Redundancy methods in ASICs

Sandro Bonacini

sandro.bonacini@cern.ch

SEU risk management

- SEU is not a destructive event
  - a given rate can be tolerated (depending on the system)
  - How often does it happen? Depends on:
    - Radiation environment
      - Which particles?
      - Which energy?
      - Which fluence?
    - Device sensitivity

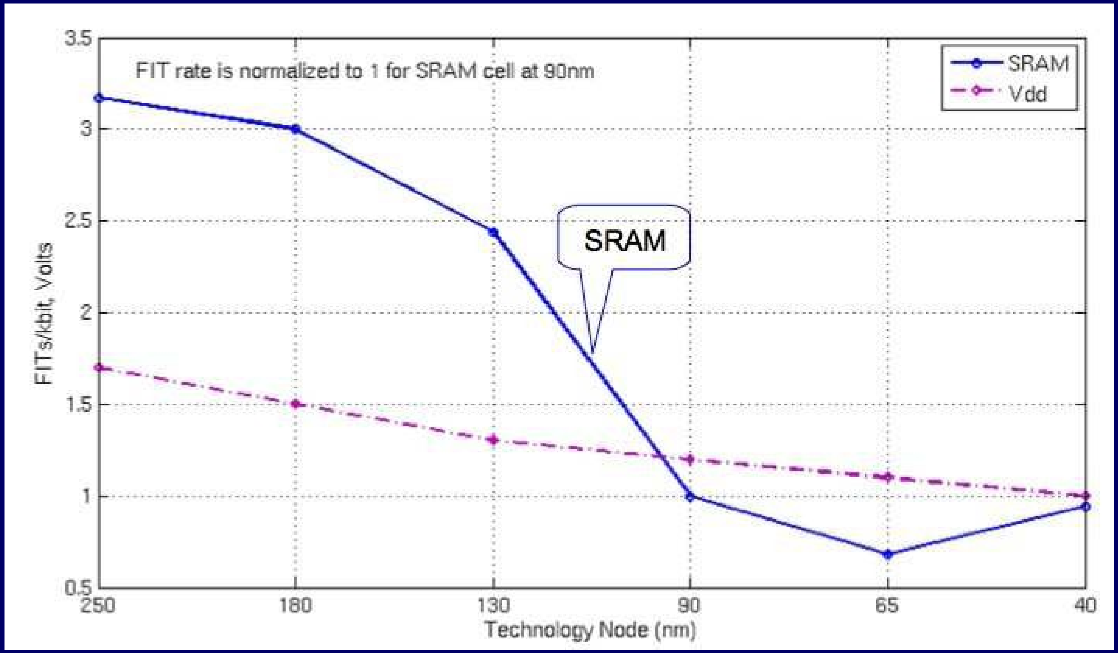
- The “safer” the circuit is designed, the more “expensive” it is (area, performance, complexity, …)
- Risk reduction comes at some cost

SEU and scaling

- FIT/bit decreases, but FIT/chip increases
  - C and VDD reduce with scaling
  - ...but smaller drain area

- All sources agree: DRAM sensitivity has been scaling down (cell area scaling has outpaced the decrease in stored charge).

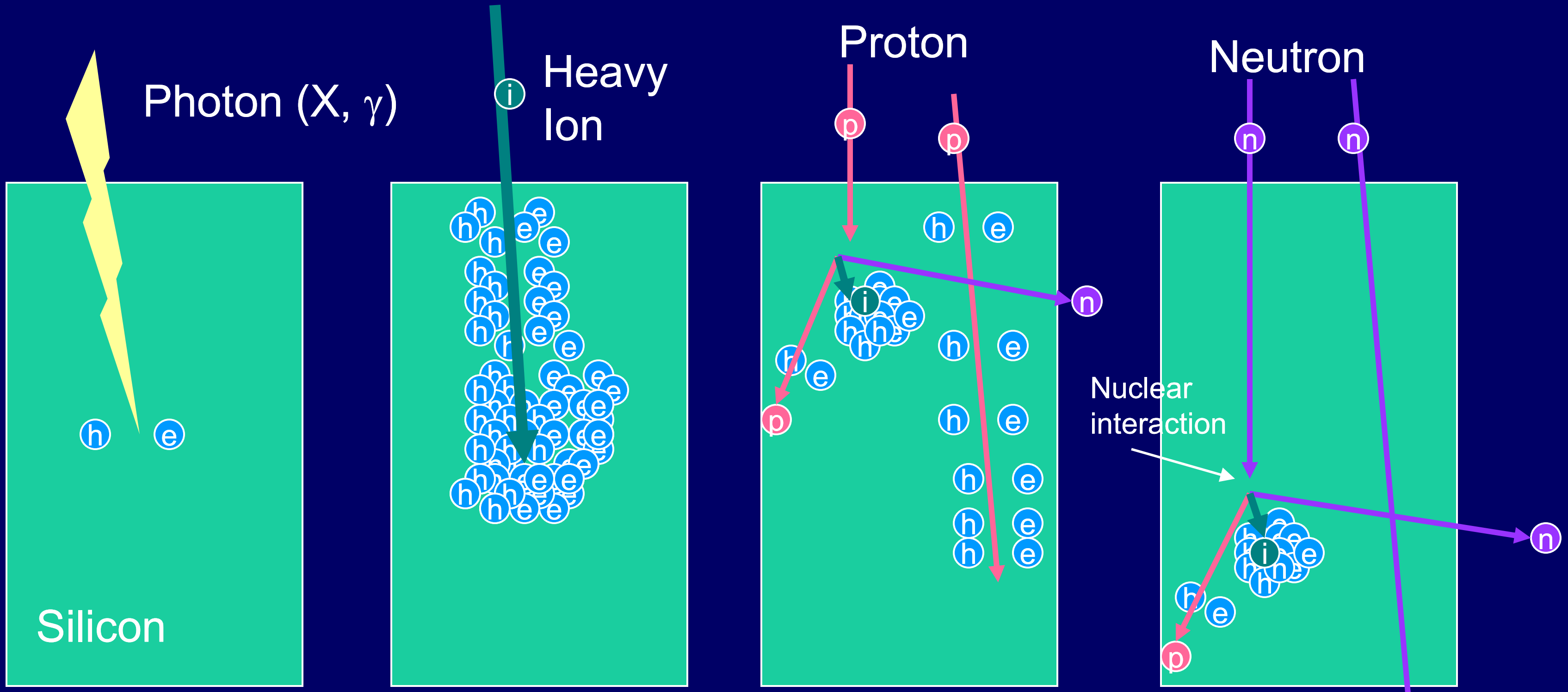
- charge collection efficiency and sensitive area are also important and change with technology generation



Dixit et al., “The impact of new technology on soft-error rates”, SELSE-6, Stanford 2010

Ionization from different radiation

- SEE: Traceable to the energy deposition initiated by one single particle, in a precise instant in time. Due to its stochastic nature, this can happen at any time – even at the very beginning of the irradiation
- Which particles can induce SEEs? In the figure below, a schematic view of the density of e-h pairs created by different radiation is shown.



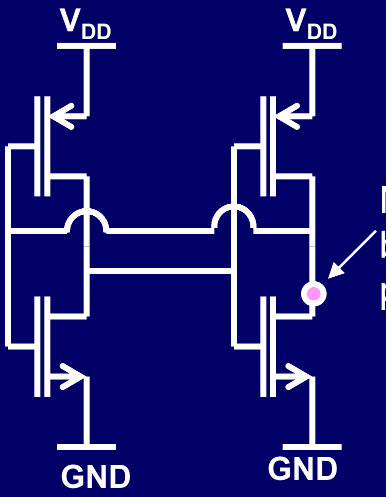
Small density of e-h pairs

Large density of e-h pairs

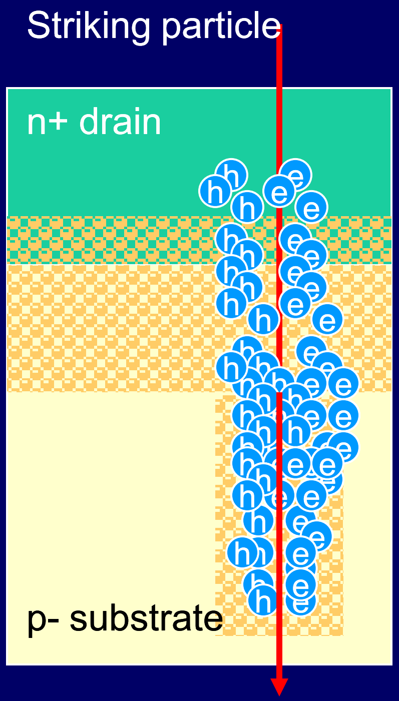
Small (proton) or no (neutron) density for direct ionization. Possible high density from Heavy Ion produced from nuclear interaction of the particle with Silicon nucleus.

Single Event Upset (SEU)

The e-h pairs created by an ionizing particle can be collected by a junction that is part of a circuit where a logic level is stored (logic 0 or 1). This can induce the “flip” of the logic level stored. This event is called an “upset” or a “soft error”.
This typically happens in memories and registers. The following example is for an SRAM cell.



Node stroke by the particle



e-h pairs in this region recombine immediately (lots of free electrons available in this n+ region)

Depletion region: e-h pairs are collected by n+ drain and substrate => those collected by the drain can contribute to SEU

High density of e-h pairs in this region can instantaneously change effective doping in this low-doped region, and modify electric fields. This is called “funneling”. Charge can hence be collected from this region to the n+ drain

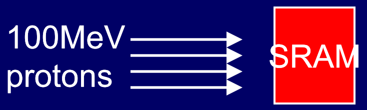
The density of pairs depends on the stopping power of the particle, or dE/dx, or Linear Energy Transfer (LET).

SEU cross-section (1)

- Sensitivity of a circuit to SEU (or in general to any SEE) is characterized by a cross-section
- The cross-section contains the information about the probability of the event in a radiation environment

Example: what is the error rate of an SRAM in a beam of 100MeV protons of flux 10^5 p/cm^{2}s?

1. Take the SRAM and irradiate with 100MeV proton beam. To get good statistics, use maximum flux available (unless the error rate observed during test is too large, which might imply double errors are not counted => error in the estimate)



2. Count the number of errors corresponding to a measured fluence (=flux * time) of particles used to irradiate

Example:
N of errors = 1000
Fluence = 10^{12} p/cm^2

Cross-section (σ)= N/F = 10^{-9} cm^2

3. Multiply the cross-section for the estimated flux of particles in the radiation environment. The result is directly the error rate, or number of errors per unit time.

If (σ) = 10^{-9} cm^2

and flux = 10^5 p/cm^{2}s

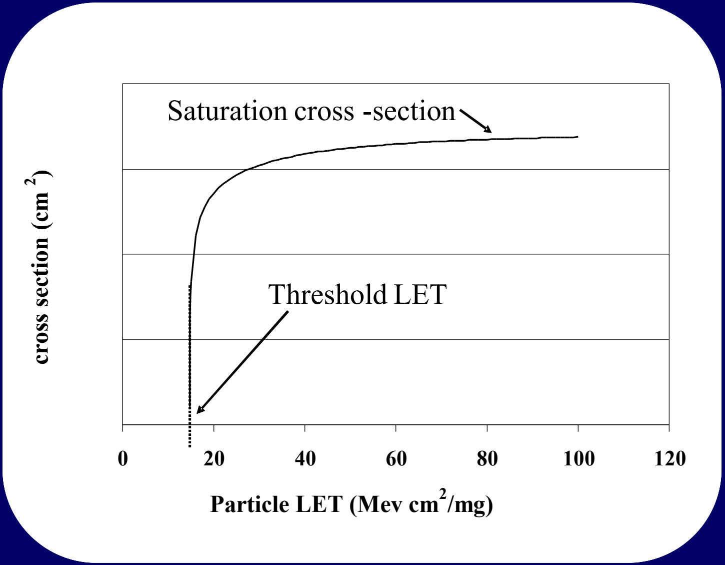
Error rate = 10^{-4} errors/s

SEU cross-section (2)

- In reality, things are generally more difficult – the real radiation environment is a complex field of particles
- One needs models to translate cross-sections measured at experimental facilities (protons or heavy ions beams) into error rates in the field
- The better the experimenter knows the sensitivity of the circuit, the better he/she can estimate the error rate in the real environment
- Heavy Ions (HI) irradiation tests are very good to probe completely the sensitivity of a circuit. With HI, it is possible to vary the LET of the particles and measure the correspondent cross-section.

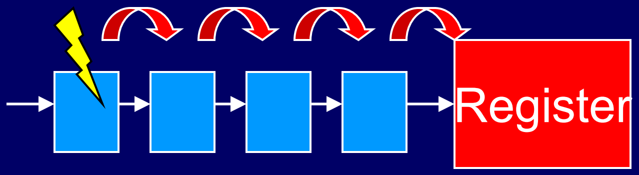
It is possible to chart the measured cross-section for different LET of the ions, as shown in the figure to the right.

A useful information to situate the sensitivity of circuits in the LHC is the maximum LET of recoils from nuclear interaction of hadrons with the Si nuclei. The maximum LET is for a Si recoil and the LET is about 15 MeVcm^{2}mg^{-1}. This information can be used to judge if a circuit for which Heavy Ion data is available will experience a high error rate in the LHC

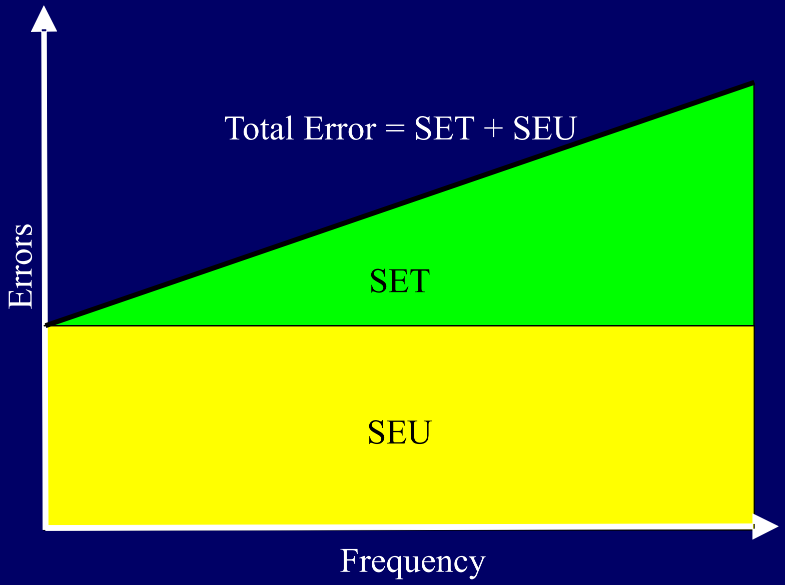


Single Event Transient (SET)

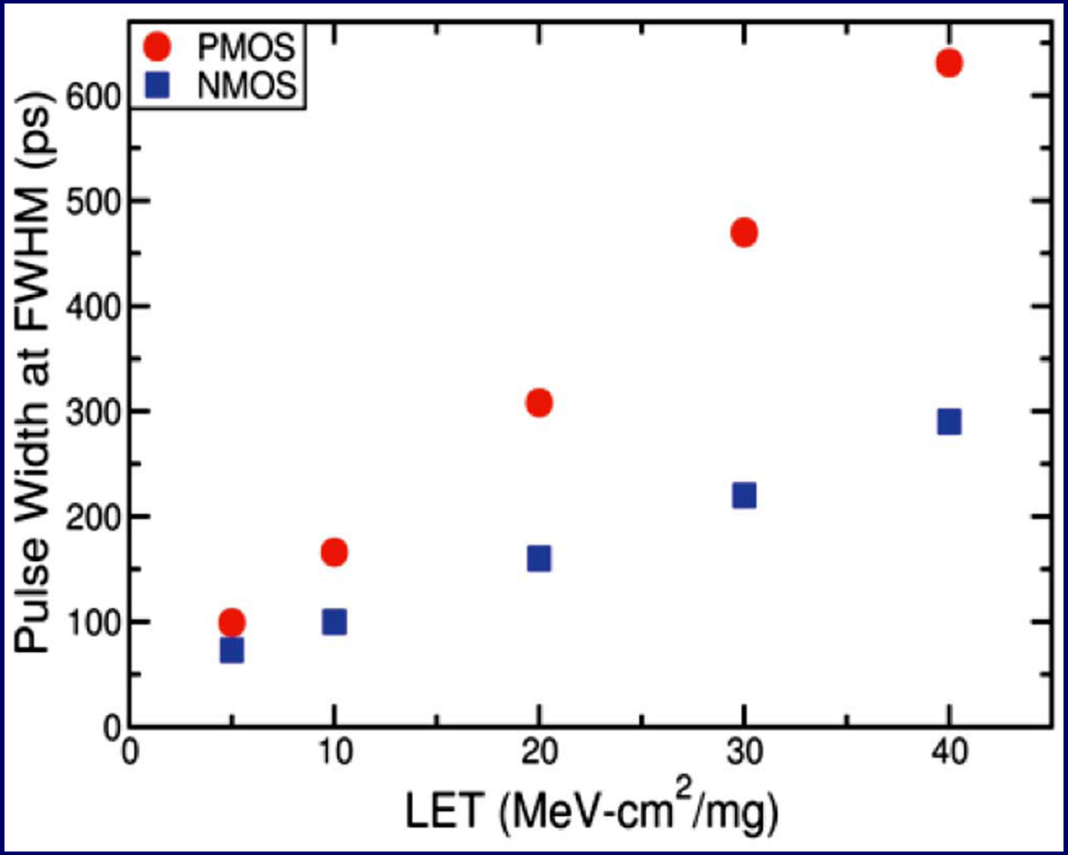
- Particle hit in combinatorial logic: with modern fast technologies, the induced pulse can propagate through the logic until it is possibly latched in a register
- Latching probability proportional to clock frequency
  - Linear behaviour with clock frequency is observed
  - more clock edges = more probability of registering errors



Combinatorial logic



SET pulse width: pFET/nFET

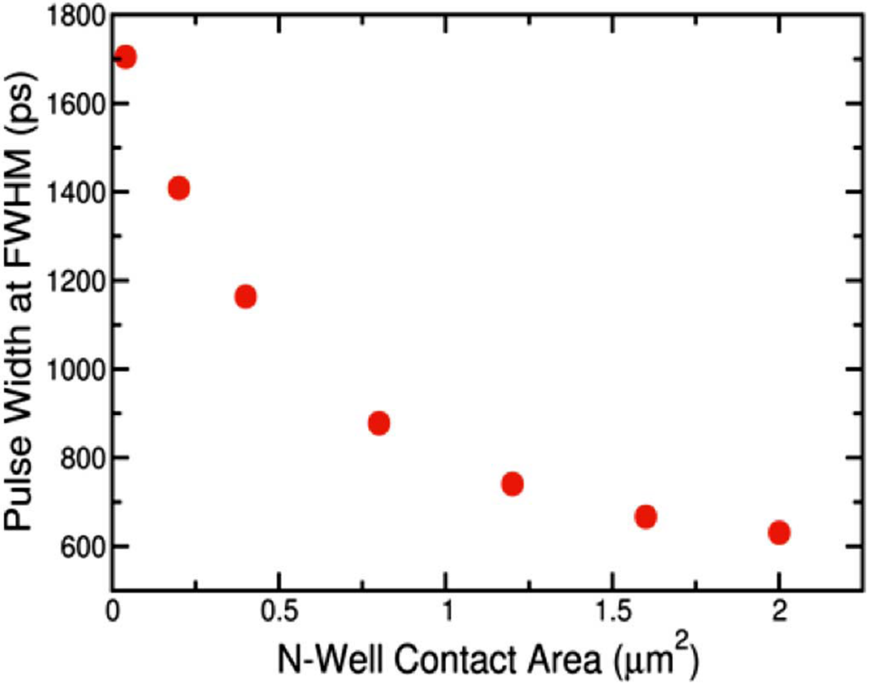


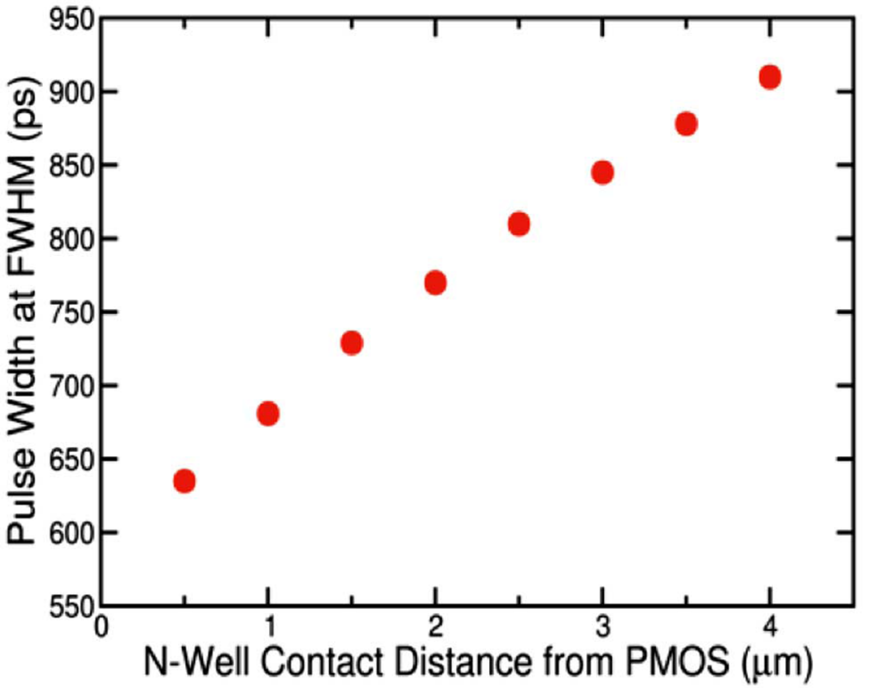
- SETs due to P-hits tend to be longer due to parasitic bipolar amplification
- SET cross-section is normally higher for P-hits
  - Area of pFET is bigger than nFET

FWHM: Full-width half-max = half-rail to half-rail

Jagannathan et al., “Independent Measurement of SET pulse widths from N-hits and P-hits in 65-nm CMOS”, IEEE Trans. Nucl. Sci. Vol. 57, No. 6, Dec. 2010

SET pulse width: n-well contact





(90 nm CMOS)

- P-hit pulse width depends on n-well contact proximity and area
  - Must lower n-well resistance to VDD
- Problem in libraries with tap-cell
  - Example in 130 nm: tap can be 35 um from cell
- Pulse width increases with temperature
  - Shuming et al. “Temperature Dependence of Digital SET Pulse Width in Bulk and SOI Technologies”, Trans. Nucl. Sci. Vol. 55, Dec. 2008

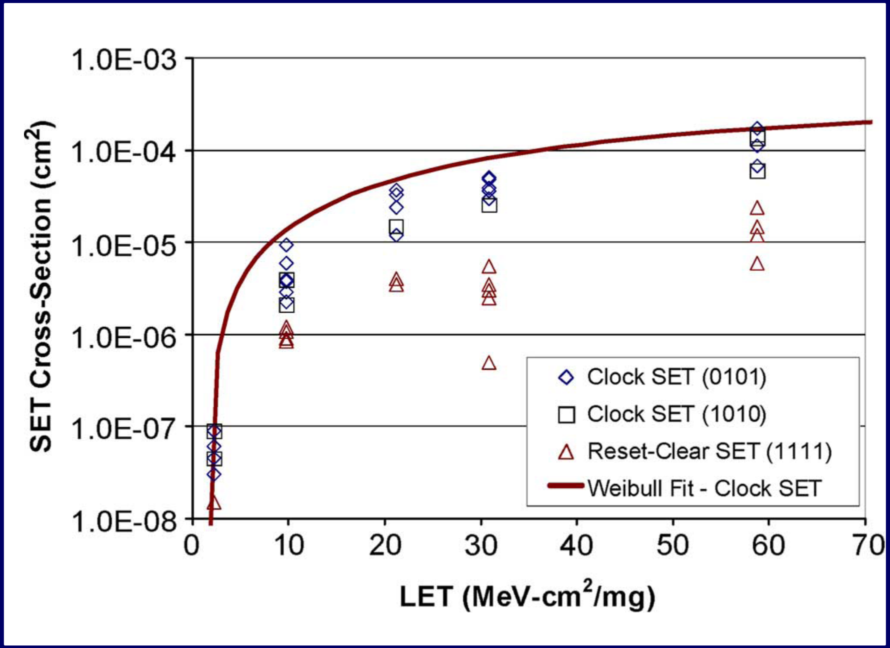
Oluwole et al., “Design Techniques to Reduce SET Pulse Widths in Deep-Submicron Combinational Logic”, IEEE Trans. Nucl. Sci. Vol. 54, No. 6, Dec. 2007

FWHM: Full-width half-max

SEFIs and hits on clock and reset

- SEFI: Single Event Functional Interrupt
  - Temporary loss of functionality due to the perturbation of control registers, clock signals, reset signals
    - Control state machines can enter in wrong state
    - Might require system reset

- Hits on clock network are increasingly important in modern technologies
  - ...as capacitance of clock branches decreased
  - Normally small cross-section



Cabanas-Holmen et al., “Clock and Reset Transients in a 90 nm RHBD Single-Core Tilera Processor”, IEEE Trans. Nucl. Sci. Vol. 56, No. 6, Dec. 2009

Solutions: SEU

- Technology level: epitaxial substrates, SOI,…
- Cell design: SEU-tolerant FFs or memories
  - Increase the critical charge by increasing the node capacitance:
    - Design larger transistors – also more driving strength
    - Add “extra” capacitors (metal-metal, or use special technology options – eDRAM like)
  - Use special architectures
- Redundancy
  - Triple Modular Redundancy (TMR): triplication and voting
  - Encoding (EDAC: error detection and correction)

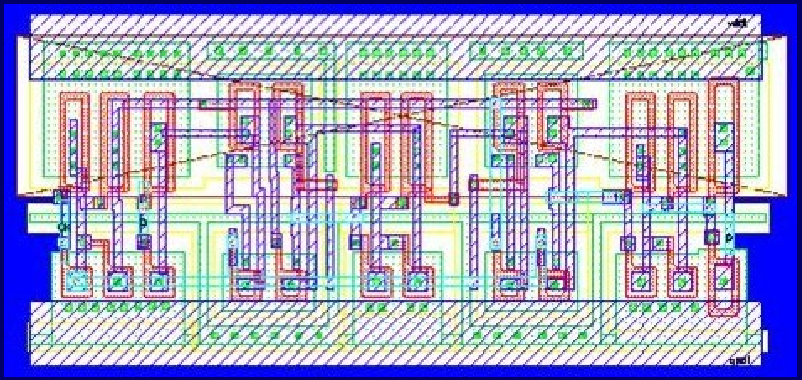
- Always to be considered at system level

Larger transistors

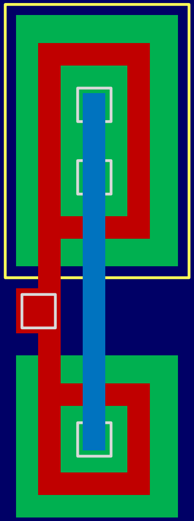
- Increase the critical charge by increasing the node capacitance:
  - Design larger transistors
    - Gives also more driving strength, therefore faster recovery

$$Q_{crit} \cong \frac{1}{2} CV_{DD}$$

$$Q_{dep} \cong \frac{q\rho_{Si} L_f LET}{E_{eh}}$$



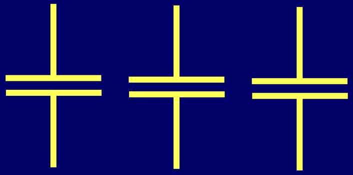
ex. in 250 nm

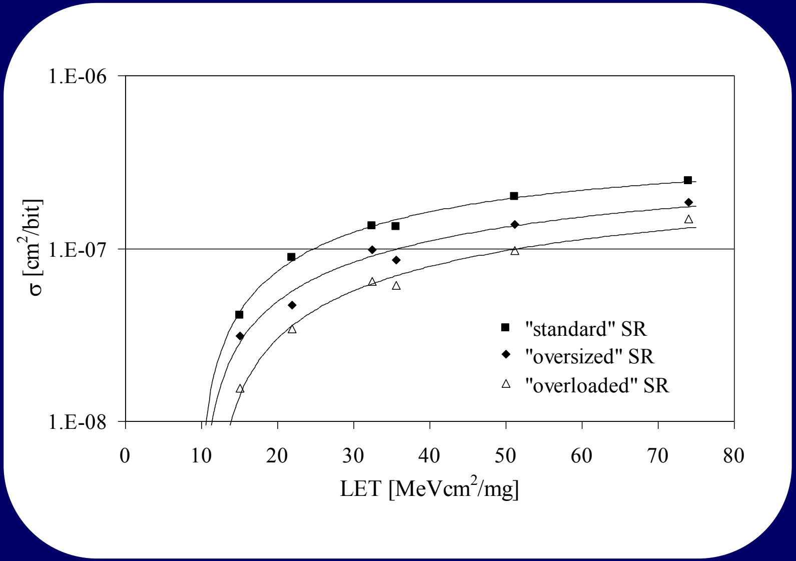


$q = 1.602 \cdot 10^{-19}$ C, electron charge
$\rho_{Si} = 2330$ m/cm^3, silicon density
$E_{eh} = 3.6$ eV, electron-hole pair energy
L_f = funnel length (in the order of 1um (?))

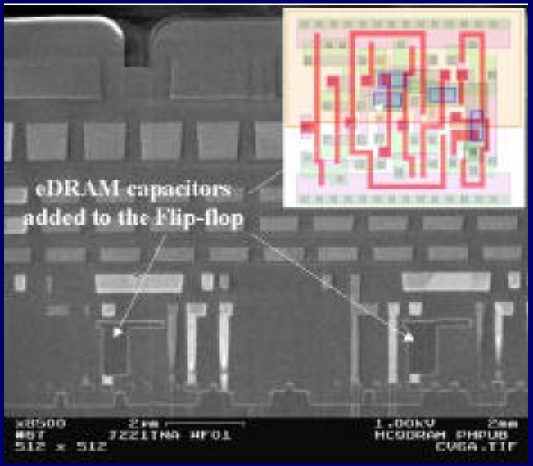
Extra capacitance

- Increase the critical charge by increasing the node capacitance:
  - ❑ Add “extra” capacitors
    - Metal/metal or eDRAM to avoid wasting area
    - Slower cell response





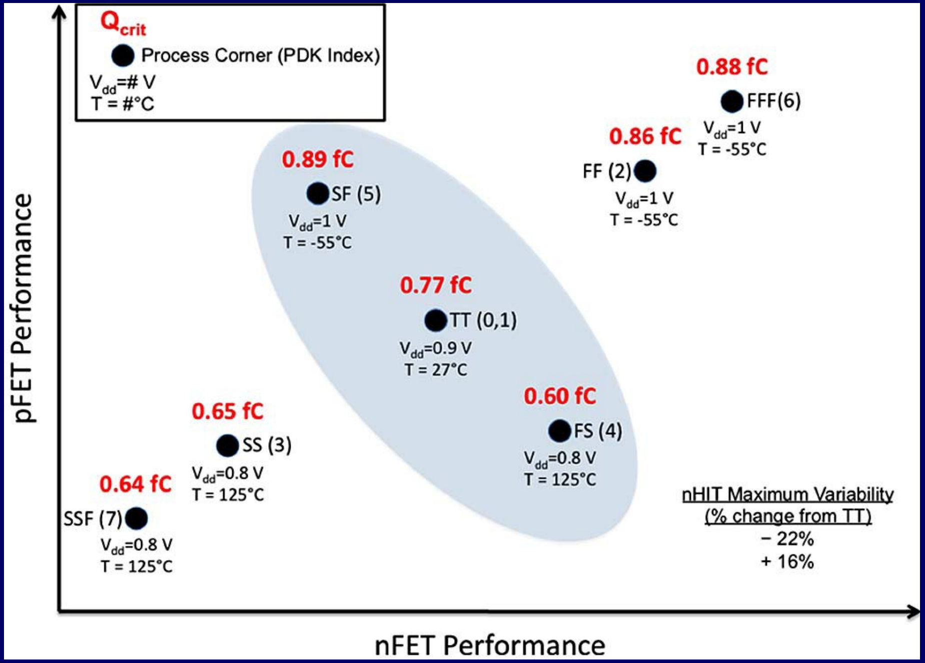
Upset rates in proton environment:
- twofold decrease for the “oversized”
- tenfold decrease for the “overloaded”



Faccio et al., “SEU effects in registers and in a Dual-Ported Static RAM designed in a 0.25 um CMOS technology for application in the LHC”, LECC, Sep. 1999

Parametric variability of critical charge

- Variability in transistors in modern technologies plays an important role in SEU response
  - cross-section/LET plots are a result of the distribution of cell sensitivities
  - Design must take into account worst corner

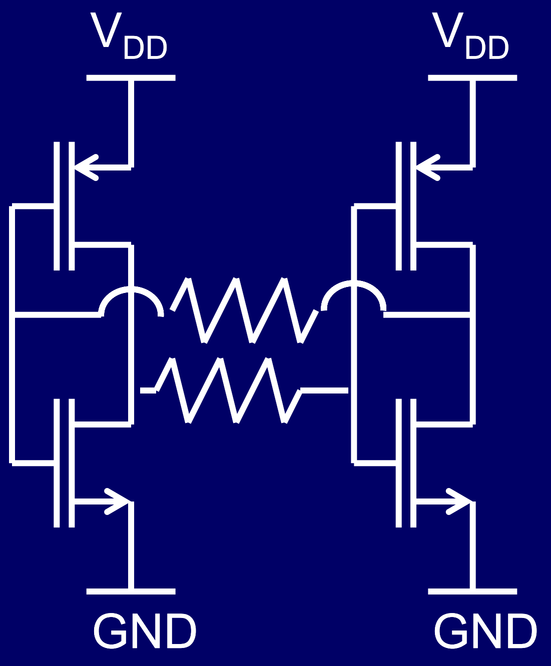


Critical charge values for strikes on nFET inverter in different corners (T, VDD, process PMOS/NMOS)

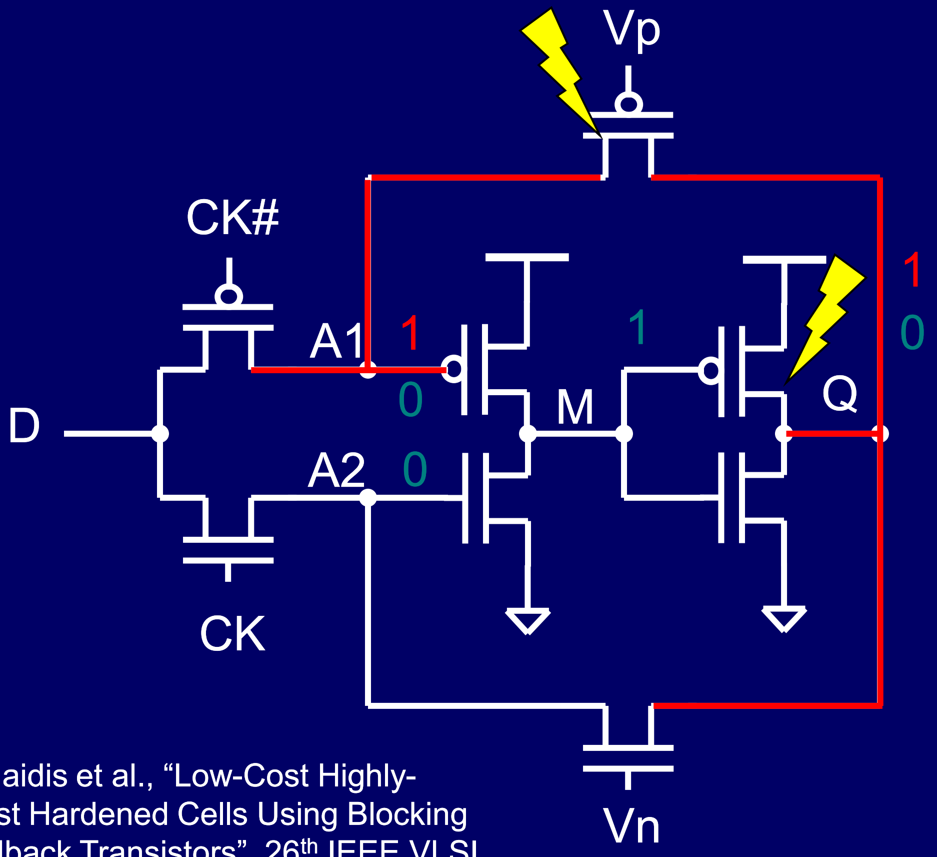
Loveless et al., “Parametric Variability Affecting 45 nm SOI SRAM Single Event Upset Cross-Sections”, IEEE Trans. Nucl. Sci. Vol. 57, No. 6, Dec. 2010

Feedback resistors

- Use special cell architectures
  - SRAM cell with added resistors, delaying the propagation of the perturbation, so allowing the cell to recover its correct state



Weak feedback transistors

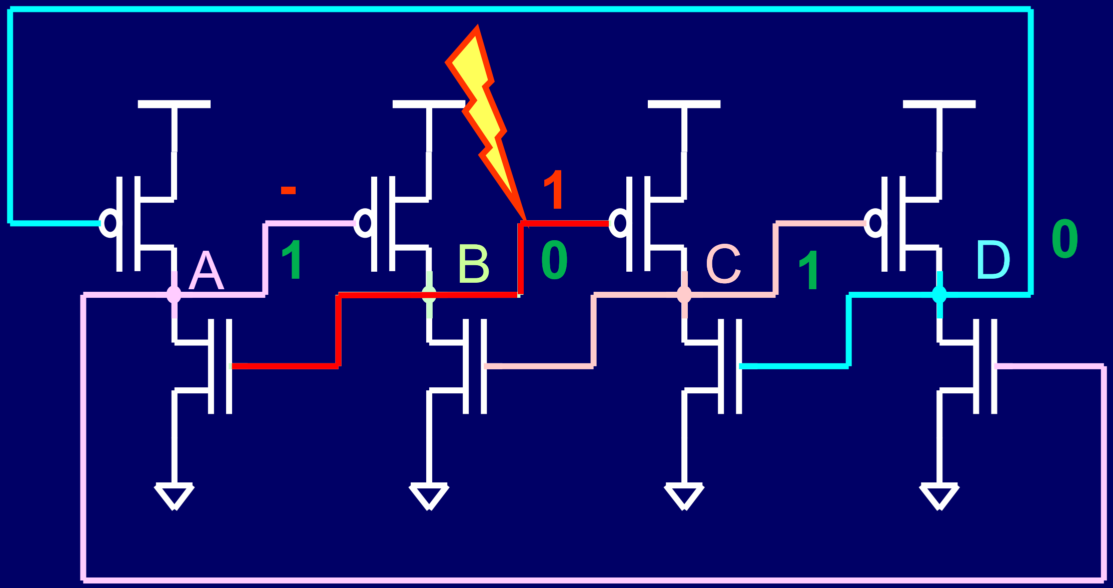


Nicolaidis et al., “Low-Cost Highly-robust Hardened Cells Using Blocking Feedback Transistors”, 26th IEEE VLSI Test Symposium, 2008

- Subthreshold feedback transistors
  - Vp > Vdd – Vtp
  - Vn < Vtn
  - Weak feedback
- Hits on input stage only bring high-Z
  - P-hits can only pull up
  - N-hits can only pull down
- Size of feedback transistor is smaller than feedback resistor

Dual-Interlocked Cell (DICE)

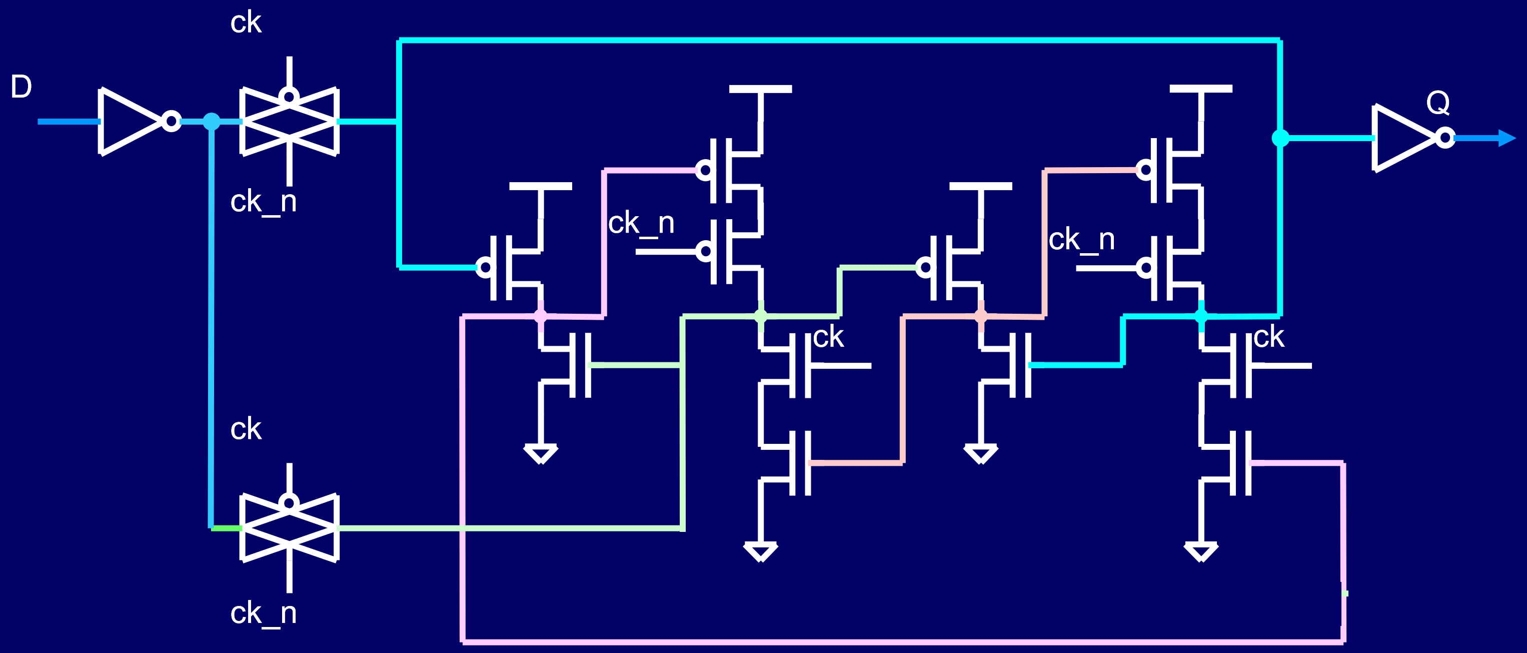
- Each memory node is linked to 2 other memory nodes.
    - Interlock
- Symmetric cell
    - Each node is equivalent to the others
- 2 stable states
- Interlock gives SEU immunity



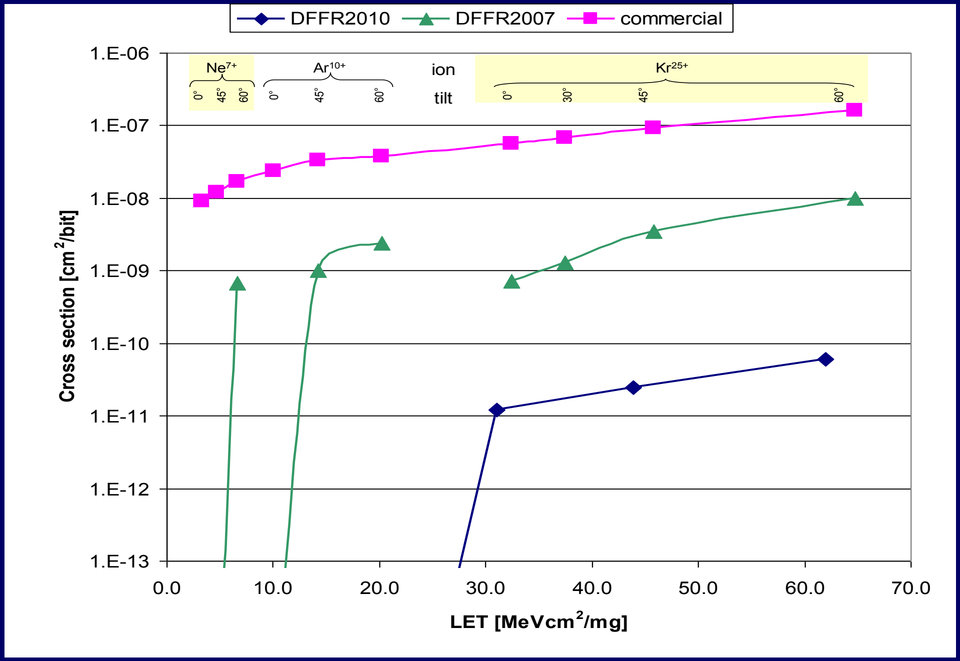
Calin et al., “Upset Hardened Memory Design for Submicron CMOS Technology”, IEEE Tran. Nucl. Sci., vol. 43, no.6, Dec. 1996

DICE latch

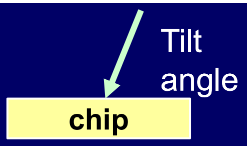
- Writing in the cell requires access to 2 nodes
- Recovery time needed after SEU / output glitch
    - Split input/outputs can avoid SETs
        - SET goes only into 1 of the nodes: it's like an SEU
- Charge collection by multiple nodes is not negligible!



Charge sharing – Multiple-node charge collection



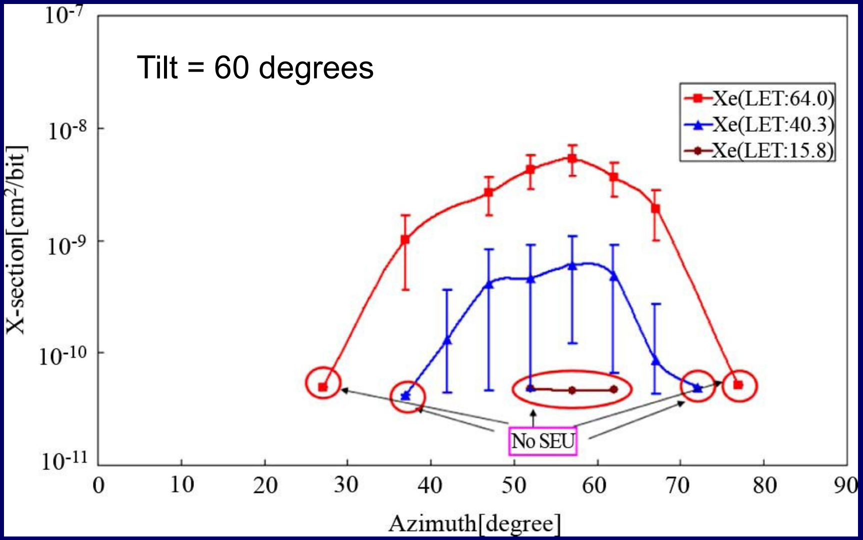
Bonacini, “Design and characterization of an SEU-robust register in 130nm CMOS for application in HEP ASICs”, 2010 JINST 5 C11019


Tilt angle

chip

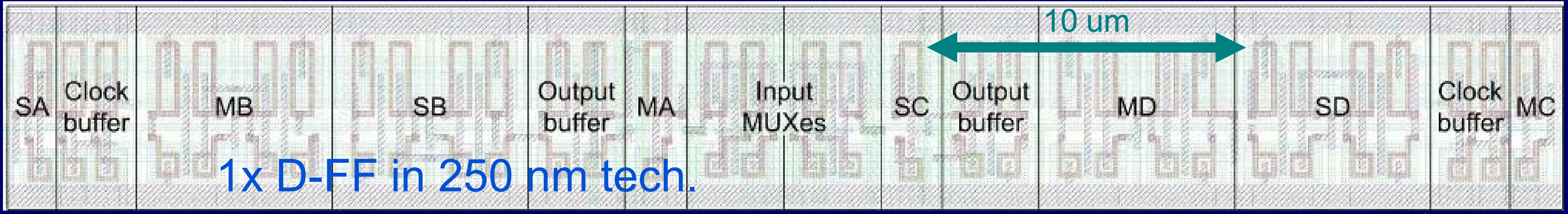
- SEU-hardened cells show a dependence of cross section data on the incidence angle
  - ❑ Due to simultaneous charge collection by multiple nodes
  - ❑ Layout matters !!

Maru et al., “DICE-based flip-flop with SET pulse discriminator on a 90 nm Bulk CMOS Process”, IEEE Tran. Nucl. Sci., vol. 57, no.6, Dec. 2010


Tilt = 60 degrees

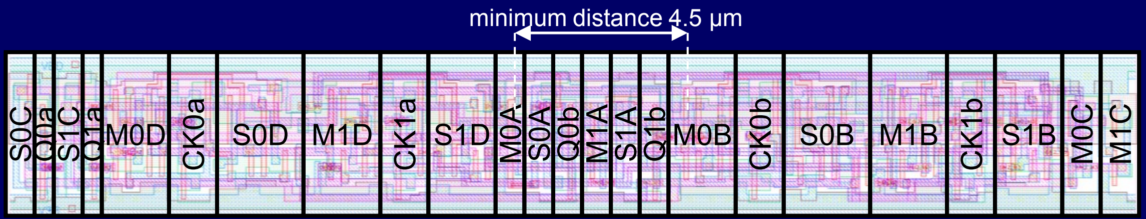
Layouts for SEU-hard FFs

- Sensitive nodes of latch must be separated
  - the more distance, the less probability of charge-sharing
  - waste of area?
- A possible solution is interleaving
  - master and slave nodes of a D-FF…. or even nodes from several D-FFs!
  - Dense interconnect !



1x D-FF in 250 nm tech.

Bonacini et al., "An SEU-Robust Configurable Logic Block for the Implementation of a Radiation-Tolerant FPGA", Trans. Nucl. Sci., vol. 53, no. 6, Dec. 2006



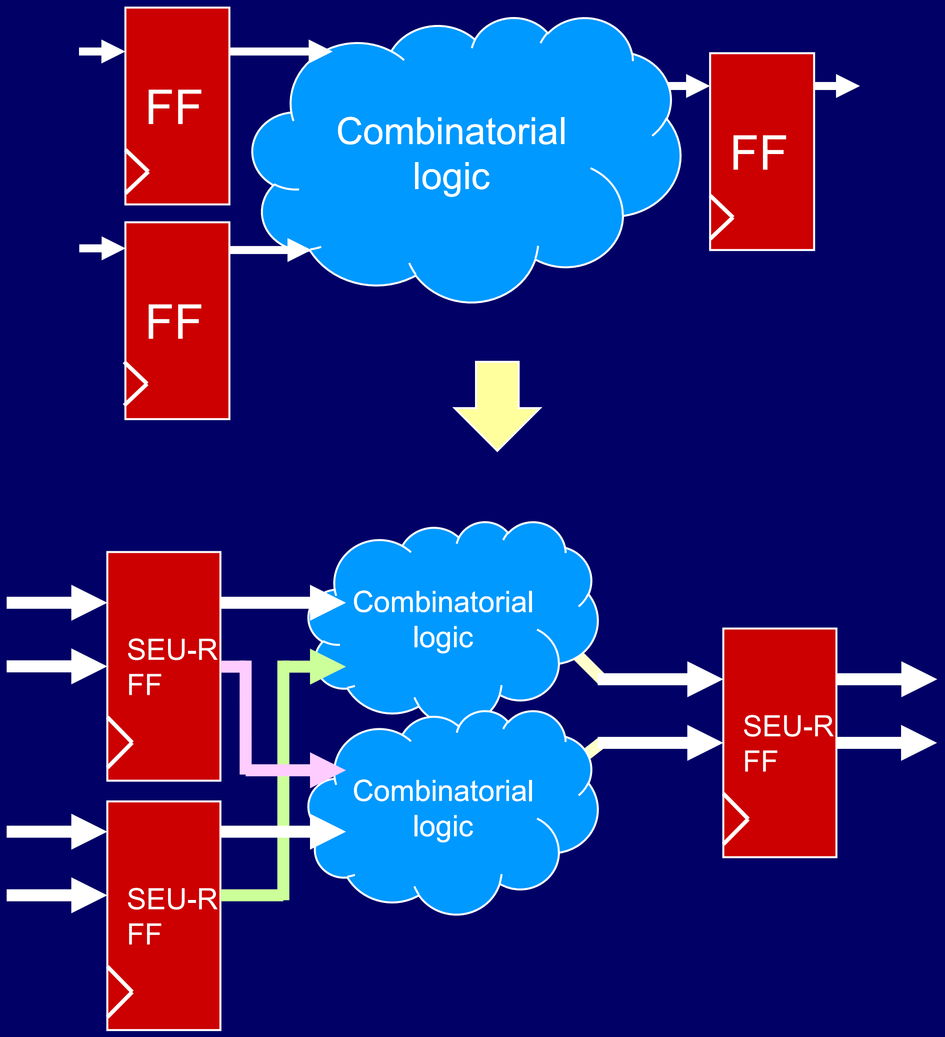
2x D-FF in 130 nm tech.

- Up to level 3 metal used fully for internal cell interconnect
  - Less routing resources for cell-to-cell wiring

Protection from SETs with DICE FFs: Duplication of combinatorial logic

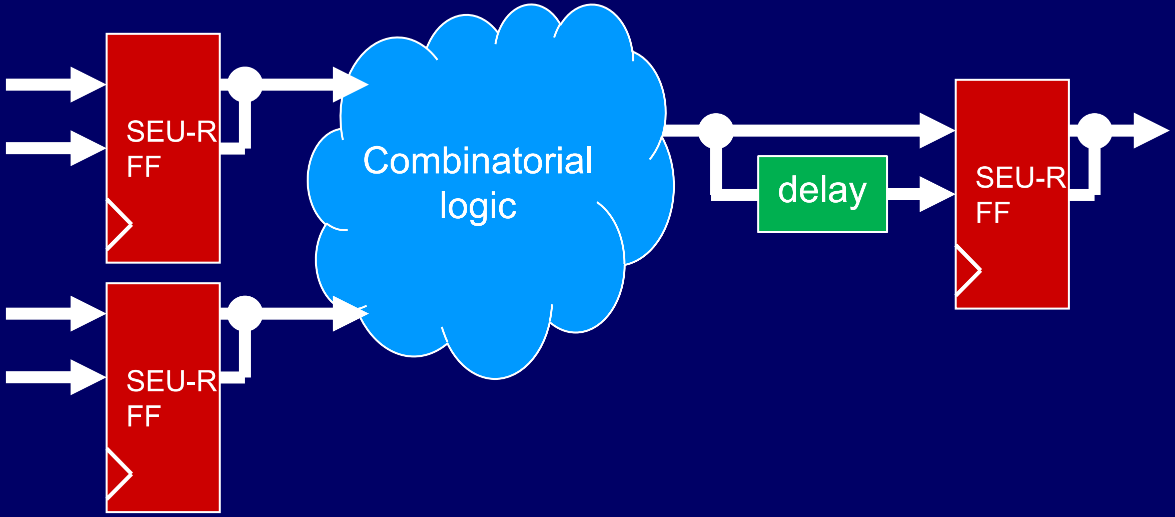
- SEU-robust FF has 2 inputs and 2 outputs
  - Can stand an SEU in one of the two inputs
  - Duplicated combinatorial logic protects from SETs
  - Clocks must be duplicated too !
  - Cannot withstand SET/SEU longer than 1 clock cycle

- With respect to Triple Module Redundancy (TMR):
  - 1x area overhead instead of 2x of TMR
  - …same applies for power consumption
  - Equal speed to TMR
  - **TMR can resist SET/SEUs longer than 1 clock cycle !!**
    - DICE registers cannot: not suitable for frequency >1GHz

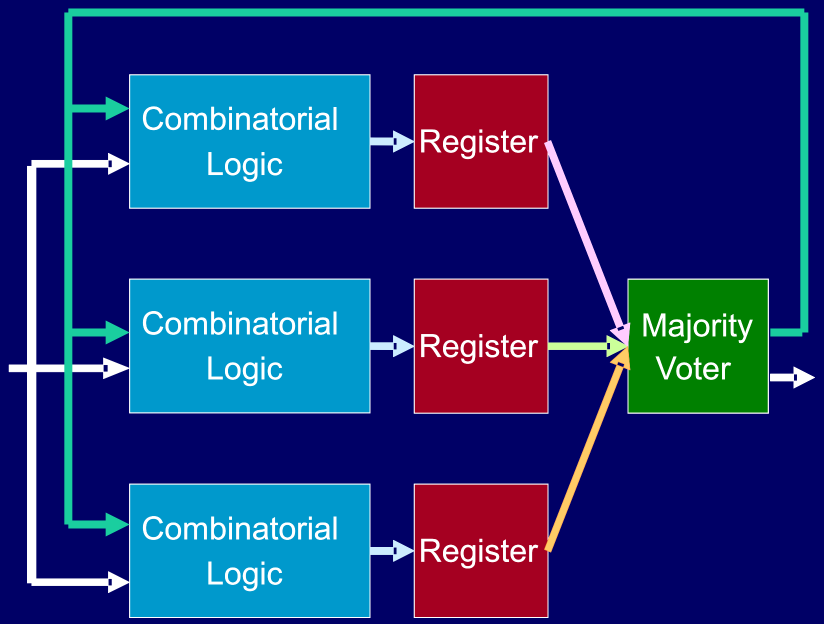


SET protection: time redundancy

- A delay unit can produce a copy of the data input of a FF
  - Redundancy in time domain
  - Timing constraints change to take into account the delay
    - Adds to the propagation delay of the logic
    - Should be ~ 1 ns, in order to account for long SETs
    - …again not suitable for frequency >1GHz



Triple Module Redundancy

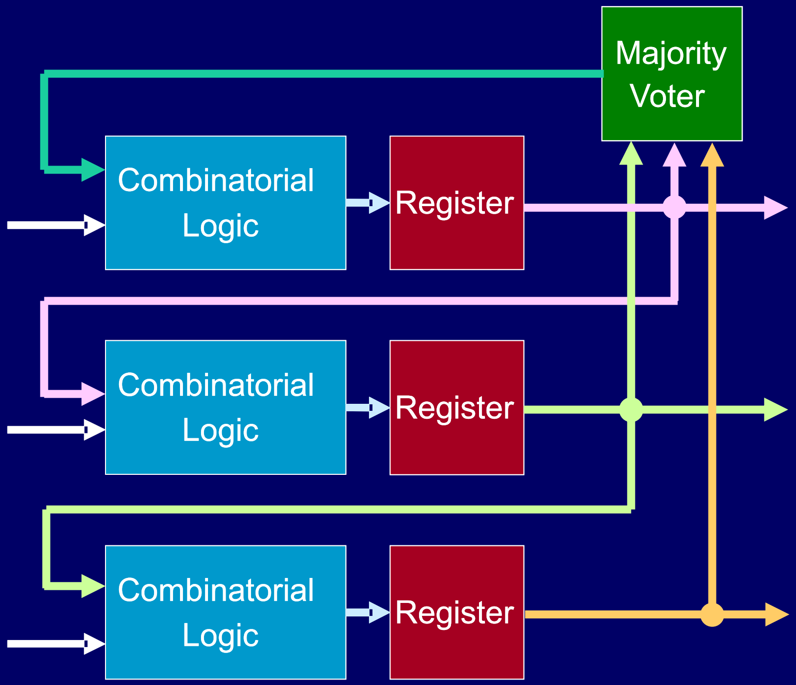


- Triplication with 1 voter

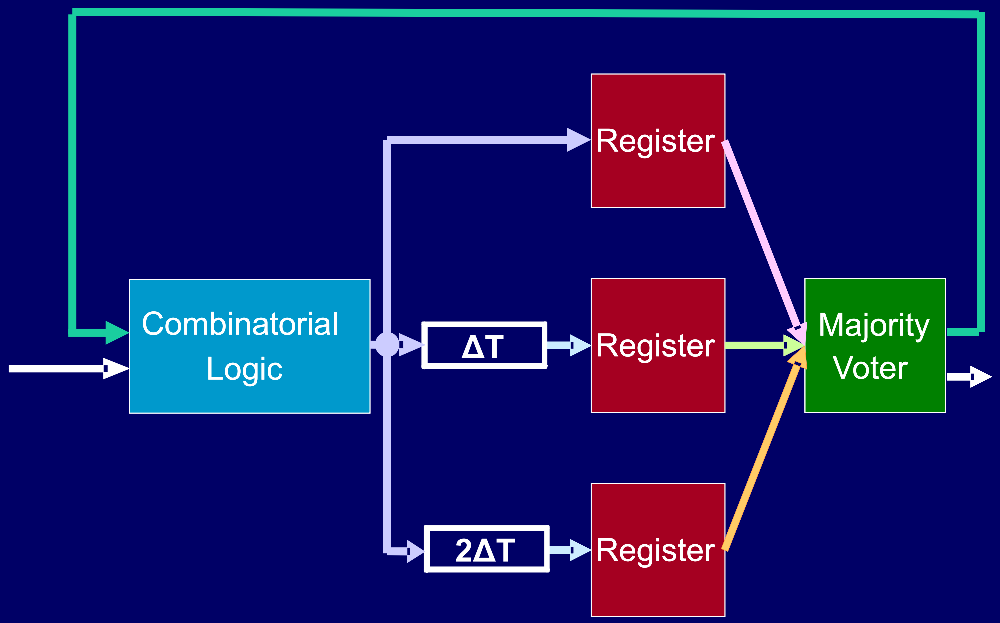
- The state machine is instantiated 3 times, with 1 voter
- An SEU can corrupt the output of one of the blocks, but majority voting restores the correct state
- An error in the voter instead corrupts the state!
  - Not SET free.

Triple Module Redundancy

- SET free structure with 1 voter
  - Voted state propagates to 1 state machine only
  - Wrong state is corrected within 3 clock cycles
- SET/SEUs longer than 1 clock cycle can still potentially bring to wrong state
  - Not suitable for clock frequencies above 1 GHz



Time redundancy with TMR

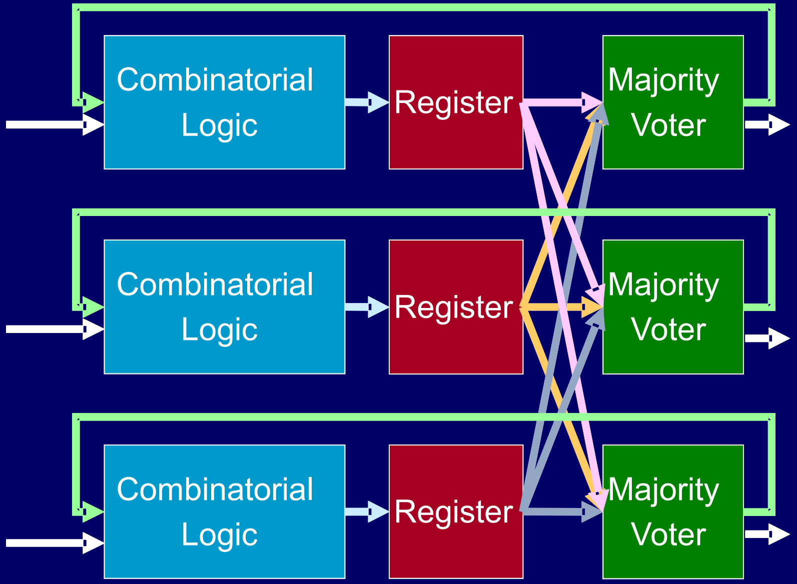


- SET protection with time redundancy
    - Timing constraints must take delay into account

TMR with 3 voters

- Triplication with 3 voters

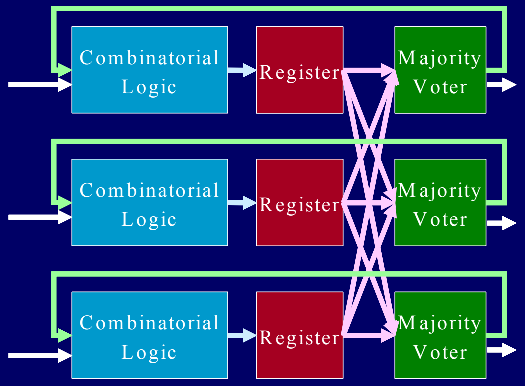
- The state machine is instantiated 3 times, with 3 voters
- An SEU can corrupt the output of one of the blocks, but majority voting restores the correct state
- An error in one of the voters is also restored
- Works at high frequency, can tolerate SEU/SET longer than 1 clock cycle

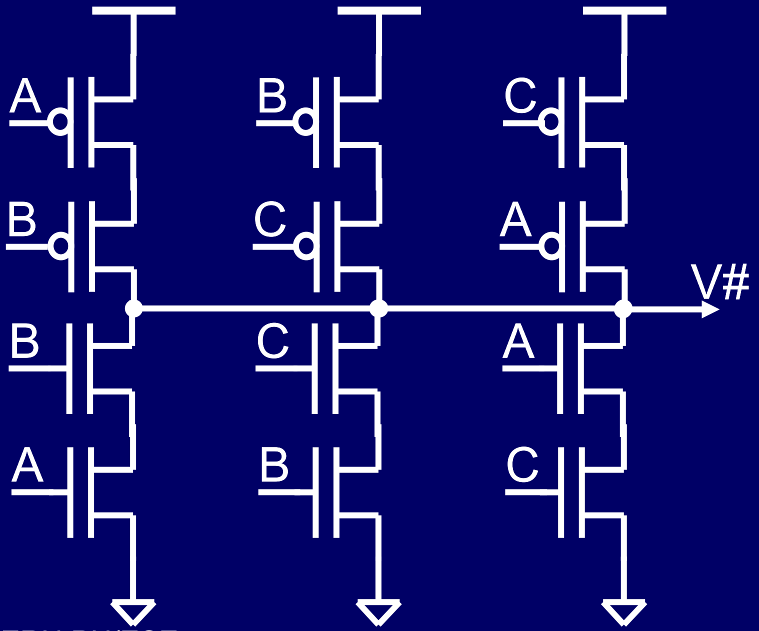


- This seems a lot of additional transistors, but sometimes the designer can find more “compact solutions”. This is shown in the following example.

Voter circuit

- A majority voter can necessitate a large number of transistors. Since 3 voters are required to protect the data from errors in the voter, the overall area penalty can be large
- One compact solution for the voter is in the schematic below:

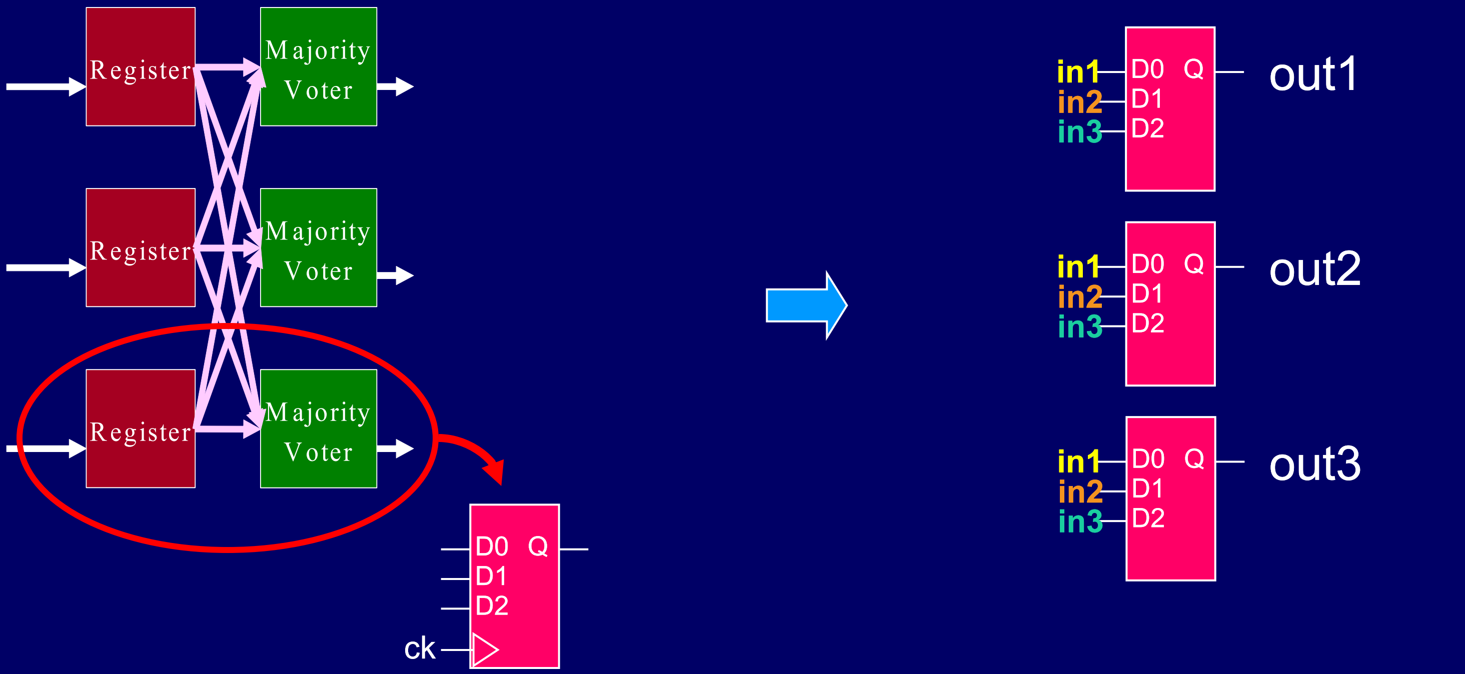




- compact layout
- 12 transistors (better than XOR+MUX style)
- Still 3 voters are needed, for a minimum of 36 transistors

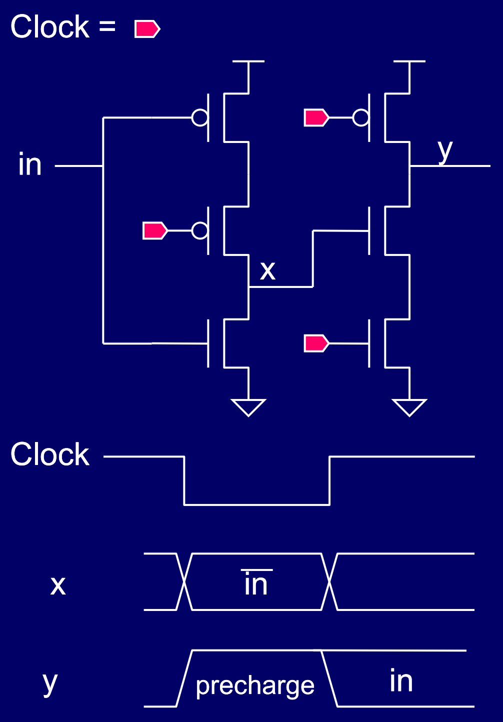
Example of TMR in registers (1)

- To save some transistors, it is possible to merge the voter in the register, and make a single cell
- Each will then have 3 inputs

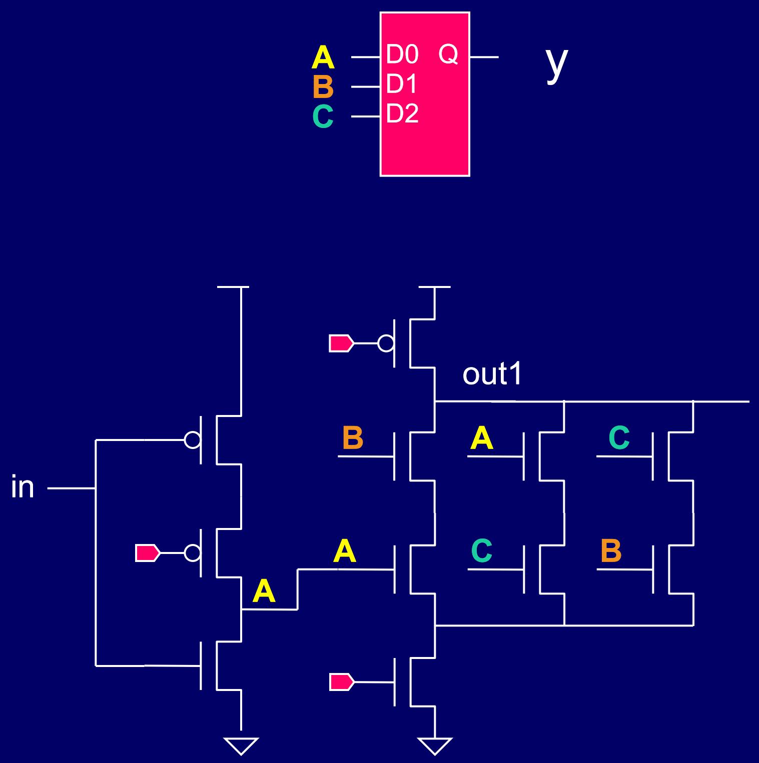


Example of TMR in registers (2)

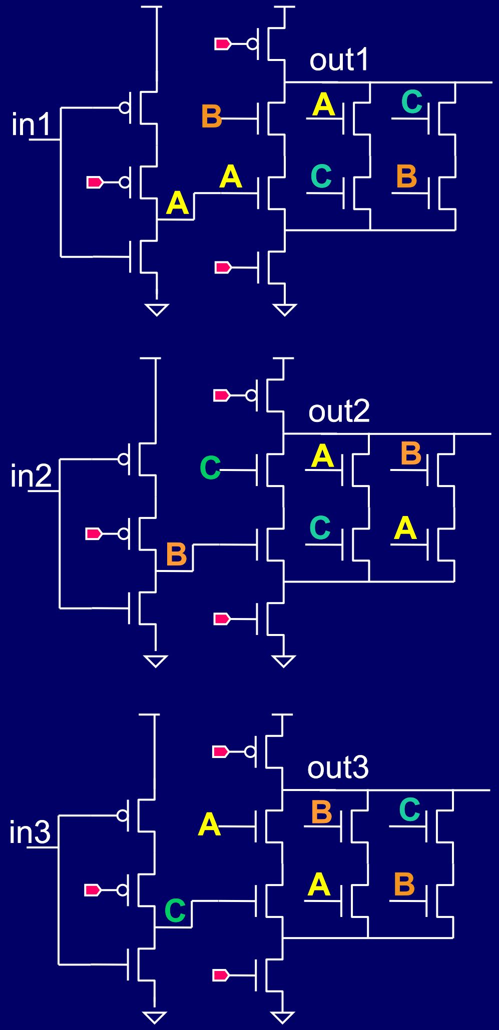
Dynamic latch:



Dynamic latch with voter (1/3):



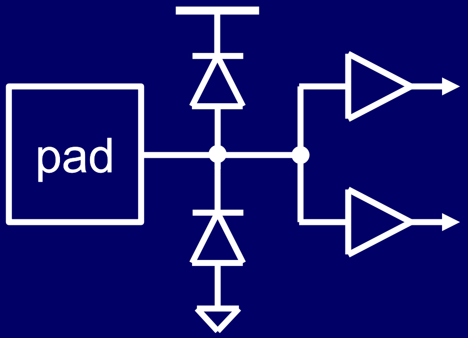
Example of TMR in registers (3)

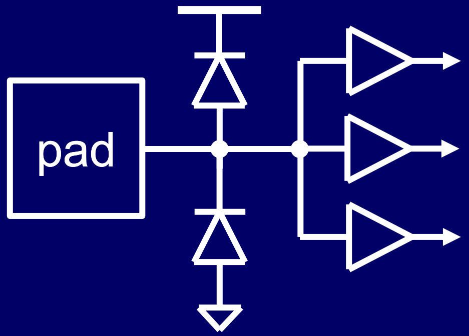


- This integration of the voter in the cell “only” costs 5 additional transistors
  - the penalty for the 3 voters is 15 transistors
- For high speed applications

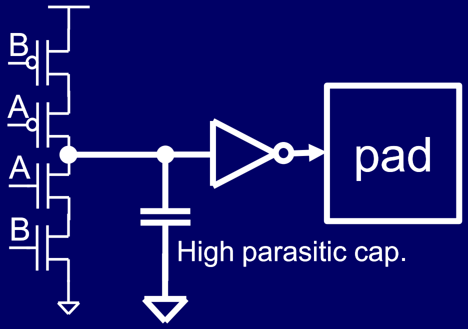
Cobanoglu et al., “A Radiation Tolerant 4.8 Gb/s Serializer for the Giga-Bit Transceiver”, TWEPP, Sep. 2009, Paris

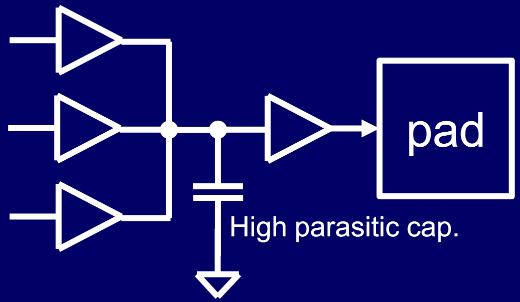
I/O pad buffer SET protection

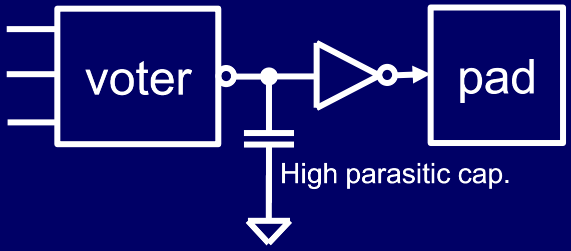




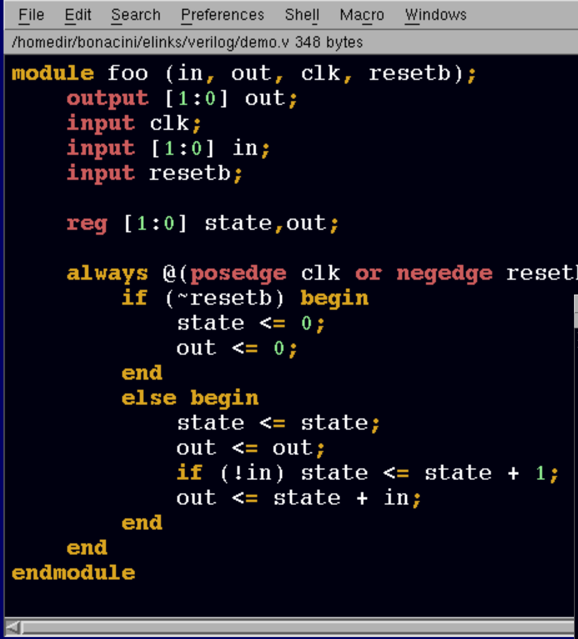
- Protection for I/O buffers might be considered
  - Especially for clock/reset networks
- Easy to duplicate/triplicate an input
- Outputs can use voter or guard-gate as pre-driver
  - Final driver has high parasitic capacitance and should be intrinsically immune to SETs



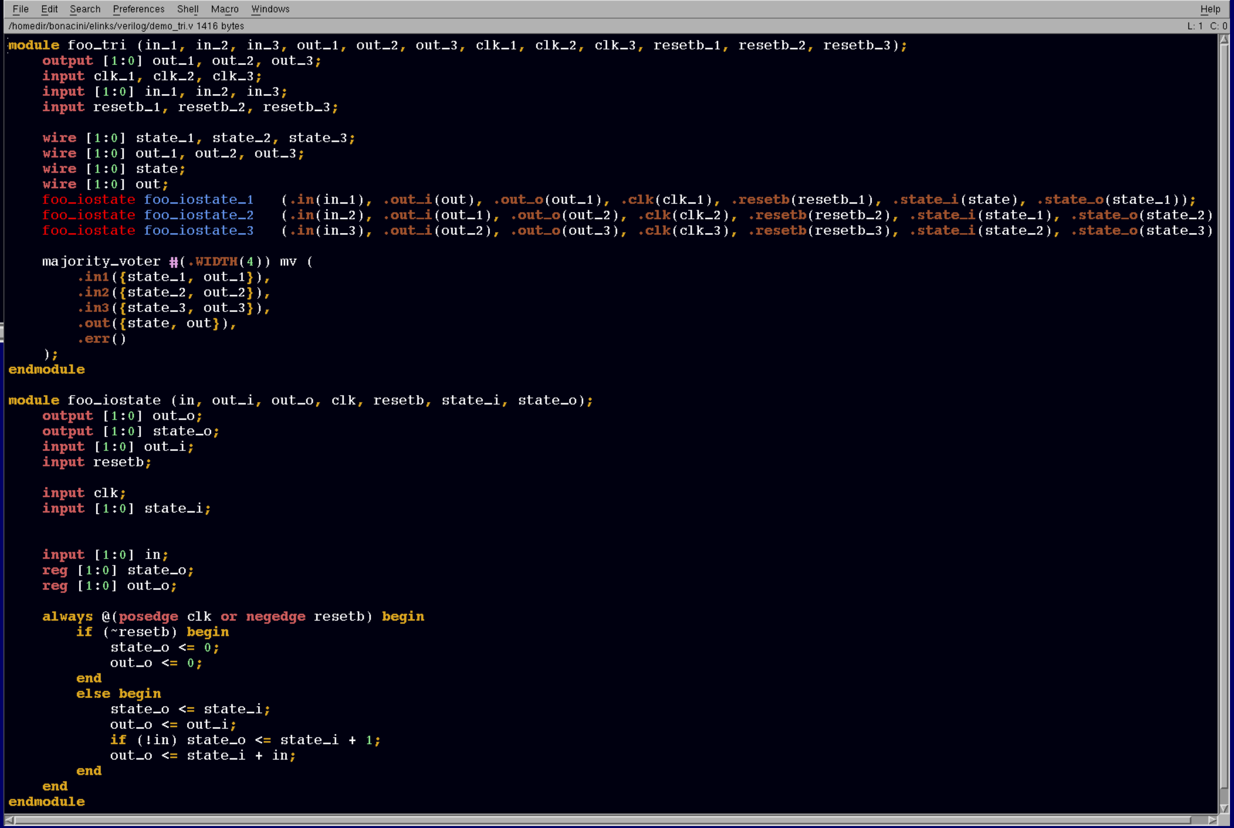




Automatic TMR implementation



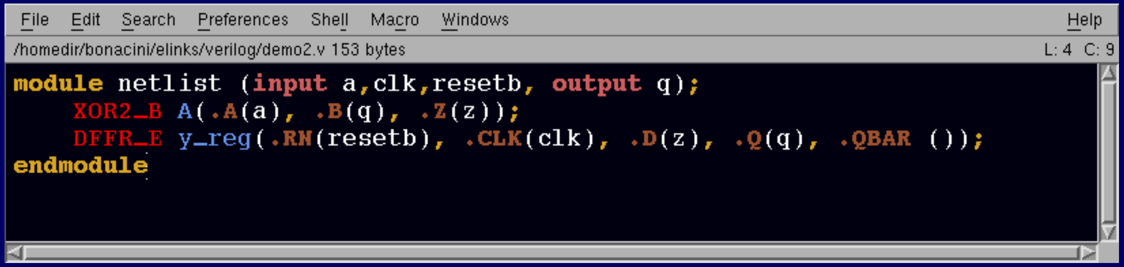
```
module foo (in, out, clk, resetb);
    output [1:0] out;
    input clk;
    input [1:0] in;
    input resetb;

    reg [1:0] state,out;

    always @(posedge clk or negedge resetb) begin
        if (~resetb) begin
            state <= 0;
            out <= 0;
        end
        else begin
            state <= state;
            out <= out;
            if (!in) state <= state + 1;
            out <= state + in;
        end
    end
endmodule
```



```
module foo_tri (in_1, in_2, in_3, out_1, out_2, out_3, clk_1, clk_2, clk_3, resetb_1, resetb_2, resetb_3);
    output [1:0] out_1, out_2, out_3;
    input clk_1, clk_2, clk_3;
    input [1:0] in_1, in_2, in_3;
    input resetb_1, resetb_2, resetb_3;

    wire [1:0] state_1, state_2, state_3;
    wire [1:0] out_1, out_2, out_3;
    wire [1:0] state;
    wire [1:0] out;
    foo_iostate foo_iostate_1   (.in(in_1), .out_i(out), .out_o(out_1), .clk(clk_1), .resetb(resetb_1), .state_i(state), .state_o(state_1));
    foo_iostate foo_iostate_2   (.in(in_2), .out_i(out_1), .out_o(out_2), .clk(clk_2), .resetb(resetb_2), .state_i(state_1), .state_o(state_2)
    foo_iostate foo_iostate_3   (.in(in_3), .out_i(out_2), .out_o(out_3), .clk(clk_3), .resetb(resetb_3), .state_i(state_2), .state_o(state_3)

    majority_voter #(.WIDTH(4)) mv (
        .in1({state_1, out_1}),
        .in2({state_2, out_2}),
        .in3({state_3, out_3}),
        .out({state, out}),
        .err()
    );
endmodule

module foo_iostate (in, out_i, out_o, clk, resetb, state_i, state_o);
    output [1:0] out_o;
    output [1:0] state_o;
    input [1:0] out_i;
    input resetb;

    input clk;
    input [1:0] state_i;


    input [1:0] in;
    reg [1:0] state_o;
    reg [1:0] out_o;

    always @(posedge clk or negedge resetb) begin
        if (~resetb) begin
            state_o <= 0;
            out_o <= 0;
        end
        else begin
            state_o <= state_i;
            out_o <= out_i;
            if (!in) state_o <= state_i + 1;
            out_o <= state_i + in;
        end
    end
endmodule
```

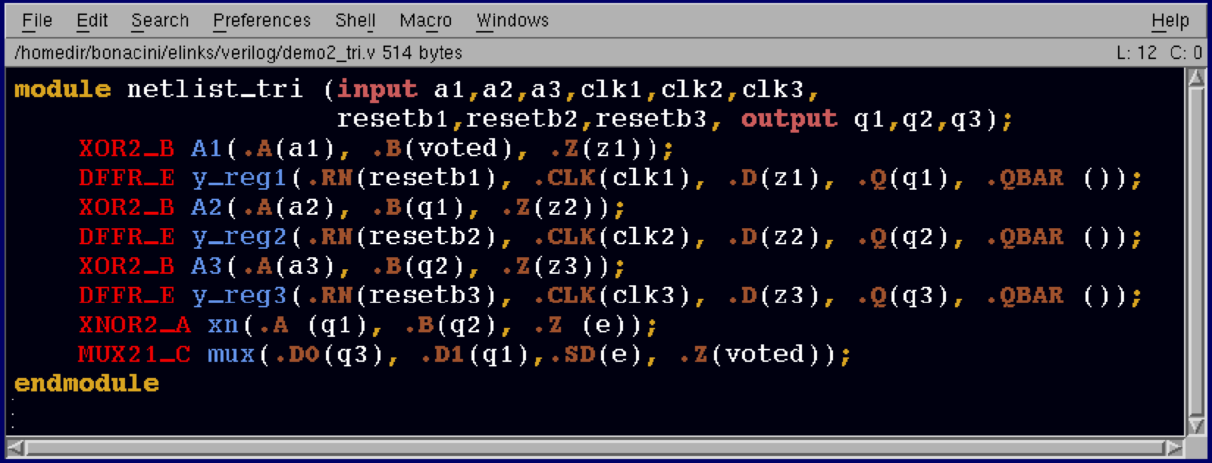
- Easy to implement TMR with a script on the RTL code

Automatic TMR implementation



/homedir/bonacini/elinks/verilog/demo2.v 153 bytes L: 4 C: 9

```
module netlist (input a,clk,resetb, output q);
    XOR2_B A(.A(a), .B(q), .Z(z));
    DFFR_E y_reg(.RN(resetb), .CLK(clk), .D(z), .Q(q), .QBAR ());
endmodule
```



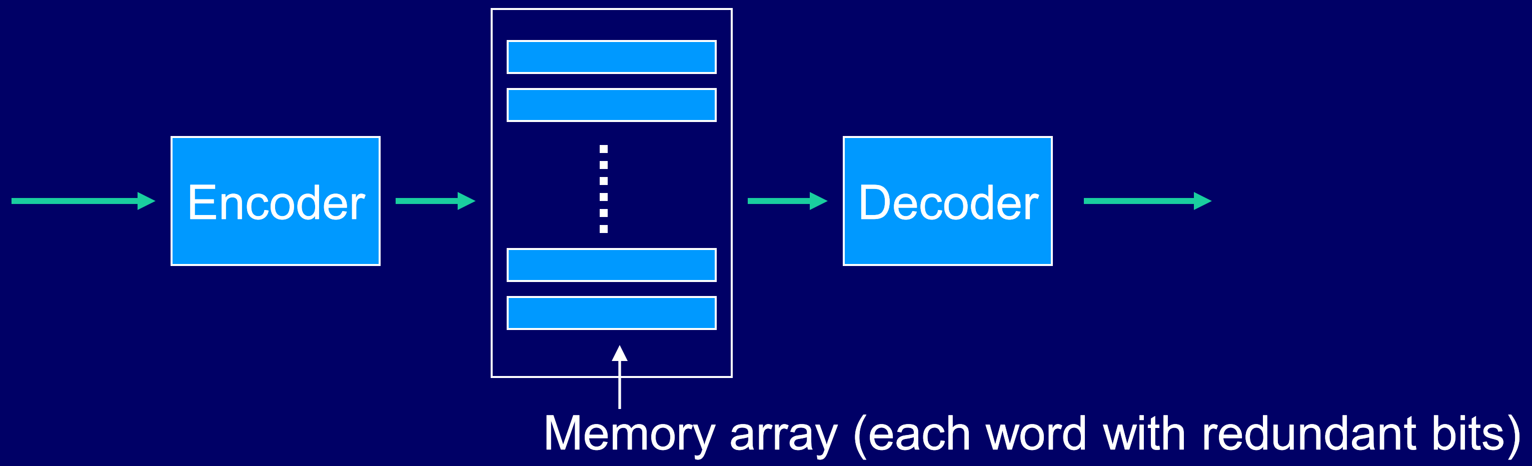
/homedir/bonacini/elinks/verilog/demo2_tri.v 514 bytes L: 12 C: 0

```
module netlist_tri (input a1,a2,a3,clk1,clk2,clk3,
                     resetb1,resetb2,resetb3, output q1,q2,q3);
    XOR2_B A1(.A(a1), .B(voted), .Z(z1));
    DFFR_E y_reg1(.RN(resetb1), .CLK(clk1), .D(z1), .Q(q1), .QBAR ());
    XOR2_B A2(.A(a2), .B(q1), .Z(z2));
    DFFR_E y_reg2(.RN(resetb2), .CLK(clk2), .D(z2), .Q(q2), .QBAR ());
    XOR2_B A3(.A(a3), .B(q2), .Z(z3));
    DFFR_E y_reg3(.RN(resetb3), .CLK(clk3), .D(z3), .Q(q3), .QBAR ());
    XNOR2_A xn(.A (q1), .B(q2), .Z (e));
    MUX21_C mux(.D0(q3), .D1(q1),.SD(e), .Z(voted));
endmodule
```

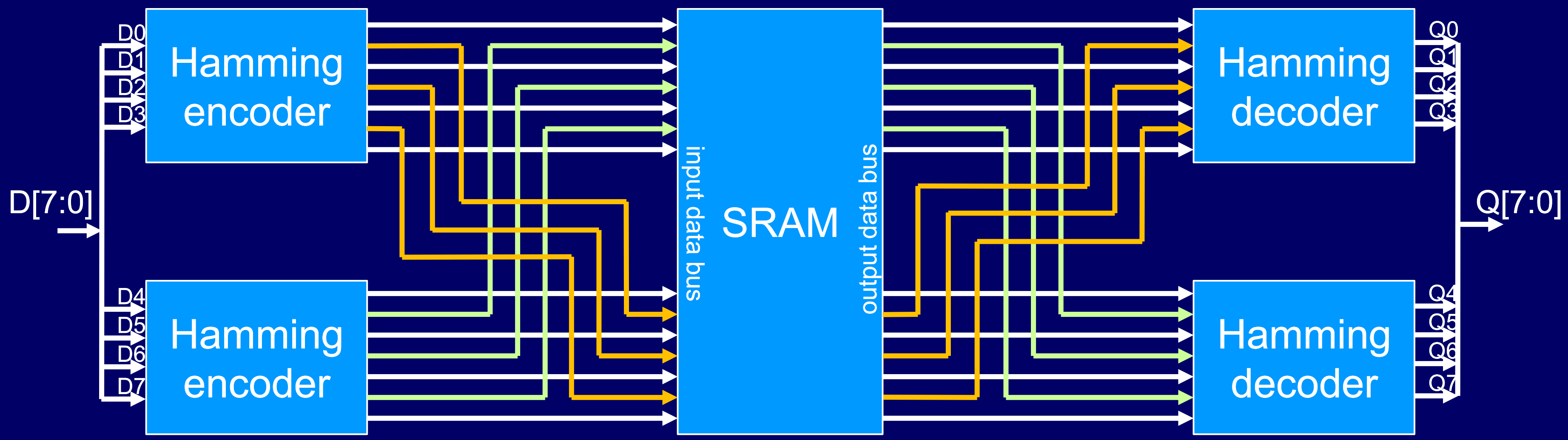
- … even easier on synthesized netlist
- But customized for cell library

Redundancy: encoding

- Adding redundant information (bits) and encoding-decoding
  - Used for data transmission and for memories
  - Requires complex encoding-decoding logic
  - Several different codes can be used (Hamming, Reed-Solomon, BCH, etc.)



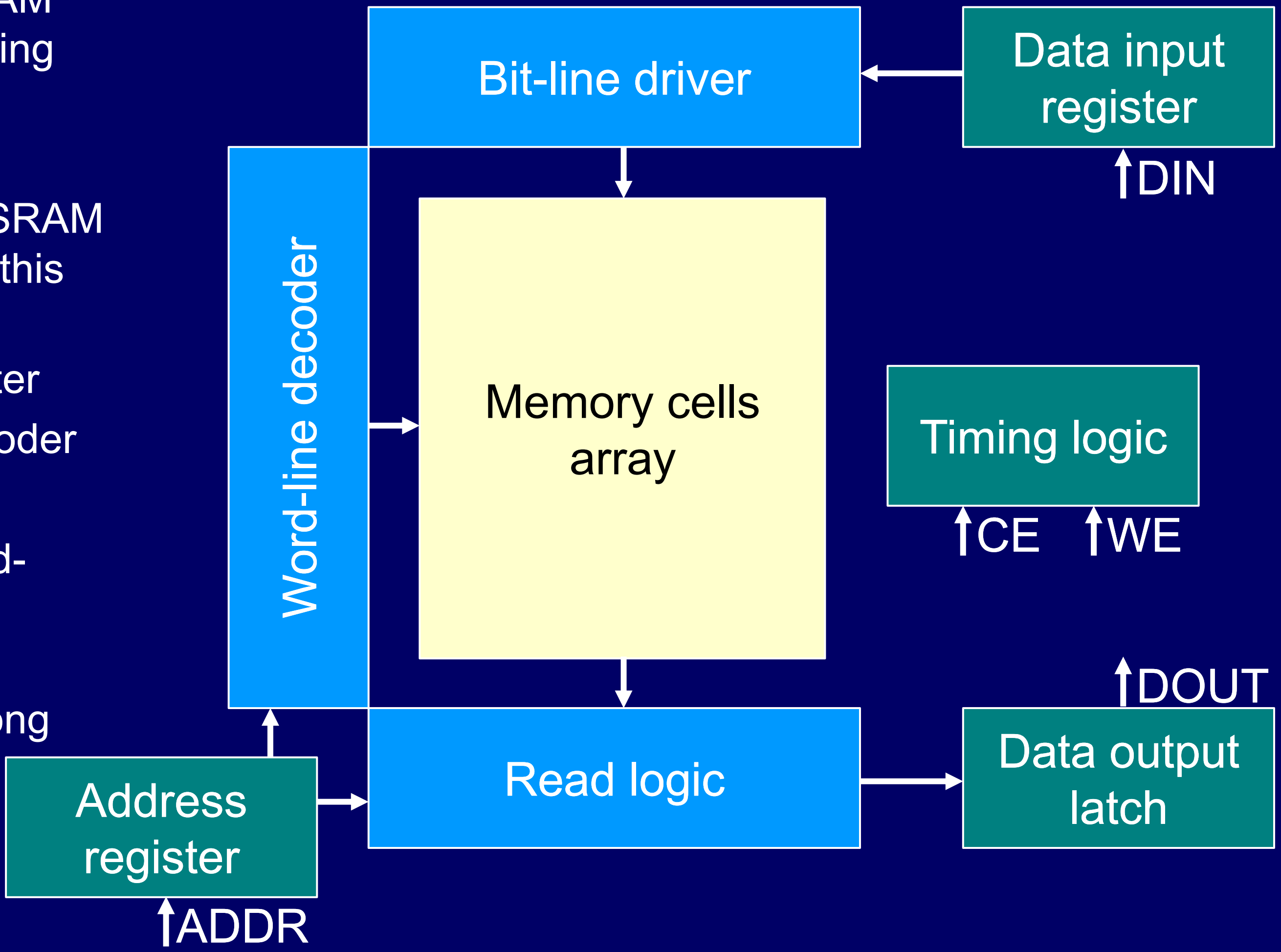
Protection of SRAM data



- Multiple Bit Upsets (MBUs) can corrupt Hamming-encoded data
  - Single error correction, double error detection
  - Interleaving 2 Hamming-coded words can alleviate this problem
    - Ok for 2-bit upsets in neighboring cells
  - … Reed Solomon codes or other if not sufficient …

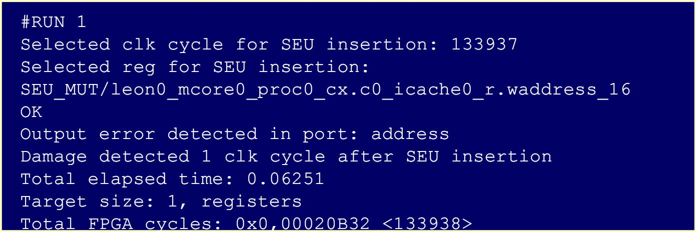
Protection of SRAMs

- Easy to protect SRAM content with Hamming encoding or other
- Some parts of the SRAM are not covered by this kind of protection
  - Address register
  - Word-line decoder
  - Timing logic
- capacitance of word-lines sufficient for protection?
- ...might write to wrong address!
- Use CRC for FIFOs ?

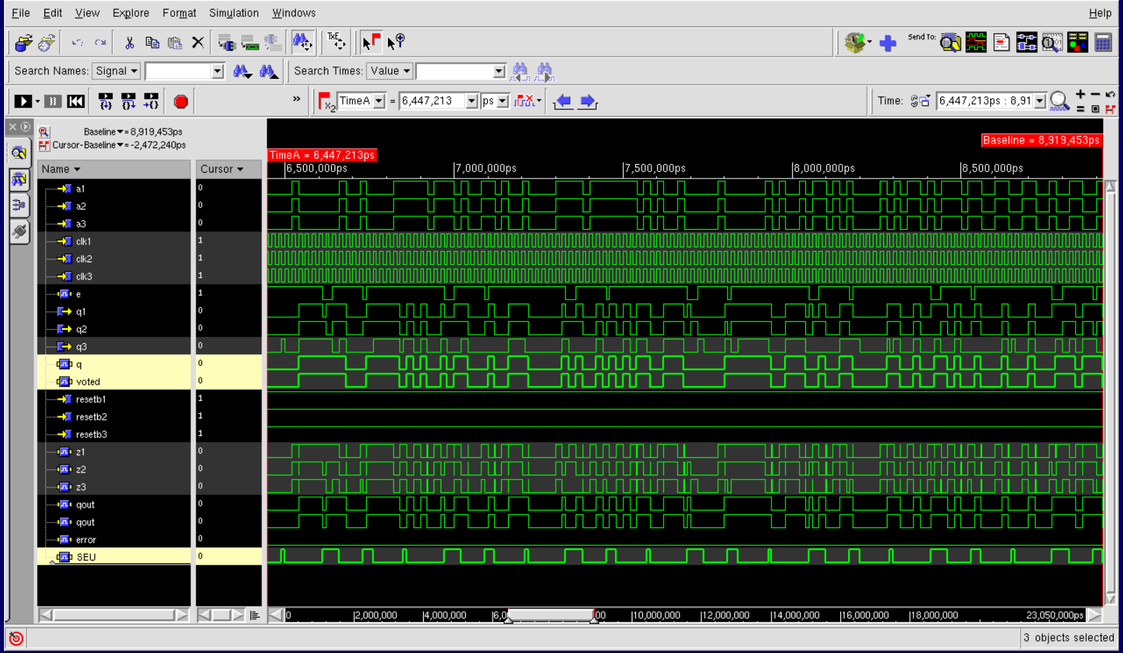


SEU simulation, emulation

- In the case of TMR SEUs / fault injections can be simulated with HDL
- ...or emulated (ex. FT-UNSHADES)



```
#RUN 1
Selected clk cycle for SEU insertion: 133937
Selected reg for SEU insertion:
SEU_MUT/leon0_mcore0_proc0_cx.c0_icache0_r.waddress_16
OK
Output error detected in port: address
Damage detected 1 clk cycle after SEU insertion
Total elapsed time: 0.06251
Target size: 1, registers
Total FPGA cycles: 0x0,00020B32 <133938>
```



- For simulation of SEU-hardened cells it’s necessary to use SPICE/Spectre
  - Current-source on the sensitive node

Further reading

- General material on radiation effects:
  - The best source is the “archive of Radiation Effects Short Course Notebooks, 1980-2002” collecting the courses given at the IEEE NSREC conference (CD sold by IEEE)

- On SEU-tolerant Cells:
  - Increased capacitance:
    - F.Faccio et al., “Single Event Effects in Static and Dynamic Registers in a 0.25mm CMOS Technology”, IEEE Trans. Nucl. Science, Vol.46, No.6, pp.1434-1439, December 1999
    - F.Faccio et al., “SEU effects in registers and in a Dual-Ported Static RAM designed in a 0.25mm CMOS technology for applications in the LHC”, in the proceedings of the Fifth Workshop on Electronics for LHC Experiments, Snowmass, September 20-24, 1999, pp.571-575 (CERN 99-09, CERN/LHCC/99-33, 29 October 1999)
    - P.Roche, F.Jacquet, C.Caillat, J.P.Schoellkopf, "An Alpha Immune and Ultra Low Neutron SER High Density SRAM", proceedings of *IRPS 2004*, pp671-672, April 2004
    - Calin et al., “Upset Hardened Memory Design for Submicron CMOS Technology”, IEEE Tran. Nucl. Sci., vol. 43, no.6, Dec. 1996
    - Bonacini, “Design and characterization of an SEU-robust register in 130nm CMOS for application in HEP ASICs”, 2010 JINST 5 C11019
    - Bonacini et al., “An SEU-Robust Configurable Logic Block for the Implementation of a Radiation-Tolerant FPGA”, Trans. Nucl. Sci., vol. 53, no. 6, Dec. 2006
    - Nicolaidis et al., “Low-Cost Highly-robust Hardened Cells Using Blocking Feedback Transistors”, 26th IEEE VLSI Test Symposium, 2008
    - Bessot, Velazco, “Design of SEU-Hardened CMOS Memory Cells: the HIT Cell”, RADECS, Sep. 1993
    - Liu & Whitaker, “Low Power SEU Immune CMOS Memory Circuits”, IEEE Trans. Nucl. Sci., vol. 39, no. 6, Dec, 1992

- On TMR and encoding:
  - Paper comparing techniques and containing references, to be used as a starting research point: S.Niranjan, J.F.Frenzel, “A comparison of Fault-Tolerant State Machine Architectures for Space-Borne Electronics”, IEEE Trans. On Reliability, Vol.45, No1, p.109, March 1996