusannaJonathanSemiFinal.pdf

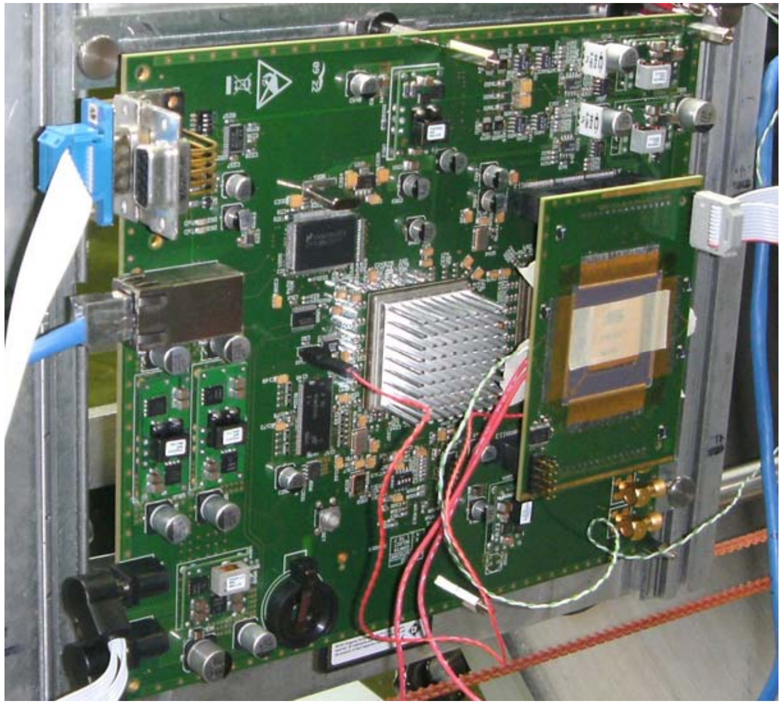


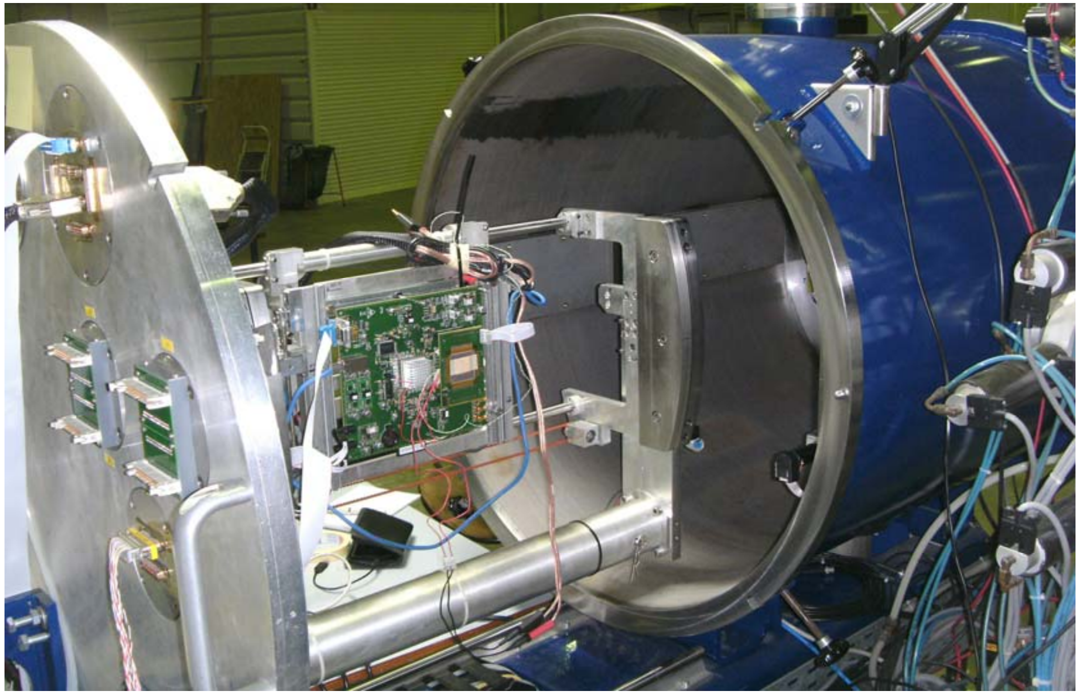


- SUSANNA: Identify sensitive bits of the configuration bit stream. → Does a bit flip lead modify the design?
- JONATHAN: Correlates the sensitive bits with a given instance in the design and identifies the most sensitive modules in the design
- Improved place & route algorithms under development

Ground radiation testing

- ***Radiation Facilities in use by ESA*** https://escies.org/ReadArticle?docId=230
  - Co-60 at ESA/ESTEC, Netherlands (total dose)
  - Californium-252 at ESA/ESTEC, Netherlands
  - Paul Scherrer Institut (PSI), Switzerland: proton irradiation
  - Louvain la Neuve (UCL), Belgium: heavy ions and protons
  - Jyväskylä University, Finland: heavy ions and protons





Conclusion

- ***SEE mitigation requires a sound Radiation Hardening Approach***
  - Identify dependability requirements and environmental conditions
  - Perform radiation analysis to define hardening concept
  - Is 100% protection of every element always necessary?
    Determine the impact of an upset at system level
    → Sometimes, selective use of SEE protection is sufficient
  - Implement, design, verify during design time, validate the final result
- ***ASIC libraries with hardened elements (flip-flops, buffers)***
  - TMR allows using commercial cell libraries, but it is difficult to implement with commercial EDA tools
  - Hardened library cells are easier to use
- ***SRAM reprogrammable FPGA require different hardening concepts***
- ***Thorough verification of the radiation hardening is required***
  - Redundancy might be removed by EDA tools
- ***Numerous tools exist to verify and validate the designs***

Contact:

Roland.Weigand [at] esa.int

http://www.esa.int/TEC/Microelectronics/

http://microelectronics.esa.int/papers/TWEPP2010-RW.pdf

Questions?