Chip Level Implementation Techniques for

Radiation Hardened Microprocessors

by

Chandarasekaran Ramamurthy

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved November 2013 by the
Graduate Supervisory Committee:

Lawrence T. Clark, Chair
Keith E. Holbert
Hugh J. Barnaby
David Mayhew

ARIZONA STATE UNIVERSITY

December 2013

ABSTRACT

Microprocessors are the processing heart of any digital system and are central to all the technological advancements of the age including space exploration and monitoring. The demands of space exploration require a special class of microprocessors called radiation hardened microprocessors which are less susceptible to radiation present outside the earth's atmosphere, in other words their functioning is not disrupted even in presence of disruptive radiation. The presence of these particles forces the designers to come up with design techniques at circuit and chip levels to alleviate the errors which can be encountered in the functioning of microprocessors. Microprocessor evolution has been very rapid in terms of performance but the same cannot be said about its rad-hard counterpart. With the total data processing capability overall increasing rapidly, the clear lack of performance of the processors manifests as a bottleneck in any processing system. To design high performance rad-hard microprocessors designers have to overcome difficult design problems at various design stages i.e. Architecture, Synthesis, Floorplanning, Optimization, routing and analysis all the while maintaining circuit radiation hardness. The reference design 'HERMES' is targeted at 90nm IBM G process and is expected to reach 500Mhz which is twice as fast any processor currently available.

Chapter 1 talks about the mechanisms of radiation effects which cause upsets and degradation to the functioning of digital circuits. Chapter 2 gives a brief description of the components which are used in the design and are part of the consistent efforts at ASUVLSI lab culminating in this chip level implementation of the design. Chapter 3 explains the basic digital design ASIC flow and the changes made to it leading to a

rad-hard specific ASIC flow used in implementing this chip. Chapter 4 talks about the triple mode redundant (TMR) specific flow which is used in the block implementation, delineating the challenges faced and the solutions proposed to make the flow work. Chapter 5 explains the challenges faced and solutions arrived at while using the top-level flow described in chapter 3. Chapter 6 puts together the results and analyzes the design in terms of basic integrated circuit design constraints.

TABLE OF CONTENTS

Page

LIST OF TABLES....vii

LIST OF FIGURES..viii

CHAPTER

1. INTRODUCTION ...1

1.1. Radiation Environment around the Earth ... 2

1.2. Charge Deposition and Soft Error Mechanisms effecting circuits ... 5

1.2.1. SEE ... 5

1.2.2. TID... 10

1.3. Radiation hardening to mitigate radiation upsets and failures... 12

1.3.1. Shielding... 12

1.3.2. Radiation hardened by process (RHBP)... 13

1.3.3. Radiation hardened by design (RHBD)... 14

1.4. Summary... 22

2. CUSTOM BLOCKS... 24

2.1. Cache... 24

2.2. Clockspine Design ... 29

2.3. Register File... 31

2.4. Summary 35

3. ASIC DESIGN FLOWS 36

3.1. Conventional ASIC design Flow 37

3.1.1. Synthesis 37

3.1.2. Physical Implementation: Floor-plan and Power-plan 39

3.1.3. Placement 40

3.1.4. Pre-CTS optimization: 41

3.1.5. Clock Tree Synthesis 42

3.1.6. Post-CTS optimization 43

3.1.7. Routing 43

3.1.8. Post-route Optimization 44

3.2. RHDB Design Flow: TMR Flow 44

3.2.1. Synthesis 47

3.2.2. Floorplanning 49

3.2.3. Placement 50

3.2.4. Optimization 52

3.2.5. Clock Tree Synthesis 53

3.2.6. Post CTS optimization 54

3.2.7. Triplication 54

3.2.8. Powerplan 56

3.2.9. Routing 57

3.2.10. Extraction and Timing 58

3.2.11. Abstract Generation 59
3.2.12. Physical Verification 60
3.3. Summary 61
4. TMR FLOW IMPROVEMENTS. 63
4.1. Pin Placement algorithm 63
4.2. Clockgater Insertion 68
4.3. Power-plan Alignment 70
4.4. Antenna Fixing 71
4.5. Clock Tree Synthesis 73
4.6. Summary 76
5. TOP LEVEL IMPLEMENTATION 77
5.1. Synthesis 78
5.2. Floor-planning 80
5.3. Placement 81
5.4. Pre-CTS optimization 84
5.5. Clock Tree Synthesis 85
5.6. Post-CTS optimization 90
5.7. Routing Phase 91
5.8. Post-Route Optimization 93

5.9. Summary 95

6. CONCLUSIONS………. 96

6.1. Summary 106

LIST OF TABLES

Table Page

1.1 Available rad-hard microprocessor designs……..…………..………………………...1

4.1 Clock skew summary for TMR blocks. 75

5.1 DMR and TMR clock tree statistics at the top-level.. 88

6.1 Timing and area summary. 96

6.2 Standard cell distribution. 98

6.3 Power space statistics at 450MHz using dhrystone vectors………………………...104

LIST OF FIGURES

Figure Page

1.1 Solar and galactic constituents of Earth's Magnetosphere. (Courtesy Nasa.gov)......... 2

1.2 Van Allen belts around Earth consisting of proton and electron belts. 3

1.3 Particle Strike showing track of charge funnel in a p-n junction................................ 7

1.4 Band diagram of TID degradation mechanism.. 10

1.5 Cross section of Inverter showing TID effects. 12

1.6 Majority Voter showing Triple mode redundancy.. 14

1.7 Temporal sampling with delta delays. 15

1.8 Multi Node charge collection without (a) and with well separation (b). 16

1.9 Double height cells showing spacing between critical NMOS devices in comparison with a single height cell of the same area. 17

1.10 A TMR Multi-bit flip-flop interleaving with spacer for spatial separation of pipelines. 18

1.11 A 5 stage DMR pipeline after [Cla2011]. 19

1.12 Redundancy scheme using DMR (Blue), TMR (Red), crossover logic (green) and custom blocks (Yellow) in HERMES. 20

1.13 Annular Gates eliminating field oxide and drain diffusion interface after [Pettit 2009]. 22

2.1 Block Diagram of the Hermes 16KB, 4-way set associative cache........................... 26

2.2 Floor plan of RHBD Cache designed for IBM 90nm process after........................... 27

2.3 Cache abstract alignment fix.. 28

2.4 A RHBD Clockspine design on IBM 90nm process after [Che2012]. 30

2.5 A DMR register file block diagram after [Cla2011]. 31

2.6 (a) Register File cell schematic, (b) Cell layout (c) Critical node separation, after [Cla2011]. 32

2.7 Register file layout with Interleaving shown. 34

3.1 A Conventional ASIC design Flow. 37

3.2 A TMR ASIC Flow. 46

3.3 In-house RC tool flowchart. 47

3.4 Synthesis Output Summary. 48

3.5 Trial placement to determine area on the left and tool generated placement on the right. 49

3.6. TMR floorplan diagram examples showing separation of 3 standard heights but varying widths. 50

3.7 Diagram showing the area highlighted (bold) available for the placement step. 51

3.8 After single redundant placement stage.. 52

3.9 Optimization results setup histogram and critical path. 52

3.10 Clock Tree Synthesized design with clock routing only. 53

3.11 Post-CTS hold fixed with buffer cells in blue 54

3.12 Step 2 GUI of the in-house tool 55

3.13 TMR placement generated from the tool and loaded with A, B and C copy and flip-flops highlighted in white. 56

3.14 Powerplan created for the TMR floorplan. 57

3.15 Routing shown for block with 5 metal layers. 58

3.16 Abstract generator used for creating layout abstract LEF file. 59

3.17 Drc verified using Calibre, Antenna violations present will be fixed at the top level 60

3.18 Calibre LVS used to verify layout versus schematic. 61

4.1 Top level drc's without pin placement program generated .lefs. 63

4.2 Pin placement algorithm flowchart. 64

4.3 Initial tool placement and initial script based placement seed input for script. 65

4.4 Single Redundant placement output from pin placement program. 66

4.6 Figure for initial seed to the script and output from script on the right 67

4.5 Triple redundant pins output from pin placement algorithm 67

4.7 Top level routed design using pin placement program generated .lefs. 68

4.8 Clockgater cell schematic explicitly coded in the RTL. 69

4.9 Powerplan TMR cell alignment. 70

4.10 Antenna violation highlighted in pink in Calibre DRC. 71

4.11 ANTENNA Calculation options and setting in the abstract generator. 72

4.12 Portion of Lef file with Antenna fixing information. 73

4.13 CTS cells with CTS only route and CTS cells in pink. 74

4.14 Symmetrical Clock Tree routing shown for all 3 logic copies in TMR block. 75

5.1 Hierarchical Modules in HERMES along with the macro blocks 77

5.2 Output summary from synthesis. 78

5.3 Floorplan showing placement of all macros in the design. 79

5.4 Shows a power-plan with TMR blocks connected by overlap and other blocks getting via towers for power from a Metal 8 -metal 7 grid. 81

5.5 DMR A (red) copy and B (blue) copy separation at placement stage. 82

5.7 Module segregation... 83

5.6 A (red) and B (blue) separation with 2 rows of decap(green). 83

5.8 Pre-CTS optimization affected cells (yellow).. 84

5.9 WNS Histogram at pre-CTS stage. 85

5.10 Clock spine CTS cartoon for skew balancing. 87

5.11 Clock Tree synthesis cells used in the design. 88

5.13 DMR Clock Tree latency histogram. 89

5.12 All the clock routing shown (DMR and TMR). 89

5.14 Design Clock Tree latency histogram. 90

5.15 PostCTS hold cells highlighted in orange. 91

5.16 Metal density in the routing phase. 92

5.17 Cells added by optimization and the corresponding A and B separation 94

6.2 Pie chart showing cell distribution.. 97

6.1 Worst timing critical path spanning multiple blocks. 97

6.3 Design overall density by stage. 98

6.4 DMR A and B copy density growth by stage. 99

6.6 Setup WNS for each stage. 100

6.5 . Failing endpoints for setup violations. 100

6.8 Hold endpoints by stage. 101

6.7 Setup TNS by stage. 101

6.10 Hold TNS by stage. 102

6.9 Hold WNS by stage 102

6.11 A and B copy of a same critical path highlighted separated by 500um. 103

6.12 Switching power density map from primetime showing module contributions. 104

6.14 Leakage Power map from primetime showing module contributions. 105

6.13 Total Power map from primetime showing module contributions. 105

CHAPTER 1. INTRODUCTION

Radiation particles of concern to the operation of any digital circuit, and consequently to a microprocessor, mainly are alpha particles, protons and cosmic rays consisting of heavy ions. This chapter will explain briefly the kind of particles and mechanisms involved in upsetting the normal operation of processors [Bar2003], resulting in machine state upsets [Ngu2005]. Design techniques for hardening digital circuits against short-term and long-term effects are also explained below in detail.

Research Implementations						
Design	Owned by	Power	Voltage(v)	Speed(Mhz)	Hardness Tested	Comments
A3300	Sandia National Labs	60mW/Mhz	4.5	10	Gamma dose of 5Mrad , LET upset threshold 28 MeV/mg/cm2	16/32 bit based on National semiconductors NS32C016 (1989)
Rad-Hard -H32	National Space Agency Japan	2W total	4.5	20	total Dose of 1.3KGy(Si)	32 bit Microprocessor(1994)
RAD600	Lockheed Martin	3W total	3.3	33	Total Dose : 2Mrad @ 170rad(si)/sec LINAC:3Grad(Si)/s Single Event: 154 Mevcm2/mg LET	32 bit RISC Microprocessor 35 MIPS throughput
RAD750	BAE Systems	6W total	2.5	133	Total Dose 1Mrad(Si) SEU immunity 1E-10 upsets/ bit-day	240 MIPS based on PowerPC 750
Commercial Chips						
Design	Owned by	Power	Voltage(v)	Speed(Mhz)	Hardness Tested	Comments
Mongoose V	Synova	n/a	n/a	15	LET > 80 MeV- cm2 / mg	radiation hardened MIPS R3000 32-bit microprocessor
Hx1750	Honeywell	n/a	5	40	Total Dose .1Mrad(Si) SER < 1x10-5 Errors/1750 days	16bit microprocessor
HXRHPPC	HoneyWell	7.6W	3.3	80	TID>300Krad(Si) SER<1.5X10-5 upsets/processor-day	32 bit RISC microprocessor based on PowerPC architechture
SPARC V8	Atmel	.7W 8mW/MIPS	1.65-1.95 Core	100	Total Dose:>300Krad , Single Event : 95Mev/mg/cm2	32bit Microprocessor 90MIPS @ 100Mhz
RAD750	IBM	2.3-14.5W @132Mhz	2.5	200	TID=1Mrad(Si) SEU <1.6E-10 errors/bit-day	>400MIPS @ 200Mhz , 32 bit microprocessor ,

Table 1.1 Available rad-hard microprocessor designs.

Circuit design technique tradeoffs have been studied and the reasons for choosing triple mode redundant (TMR) logic design techniques are also been elucidated. Table 1.1 lists the available radiation hardened processors where a clear lack of speed is evident.

*1.1. **Radiation Environment around the Earth***

Fig. 1.1 shows the solar and galactic constituents affecting earth's magnetosphere. The radiation profile around the earth comprises of 4 basic radiation sources:

1. Plasma
2. Trapped Particles
3. Solar Particles
4. Galactic Cosmic Rays

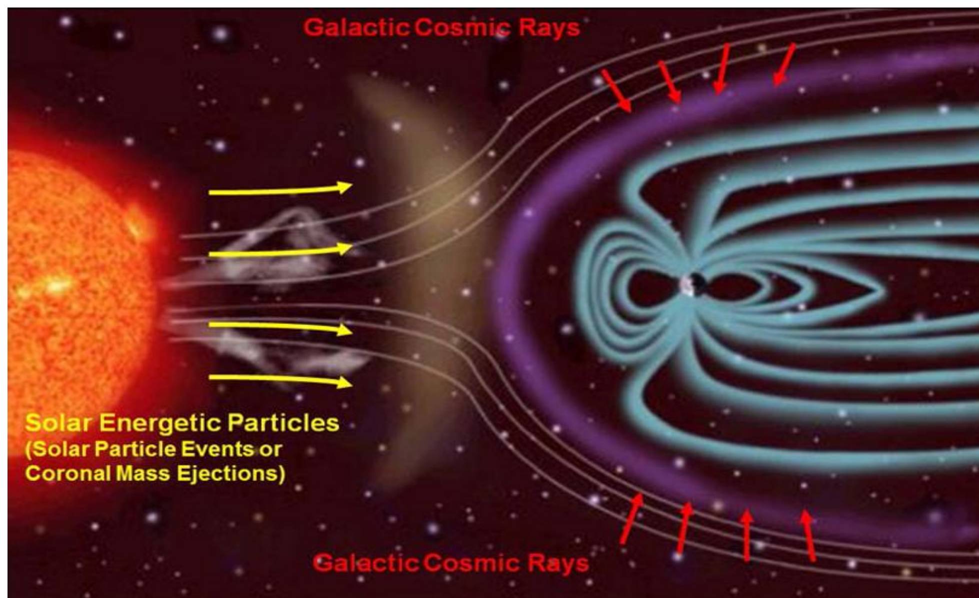

Fig. 1.1 Solar and galactic constituents of Earth's Magnetosphere. (Courtesy Nasa.gov)

Plasma is the neutral cloud of charges in the space which is controlled by earth's magnetosphere in the vicinity of earth. It contains electrons and ions at relatively low energy levels compared to ionizing radiation around earth. It is a discontinuous region with varying radiation and energy constituents. These regions are named as

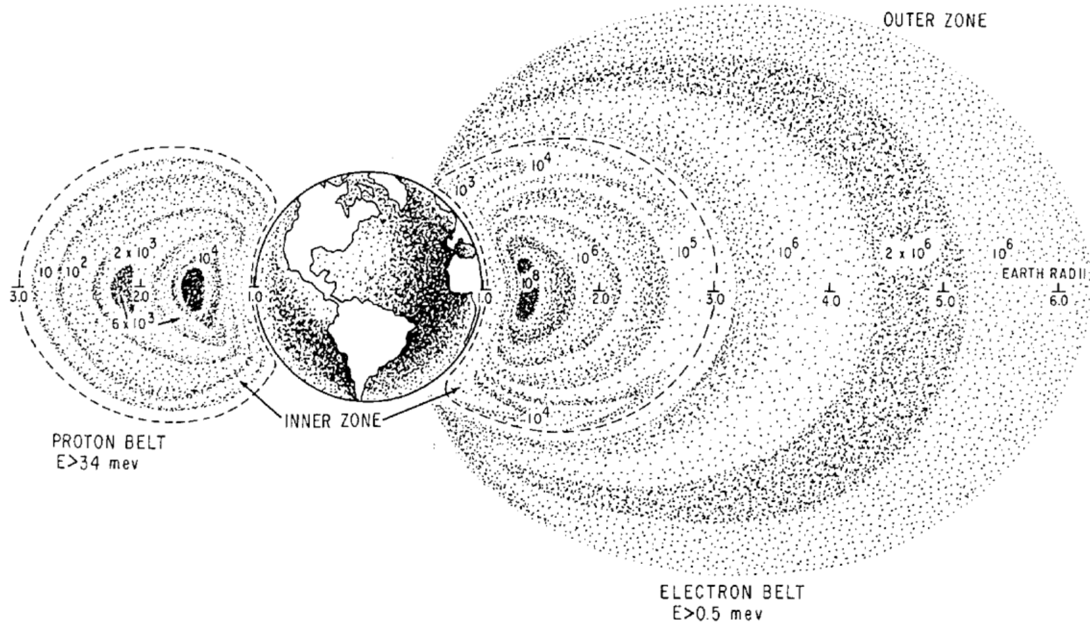

Fig. 1.2 Van Allen belts around Earth consisting of proton and electron belts.

Ionosphere, Magnetosphere, Magnetosheath and Interplanetary Space. The potentials on these regions usually increase from a few tenths of a volt to a few volts. Plasma is hugely dependent on the solar cycles since the cycles can create huge magnetic fluxes that cause spikes in radiation levels. Plasma is not known to cause any significant radiation effects on the functioning of digital circuitry.

Trapped particles generally refer to the particles trapped in earth's Van Allen radiation belt [Sta1988] and primarily comprise of protons trapped in the inner belt and electrons trapped in the outer belt as shown in Fig. 1.2. Electrons are present in the inner belt but their low energy(less than 7MeV) as compared to the protons (500MeV) renders them less harmful. The inner and outer belts are described in terms of the radial distance from earth (RE). Radial distances (L) greater than 2.5RE constitute the outer belt and those less than that constitute the inner belt. The electron flux and energy in the outer belt is significantly higher than that in the inner belt and can normally be as high as 7MeV [Gus1996]. It can increase by several orders of magnitude during magnetic storms rendering them more dangerous during such events.

Solar Particles are the byproducts of solar flare cycles [Bar2003] that follow an eleven year period. Earth's magnetic field provides great protection from these flares. Nevertheless, a variety of particles still manage to enter the earth's atmosphere. The solar flares are comprised chiefly of protons (around 90%) with alpha particles, heavy ions and electrons making up the rest. Protons from a solar flare have energies from the range of 10Mev to 1 GeV. These high energy protons are very critical as they have high fluence and shielding against them is very important to reduce the total dose impact on any device in space. Solar activity, i.e. the concentration of protons and electrons, varies based on the solar cycle (solar min or max). It is important to note that heavy ions from solar flares have less flux than those from galactic cosmic rays.

Galactic cosmic rays (GCR) as the name suggests are particles carried to the solar system from external galactic sources. The atmosphere shields the earth from these galactic rays just as it does against solar flares, but their effects intensify as we reach farther from the equator in terms of inclination and also higher in altitude. The galactic particle concentration is inversely proportional to the solar flare based particle concentration because the latter is boosted by solar maximums of solar flare cycles. GCRs comprise of 85 % protons, 14% alpha particles and 1% nuclides as well as heavy ions of Hydrogen, Helium, Carbon, Oxygen of energies around 1GeV and Iron with energies of 10GeV [Fre1996]. Galactic particles are low flux, high energy particles, Hence they are much more difficult to shield against.

*1.2. **Charge Deposition and Soft Error Mechanisms effecting circuits***

Radiation can cause complete malfunction, parameter degradation and momentary upsets in electronic devices [Ker1988]. The type and severity of the interaction depends on the charge, flux, energy, speed and mass of the incoming particle as well as the mass, density and atomic number of the target material. The effects of radiation can be instantaneous, e.g., single event transients (SET) [Mav2002], single event upsets (SEU), multi-bit upsets (MBU), single event latch-up (SEL) or long term i.e., total ionizing dose (TID) [Bar2006].

*1.2.1. **SEE***

Single event effects (SEE) are non-reproducible hardware logic upsets that are produced by alpha particles, neutrons, protons and heavy ions, all of which are present in

abundance outside earth's atmosphere [Kar2004]. The charge deposited by these particles results in production of electron hole pairs by drift and diffusion mechanisms and if the amount of charge deposited is sufficient, it can result in upsetting the logic value stored on a storage element.

It is also important to note that alpha particles are also present in earth's atmosphere. Packages that encase IC's are common sources of alpha particles, consequently the studies of effect of alpha particles is not an area limited to rad-hard applications. Neutron interactions are mostly kinetic because of the fact that they do not hold any charge [Sag2005]. They can still produce electron hole pairs but the probability of such an event is low. Whenever a particle like proton, electron or heavy ion hits the device material it loses its kinetic energy because of columbic interactions. The particle strike results in either the release of electron hole pairs or in the nucleus getting knocked out of the atomic lattice. Both cases produce charge to upset the node value thus producing a soft error [Ngu2005].

As mentioned, Soft errors are transient and non-destructive in nature. The value on a storage element in architectural terms translates to the stored state of the machine. These states can be machine (architectural) or non-machine states. Upsetting an architectural state is absolutely critical whereas upsetting a non-architectural state can be less critical if it does not cascade into an architectural state error. The same strike can be destructive and permanent in nature like single event latch-up (SEL) [Dod2003] or single event gate rupture (SEGR), these will be explained in detail subsequently. The type of

effect is determined by energy of the particle striking and the physical characteristics of target material.

$$LET = \frac{1}{\rho}\frac{dE}{dx}\ (MeV - {cm^2}/{mg}) \qquad (1.1.)$$

The measure of this interaction is described in a terms of linear energy transfer (LET) or stopping power of the material given by equation (1.1), which refers to energy lost by the particle per unit length in the material before coming to a complete stop or exiting out, its unit is MeV-cm^2/mg. LET is a function of the energy and mass of the particle and density of the device, where (dE/dx) is the energy lost per unit length and ρ is the density of the material in mg/cm^3 . The maximum LET value near the end of particle's range is called the Bragg peak [Hse1981]. Charge interaction can cause direct or indirect ionization. Direct ionization corresponds to generation of electron-hole pairs across the track of incidence. Displacement damage as a result of the displacement of the nucleus from the lattice because of an inelastic collision resulting in a track of charge is

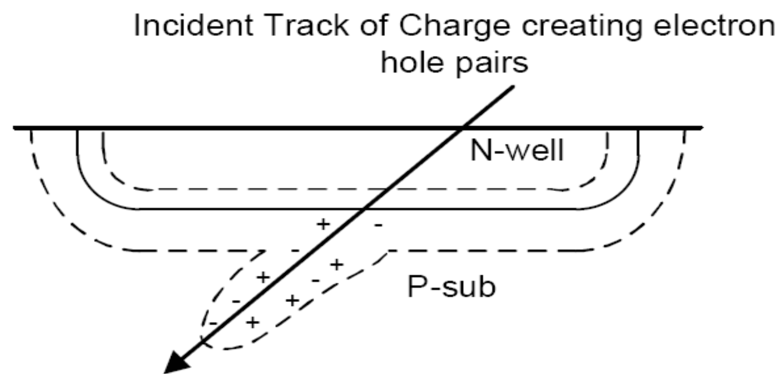

Fig. 1.3 Particle Strike showing track of charge funnel in a p-n junction.

also by product of direct ionization. Indirect Ionization is the process of neutrons transferring their kinetic energy to the atom in the lattice and thereby producing secondary electrons capable of producing ionization as they moves inside the lattice and lose their kinetic energy. Neutrons undergo capture by the nucleus of the lattice atom and result in generation of charged secondary such as alpha particles and oxygen nuclei which are also called secondary ions. Single event effects are a consequence of a single particle strike. Most heavy ions and protons cause direct ionization whereas neutrons generally cause indirect ionization owing to their neutral charge.

SEE response in Fig. 1.3 is generally characterized by three stage process of charge generation, charge collection and charge removal through circuit response. It is known through experimental data that an electron-hole pair is produced by every 3.6eV energy equivalent particle incidence on the silicon surface. We know that the density of silicon is 2328mg/cm^3, thus we can calculate using equation (1.1) that an LET of 97MeV-cm^2/mg corresponds to charge deposition of 1pC/um. Hence the charge collected is usually formulated as

$$Q = 0.01036 \times \text{LET } {}^{pc}/_{um} \,. \qquad (1.2)$$

The charge produced can range from (10-100 fC) and can only be removed by carriers drifting in response to the applied or built-in fields in the device, through the diffusion mechanism under carrier gradient concentrations or through carrier recombination in silicon [Dod2003]. A reverse biased p-n junction is most susceptible to particle strikes. In terms of device characteristics, the sensitivity to an upset of a specific

device is given by the critical charge (Q_{crit}), which is the amount of charge that needs to be deposited to upset a given node and cause a single event effect, i.e., a node state flip.

A single even transient (SET) is the momentary voltage spike resulting in logical value change of a combinational node due to a particle strike [20-22]. This spike can propagate and eventually dies down inside the circuit because of charge being taken out through the driving circuit. Propagation of the spike to a sequential element and subsequent capture is can upset an architectural state. This is in contrast to a single event upset (SEU), which is an upset on a latch or a flip-flop [Axn1986]. The same upset can happen within a memory cell due to the flipping of storage nodes in an SRAM [Sag2005] or a DRAM. The particle strike, if it has sufficient energy can also lead to a multiple bit upset (MBU) in a memory as the particle passes through multiple cell diffusions. To be described as multi-bit upset, a single strike should upset two or more bits in a single clock cycle [Mus1996].

Another effect, which needs a hard reset of the device to correct, is a single event latch-up (SEL). SEL is activation of the parasitic silicon controlled rectifier (SCR) inside a CMOS device in response to a high energy particle strike, resulting in a positive feedback leading to high short-circuit currents in a device. This high current can lead to clamping of the chip to supply or ground making it dysfunctional.

A single high energy strike directly to the gate can also lead to gate-oxide damage, which is called single event gate rapture (SEGR). With the shrinking of the gate oxides over the technologies this effect is very critical at lower technology nodes. These

failures are hard as the rupture is permanent. Even a reverse-biased p-n junction diffusion can breakdown by Zener or avalanche mechanisms due to a high-energy particle strike.

1.2.2. ***TID***

Total Ionizing dose is the degradation of MOS transistor parameters, specifically the threshold voltage (V_{th}), with gradual exposure to radiation [Bar2006]. A proton dose produces electron hole pairs on interaction with silicon substrate, the electrons get swept to the interface because of higher mobility but the holes get trapped in the oxide around the field area and hence turn on parasitic transistors around the field oxide and silicon-on-insulator (SOI) buried oxide. Threshold voltages of these NMOS parasitic devices decrease with time due to accumulation of holes in the oxide-traps. This also modifies the threshold voltage of the primary channel. Shift in threshold voltage and formation of parasitic channel results in the increase of standby current (I_{sb}) in the MOS transistor. The

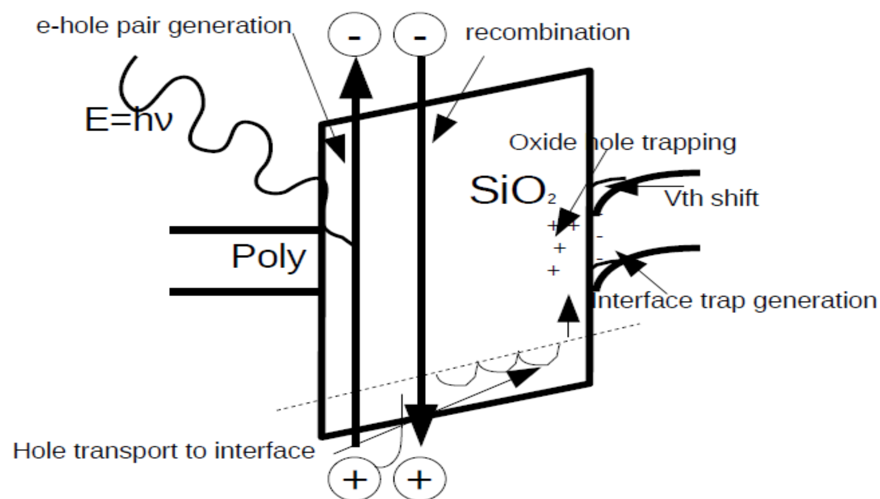

Fig. 1.4 Band diagram of TID degradation mechanism.

Si-SiO2 band diagram is showed in Fig.1.4. The mechanism involved in creation of free charge that can cause oxide contamination and degradation are as follows.

1. Electron-Hole pair generation due to incident radiation:

Any incident radiation like protons, hot-electrons, heavy ions with sufficient energy can produce electron-hole pairs across the track of their incidence. The amount of charge produced depends on linear energy transfer (LET) of the incident particle and band gap characteristics of the target material.

2. Recombination mechanisms of generated charge:

The free charge produced by the mechanism above can recombine back to exhaust itself. The fraction of recombination depends on quantum effects like columnar and geminate recombination. Recombination also depends on the energy that particles acquire on resulting displacement from the particle strike. Holes left free after recombination process can produce positively charged ions that cause threshold shifts.

3. Transport of free carriers to the Si-SiO2 interface:

The transport of the aforementioned charged particles to the interface can result in either trapping of holes in the oxide defects resulting in extra charge on the oxide [Bar2009] thereby shifting interface potential, altering the threshold voltage. It can also result in breaking of Si-SiO2 bonds on the interface and release of holes that hop around the interface resulting in mobile traps. Since the mobility of electrons and holes are of

different orders, the recombination time constant of holes is much longer than that of electrons. The accumulation of holes inside the oxide results in decrease of threshold voltage of NMOS transistors and increase of threshold voltage of PMOS transistors represented in Fig. 1.5.

1.3. ***Radiation hardening to mitigate radiation upsets and failures***

Radiation hardening can be incorporated by techniques applied at the device level, the system level and the circuit level [Ker1988]. The basic types of radiation hardening techniques are as follows.

1.3.1. ***Shielding***

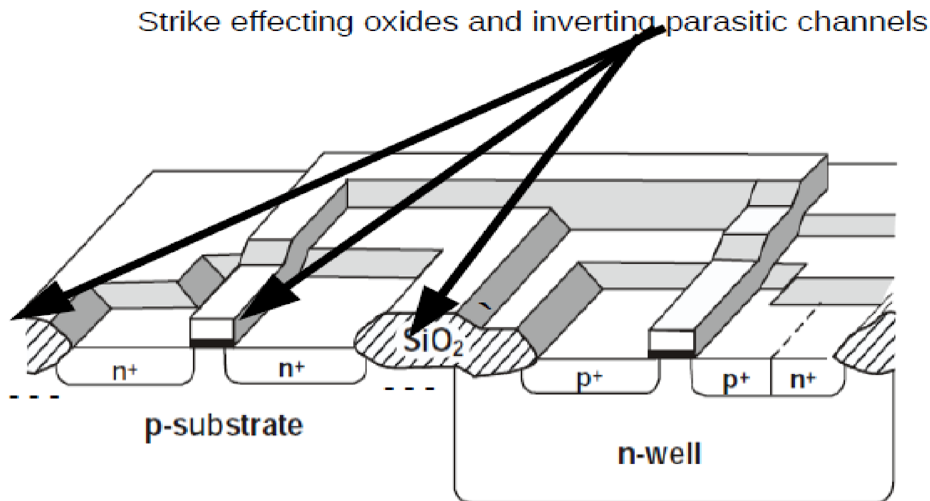

Fig. 1.5 Cross section of Inverter showing TID effects.

Shielding is a basic method to shield the devices from radiation particles by encapsulation in a shielding material. This method is generally ineffective because most of the particles have energies high enough to penetrate most shields in space.

1.3.2. ***Radiation hardened by process (RHBP)***

RHBP is the process of hardening the device by modifying fabrication process steps. Process steps are changed to modify parameters that are affected by TID and SEE. Care is taken to ensure that performance of device under normal operating condition does not change. RHBP usually involves changing integrated circuit fabrication steps to address TID or SEU concerns such as silicon nitride passivation layer and thinning field oxide for fixing oxide threshold shifts [Yos1994]. Resistive hardening [Ber2001] involves adding intentional resistances on to the storage nodes of sequential elements thus making the time constants of these circuits larger resulting in larger Q_{crit} of these circuits, and consequently making them harder to upset. The fastest commercially available rad-hard microprocessor [Rad750] uses 0.15μm rad hard bulk process to harden the design. Though RHBP is an easy method to achieve hardness it is not amenable to easy design fixes on the fly and it is not commercially viable owing to the lack of mass production, i.e., commercial demand. RHBP also lags the commercial processes by at least two generations in the semiconductor industry [Lac2000].

As the integrated circuit fabrication technology advances in terms of gate length nodes, the Q_{crit} required to upset the storage nodes decreases due to the inherent scaling with every new process, thus RHBP techniques become more difficult and expensive.

Thus, radiation hardened by design (RHBD) techniques using commercially available processes with no changes in process steps is increasingly attractive today.

1.3.3. ***Radiation hardened by design (RHBD)***

Radiation hardened by design circuit techniques are adopted to harden the devices using a non-rad-hard commercially available process [Lac2000]. This minimizes the cost of producing radiation hardened circuitry and also gives the designer control of the design to harden the circuits depending on the chip functionality.

A layout based method is to increase the sizes of transistors such that critical node

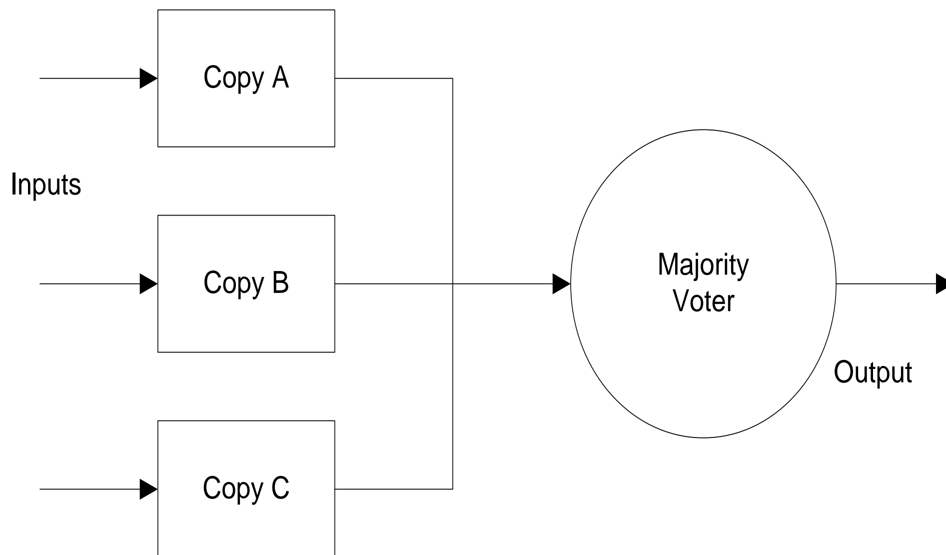

Fig. 1.6 Majority Voter showing Triple mode redundancy.

capacitance increases thus requiring higher Q_{crit} charge to upset the node hence making it less susceptible to radiation induced errors [Dod1995].

Other widely used technique is logic based hardware redundancy using a majority voter gate that is the basis of a triple mode redundant (TMR) logic system [Hin2011] [Hin2009] shown in Fig. 1.6. In a TMR redundancy scheme the same logic is replicated thrice and separated spatially to protect it against a worst case scenario of a MBU. A majority voter votes out the wrong value and the output always has the correct value.

Another method of filtering radiation errors is called temporal hardening [Mav2002] [Wea2004] shown in Fig. 1.7 where the signal is delayed twice and then voted against the original signal. The delayed value is greater than the experimentally calculated SET value so that the delay can always outlast an SET. The result of this combination is that even when the device has a radiation strike, there is a delayed version of the signal that does not get upset.

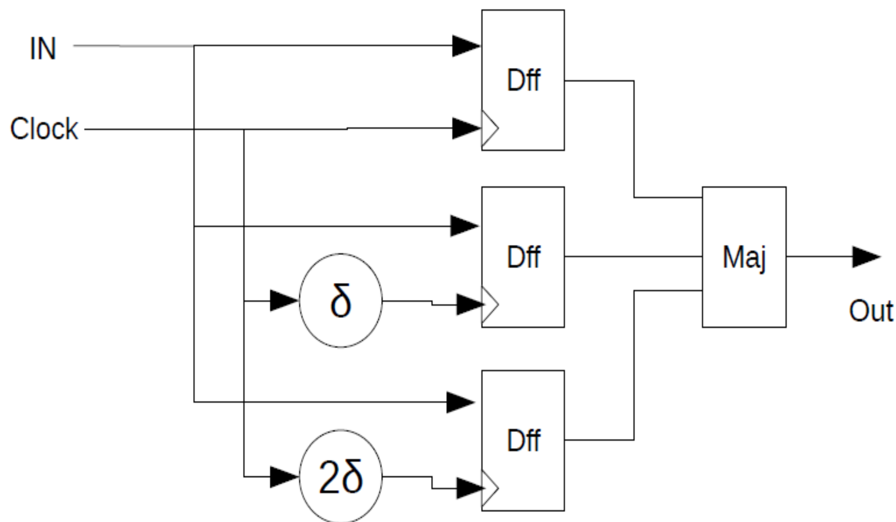

Fig. 1.7 Temporal sampling with delta delays.

Therefore the design always maintains the previous value till such time that it can give an updated correct value. The disadvantage with temporal hardening is the delay penalty that must be incurred even in absence of a radiation hit and since the values of the delays are the order of 300-400ps, system's maximum frequency is limited.

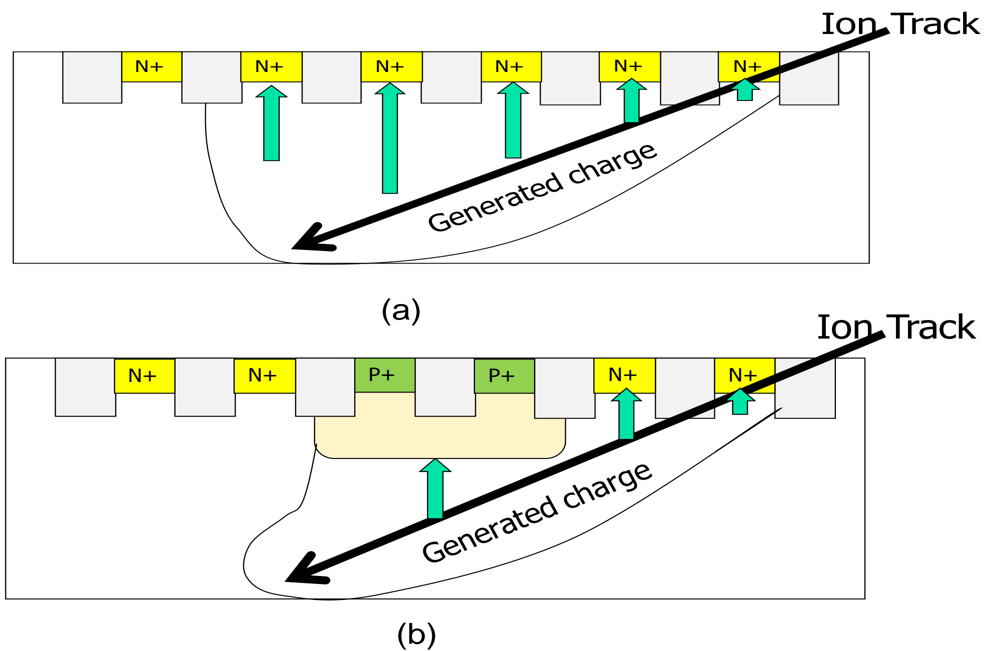

Fig. 1.8 Multi Node charge collection without (a) and with well separation (b).

Both techniques TMR and temporal need some spatial separation whereby they are protected against multi-node single event upset (MNSEU) [Knu2006] [Uem2010]. MNSEU is as the name suggests, an upset spanning multiple critical nodes because of a single radiation strike shown in Fig. 1.8. Critical charge collection across multiple nodes with and without well isolation is shown.

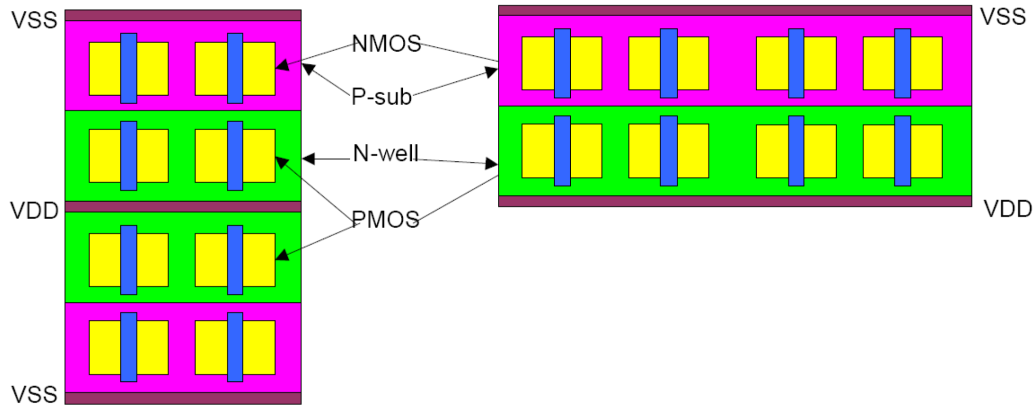

Fig. 1.9 Double height cells showing spacing between critical NMOS devices in comparison with a single height cell of the same area.

The same kind of upset can also occur by multiple radiation strikes on two critical nodes in a circuit but the probability of such a hit is infinitesimally low. Fig. 1.9 shows double height cells (DHC) with a separation based on critical and cancelling areas produce high immunity to upsets. The DHC have an immunity of 99.3% that is much better than single height cells with a shared well (90%). Well spacing in Fig 1.9 clearly shows that critical areas are spaced apart increasing radiation soft-error immunity.

Similar techniques are proven on both TMR [Hin2011] and temporal flip-flops [Knu2006] [Sha2011] where multi bit cells are interleaved in an inline or multi-height cell, thereby saving area. These cells can then be seamlessly used in the CAD flow.

In case of TMR hardening with multi-bit, multi-height cells [Hin2011] it has been shown that adding a spacer cell between two groups of flip-flop pipelines is very effective as long as the strike cannot span the height of the spacer cell thereby protecting against a MNSEU as shown in Fig. 1.10. This technique has been employed in the

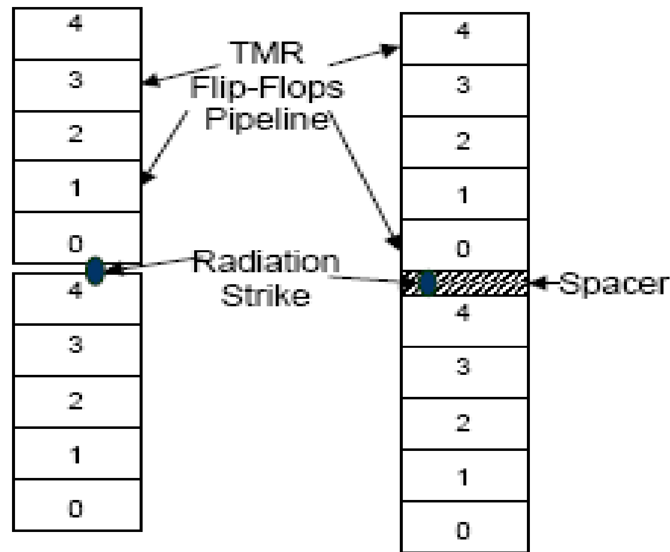

Fig. 1.10 A TMR Multi-bit flip-flop interleaving with spacer for spatial separation of pipelines.

reference design sequential elements and is an effective method for ensuring hardness by spatial separation. The performance comparison of the two techniques clearly yields that when performance is the target TMR logic is definitely the best option although leading to a clear area penalty. Power to a first order is definitely much lower in the temporal counterpart [Sha2011] but as voltage is scaled down there are some obvious gains in TMR logic as the supply voltage reduces to 0.8V [Hin2011].Such scaling is not practical in temporal designs owing to serious degradation of the sequential element dead times.

These were the key factors in the choice of TMR hardening for the reference design since speed with rad-hardness was the main design target.

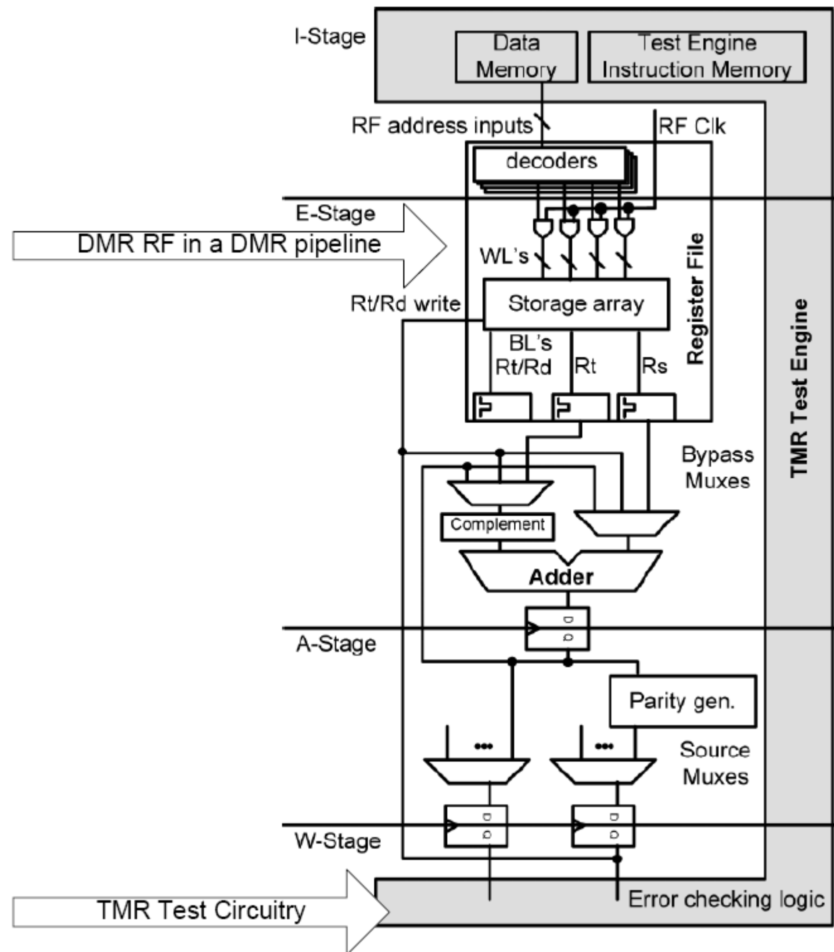

Fig. 1.11 A 5 stage DMR pipeline after [Cla2011].

Dual mode redundancy [DMR] logic scheme is another error detection scheme which can be combined with write back features to allow error detection and subsequent pipeline stalls which protect against architectural upsets due to propagation through the pipeline in subsequent clock cycle. A DMR processor scheme with a DMR register file

(RF) has been demonstrated in [Cla2005]. It is a DMR pipeline in a 5-stage pipeline microprocessor design, where DMR RF is used in conjunction with dual redundant ALU/bypass logic for data path error detection and correction based on parity generation and write back.

This scheme demonstrates the DMR processor which in conjunction with triple mode redundant logic portions is the basis of the rad-hard processor architecture. Fig. 1.11 shows the DMR pipeline described in [Cla2005] with a TMR test structure for validation of error correction on silicon using proton and heavy ion beam testing.

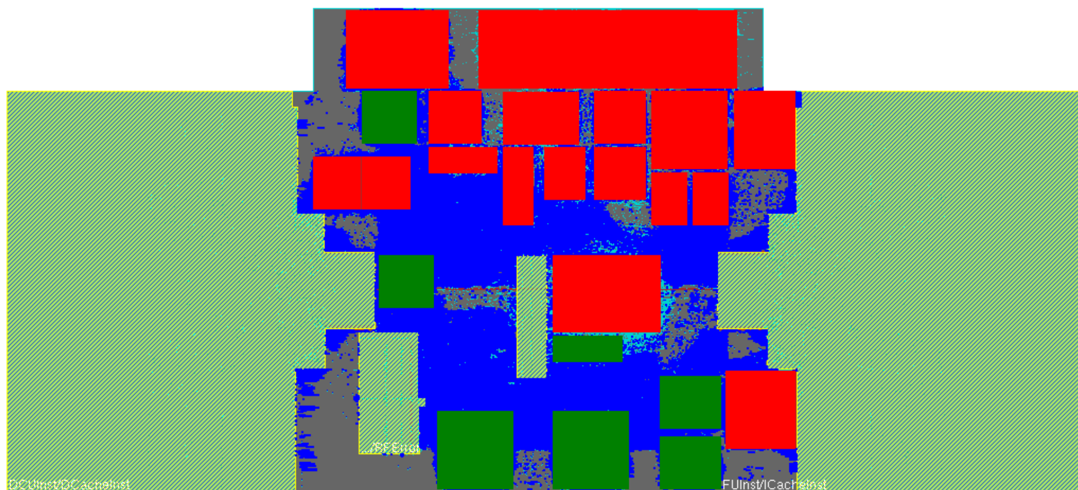

Fig. 1.12 Redundancy scheme using DMR (Blue), TMR (Red), crossover logic (green) and custom blocks (Yellow) in HERMES.

Register file designed will be explained in detail in the second chapter.

Processor operation works on speculative pipeline states. A speculative state is converted to an architectural state only after it has been ascertained that the state did not have an upset. In case of an upset the last know good state is restored. A combination of

TMR and DMR logic in conjunction with crossover logic (DMR to TMR) and (TMR and DMR) is used as the RHBD implementation technique in the reference design 'HERMES'. Fig 1.12 highlights the logic portions in the processor which are based on the logic redundancy schemes explained. Chip level physical realization of these techniques to achieve a radiation hardened processor is the main goal of this thesis.

Another method using logic redundancy is error correcting codes (ECC). One such method is error correction and detection scheme (EDAC) where redundant bits are added to the memory and the system can detect and then correct errors in the data using schemes like parity, Hamming codes or other such error correction codes [Che1984]. In the cache designed for this microprocessor implementation a parity scheme along with interleaving where for every 8 bits is used. Parity is calculated and this value is checked on each access to determine sanity of data. System level techniques are also used where microprocessor maintains checkpoints to monitor the system state and detect faulty states because of radiation hits. After detection there can be recovery of the system to a previously known valid state.

TID hardening techniques are different from hardening against soft errors since the mechanisms are long term in nature and need to be handled differently. Edgeless transistors shown in Fig. 1.13 are used to protect the devices against TID susceptibility. All the standard cells in HERMES use edgeless NMOS transistors.

MOS transistors have interface of source and drain to the field oxide. A particle strike causes the oxides to have excess charge and may invert silicon below to create a

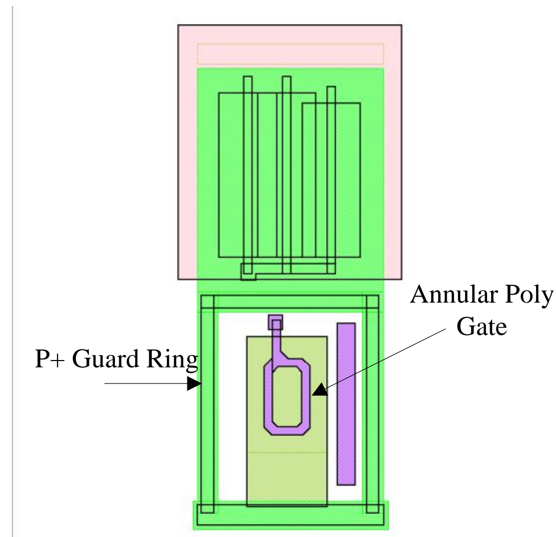

Fig. 1.13 Annular Gates eliminating field oxide and drain diffusion interface after [Pettit 2009].

parasitic channel. Edgeless transistors do not have a gate oxide/field oxide interface to the drain diffusion and since the diffusion on the outside has the same potential, leakage currents are eliminated. Along with the special annular transistor, a p+ guard ring is also implemented in the gates leading to creation of back to back reverse-biased diodes that minimize leakage current. Destructive radiation effects cannot be guarded against and redundancy at circuit or system level is the only possibility of correction.

1.4. ***Summary***

First chapter explained the radiation environment that is required to understand the need for rad-hard circuit design and demonstrated mechanisms involved in radiation-induced failures. Techniques used in hardening circuits to ensure their continued functionality in presence of radiation are also discussed. The reasons for choosing TMR hardening were explained with tradeoffs in comparison to temporal

hardening. DMR and TMR logic hardening techniques being used as RHBD methodology to harden Hermes microprocessor are elucidated.

CHAPTER 2. CUSTOM BLOCKS

Any large ASIC contains macro block abstractions which either contain custom circuitry like memory, PLL, Serializer/Deserializer or some third party black-box intellectual property(IP). These design constituents limit the design performance and add constraints in terms of floor planning, placement, power planning and timing. For successful full chip implementation the physical and logical integration of these constituents is critical. These custom designs were implemented over the period of last few years are now being used as constituents in the current chip level implementation [Yao2011] [Cla2011] [Che2012]. In this chapter, custom blocks used in HERMES are explained and the constraints they impose on full chip integration are discussed.

2.1. ***Cache***

Memory is an important part of any processing system. Ideally any processing system requires an unlimited memory at an extremely low access time. Since such a system is not physically realizable a memory hierarchy with multiple levels of hierarchy of storage varying density and speed is utilized. The density increases and the speed decreases as we move away from the processor core. A register file is the fastest and the smallest memory in such a hierarchy. A cache is the first level of memory outside of the core, followed by off processor main memory and finally disk memory. Memory hierarchy and specifically cache relies on temporal and spatial locality to facilitate memory access. Spatial locality means “that items whose addresses are near one another tend to be referenced close together in time” [Patterson]. Temporal locality implies that

“recently accessed items are likely to be accessed in the near future” [Patterson]. Thus a cache stores most recently used data and instructions.

Soft errors in caches are of concern not only in radiation-hardened designs but even at ground levels. Neutrons have been identified as the major cause of cache upset, creating secondary ions which cause SEE induced soft errors in memories [Lambert 04]. Since the cache is an array, accumulation of SEE induced errors to cause a multi-bit error is likely. To protect against these soft errors, error correcting codes (ECC) and scrubbing techniques are used [Sla2005] [Muk2004]. The measure of effectiveness of these codes is mean time to failure (MTTF) which must be an acceptable value for a cache. EDAC is added to improve SER so that it becomes insignificant in the overall design MTTF in low-latency L1 caches and register files [Moh2006]. Fixed interval scrubbing involving error checkers periodically accessing cache blocks and removing single-bit errors is used on large caches to reduce temporal double-bit error rates [Muk2004]. Lightweight EDAC (LEDAC) is also employed in rad-hard memories [Moh2007]. LEDAC contains a two dimensional parity checking comprising of row and column check bits. Row check-bits flag an error in the rows and column check-bits locate and correct the bits.

A radiation hardened cache in 90nm IBM G technology was implemented, fabricated and tested to show excellent hardness [Yao2011]. The design is used as instruction and data L1 caches in the processor design ‘HERMES’. It is a 16KB, 4-way set associative cache using write-through and no-write allocate policies. This gives the cache a reasonable hit rate without being complicated design. The cache has 1024 lines

with 16 bytes in each line. It is virtually-indexed and physically-tagged. Each line consists of four words in the same set and way. 256 cache lines and 4 ways make it a 16kB cache.

The block diagram of the organization is shown Fig. 2.1. The cache supports four operations namely lookup, read, write and global invalidation. In the lookup operation, 4 ways of the cache are read and the tag address is compared with the physical address from translation look-aside buffer (TLB). In case of a match the selected way is output and the hit flag is set. A fetch or load operation is also supported. The read operation can read a given set and way specified inside the tag or data array. In case of a write operation a specified set or way is written into the data or tag.

The minimum unit written is a byte but in case of a line fill the whole line is written. In the global invalidation operation the whole cache is invalidated. Figure 2.2 shows the implemented floor plan screenshot of the cache design implemented, where it

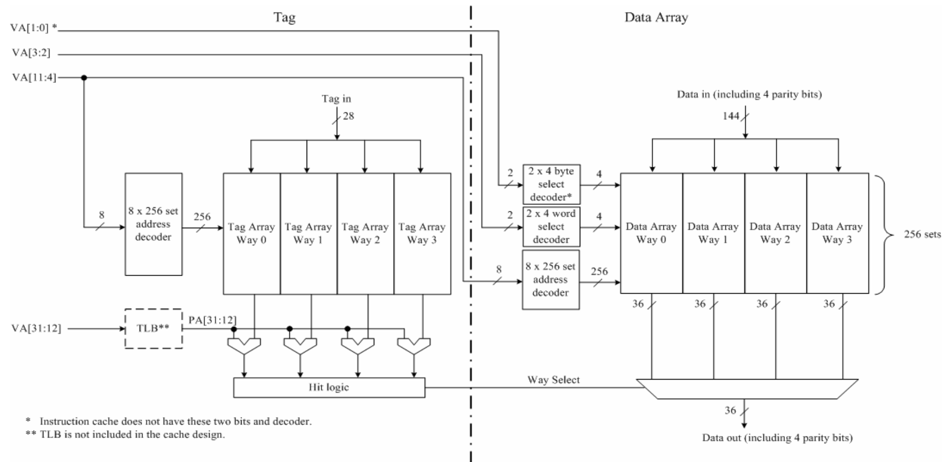

Fig. 2.1 Block Diagram of the Hermes 16KB, 4-way set associative cache.

can be seen that the data array is divided into 2 halves each storing one half word. The tag is in the middle to minimize the hit timing path to maximize speed and minimize power. In each data array half there are 4 words comprising of 4 banks. Each bank has a top and bottom sub-bank comprising of 32 rows and 72 columns of SRAM cells. There is

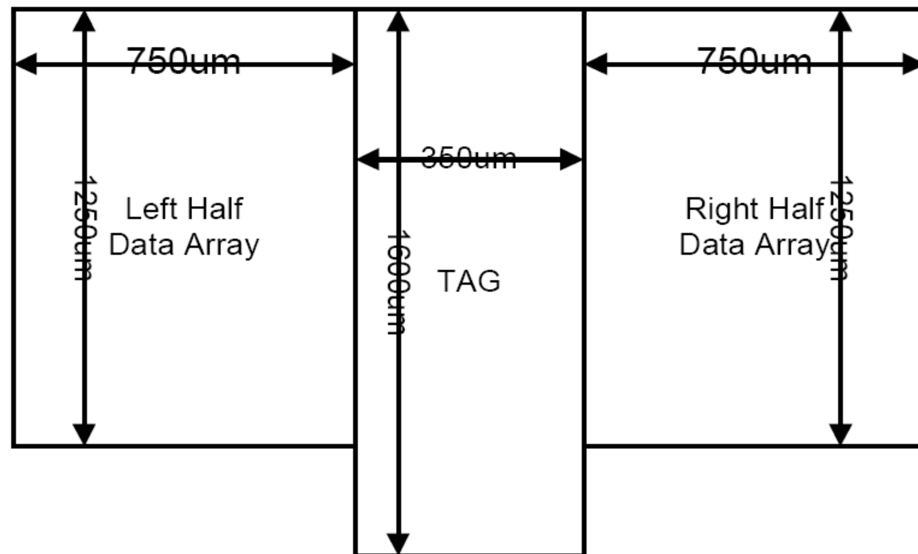

Fig. 2.2 Floor plan of RHBD Cache designed for IBM 90nm process after

parity protection for each byte and hence the total size is 18 bits (16 data+ 2 parity).

Radiation hardness is achieved using layout interleaving to maintaining critical node spacing between cells. 28 bits of the tag are divided into four parity groups. The distance between the bits of same tag parity group is width of 3 SRAM cells. Valid bits are SEE protected by dual-redundancy. On every read, dual redundant valid bits are read out and checked. If they do not match the output is considered invalid. Cache uses multiple error detection domino circuits, such as the word line encoder, word line nor checker, read checker, write checker, write enable encoder and write-enable nor checker.

Most of these circuits are a combination of ones-catching domino circuits that detect radiation induced errors and set an error flag used by the processor to generate fault trap. Detailed discussion of their operation is beyond the scope of this thesis.

Test chip 'TC19' with the cache defined above achieves speeds of above 1 GHz at irradiations of 2Mrad TID and 219MeV-cm^2/mg^2 LET. Its max power consumption at

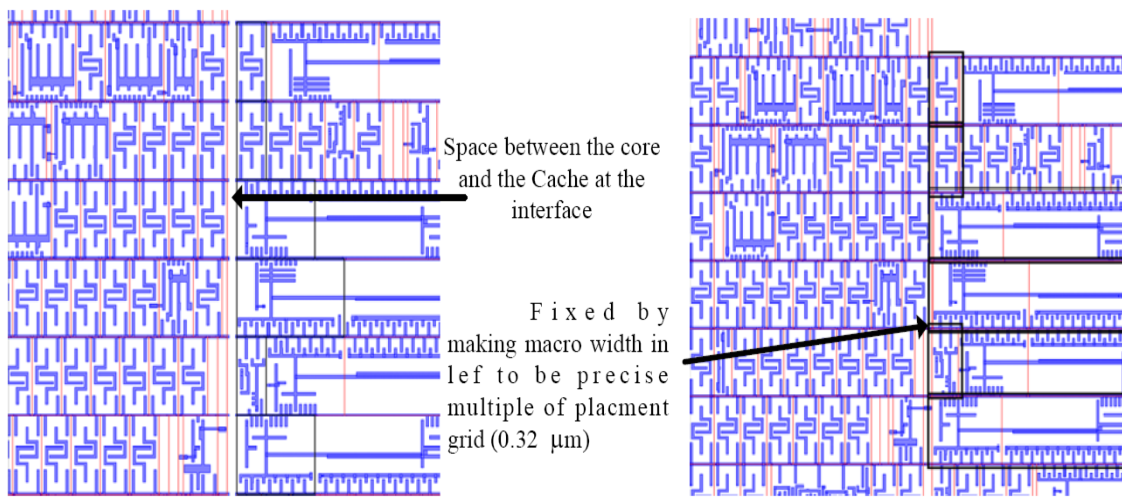

Fig. 2.3 Cache abstract alignment fix.

1GHz was tested on silicon to be 226.1mW at a VDD of 1.45V.

Some issues were encountered during the integration of the cache [Yao2011]. Abstract timing model (.lib) for the cache was generated with multiple Hspice simulations. It took multiple iterations to get the timing abstract properly modeled timing abstract. Also, creating a physical abstract (.lef) file was a challenge owing to the huge size of this block. Tool runtime issues and tool specific settings had to be tweaked to get a proper physically accurate abstract file.

While physical integration in the floor plan abstract alignment issues were encountered and subsequently solved with iterations on the abstract (.lef) generation. To fix this issue, abstracts were regenerated with the width of the cache macro to be a multiple of the standard cell placement grid which is 0.32µm. Fig. 2.3 shows the results of such realignment based on lef regeneration. Rectilinear size of the macro posed some challenges since the macro width and height need to be an integral multiple of the placement grid and standard cell height (4.48µm) respectively.

2.2. ***Clockspine Design***

Clocking is central to the performance of a microprocessor system. For a rad-hard processor not only is the performance of the clocking system important, but also its capability to work successfully in a radiation environment. Radiation strikes on the clock can cause clock SET's, which propagate and result in inadvertent clocking of the flip-flops or latches causing SEU's. The jitter performance of the clock may degrade considerably due to radiation hits. The HERMES clock spine design has been fabricated at 90nm processes and tested to be error-free to 100MeV LET.

The clockspine design takes the clock from PLL, distributes it to the whole chip and at the same time makes use of enable signals to control clock selection and gating. The clock from the PLL is carried to the spine using 102 distributed buffers that are spatially separated. A single particle strike on any individual buffer therefore produces less than 1% jitter on the clock outputs. Fig 2.4 shows the physical design of the clockspine.

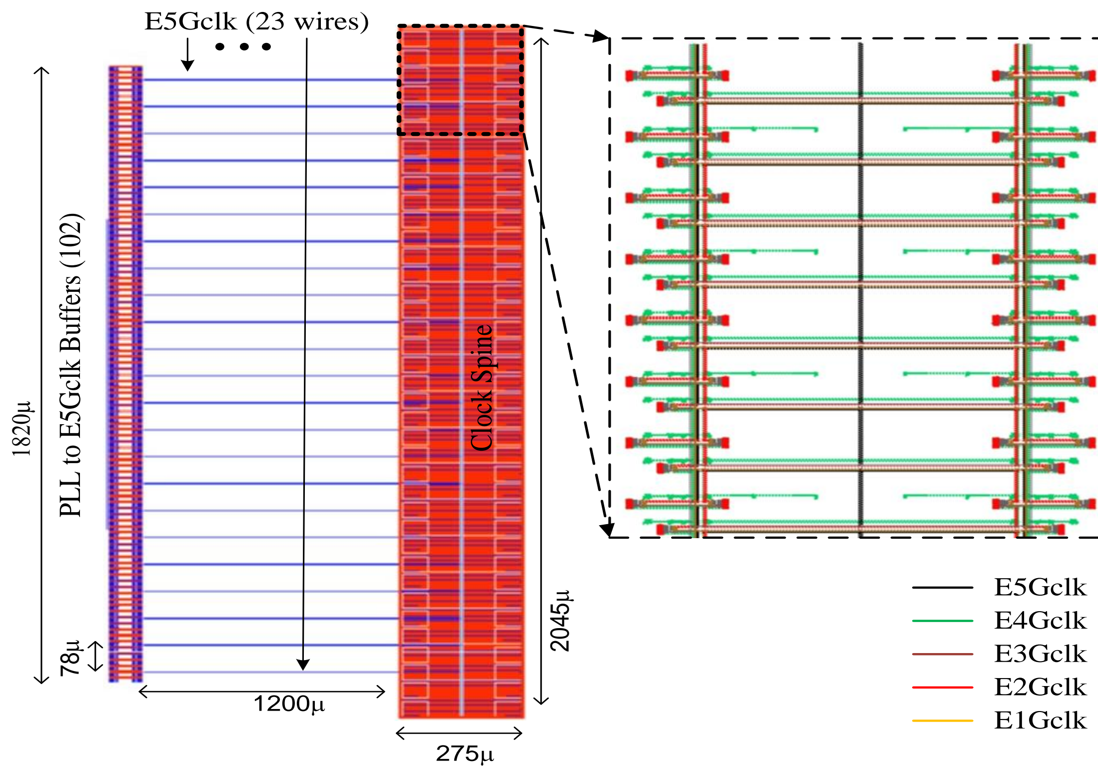

Fig. 2.4 A RHBD Clockspine design on IBM 90nm process after [Che2012].

The spine itself is designed in two stages that provide different protection mechanisms namely, global clock distribution and local clock distribution. In the global clock distribution network, the clock at the spine input is buffered to the local clock leaf nodes using 5 inversion stages. Each of these global nodes is driven from the previous stage by 38 spatially dispersed inverters that also provide sufficient drive fan-up to supply the large number of local clock networks.

The large capacitance on each of the global clock nodes makes them essentially immune to SET induced glitches. The local clock networks produce the clocks for each of the chip logic sub blocks and each may be enabled or disabled as needed to conserve

power. The local clock networks cannot be capacitively hardened like the global clock nodes and hence, the local clock networks employs XOR based SET checkers that check for strikes on clocks and clock-enables, erroneous clocks and clock glitches. Once an error is detected, an error flag is set that tells the microprocessor that a clock upset has occurred and the machine state is flushed. The clock spine provides 64 output clocks that are spatially separated.

*2.3. **Register File***

A 32-bit DMR register file (RF) has been implemented and tested to show successful hardening [Cla2011]. The datapath is also DMR for compatibility with the RF. SEE protection is provided using critical node separation combined with bit interleaving and parity. Interleaving protects the RF from MBU's. Parity detects the errors and dual mode redundancy provides a corrected copy for SEU correction. Register file hardening

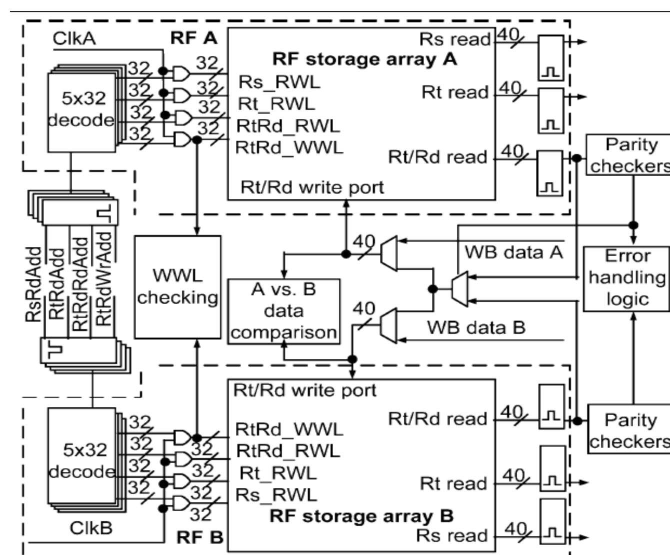

Fig. 2.5 A DMR register file block diagram after [Cla2011].

is challenging since it is a part of the pipeline timing critical path. Using EDAC schemes incur a timing and area overhead on the register file design. Moreover, EDAC cannot protect against erroneous data from SET's either in the RF itself or the ALU/bypass circuitry which consumes the data in the RF. RF design described is part of the five stage pipeline processor. A DMR RF block diagram is shown in Fig. 2.5.

The data path can be erroneous due to SEU's in the RF or because of SET's in the RF readout logic or the DMR pipeline logic. Error detection is trivial in DMR but correction is not because there is no way of telling instantaneously which copy is incorrect. The RF has parity detection on a nibble basis which requires twice as many parity bits compared to byte parity. When an SEU is detected the data from the copy that

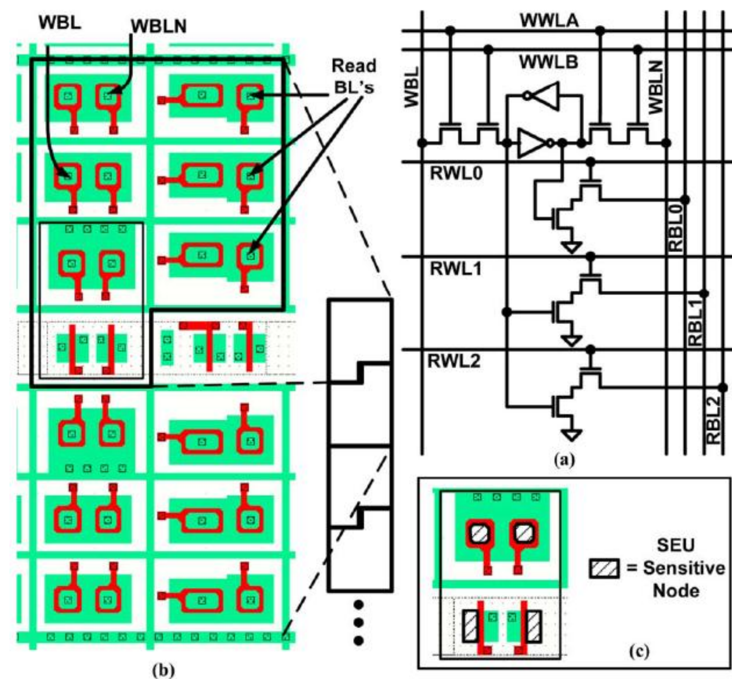

Fig. 2.6 (a) Register File cell schematic, (b) Cell layout (c) Critical node separation, after [Cla2011].

is correct is written back into the register file. To correct the data the RF has an extra Rt/Rd read port which reads the copy of destination registers to be updated next cycle write-back stage. All datapath transactions which update the RF are checked for DMR parity and in the event that they mismatch an exception is produced in the processor. Therefore an overwritten RF store can be cancelled if an upset is detected and the original data is written back to the RF. While the incorrect data is written the processor pipeline is stalled to stop the erroneous data from propagating.

Write word line (WWL) error detection scheme is used to detect erroneous or incomplete WWL assertions. The same write-back is used to restore the prior value from the register entry intended to be written into the RF and the previous correct architectural state is restored. The A and B copy parities are checked to determine which copy of data is sane and written back. Accumulated errors or errors originating from multi-particle hit at the same parity group in the A and B copy cannot be corrected by the write-back scheme and need background scrubbing [Cla2011]. Opportunistic scrubbing scheme is also implemented which uses cycles of inactivity in RF and checks parity in all the registers sequentially and corrects upsets that can potentially lead to accumulated strikes causing irrevocable errors.

Register file cell schematic and layout are shown in fig 2.6(a) and 2.6(b) respectively. Annular transistors protect against TID. Annular diffusions are also shared to minimize the capacitive loading on the BL's also SEU critical nodes are separated to minimize MNSEU probability shown in Fig 2.6(c).

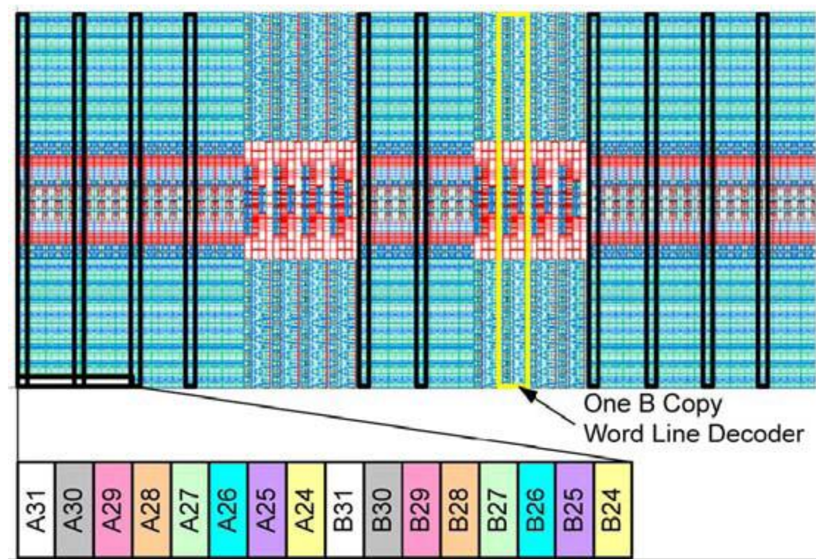

Fig. 2.7 Register file layout with Interleaving shown.

DMR read and write wordlines are also spatially separated because matching incorrect assertions of the wordlines would not be detected. Interleaved RF layout is shown in Fig. 2.7.

RF SEE testing was done at Berkeley labs with static and dynamic testing of the RF using broad beam ion testing. Heavy ion testing with Boron, oxygen, neon, argon and copper ions and max LET of 21.17Mev-cm^2/mg at angles of 0 and 70 degrees was performed. The testing was done at speeds of 100 and 200 MHz. Proton beam testing was done with energies of 49.3 and 13.5 MeV at 60 MHz. All these tests produced acceptable error rates to prove the success of the DMR design strategy.

2.4. ***Summary***

Cache, clockspine and register file are used in the microprocessor top-level design, and the performance of the whole system is dependent on the performance of these sub-blocks in terms of speed, area and radiation hardening. Since these parameters are proved on silicon, similar or derived design techniques have been used for hardening the top-level design.

CHAPTER 3. ASIC DESIGN FLOWS

A clean design flow is essential to reproducible design cycles and fast turn-around design times. The general ASIC design flow is defined in Fig 3.1. This chapter describes the basic ASIC design flow and compares it with the RHDB specific design flow developed in this work. A brief introduction to the triple mode redundant (TMR) flow [Hin2011] is also described. This chapter also explains how these design flows are used in tandem to realize a complete RHBD microprocessor from register transfer language (RTL) to graphical data storage information interchange (GDSII). The design methodology described in this thesis is a combination of RHBD top-level ASIC flow combined with TMR level flow used to create the TMR hard logic modules.

These modules are then used in the top-level flow to produce a complete chip, implementing a bottom-up design strategy. The microprocessor design is partitioned into logical sub-blocks that implement a MIPS core with individual instruction and data caches. Design is implemented as TMR blocks and DMR logic. There are crossover blocks which are the interface between the TMR and DMR domains. DMR logic is part of the top-level implementation flow whereas the TMR blocks are implemented using the TMR flow described in this chapter.

TMR blocks are used to correct errors which are detected in the DMR blocks. The architectural functionality of these independent sub-blocks is beyond the scope of this thesis and is not discussed further.

*3.1. **Conventional ASIC design Flow***

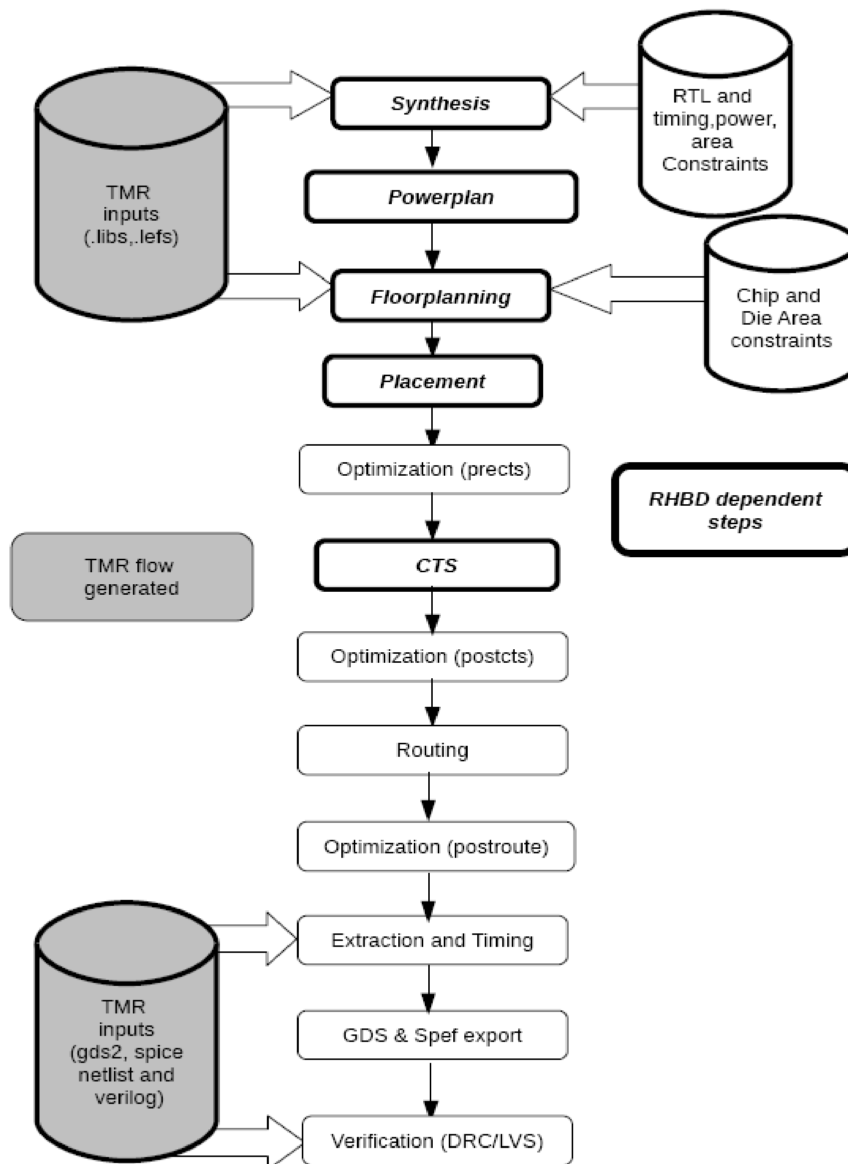

Fig. 3.1 A Conventional ASIC design Flow.

*3.1.1. **Synthesis***

The conventional design flow shown in Fig 3.1 starts with the RTL code of the design implemented in a hardware description language such as VHDL or a Verilog. The

top-level RTL describes the complete design logically. This RTL code is synthesized to an equivalent functional gate level design. Synthesis is the process of mapping generic digital logic gates to technology specific CMOS gates to implement the logical function specified by the RTL using a synthesis tool. The synthesis tool generates functional equivalents of the RTL code in terms of logic gates under timing constraints called standard design constraint (.sdc) and other design options. The quality of synthesis is determined by these design constraints and can be optimized for area, timing or power.

Multiple design options can also be specified to the synthesis tools for determining the design–for- test (DFT) related design features that may be needed for creating a testable design. Options for hierarchical modifications and preservation of hierarchical designs are used to manage design hierarchies. Most of these design options are proprietary to the tool sets used (Cadence, Synopsys, Mentor etc.). Nevertheless similar features are provided.

The output of synthesis is a gate-level netlist that instantiates technology specific gates from vendor provided foundry libraries that determine their process, timing and other physical parameters. The liberty timing (.lib) and liberty exchange format (.lef) views are abstract timing and physical views respectively that help in reducing design complexity and managing large design databases. A .lib file contains the timing, power and functional information of a logical gate and its behavior to input stimuli and output loads. The liberty exchange format (.lef) specifies abstract views of the cell layouts

representing only inputs, outputs, metal blockages and boundary information. Together they allow reduced, but significant information to the CAD tools.

3.1.2. ***Physical Implementation: Floor-plan and Power-plan***

Once the gate-level netlist has been successfully generated, the design constraints specified by the synthesis tool are used as inputs to the next step that involves creating the physical database. Power-planning, which involves formulating the spatial location of metal lines for the power and ground supplies for the design is crucial and is implemented in an early phase of the physical implementation. In ASIC designs this is assisted by the definition of standard cell rows, meshes and rings if necessary. The actual layout also depends on the type of packaging used. Using wire bond packaging requires power provided by multiple explicit power and ground pads in the I/O ring. This means connections need to be made to meshes/rings from the pad pins in contrast, for flip-chip packaging, power and ground planes are orthogonally connected to the meshes from the top through vias. Power-planning needs to take into account the total current and power budget to calculate the width of the power lines to be able to accommodate the required current within the acceptable supply voltage range as per chip specifications. The power-plan should also make sure that any macros in the design are well strapped to the power lines, avoiding supply IR drop issues and electro-migration issues. Having a large number of vias to tap in to the power grids avoids such failures and falls in the purview of design-for-manufacturability techniques (DFM). Designs with multiple power and ground supplies need to be well isolated to prevent accidental shorts or coupling issues. Digital and analog power and ground supplies need to be spatially isolated as well to avoid noise

and substrate coupling. Upon successful completion of power-planning, the design floor-plan is created. Floor-planning is the process of efficient allocation of silicon die area among all the sub-modules of the design to create optimal arrangements. This is an iterative process that involves converging on the optimal size of core and die for the best possible performance under given area, speed and power constraints. Floor planning involves placement of macros and partitions based on connectivity in the RTL and architecture. Ideal floor-planning results in minimizing routing and placement congestion as well as meeting timing, area and power constraints. Pads for the chip and I/O's for sub-blocks are also placed during this step. If the design is hierarchical then design partitioning happens at this step. Partitioning is done to manage design complexity in case of large designs and enable faster turn-around times and design closure.

3.1.3. ***Placement***

Upon successful generation of an optimal floor plan, the design gates, instantiated as standard cells are placed as per the floor plan in the next step called placement. Silicon area for larger blocks is budgeted in the floor plan step and allocated at specified locations for maximum resource utilization. Such larger blocks may also be pre-placed in the floor plan step. The placement tool then places the standard cells using the optimizing placement algorithms. Macros and pad/pin placements are used as a seed to the algorithms for efficient placement of the cells. The emphasis during placement may be to optimize timing or may be congestion driven as per the design needs. Modern tools have multiple options that are fine-tuned for complex design specifications that range from setting densities, clearances and groups to guide the placement algorithm. The placement

step also accommodates checking congestion statistics and timing statistics to gauge the quality of any placement done. A great deal of designer effort is required to produce a quality placement and is generally done multiple times before a satisfactory placement has been reached. This step contributes heavily to the overall quality of the final design in term of design metrics such as power, performance or area.

3.1.4. ***Pre-CTS optimization:***

The placed design from the previous stage can be further optimized for meeting important design aspects such as timing goals and power budgets. Timing paths across the chip are determined in relation to the system clock(s). Clock distribution networks, i.e. clock trees, are crucial to balancing skews on such clocks across all sequential elements in the chip and this contributes to the overall speed of operation. Consequently at this stage it is crucial to optimize the design, firstly for timing and then for area and power if so constrained. This stage of optimization utilizes complex Boolean algorithms and like most steps explained here is specific to the toolsets being used. The optimization tool determines the worst case timing path, called the critical path, based on the worst case negative slack (WNS) in each path under given constraints. The optimization is then carried out by minimizing the total negative slack (TNS) which is the sum of all WNS in the design. The optimization starts with the setup timing paths at the first optimization step s called pre-CTS optimization. This is done prior to designing the clock trees or clock-tree synthesis (CTS). The CTS operation will be explained in further detail later in this chapter. The pre-CTS optimization engine also performs design rule violation (DRV) fixing to fix discrepancies in the design regarding maximum capacitance, transition and

fanout constraints. The pre-CTS optimization tool algorithm takes the worst critical path and performs operations such as remapping, upsizing and downsizing, buffering, or swapping pin locations to achieve the best result. Once the given primary timing constraints have been sufficiently met, there may be additional steps for reclaiming area. Pre-CTS optimization can only correct setup violations since there are no hold constraints with a clock tree not yet being built. After acceptable timing numbers (WNS, TNS) have been achieved the next phase of the design flow would be clock tree synthesis (CTS).

3.1.5. ***Clock Tree Synthesis***

Clock tree synthesis (CTS) involves building a balanced clock tree to provide clocks to sequential elements (flip-flops, latches and macros) in the design. Clock trees are mostly balanced H-tree configurations or mesh structures that are designed with stringent design goals to minimize skew between different leaf cells while keeping the overall clock delay within a desired range. The mesh structure gives a superior performance in terms of lower skew but consumes more power due to the large number of buffers, greater node capacitances and multiple driven nodes in the clock tree while the H–tree based clock distribution networks have larger skews at lower power costs. This design follows an integrated h-tree based and mesh based implementation with global clock distribution nodes being mesh structures confined to a dedicated clock spine that were balanced by hand, while H-trees fan-out from different points in the spine to different modules and sequential elements based on physical proximity and logical functionality. The clock skew generated during this phase of the design determines the hold time fixing requirements for subsequent steps. Clock trees are usually built with

buffers or inverters and may also have integrated clock gating (ICG) cells that shuts the clock to gated portions of the design for power saving.

3.1.6. ***Post-CTS optimization***

After the clock tree has been synthesized that meets the required skew, the next step of the flow, called post-CTS optimization involves fixing for remaining design rule violations (DRV), as well as setup and hold violations. The design is constrained with proper min delay constraints to achieve the best possible performance. Min-delay time, along with the skew between flip-flops, determines the amount of hold buffering required. The statistics involved in making the design decisions for this specific design are explained later. Once post-CTS optimization provides acceptable numbers meeting the setup, hold, and DRV constraints, we proceed to the actual routing of design signals.

3.1.7. ***Routing***

Once CTS is completed the design has be routed. Routing in encounter uses the proprietary Nanoroute feature. A routing algorithm tries to find Manhattan routes for all the nets in the design. As it performs routing, Nanoroute has to ensure that no shorts, spacing, and min area violations are created. To do this it creates routing tracks called Gcells. The whole design is partitioned into these Gcell tracks which allow early calculation of congestions statistics. Based on these congestion numbers the designer can ascertain the routing density and congestion of the design. The likelihood of Nanoroute failing or creating DRCs is inversely proportional to the congestion values. Once routed

we can run verifyGeometry and verifyConnectivity to run encounter based design-rule-check and layout versus schematic checks to ascertain design consistency and feasibility.

3.1.8. ***Post-route Optimization***

Once the design is routed post-route optimization is carried out. Again, optimization algorithms optimize the design for setup, hold and DRV violations. However this time on detailed and most importantly, extracted routes rather than estimated. This stage, being rigorous and detailed has much longer run times. After post-route optimization, usually silicon integrity (SI) optimizations for sub-90nm designs are carried out as they are more susceptible to noise coupling. SI optimization usually involves spacing critical nets farther apart, increasing their width, increasing victim net gate drive and decreasing aggressor net gate drive.

3.2. ***RHDB Design Flow: TMR Flow***

Fig 3.1 shows a flow chart where the steps marked in bold indicate steps where RHBD specific design decisions play an important role. This usually involves modifying the general flow to perform special design functions that facilitate hardening while maintaining performance. The exact design decisions taken will be explained in detail in chapter 5. The bold boxes show the inputs of TMR flow to the top-level RHBD flow. The design steps necessary to generate the TMR dependent inputs to the RHDB top-level flow are illustrated in Fig. 3.2. These inputs are the .lib (timing abstract), .lef (physical abstract), GDSII (layout), Spice netlist (transistor level connectivity) and Verilog (gate level connectivity).

These inputs are used at various stages in the top-level flow for design import, signoff timing and signoff verification. The design steps in the TMR flow are similar to top-level flow with specific changes to accommodate radiation hardening. The RHBD flow incorporates separation for dual mode redundant (DMR) logic. Special CTS is performed by providing independent clock trees that are less susceptible to strikes. To route TMR blocks successfully, a special pin placement algorithm is incorporated into the TMR flow. All such modifications/improvements are described in the next two chapters.

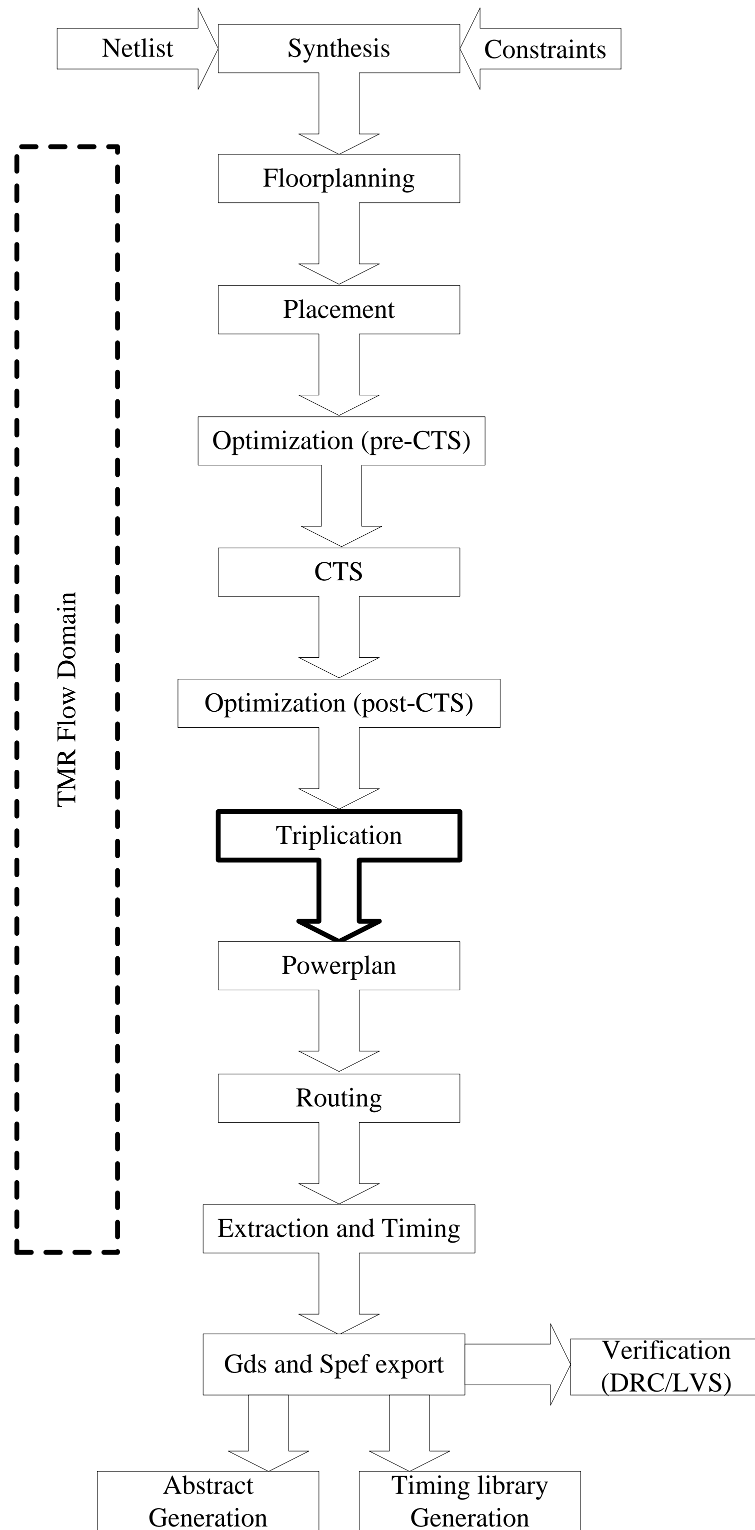

Fig. 3.2 A TMR ASIC Flow.

3.2.1. ***Synthesis***

Synthesis for the TMR modules is identical to that of normal logic, where the RTL code and the synthesis constraints are used to create a technology specific gate level netlist. This gate level netlist is the input to the physical implementation. We use an in-house tool developed here at ASU shown in Fig. 3.3 to get implement the TMR flow steps [Hin2011]. The aforementioned tool uses standard tools and scripts to perform TMR radiation hardening along with the usual design flow steps to get TMR blocks

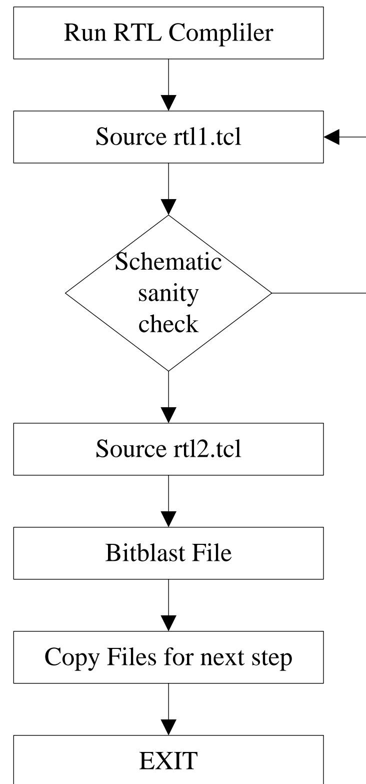

Fig. 3.3 In-house RC tool flowchart.

without compromising performance from timing standpoint.

```
  Generated by:           Encounter(R) RTL Compiler v09.10-p104_1
  Generated on:           Nov 16 2013  03:42:18 PM
  Module:                 IEArchInterfaceST
  Technology libraries:   rhbd_celllib
rhbd_celllib_lvt
Operating conditions:   typical (balanced_tree)
Wireload mode:          enclosed
  Area mode:              timing library
============================================================
        Timing
        ------
Tracing clock networks.
Levelizing the circuit.
Applying wireload models.
Computing net loads.
Computing delays.
Computing arrivals and requireds.


        Slack               Endpoint              Cost Group
------------------------------------------------------------
-1240ps IEArchInterfaceST/IESelCP0DataLoCAH GClk

        Area
        ----
     Instance      Cells  Cell Area  Net Area   Wireload
-------------------------------------------------------------
IEArchInterfaceST   2495      28265         0     <none> (D)
(D) = wireload is default in technology library
        Design Rule Check
        -----------------
Initializing DRC engine.
Max_transition design rule: no violations.
Max_capacitance design rule: no violations.
Max_fanout design rule (violation total = 33.000)
Worst violator:
Pin               Fanout           Max     Violation
-----------------------------------------------------
g75498/o          65.000       32.000        33.000
```

Fig. 3.4 *Synthesis Output Summary.*

The tool reads the netlist generated by the synthesis tool and performs bitblasting, a post processing design step to sanitize the netlist file and also adds clockgaters based on sequential fan-out in the verilog file. The tool then generates an output netlist that is used for physical implementation. Synthesis output summary in RTL compiler from Cadence is shown in Fig 3.4.

3.2.2. ***Floorplanning***

This step involves ascertaining the area required to implement each TMR module. The TMR blocks only have standard cell logic including TMR flip-flops that span multiple cell rows. The area of the cell is determined by placing a single redundant version of the logic.

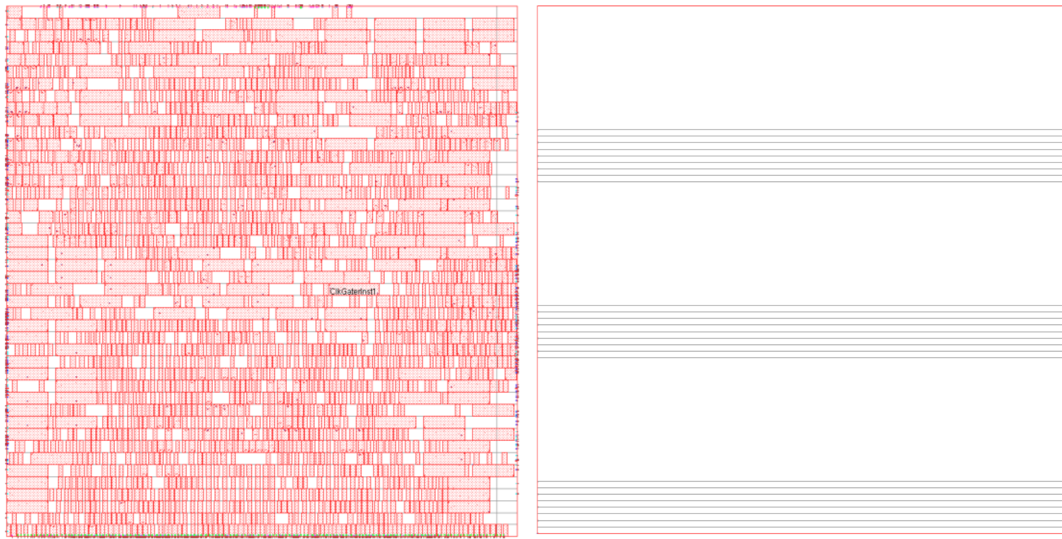

Fig. 3.5 Trial placement to determine area on the left and tool generated placement on the right.

Once the initial area is determined by a trial run, a new floorplan three times the size for the triple redundant module is created with the standard height (120.96u, 241.92u, 362.88u high for the three copies in this case) based on the block to get a size as close as comfortable form factor implementable in top level flow shown in Fig. 3.5.

These three heights are a multiple of 27 standard cell heights. The floorplan is then divided into three stripes of A, B and C regions of logic for the given module that are arranged in 3 possible combinations to provide spatial separation with one row of

filler providing additional isolation on each stripe to protect against multi-node upsets. Example placement possibilities are shown in Fig. 3.6 with the help of a diagram. This step happens by using the 'Spread floorplan' program after saving the example floorplan.

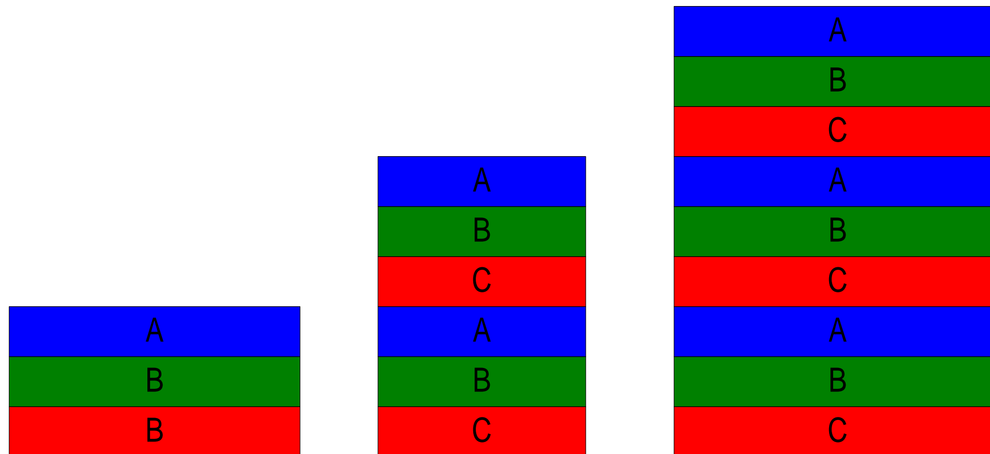

Fig. 3.6. TMR floorplan diagram examples showing separation of 3 standard heights but varying widths.

3.2.3. ***Placement***

During the placement stage, the design standard cells are allowed to place in a single redundant version of the above floorplan. As shown in Fig 3.7, cells are only allowed to be placed in the stripe outlined in bold. Other redundant copies are arranged by the custom in-house tool which copies the placement twice and merges the flip-flops into multi height self-correcting versions. Once the placement is done the quality of placement in terms of congestion and timing are checked and can be iteratively placed until desired goals are achieved.

The module pins are also placed during this step. In addition to the tool's initial pin placement, additional steps are carried out to guide the tool for better results as will be explained in the subsequent chapters.

At this stage, the sequential elements (flip-flops) used for these placements are single redundant versions that are later converted to specially designed triple redundant versions by a special script once clock tree synthesis has been completed. Fig 3.8 shows the single redundant placements of the TMR flow.

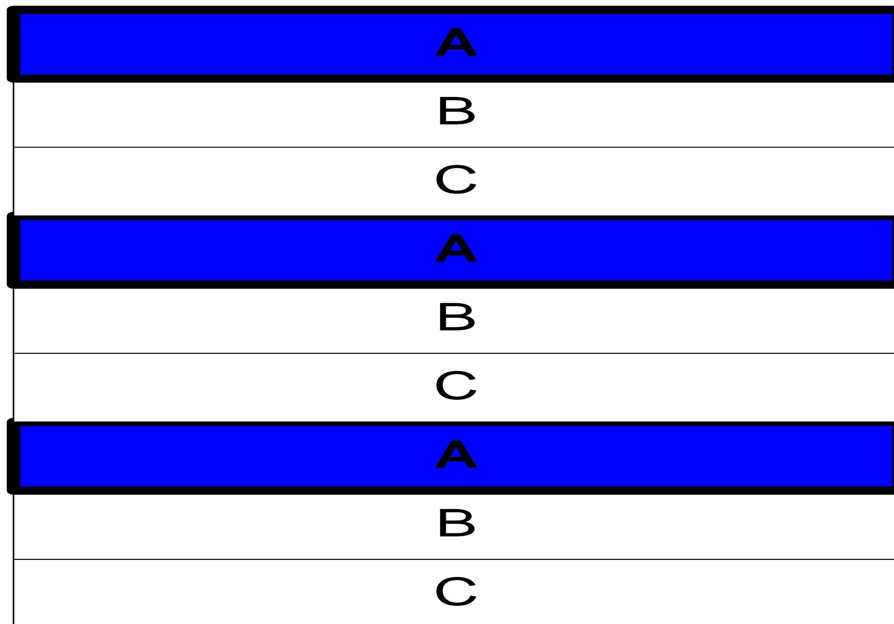

Fig. 3.7 Diagram showing the area highlighted (bold) available for the placement step.

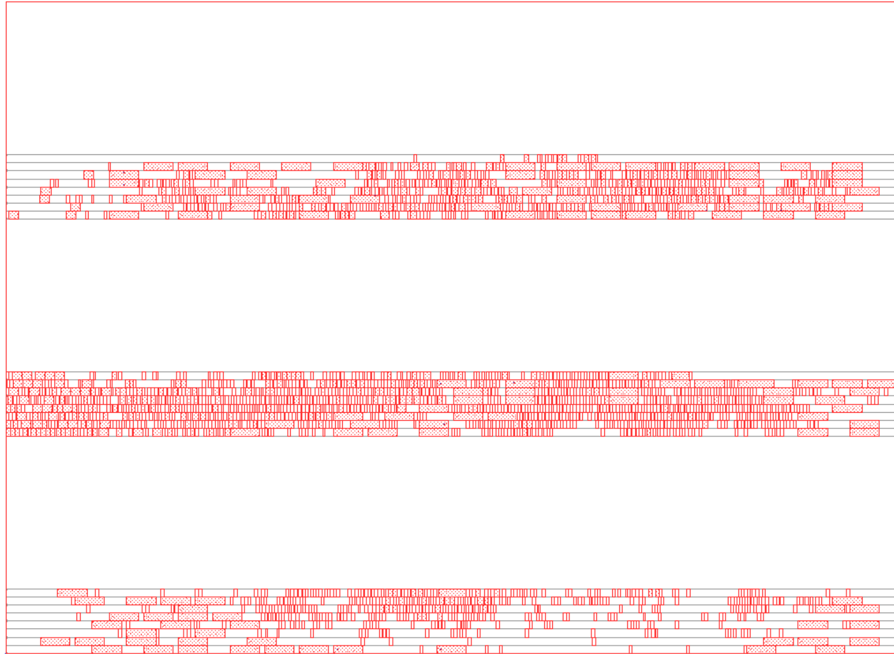

Fig. 3.8 After single redundant placement stage..

3.2.4. ***Optimization***

The optimization step is unchanged from the standard ASIC flow previously

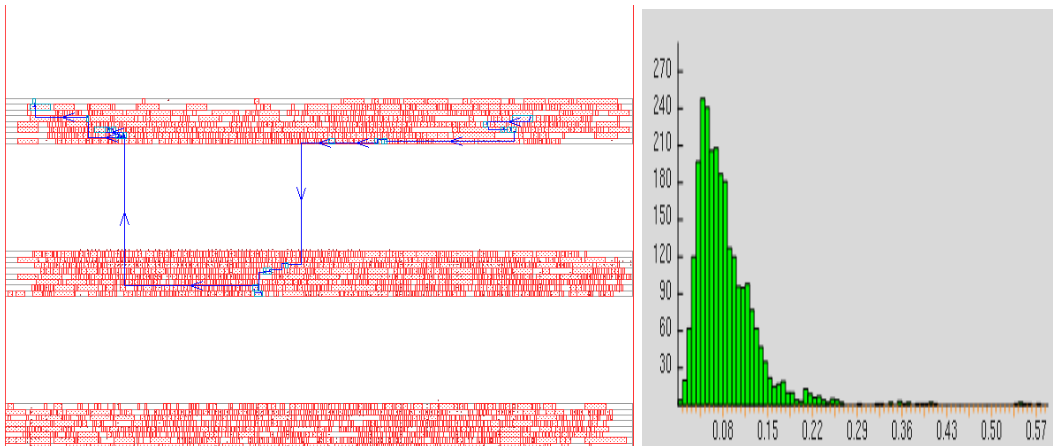

Fig. 3.9 Optimization results setup histogram and critical path.

explained. Fig 3.9 shows pre-CTS WNS histogram and the critical path spanning the A placement copy.

*3.2.5. **Clock Tree Synthesis***

The CTS stage is the same as in the main flow but special care was taken to get the skew in all the blocks to be lower than 20ps. Since the global clock routing will add to the total skew this is required to maintain the global skew of the chip to the +/- 75ps range. Again the clock tree is built for a single redundant version (the A copy) with the pin placed right in the center to achieve lower skew by a more perfectly balanced H-tree. Fig. 3.10 shows only the clock routing for one such block implemented. Sequential elements clearly span all the 3 stripes and hence the skew balancing has to factor this spacing.

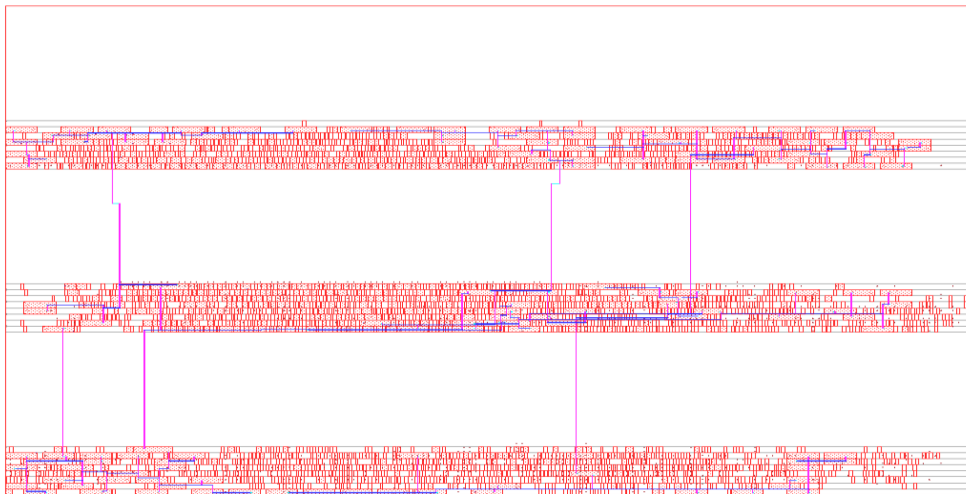

Fig. 3.10 Clock Tree Synthesized design with clock routing only.

3.2.6. ***Post CTS optimization***

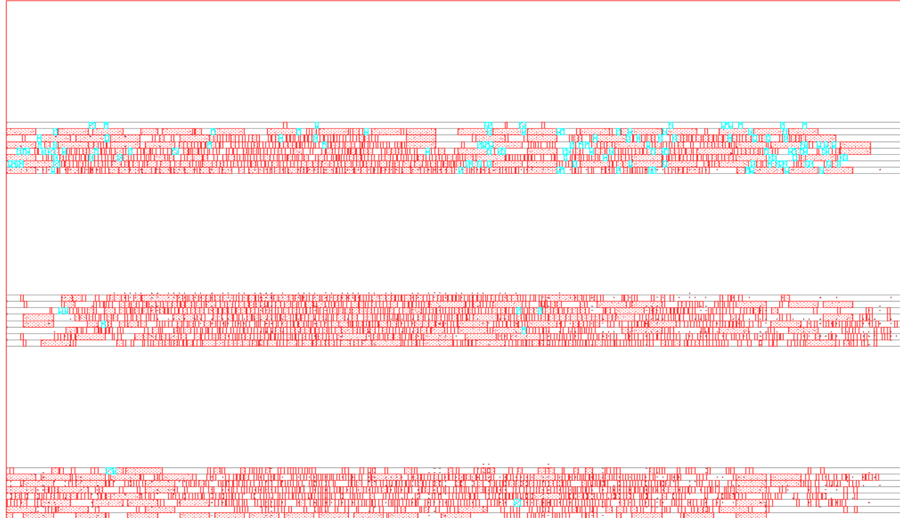

Fig. 3.11 Post-CTS hold fixed with buffer cells in blue

This step is similar to that explained previously where setup, hold and design rule violations (DRV) related issues are fixed in postcts mode with clock tree delays taken into account. The extra min delay margin added during this project for TMR blocks is 150ps. Hold buffers highlighted in bold shown in Fig. 3.11 are added to fix hold violations based on path skew.

3.2.7. ***Triplication***

This special step is added to the conventional design flow for radiation hardening. It involves converting the existing design database to a TMR version of the design as explained previously. The script as part of the in-house tool is shown in Fig. 3.12.

The tool modifies the floorplan, verilog and placement files to provide three redundant copies shown in Fig 3.13. Thus the outputs are the same files with a _TR extension. After triplication the new TMR design is loaded with the new files and we proceed to the next step.

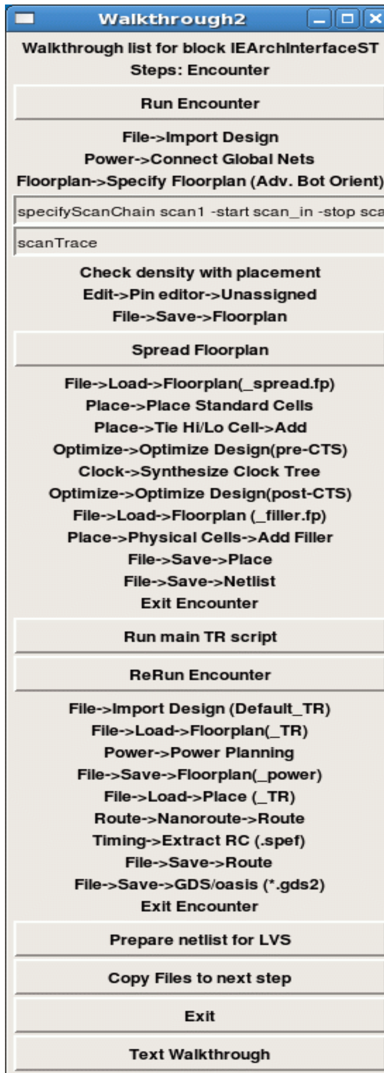

Fig. 3.12 Step 2 GUI of the in-house tool

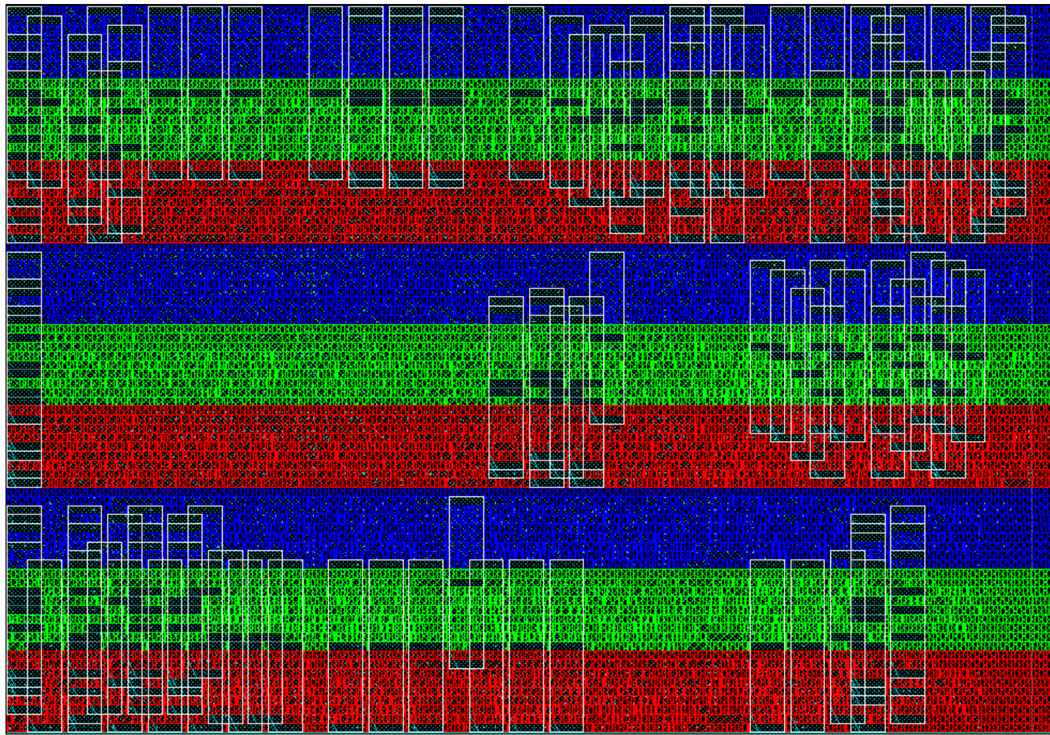

Fig. 3.13 TMR placement generated from the tool and loaded with A, B and C copy and flip-flops highlighted in white.

3.2.8. ***Powerplan***

The next step is power-planning, where the meshes and standard cell rows are created to provide power and ground to the standard cells and TMR flip-flops. Care was taken while designing the power-plan to align it with the power-plan of the flip-flops.

This is done to provide a robust power supply to the gates so as to minimize IR drop without conflicting with the TMR internal routes, which provide the self-correct mechanism's feedback signals. Metal 7, Metal 6, Metal 5, Metal 4 and Metal 3 are used to create the power-plan shown in Fig 3.14.

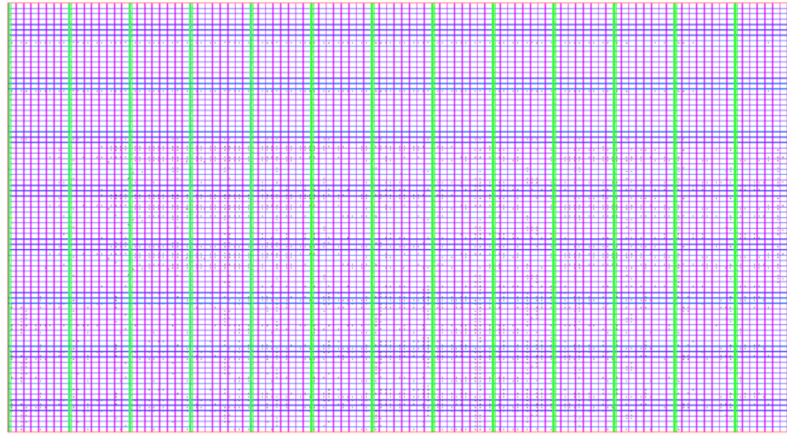

Fig. 3.14 Powerplan created for the TMR floorplan.

3.2.9. ***Routing***

The routing step follows that described in the top-level flow. The global routes are converted into detail routes during this process. Clock routes are given special status and they are routed first. This enables then to have lowest possible delay and skew. Then, the rest of the nets are routed to give the best possible timing delay with no or minimal spacing and short violations.

Figure 3.15 shows the routed block excluding the power routes for clarity. We can see that the routing is much denser in the interior owing to the placed and optimized cells being clustered. The bottom left skew is due to the pin placements.

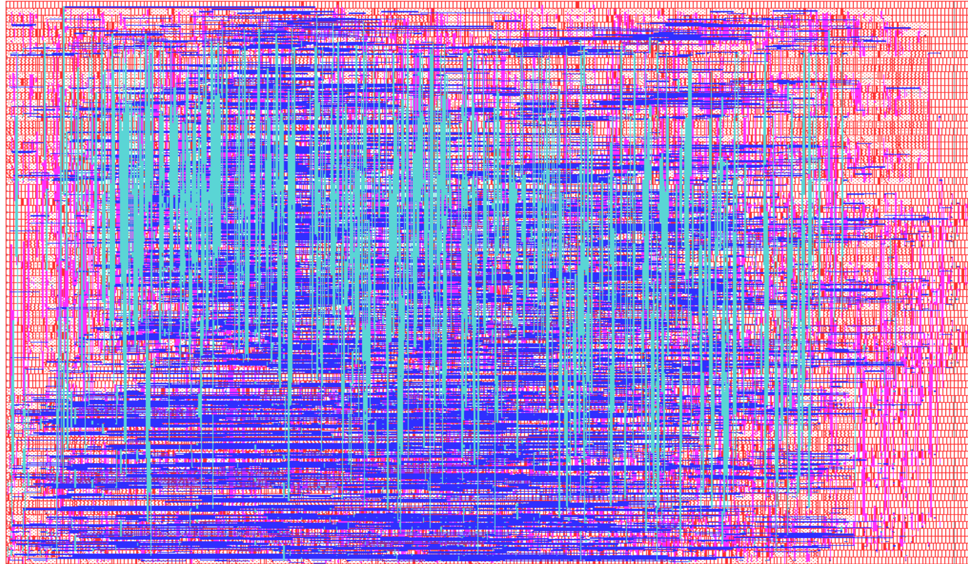

Fig. 3.15 Routing shown for block with 5 metal layers.

3.2.10. ***Extraction and Timing***

Once we have the routed database we extract the database containing parasitic (capacitance and resistance values) to be used for signoff timing calculation in Primetime. The triplicated verilog generated from the tool for the TMR module is passed to Primetime. The timing analysis uses the standard parasitic exchange format (.spef) to estimate the performance of the block, after which we also generate timing library file (.lib), i.e., the abstract timing model of the TMR module. This completed module will be instantiated as shown in the top-level design. The TMR tool provides a custom built GUI on which this process of timing can be facilitated [Hin2011].

3.2.11. ***Abstract Generation***

The GDSII is exported from Encounter and given to the Abstract Generator to create a lef file to be used in the input physical design implementation. The Abstract Generator tool shown in Fig 3.16 uses the GDSII layout and technology information to generate the lef file. Various options are set to generate the lef file in the required format. Details are explained in subsequent chapters.

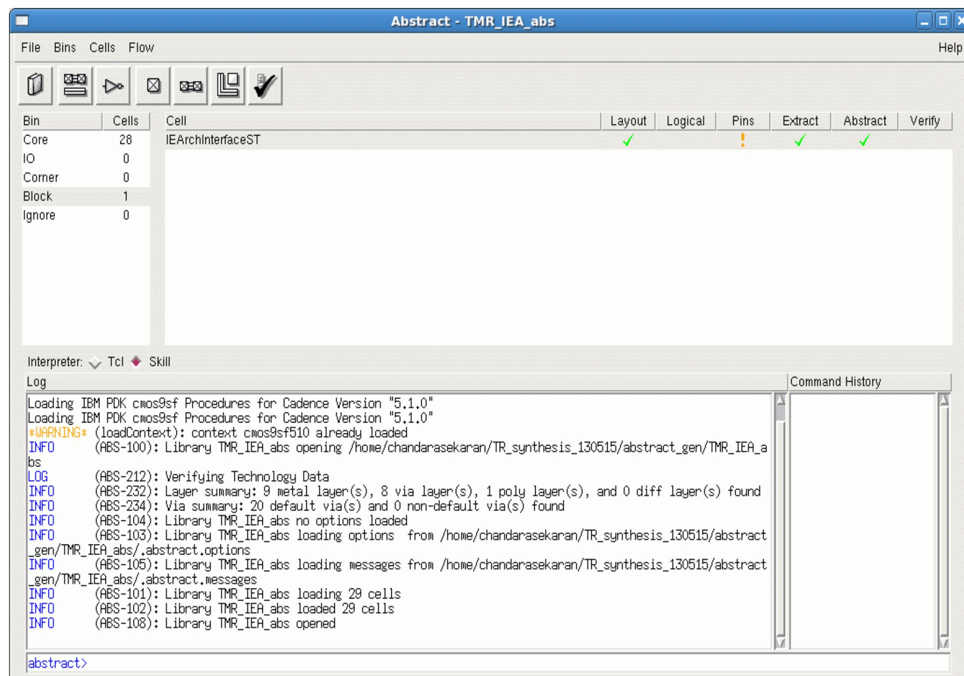

Fig. 3.16 Abstract generator used for creating layout abstract LEF file.

3.2.12. ***Physical Verification***

DRC and LVS for the TMR blocks are verified in the same fashion as explained in the basic flow. GDSII and spice netlists are used by Calibre DRC and LVS tools to run the required checks and verify the database physical quality. Output summary is shown in Fig. 3.17 and Fig. 3.18.

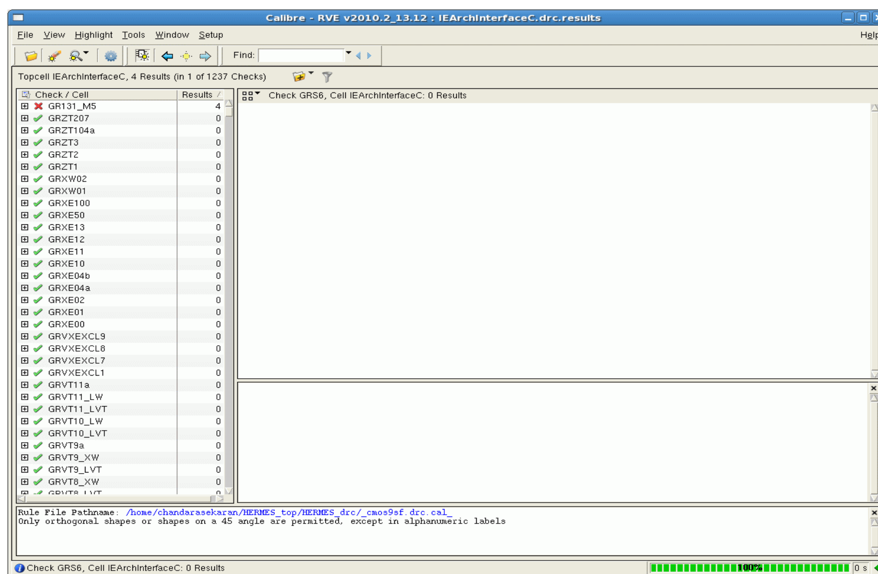

Fig. 3.17 Drc verified using Calibre, Antenna violations present will be fixed at the top level

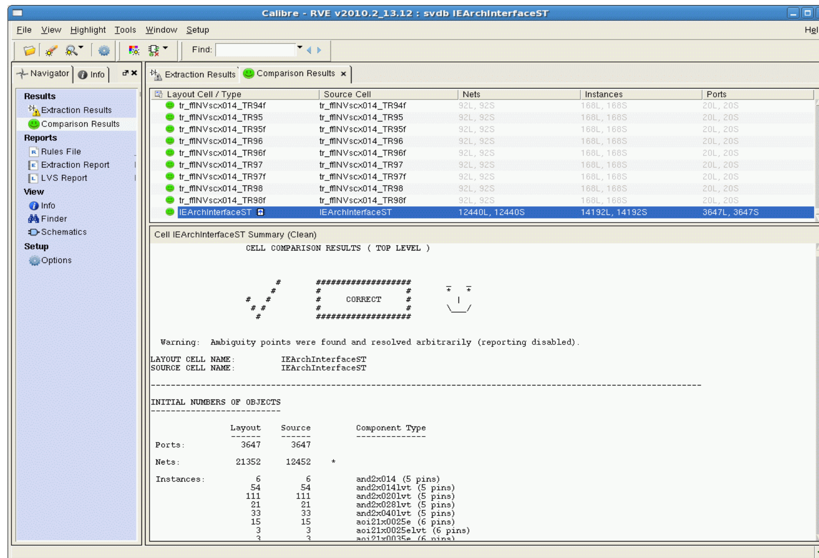

Fig. 3.18 Calibre LVS used to verify layout versus schematic.

*3.3. **Summary***

This chapter explains the conventional ASIC design flow for digital integrated circuit design. Steps of the flow modified by RHBD constraints are explained. For the HERMES processor, the TMR flow described is used to generate 20 TMR modules that are used in the top-level RHBD flow. The .lef, .lib, .gds, spice netlist and verilog files are passed on to the top-level flow and analysis. In course of this work, a number of improvements were made to the original flow to ensure the TMR blocks generated were easy to integrate at the top-level. These improvements are explained in detail and the

impact of the changes to the top-level flow and TMR flow will be explained in next chapter.

CHAPTER 4. TMR FLOW IMPROVEMENTS.

4.1. *Pin Placement algorithm*

Radiation hardened design is primarily obtained by redundancy, either spatial or temporal and in the former case results in increase in the amount of logic in a unit circuit. Moreover, the pin density and routing density also increase. Since the successful physical realization of any logic circuit is dependent on these factors, pin placement and routing are bottlenecks to fast design closure. A design that cannot be routed successfully is physically unrealizable.

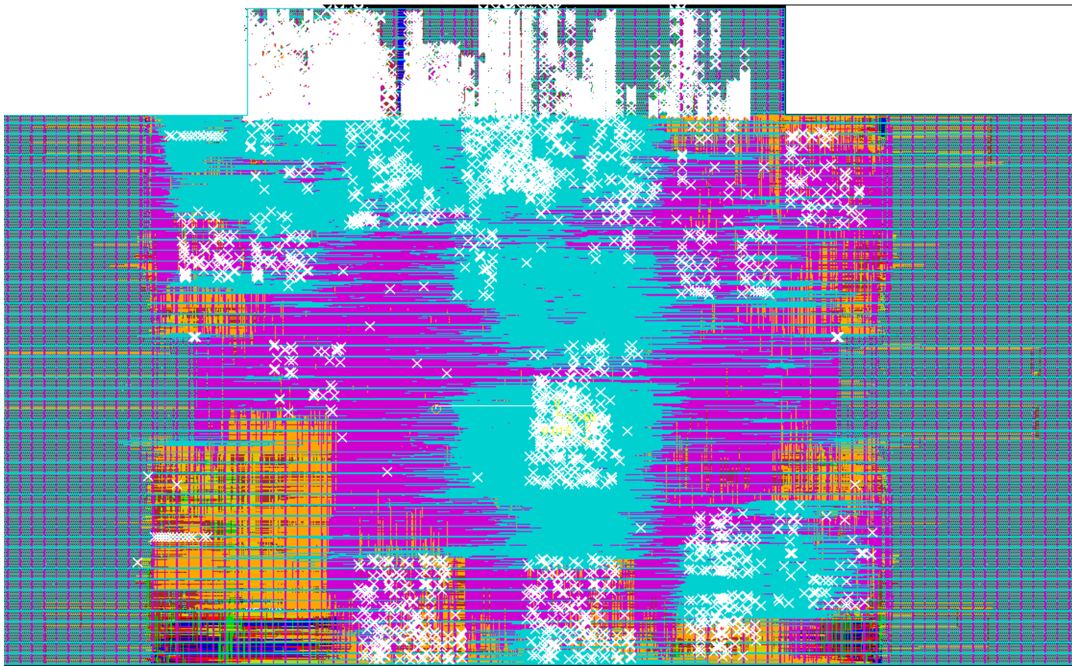

Fig. 4.1 Top level drc's without pin placement program generated .lefs.

We encountered the problem of large design rule check (drc) because of the above reasons while dealing with TMR blocks implemented at the top-level. The number of

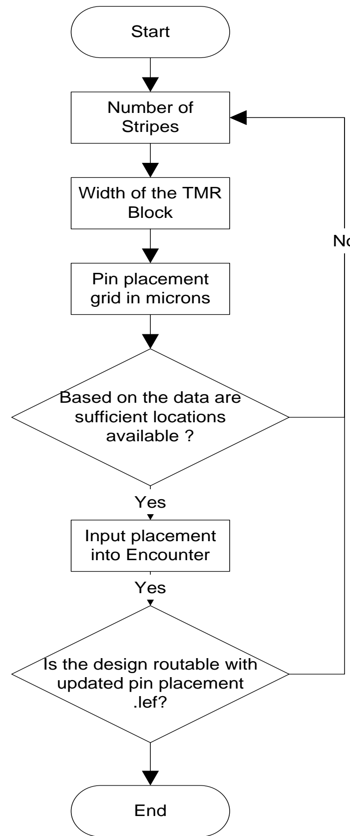

Fig. 4.2 Pin placement algorithm flowchart.

drc's at the top is directly proportional to the routing density at the lower metal layers and the pin density in TMR blocks. Fig 4.1 shows a snapshot of the top level routed design which shows more than 10000 drc's, which makes the design unrealizable.

Further analysis of the violations showed that the pins placed were not on the routing grid and thus made it harder for Encounter to route to. To solve this problem we devised the pin placement algorithm which created optimum pin placements at properly

spaced routing grids thus facilitating routing. A flowchart describing the program is shown in fig 4.2.

Inputs to the program are initial seed placement based on the fan-ins and fan-outs of the TMR block in question shown in Fig 4.3. This is written out from Encounter in a

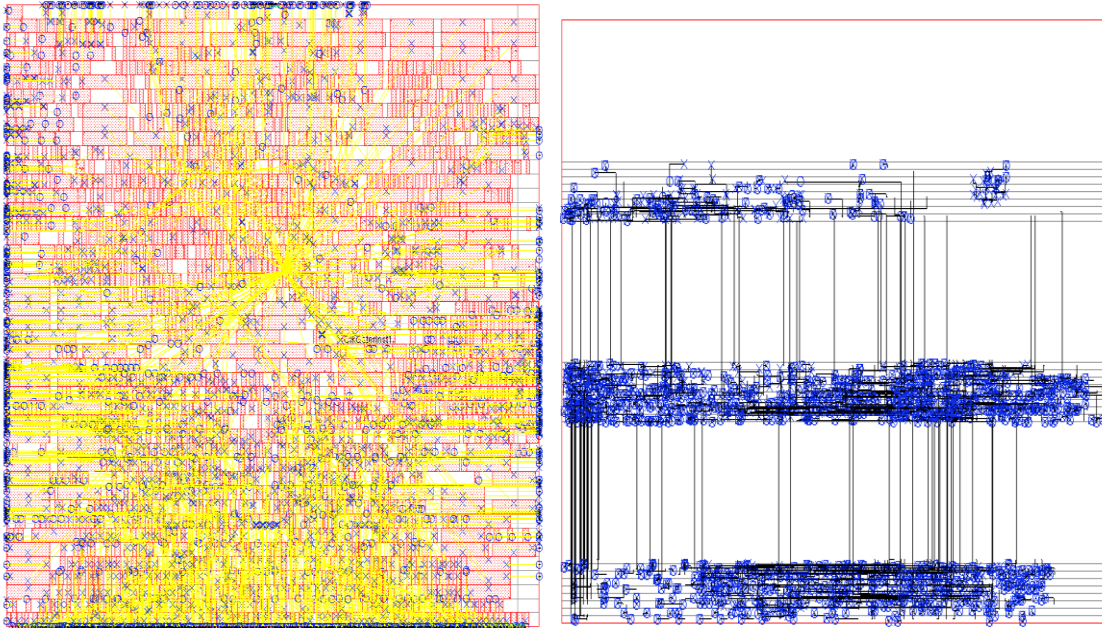

Fig. 4.3 Initial tool placement and initial script based placement seed input for script.

text format. We specify the number of stripes in the TMR block (1/2/3 as described in chapter 3). We also specify the grid on which these pins will be placed in microns. Iteratively this could be increased if the routing congestion was still high at the top level. The pin placement program creates optimal pin placement locations in text format. These pin placement are spaced apart to facilitate congestion free routing.

The pin placement program had to be integrated into the specialized TMR flow since the TMR flow operates at the single and triplicated domains. Thus the input to the

Fig. 4.4 Single Redundant placement output from pin placement program.

program was a single redundant pin placement copy which contained only the A copy pins. The pin placement program gives us two text files as outputs which are single and triple redundant pin locations for the TMR design.

Single redundant version is used before triplication and triple redundant version after triplication has been achieved. This ensures that optimum placements are not changed between triplication. Figure 4.4 and 4.5 show the single and triple redundant pin placement outputs from the program.

Figure 4.6 shows a cartoon representation of the input to and output from the program. We can see that the output is well spaced pin placements which proved to facilitate routing.

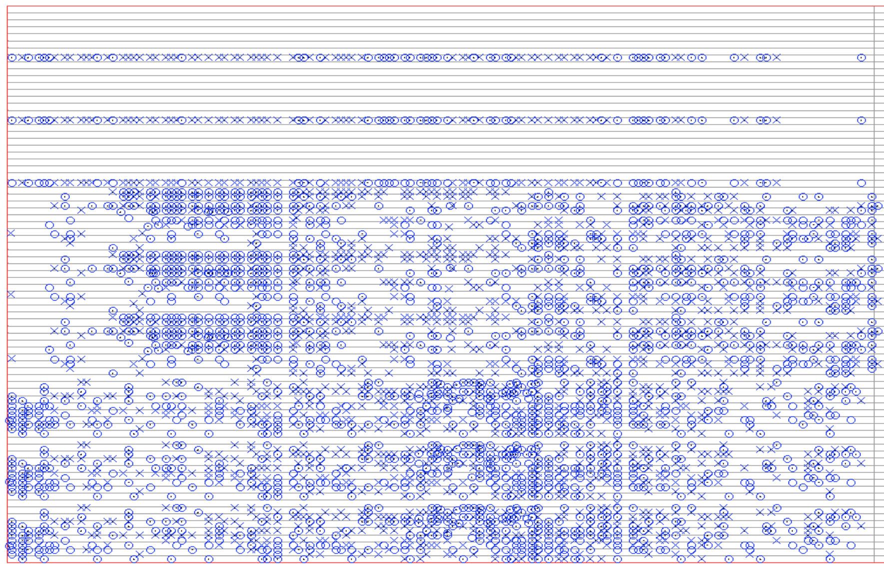

Fig. 4.6 Triple redundant pins output from pin placement algorithm

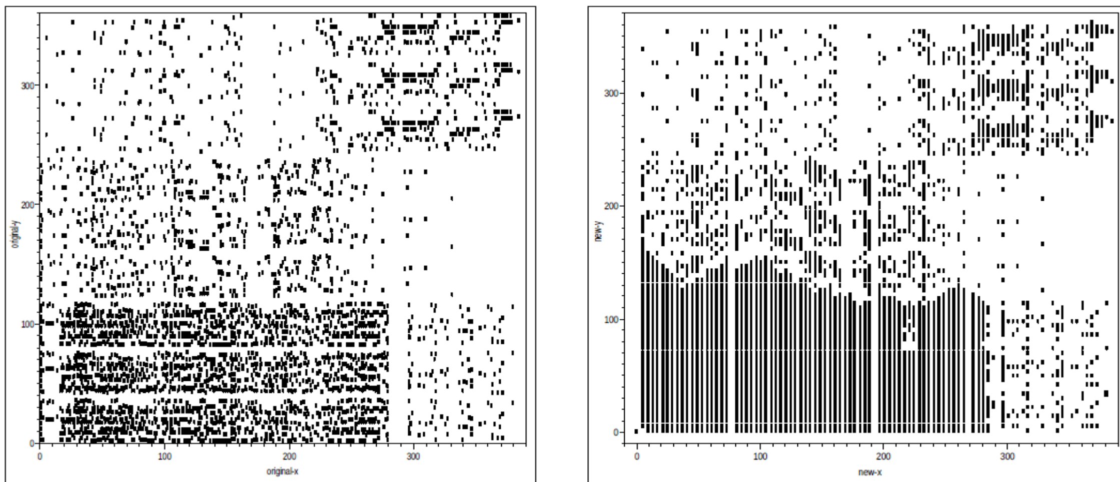

Fig. 4.5 Figure for initial seed to the script and output from script on the right

The blocks are then implemented and tested with these new pin locations. The result of the placement algorithm is that many of the TMR blocks with high pin density which were un-routable previously, were routable with the new .lefs generated with this

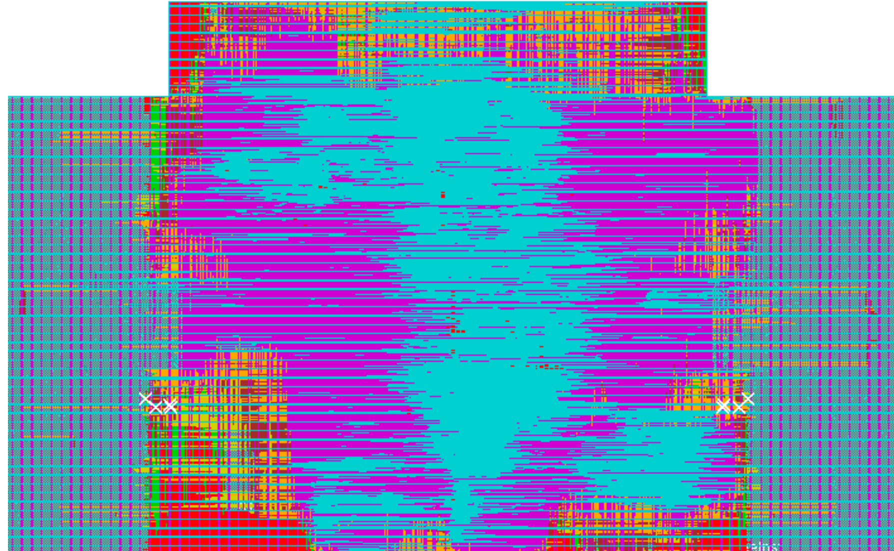

Fig. 4.7 Top level routed design using pin placement program generated .lefs.

pin placement flow.

Fig 4.7 shows the top level routed design with very few design rule check violations, and these violations are not on the TMR blocks. All the TMR blocks were clean of design rule violations at the block and top level with this pin placement implementation.

4.2. ***Clockgater Insertion***

Clock gating cells are inserted to reduce clock switching power in any design. They reduce activity factors on clock node which otherwise would switch every cycle.

Clock nodes are high capacitance nodes in the design hence gating the transition on them is an effective strategy. This results in reduction of switching power dissipation giving considerable power savings. The insertion is controlled by the synthesis tool which looks for a template containing a clock ANDed with an enable signal and a latch and then converts it into an integrated clock gating (ICG) cell. This happens in a seamless fashion but in the HERMES the clock gating cells are explicitly coded because the RTL compiler

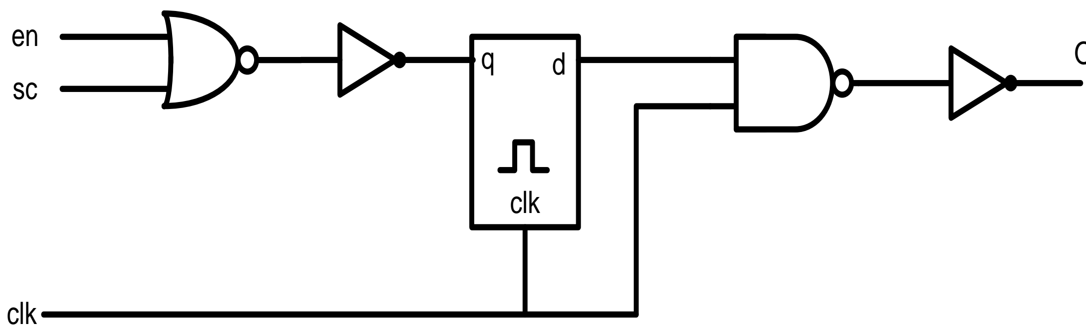

Fig. 4.8 Clockgater cell schematic explicitly coded in the RTL.

was unable to automatically infer these cells. Cloning of the clock gate cells, which is essential for fixing maximum capacitance and fanout violations is done in Encounter with correct clock loads in place. This ensures the best possible clock gating insertion without formal verification issues. Shown below is the structure of the clockgaters used in the design. Fig. 4.8 shows the schematic of such a clock-gating cell.

4.3. ***Power-plan Alignment***

In the implementation of the TMR blocks the power-plan needs to be perfectly aligned with the TMR flip-flop internal VDD/VSS connections. Exact overlap ensures

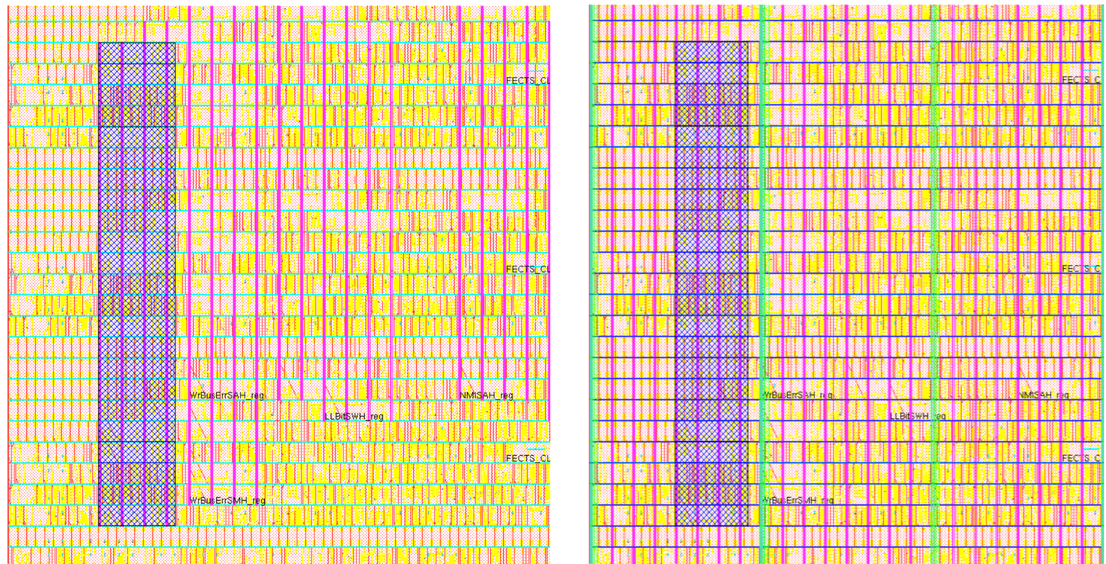

Fig. 4.9 Powerplan TMR cell alignment.

the best possible power connection as opposed to the connection that happens through the standard cell rail. The alignment is ensured by the placing the TMR flip-flops on the same grid as of the power-plan, which is set to 8.96µm increments.

Figure 4.9 shows the alignment of the power-plan to the internal power pin. The top-level power plan grid also needs to align with the TMR block grid on metal 7 as explained in the top-level flow.

4.4. ***Antenna Fixing***

Plasma etching process of the wafer fabrication is used. It has a side effect wherein the etching results in charge collection on the metals and hence charging damage. The conductors, which are supposed to be connected to the poly shown in Fig. 4.10, are not connected until later in the metallization process. The lack of a connection to the substrate results in the metal acting as a capacitor and the charge builds up during the plasma etching process.

Charge on the metal conductors connected to a gate results in high stress fields on the gate oxide and can result in the damage of the oxide and eventually, a gate-oxide punch-through. These effects on the reliability of the circuit are called antenna effect. It can be fixed by either metal layer hopping when they are beyond a certain ratio given by equation (4.1), thus never building enough charge to destroy the gate. It can also be fixed by addition of an antenna diode which can provide a path to ground during a high current

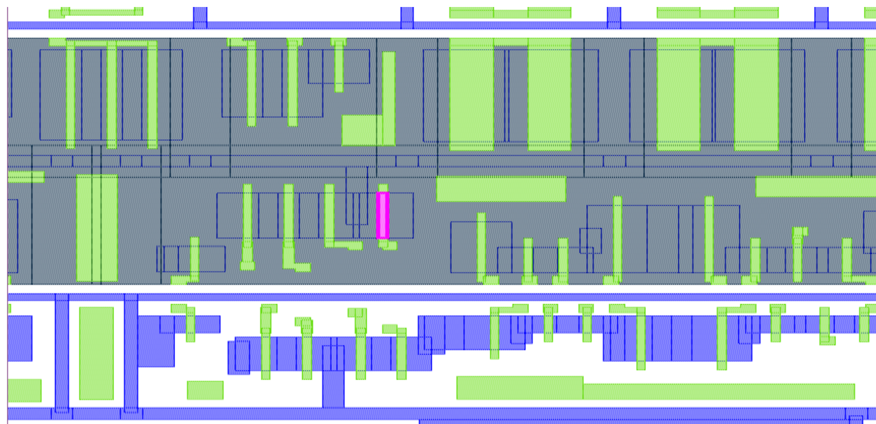

Fig. 4.10 Antenna violation highlighted in pink in Calibre DRC.

event to alleviate the antenna effect. We used both fixes in this design. The equation below defines a violation.

$$\text{AntennaAreaRatio} = \frac{(\text{Area of the metal connecting the gate oxide})}{(\text{Area of the gate oxide over the channel})} \tag{4.1}$$

For every process the given antenna area ratio cannot be violated thus either by layer hopping or by insertion of the antenna diode this violation is fixed. This information is calculated from the technology lef file and the cell and macro lef files. Without this information it would not be possible for the tool to calculate these errors. The Abstract generator flow had to be modified to give the correct values to the fields of

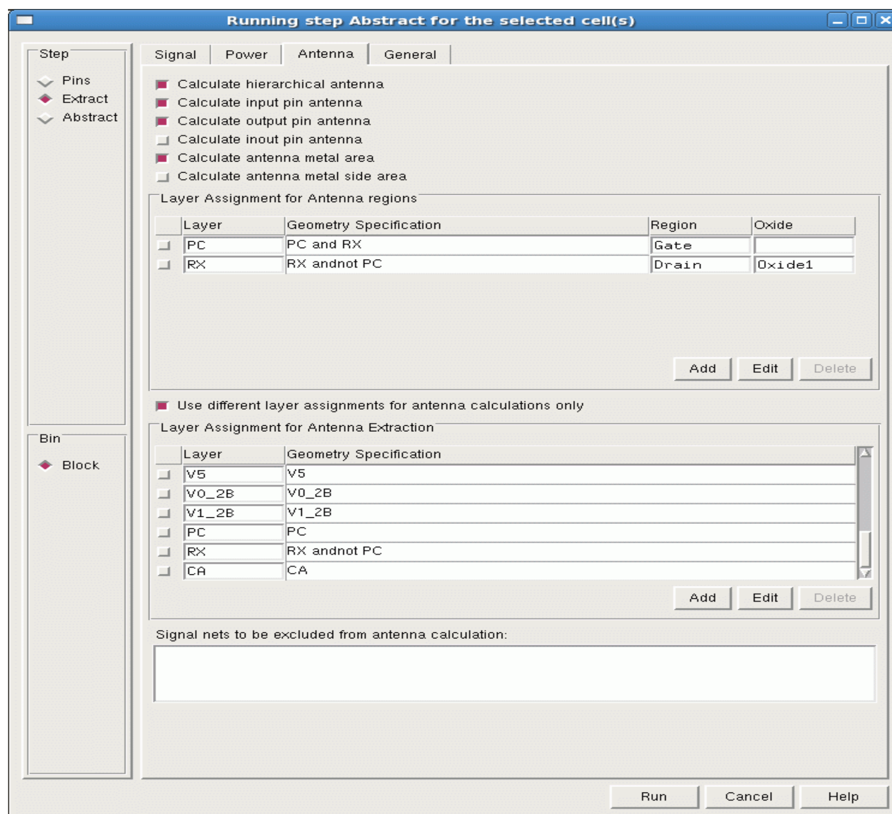

Fig. 4.11 ANTENNA Calculation options and setting in the abstract generator.

ANTENNADIFFAREA and ANTENNAGATE in the lef files, shown in fig. 4.11. Figure 4.12 shows a portion of the lef file that shows this information. The Abstract generator tool options had to be tweaked to calculate the values correctly.

```
    CLASS CORE  ;
    SIZE 0.32 BY 4.48 ;
END CoreSite

SITE CoreTRSite
    SYMMETRY Y  ;
    CLASS CORE  ;
    SIZE 9.6 BY 4.48 ;
END CoreTRSite

MACRO xor2x010
    CLASS CORE ;
    FOREIGN xor2x010 0 0 ;
    ORIGIN 0.00 0.00 ;
    SIZE 6.40 BY 4.48 ;
{   SYMMETRY X Y ;                               }
    SITE CoreSite ;
    PIN o
        DIRECTION OUTPUT ;
        ANTENNADIFFAREA 1.86  LAYER M1  ;
        PORT
        LAYER M1 ;
                POLYGON  6.20 3.17 6.16 3.17 6.16 3.73 6.20 3.73 6.20 3.89 6.16 3.89
                 6.16 4.24 6.00 4.24 6.00 3.17 1.44 3.17 1.44 1.52 1.05 1.52
                 1.05 1.08 1.21 1.08 1.21 1.36 1.60 1.36 1.60 3.01 4.42 3.01
                 4.42 2.73 4.58 2.73 4.58 3.01 5.23 3.01 5.23 2.73 5.39 2.73
                 5.39 3.01 6.00 3.01 6.00 0.88 5.03 0.88 5.03 0.72 6.00 0.72
                 6.00 0.24 6.16 0.24 6.16 2.73 6.20 2.73 ;
        END
    END o
    PIN b
```

Fig. 4.12 Portion of Lef file with Antenna fixing information.

*4.5. **Clock Tree Synthesis***

The clock tree synthesis explained in the flow section was implemented in the TMR blocks. A TMR clock tree as implemented in the TMR flow is different from a normal clock tree since the cells can sit in 3 spatially separated portions of the chip for a single copy of the TMR circuit.

The clock tree is also implemented with very low skew target of 20ps. The clock pins for the TMR blocks were placed on top of the block instead of the periphery just like other pins, though close to the center to facilitate ideal H-tree implementation shown in Fig. 4.13. Clock-gates inserted during synthesis were cloned to provide best possible placement and skew. To this effect clock gate aware placement was run which assigns clock-gates based on their sequential loads. After the triplication the clocks are routed first to ensure shortest possible routes, consequently minimize skew.

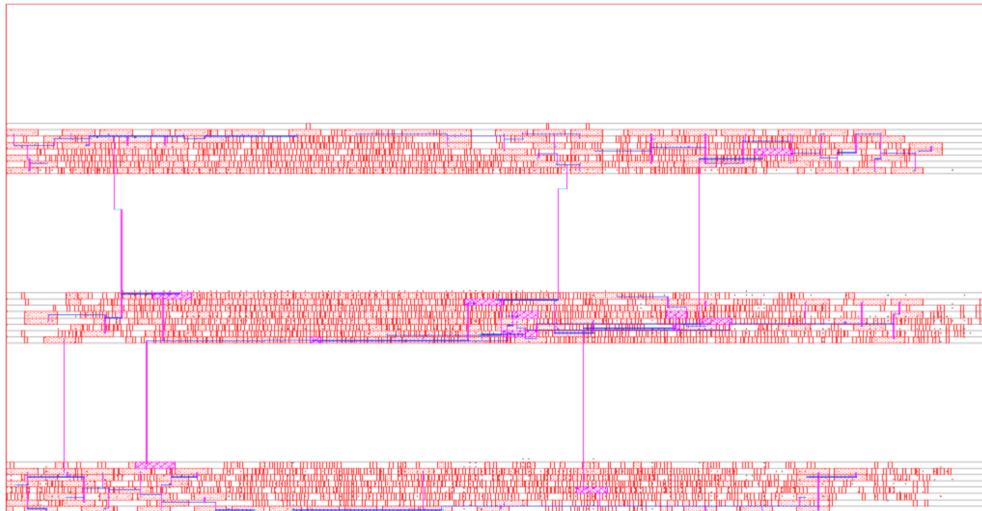

Fig. 4.13 CTS cells with CTS only route and CTS cells in pink.

Non default rules were used for these routes which use double width and double spacing rules to provide a robust clock tree with protection against signal integrity (SI) issues. The triple redundant version shows three wires with almost identical clock routing shown in Fig. 4.14. Clock trees with less than 20 ps skews were built for all 20 TMR blocks and the summary is shown in the table 4.1.

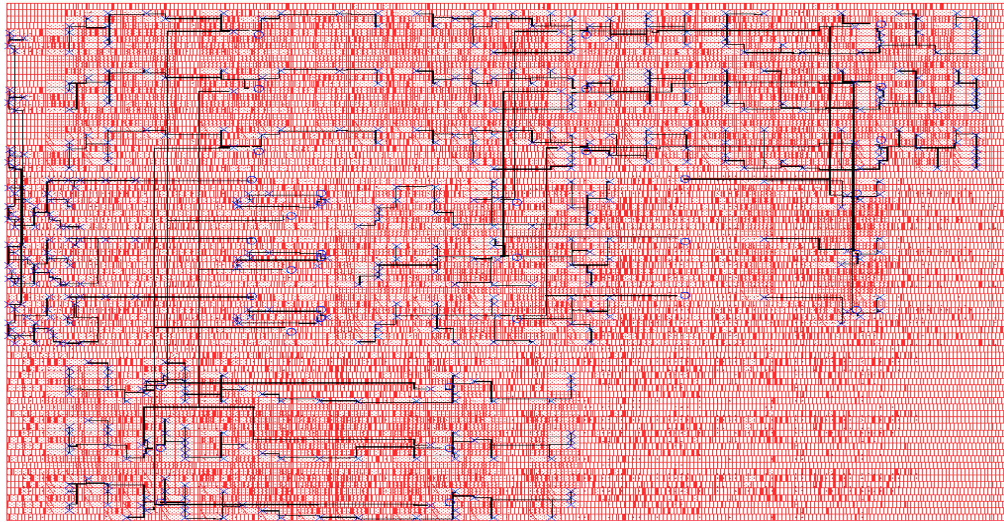

Fig. 4.14 Symmetrical Clock Tree routing shown for all 3 logic copies in TMR block.

Blocks	Max Skew (+/-)(ps)	Internal Stages	Min Latency(ps)	Max Latency(ps)
BIUInst/BICtlAdrDatCInst	15	8	220	240
C0UInst/C0CacheRegsCInst	10	6	140	160
C0UInst/C0ErrorRegsCInst	5	4	140	160
C0UInst/C0RFRegsCInst	5	4	170	190
C0UInst/C0SystemRegsCInst	5	4	120	140
C0UInst/C0TLBRegsCInst	10	6	150	170
C0UInst/C0TimerRegsCInst	10	8	180	290
DCUInst/DCFBDualTripleCrossCInst1	10	6	180	200
DCUInst/DCFBDualTripleCrossCInst2	10	6	180	200
DCUInst/DCFillBufCInst	10	6	170	190
DCUInst/DCStoreBufCInst1	10	6	170	190
DCUInst/DCStoreBufCInst2	10	6	170	190
IEUInst/IEArchInterfaceCInst	10	6	160	180
IEUInst/IEDualTripleCrossCInst1	10	6	140	160
IEUInst/IEDualTripleCrossCInst2	10	6	140	160
IEUInst/IEPCPipeCInst	10	6	150	170
IFUInst/IFDualTripleCrossCInst1	10	6	160	180
IFUInst/IFDualTripleCrossCInst2	10	6	160	180
IFUInst/IFFillBufCInst	10	6	190	210
MDUInst/MDCtlCInst	10	6	150	170
MDUInst/MDDualTripleCrossCInst1	5	4	140	160
MDUInst/MDDualTripleCrossCInst2	5	4	140	160

Table 4.1 Clock skew summary for TMR blocks

4.6. ***Summary***

Specific RHDB constraint driven improvements made to the TMR implementation flow and the results of such changes have been elucidated in this chapter. These changes enabled the successful implementation of the reference design HERMES and laid guidelines for future implementations.

CHAPTER 5. TOP LEVEL IMPLEMENTATION

This chapter discusses the physical implementation of the radiation hardened HERMES microprocessor. The HERMES hierarchy is shown below in Fig. 5.1 where custom blocks are shown without bold boundaries. The techniques specific to an RHBD flow are elucidated along with their effects on radiation hardening. Custom hard blocks are shown highlighted in bold. The top-level implementation is an integration of the TMR and DMR logic hardening methodologies.

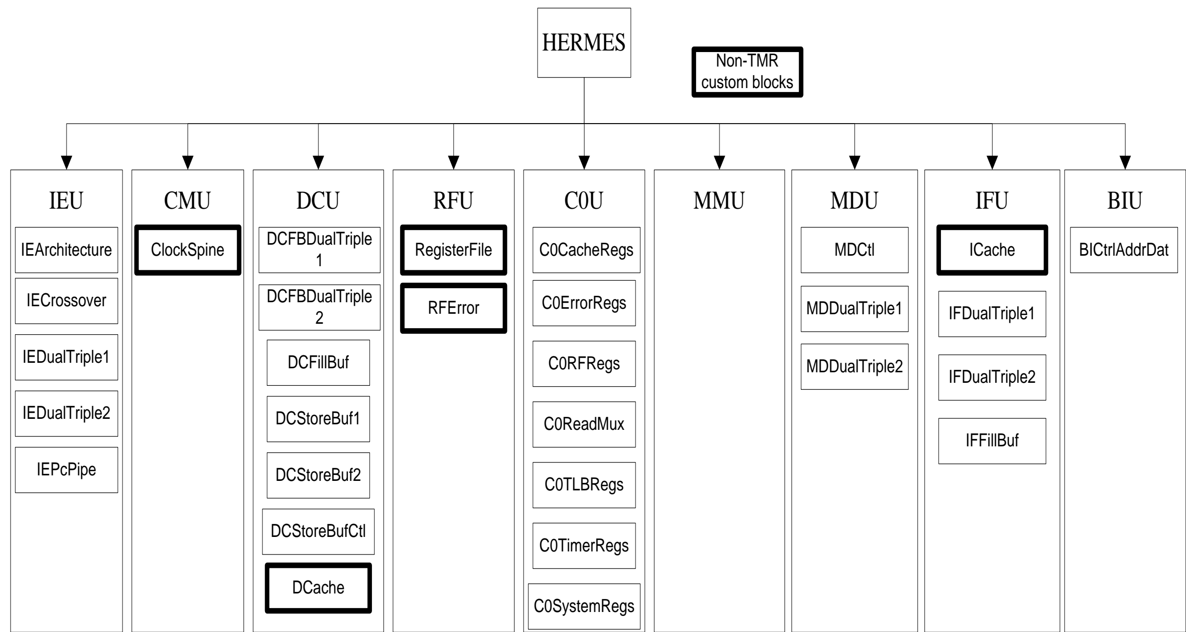

Fig. 5.1 Hierarchical Modules in HERMES along with the macro blocks

There are crossover circuits which check correct signaling between the DMR and TMR logic. Cache, clock-spine and register file blocks are the hard blocks that were described in detail in chapter 2. The primary goals of maximizing speed while maintaining rad-hardness were the primary constraints to the top level implementation.

Other ASIC chip-level constraints like area (placement and routing densities) and power were secondary optimization goals. Core area of Hermes (excluding the pads) is 11.22mm^2.

5.1. Synthesis

```
       Operating conditions:   typical (balanced_tree)
Wireload mode:          enclosed
  Area mode:              timing library
============================================================
        Timing
        ------
 Slack                   Endpoint                  Cost Group
---------------------------------------------------------------------
 -148ps DCUInst/DCCtrlSDInstB/nWrW3B1ParitySMH_reg/d GClk


               Area
       ----
Instance  Cells  Cell Area  Net Area   Wireload
----------------------------------------------------
HERMES    44401    3113797         0     <none> (D)
 (D) = wireload is default in technology library
        Design Rule Check
        -----------------


       Max_transition design rule: no violations.
Max_capacitance design rule (violation total = 1072.8)
Worst violator:
Pin                                   Load (ff)           Max     Violation


----------------------------------------------------------------------------


IFUInst/ICacheInst/TagWay2HitBCXH          168.6         100.0          68.6


Max_fanout design rule: no violations.
```

Fig. 5.2 Output summary from synthesis.

Synthesis, as explained in chapter 2, is the primary step for the ASIC flow. The top level RTL, along with the design constraints, are input into the synthesis tool RTL Compiler for synthesis. TMR and hard block library files are provided as inputs. The technology information is provided as standard cells' library files, which are used to map

the design to technology specific gates. Options for high effort timing optimization are used. The primary synthesis objective is speed and we extensively used low voltage threshold (LVT) libraries that are the fastest gates but with a penalty of higher leakage. The synthesized design has a standard cell area of 3325494um^2. The post-synthesis WNS is 1.453ns, which translates to a speed of 690Mhz. TMR block library speeds are re-checked in the top level synthesis and excessively slow TMR blocks were re-synthesized to optimize the ‘input to register’ and ‘register to output’ paths. Iteratively, the speed of multiple TMR blocks was improved based on multiple TMR level runs. Constraints were tweaked to achieve the same by constraining the ‘input to register’ and ‘register to output’ paths on slow blocks. Fig.5.2 gives the textual summary output from RTL compiler that gives statistics of area power and timing.

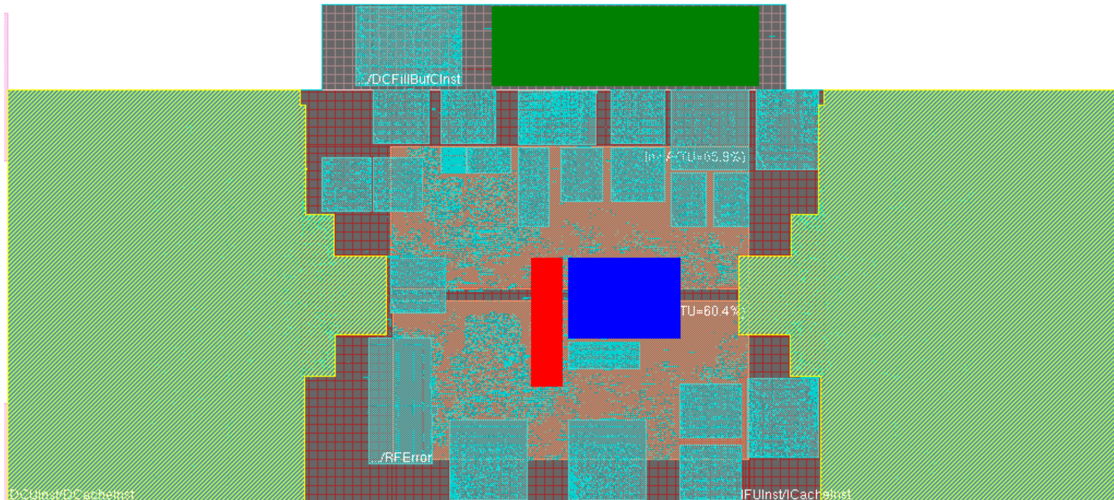

Fig. 5.3 Floorplan showing placement of all macros in the design.

5.2. **Floor-planning**

Floor-planning is the process of placing hard macros in Encounter to achieve an optimal timing and congestion performance. Since all commercial chips are expected to meet a speed requirement, timing is the primary driving force in floorplanning. Designs that cannot be routed repeatedly by the automatic place and route (APR) tools, like Encounter, cannot be reproduced in a given time-frame with confidence. Thus floor-planning also concentrates on congestion alleviation aspects. In the floorplan shown in Fig. 5.3 we can see that clockspine (red), which is the central clock unit, is placed in the center to facilitate a low skew clock tree. The module BIU (green) is the bus interface unit and is the main I/O block; therefore it is placed on the periphery to facilitate easy bus access.

The IEArchitecture block (blue) interacts with all blocks in the design and is placed in the center to minimize cross-chip routing. The caches are the biggest blocks on the chip and are placed on the sides to ensure easy data flow (yellow). This also matches their interfaces. These decisions were taken over a period of multiple floorplan iterations and placement runs to arrive at the final floorplan.

Floorplanning cannot be done in isolation without doing placement and global routing to estimate timing and congestion, therefore they are inter-dependent processes. The floor-plan in Fig. 5.3 shown contains two fences (orange) and some other density screens used to guide placement. These will be explained subsequently. Power-planning is also part of floor-planning since the placement of hard blocks is in part driven by the

power-plan as well. As shown in the Fig. 5.4, the internal power-plan is on a grid that is a multiple of the top-level power-plan grid. The blocks get a certain number of valid placement locations that are multiple of the power-plan grid on metal 7 that is equal to 38.4um.

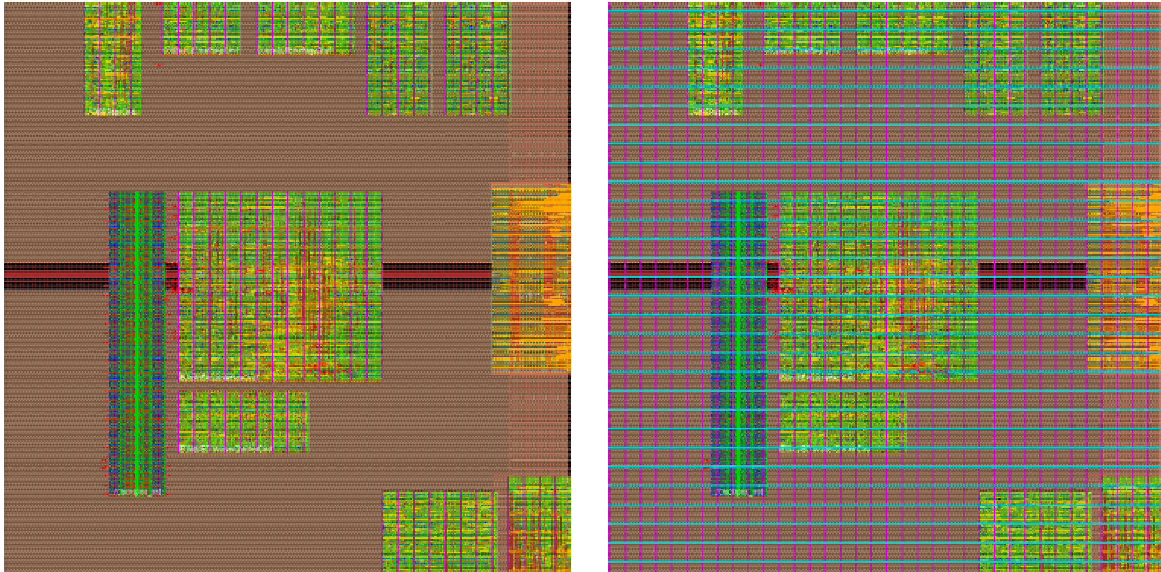

Fig. 5.4 Shows a power-plan with TMR blocks connected by overlap and other blocks getting via towers for power from a Metal 8 -metal 7 grid.

The alignment of power-plan serves two-fold purpose. It ensures that there is a robust power connection to the TMR blocks. Secondly, the top-level connections are determined by overlap and not by any explicit connection between the macro and power-plan grid.

5.3. ***Placement***

Placement is the process of placing standard cells. For the rad-hard implementation of a chip, logic functionality is of utmost importance. We have already

explained that the TMR logic has separation and voting to correct errors. The top level logic in the design is implemented as DMR logic as explained in the first chapter. DMR logic has two copies of the same circuit to ensure an error flag when the outputs of the logic do not agree.

It is therefore important that logic be separated spatially to ensure hardness and

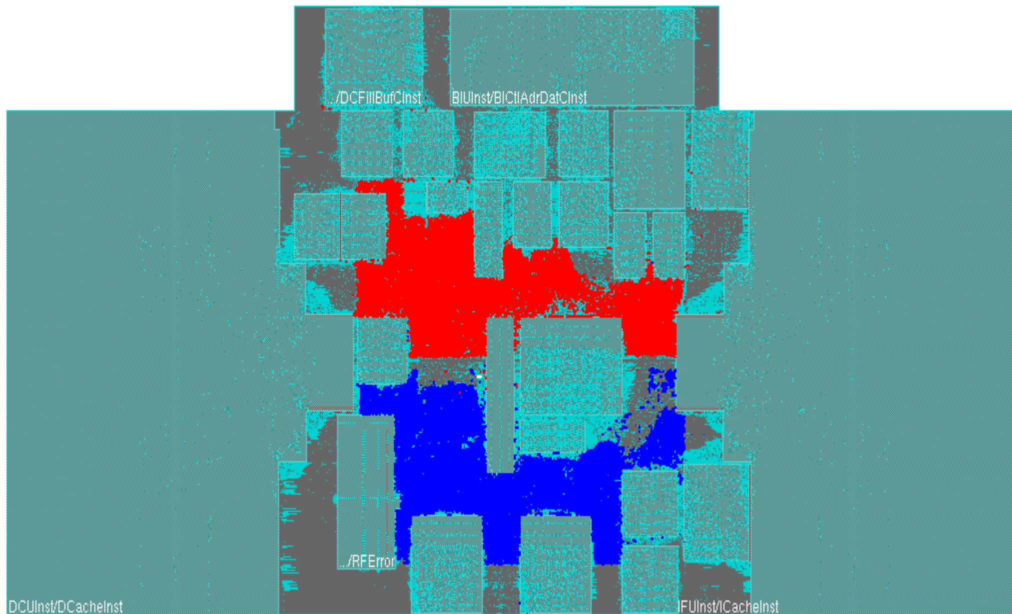

Fig. 5.5 DMR A (red) copy and B (blue) copy separation at placement stage.

protect against multiple node collection upsetting the redundant logic cone. Figure 5.3 shows an orange fence which ensures that logic can only have restricted placement. In this case, they are the two redundant copies of the DMR logic. Figure 5.5 shows the two modules highlighted n red and blue with exclusive placement within the A and B fences.

We can see A and B copy of the DMR module logic in Fig. 5.6 within fences that are separated by two rows of decoupling capacitors (decap) cells to ensure hardness. It

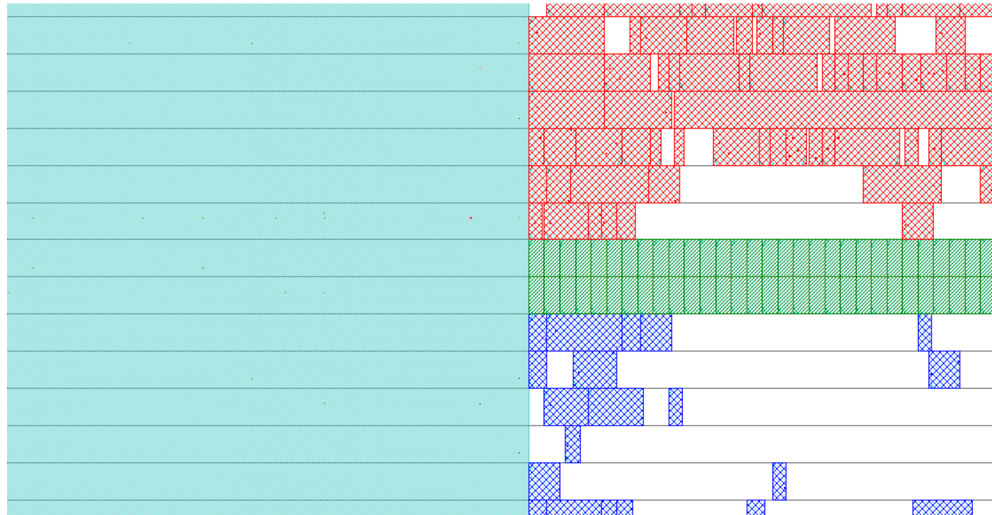

Fig. 5.7 A (red) and B (blue) separation with 2 rows of decap(green).

has been proved on silicon that two rows equivalent of cell separation by decap can be effective against MBU's. Other screens include buffer only screen where we can constrain the placer to have only buffers placed in an area.

Placement blockage screens (dark red), which disallow placement in those region,

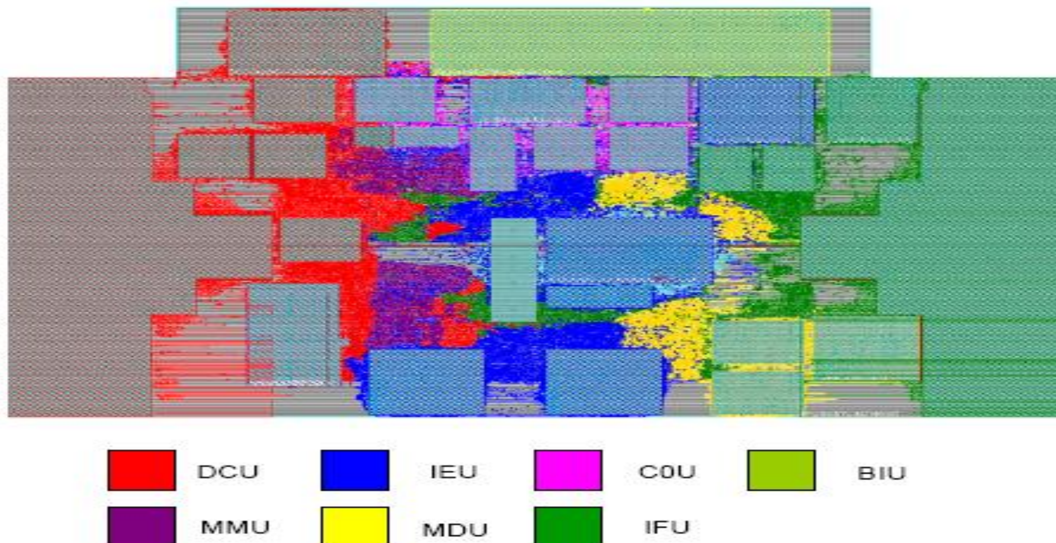

Fig. 5.6 Module segregation

guides the placement algorithm. Density at placement is usually maintained at 65% to allow sufficient area for subsequent steps. However, in HERMES we start with a fence density of around 50% since greater area is required for optimization and clock tree synthesis.

Logic A and B interacts with most TMR blocks and hence the routing congestion is higher than in a conventional design. Figure 5.7 shows the logical separation of the architectural modules. Based on this the quality of module placement can be ascertained.

*5.4. **Pre-CTS optimization***

Pre-CTS optimization, as explained in chapter 2, involves optimizing timing of the design in terms of setup and hold violations. Top-level optimization adds/modifies around 9684 cells out of 90762 during pre-CTS optimization. Fig. 5.8 shows standard cells added and modified at the pre-CTS optimization stage in yellow.

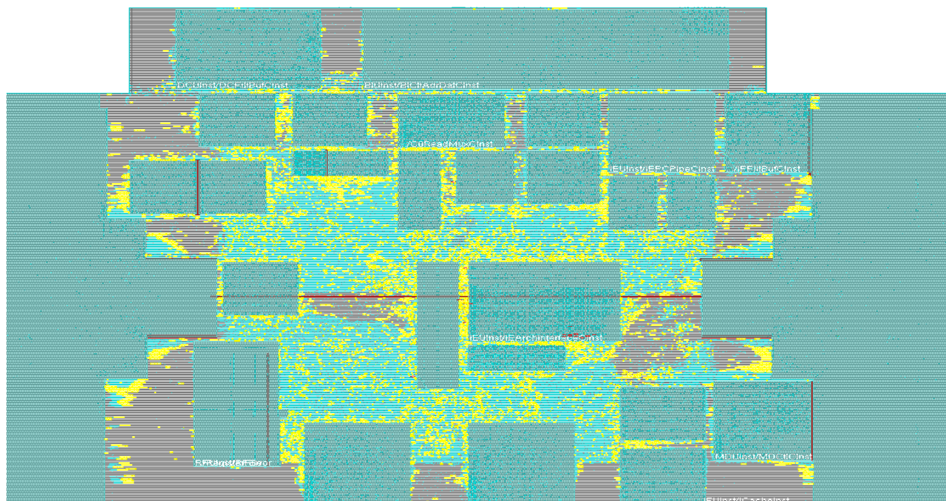

Fig. 5.8 Pre-CTS optimization affected cells (yellow)

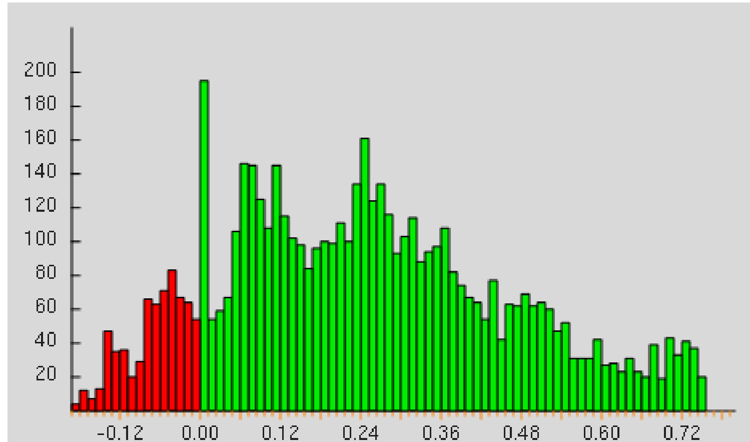

Fig. 5.9 WNS Histogram at pre-CTS stage.

It can be seen that the design violates the speed target of 500MHz (2ns timing window) by 188ps at the pre-CTS stage. There are 4879 failing critical paths. Pre-CTS histogram is shown in Fig 5.9.

5.5. ***Clock Tree Synthesis***

Building a radiation hardened clock tree is critical for rad-hard performance. Setup and hold timing budgets are extracted based on the skew values for the top-level clock tree. The top-level clock tree implementation goal is skew minimization. To that end the TMR clock tree values are already known from the TMR implementation of the blocks. Clock trees are then built for the DMR logic at the top-level HERMES implementation.

These clock standard cell logic at the top-level and H-trees are built to the clock pins traced through the clock gating cells. The clock gates are already spatially separated

TMR
DMR

Clock	max	min	levels	Skew (+/-)	Blocks Connected				
Gclk[14]	428	394	10	17	DCDTLBSDInstA	DCErrorSS	RFCtrlSDInstA		
Gclk[17]	415	364	9	25	IFCtrlSDInstA	IFITLBSDInstA	IFErrorSSInst	MDMACDivSD	IEBypassSD
Gclk[18]	438	359	13	28	MMJTLBCAMDDInstA	MMJTLBDataDDIn	MMJTLBMiscSDInst	IECtrlSDInstA	
Gclk[48]	425	392	11	17	IECtrlSDInstB	IFITLBSDInstB	IFCtrlSDInstB	MDMACDivSD	
Gclk[49]	441	361	11	29	MMJTLBCAMDDInstB	MMJTLBDataDDIn	MMJTLBMiscSDInst	IEBypassSDIn	
Gclk[60]	431	338	10	30	RFCtrlSDInstB	DCDTLBSDInstB	DCCtrlSDInstB		
Gclk[0]	263	256	5	5	BICtlAdrDatC	C0SystemRegsC	C0TimerRegsC	IEArchInterfac	
Gclk[1]	264	251	4	6	BICtlAdrDatC	C0SystemRegsC	C0TimerRegsC	IEArchInterfac	
Gclk[2]	258	254	4	5	BICtlAdrDatC	C0SystemRegsC	C0TimerRegsC	IEArchInterfac	
Gclk[3]	267	262	4	3	BICtlAdrDatC	C0ErrorRegsC	IEPCPipeCInst		
Gclk[4]	252	237	5	8	DCFillBufC	C0TLBRegsC	DCFBDualTripleCross	DCStoreBufCInst1	
Gclk[5]	275	257	5	9	DCFillBufC	C0TLBRegsC	DCFBDualTripleCros	DCStoreBufCI	
Gclk[6]	273	263	6	5	DCFillBufC	C0TLBRegsC	DCFBDualTripleCros	DCStoreBufCI	
Gclk[7]	234	228	4	4	C0CacheRegsC	C0RFRegsC	DCFBDualTripleCros	DCStoreBufCI	
Gclk[8]	236	236	3	1	BICtlAdrDatC	C0ErrorRegsC	IEPCPipeCInst		
Gclk[9]	277	263	4	5	BICtlAdrDatC	C0ErrorRegsC	IEPCPipeCInst		
Gclk[10]	251	239	4	5	IFDualTripleCrossCInst	IFDualTripleCross	IFFillBufCInst		
Gclk[11]	258	242	4	8	IFDualTripleCrossCInst	IFDualTripleCross	IFFillBufCInst		
Gclk[12]	239	228	4	5	C0CacheRegsC	C0RFRegsC	DCFBDualTripleCros	DCStoreBufCI	
Gclk[13]	237	222	4	8	C0CacheRegsC	C0RFRegsC	DCFBDualTripleCros	DCStoreBufCI	
Gclk[16]	248	240	4	5	IFDualTripleCrossCInst	IFDualTripleCross	IFFillBufCInst		
Gclk[19]	242	242	4	0	IcacheInst (TagGClkA1)				
Gclk[23]	235	235	3	0	DcacheInst				
Gclk[24]	240	240	3	0	IcacheInst (TagGClkA2)				
Gclk[25]	296	258	6	19	IFCtrlSDInstA				
Gclk[26]	253	253	4	0	IcacheInst				
Gclk[27]	275	275	2	0	IcacheInst				
Gclk[28]	231	230	4	1	DcacheInst				
Gclk[30]	220	220	3	0	DcacheInst				
Gclk[31]	242	242	2	0	DcacheInst				
Gclk[32]	240	232	3	5	IFCtrlSDInstA				
Gclk[33]	232	232	3	0	IcacheInst (TagGClkB)				
Gclk[34]	260	246	5	7	IFCtrlSDInstB				
Gclk[35]	231	231	3	0	IcacheInst				
Gclk[36]	236	225	3	5	DCCtrlSDInstA				
Gclk[37]	217	217	1	0	DcacheInst (TagGClkB)				
Gclk[38]	234	232	4	1	DCCtrlSDInstB				
Gclk[39]	226	226	3	0	DcacheInst				
Gclk[40]	231	231	2	0	IcacheInst				
Gclk[41]	258	239	5	10	IFCtrlSDInstB				
Gclk[44]	226	226	2	0	DcacheInst				
Gclk[45]	245	242	5	1	DCCtrlSDInstB				
Gclk[50]	267	257	3	5	MDDualTripleCrossCIns	MDDualTripleCrossCInst2			
Gclk[51]	271	267	3	2	MDDualTripleCrossCIns	MDDualTripleCrossCInst2			
Gclk[56]	276	264	3	5	MDDualTripleCrossCIns	MDDualTripleCrossCInst2			
Gclk[57]	256	245	4	5	IEDualTripleCrossCInst	MDCtlCInst			
Gclk[58]	264	259	4	2	IEDualTripleCrossCInst	MDCtlCInst			
Gclk[59]	259	253	4	3	IEDualTripleCrossCInst	MDCtlCInst			
Gclk[61]	212	212	5	0	IEDualTripleCrossCInst1				
Gclk[62]	208	208	2	0	IEDualTripleCrossCInst1				
Gclk[63]	236	236	3	0	IEDualTripleCrossCInst1				

Table. 5.1. DMR and TMR clock tree statistics at the top level.

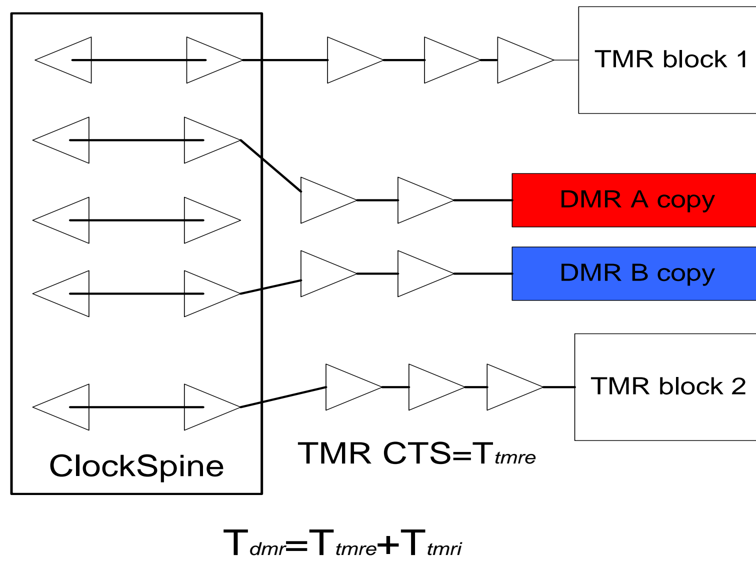

Fig. 5.10 Clock spine CTS cartoon for skew balancing.

based on their A or B copy association. DMR clock trees are built first and a constraint on the total latencies are reached for the whole design since global not local skew minimization is the goal.

The shorter clock H-trees to the TMR and other hard blocks comprehend the total latency of the DMR clock trees as shown in the cartoon in Fig. 5.10. T_{tmri} subtracted from T_{dmr} is the resulting latency to which clock trees should be built to the TMR blocks. Latency values of clock tree and the skew numbers are also shown in Table 5.1.

A, B and C clocks to single TMR blocks cannot be shared but they can be grouped with other TMR clocks. Also, any clock with a unique enable has to be separately connected, as is the case with the cache clock pins. The TMR block clocks are also grouped together, based on proximity, to drive the clock from one of the 64 clockspine pins. The assignment to the TMR and DMR clocks is based on the enables that control the 64 clock outputs; therefore not all clocks can be mixed.

Fig. 5.11 shows the relatively low number of buffers added to build the top level

Fig. 5.11 Clock Tree synthesis cells used in the design.

clock tree. As explained earlier clocks are routed first to ensure that they get the largest share of routing resources to ensure lowest possible skew and lowest possible capacitive coupling. Fig 5.12 shows the clock-only routing shown in the HERMES core.

DMR and the design clock tree latency histograms are shown in Fig. 5.13 and Fig 5.14. These show the excellent latency spread achieved by the aforementioned CTS strategy. The spread for the DMR clock tree is from 440 to 540 ps resulting in a +/- 50ps

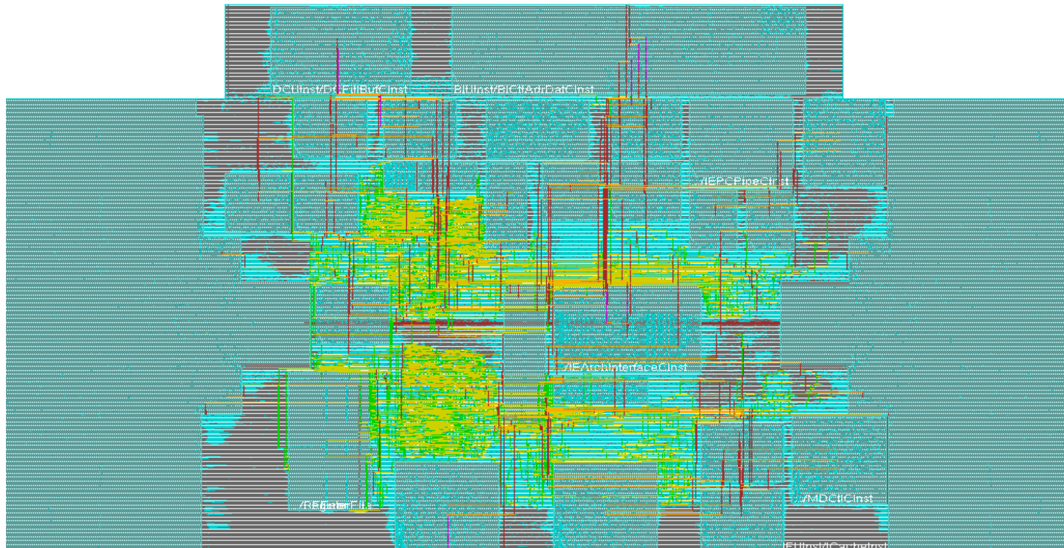

Fig. 5.13 All the clock routing shown (DMR and TMR).

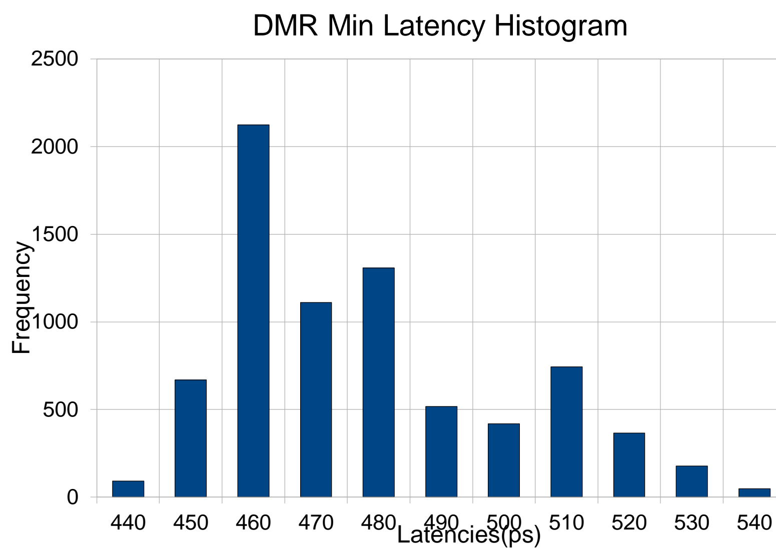

Fig. 5.12 DMR Clock Tree latency histogram.

skew. The same number for all clocks in the design is 430 to 580 ps with a resulting skew of +/- 75ps.

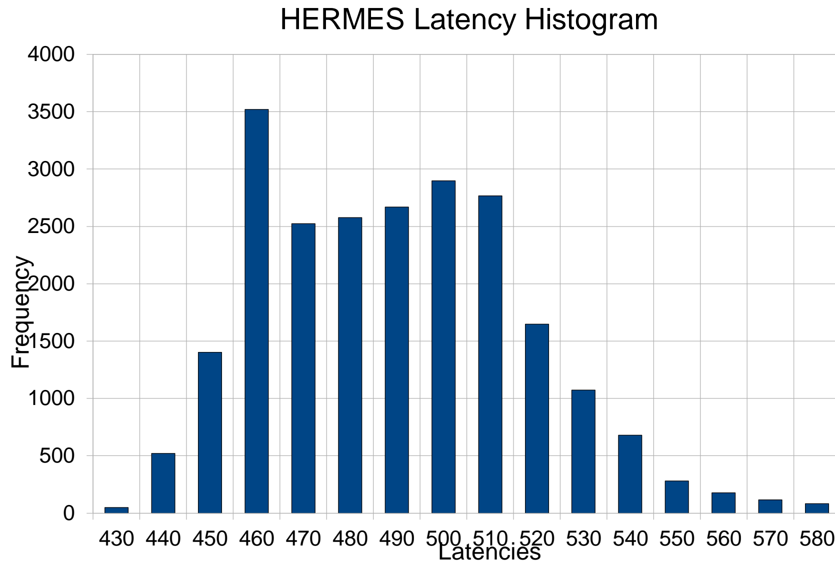

Fig. 5.14 Design Clock Tree latency histogram.

5.6. ***Post-CTS optimization***

Once the clock tree synthesis is completed, post-CTS hold and setup optimizations fix timing violations. Setup violations are fixed before hold violations to make sure that fixing setup does not degrade the more important hold fixes. Fixing hold violations can lead to degradation of setup margin but that is acceptable since hold violations, unlike setup violations, cannot be fixed post fabrication by reducing the chip clock frequency. The hold margin fixed for this design is 150ps, based on a global skew

on the order of +/- 75ps providing a net margin of 50ps. The number of hold buffers added and the timing statistics are described in the final chapter. The TMR blocks have 100ps hold margin. Hence, the net 250 ps of hold margin in the TMR paths make them more robust. The post-CTS hold fixing cells are shown below in Fig. 5.15.

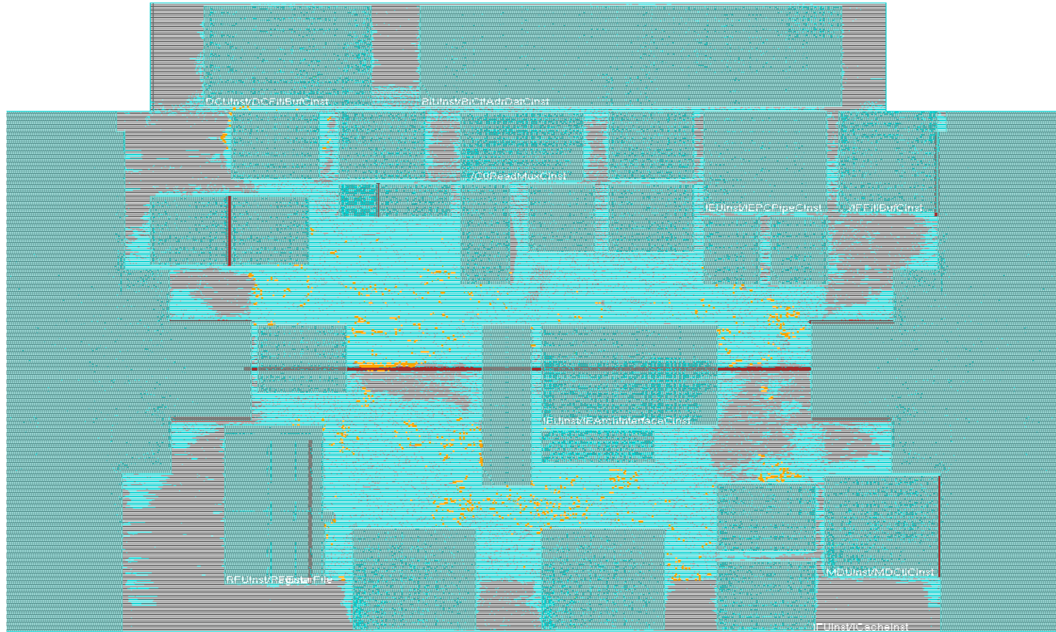

Fig. 5.15 PostCTS hold cells highlighted in orange.

5.7. ***Routing Phase***

After the post-CTS optimization, HERMES was routed using the detailed router, in this case Nanoroute ultra in Encounter. It takes into consideration of the total congestion statistics of the chip by analyzing routes available to the router and then routes the design.

There were no RHBD specific constraints given to the router, except for the CTS routing that are routed on priority with double width double spacing rules.

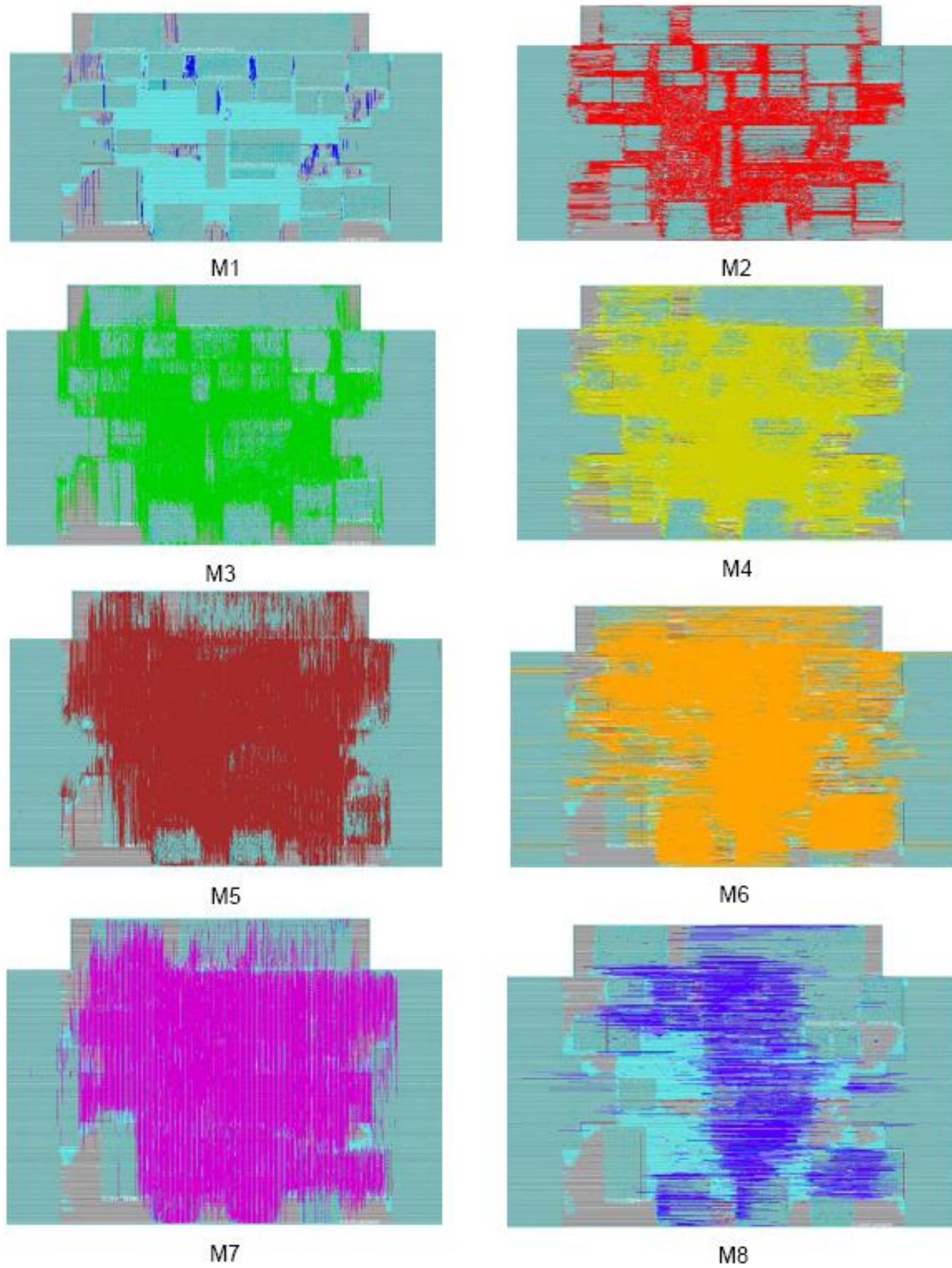

Fig. 5.16 Metal density in the routing phase.

In the initial runs the router had over 100000 DRC errors after routing because of the TMR pin congestion and macro alignment. These issues were resolved with the pin placement algorithm described in the TMR flow improvements chapter. The router also fixed antenna violations by layer hopping. Remaining violations needed antenna diode cells to be rectified. The routed database with all the metal layer routing is shown in Fig. 5.16. Metal 1 and Metal 8 are not fully utilized, latter because of restriction placed on the router and former because of lower routing congestion in the design. Note that post-Route optimization stage will increase the density of the routing further.

5.8. ***Post-Route Optimization***

Post-Route optimization is run after the routing phase. It involves near sign-off level timing calculations using high effort extraction using ICE caps extraction files and turbo-QRC (TQRC) feature in Encounter. The resulting slews and clock delays are the most accurate possible using this tool. Comparison of these values with the pre-route numbers is a worthwhile exercise at this level to figure out the correlation in the tool. Large variations imply that there could be inaccuracies in input extraction files and tool settings on extraction and timing modes.

The final pre-CTS congestion numbers were 0 for horizontal and 0.03 for vertical congestion at pre-CTS stage. This correlated well with the very low drc (~600) count at postroute, most of which were antenna violations. Tool options on slew propagation and clock propagation should be double checked at this stage to guarantee accuracy. Fig. 5.17

shows the summary of optimization based setup and hold cell additions along with A and B copy cell modifications.

Post-route optimization runs the same algorithms with more accurate values to minimize setup and hold WNS and TNS based on the constraints provided. Signal integrity optimization was not run. A copy and B copy cells are placed within their respective fences even after optimization, though there is an issue with cells from other hierarchies or cells added by the tool that do not explicitly get attached to the A or B fences. This clearly shows that though very hard from a separation point of view, some

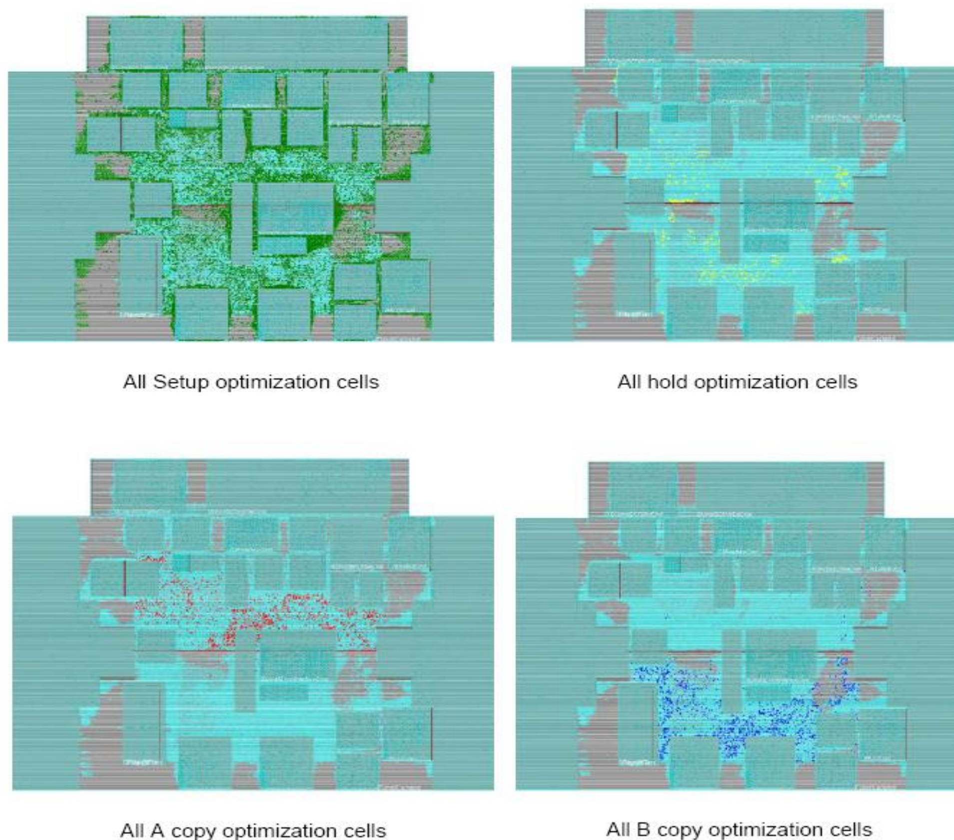

Fig. 5.17 Cells added by optimization and the corresponding A and B separation

buffers or other optimized cells added might not be in a good placement location from a DMR separation standpoint. Nonetheless, the probability of a particle striking a critical path in A copy and its corresponding B equivalent is very low. In the next chapter we will discuss the results in terms of area, speed and power. Techniques used for hardening are also summarized and their efficacy will be discussed with emphasis on possibilities for further improvement.

5.9. ***Summary***

Top level implementation of HERMES is explained in this chapter. Design decisions which enabled successful implementation of HERMES in terms of constraints of area, timing, power and radiation hardness have been discussed.

CHAPTER 6. CONCLUSIONS

HERMES is implemented and the summary of the design parameters is shown below in table 6.1. Progression of the design from floorplanning to postroute optimization is compared. Density of the design clearly increases between steps gradually, owing to cells that are added and upsized for timing fixing.

Stage	Setup			Hold			Area(density %)		
	WNS	TNS	violating paths Out of 15243	WNS	TNS	violating paths Out of 15243	Total 2276489 um^2	InstA 560727 um^2	InstB 618749 um^2
Synthesis	557.000	0.000	0.000	n/a	n/a	n/a		n/a	n/a
Placement	-79.574	-49583.100	8930.000	n/a	n/a	n/a	27.000	52.400	47.500
Pre-CTS	-0.188	-45.827	671	n/a	n/a	n/a	48.000	59.900	55.800
CTS	-0.718	-283.169	1610	-0.236	-15.144	109	51.320	64.300	60.900
Post-CTS	-0.220	-56.679	558	-0.049	-0.110	13	52.300	66.100	61.300
Route	-0.399	-188.878	1538	-0.039	-0.767	95	52.700	67.100	61.500
Postroute	-0.248	-56.361	667	-0.003	-0.006	4	52.300	67.100	61.500

WNS Worst Negative Slack

TNS Total Negative Slack

Table. 6.1. Table showing timing and Area summary

The design has an operational frequency of 450Mhz, which is a 10% degradation on the expected design target frequency of 500Mhz. Loss of speed can be attributed to speed of some individual TMR blocks and their physical placement that was sub-optimal owing to area and routing constraints shown in Fig. 6.1. TMR block placements were

restricted because of power-plan constraints and area bottleneck. In hindsight, the aspect ratios of the blocks were also sub-optimal.

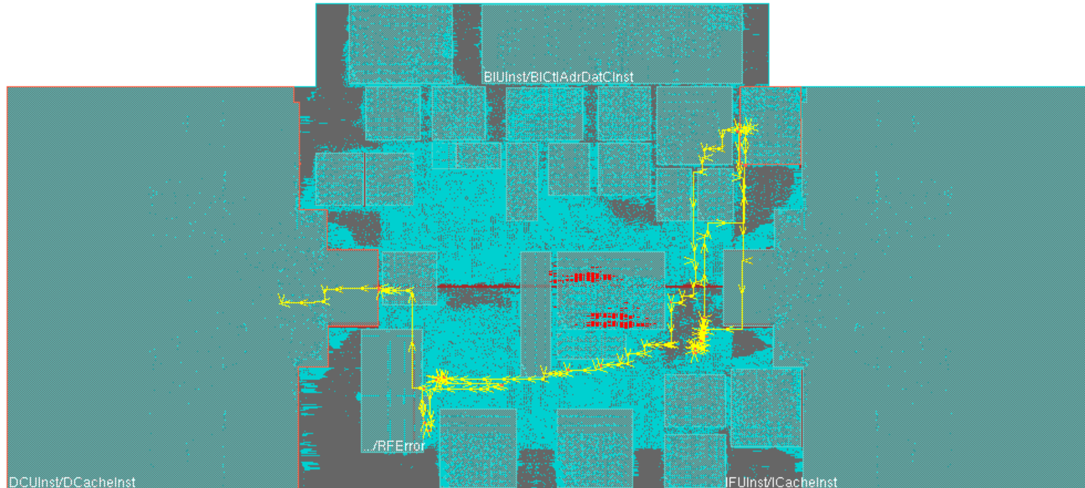

Fig. 6.2 Worst timing critical path spanning multiple blocks.

Routing density was consistent except for metal 1. Metal 1 was used minimally

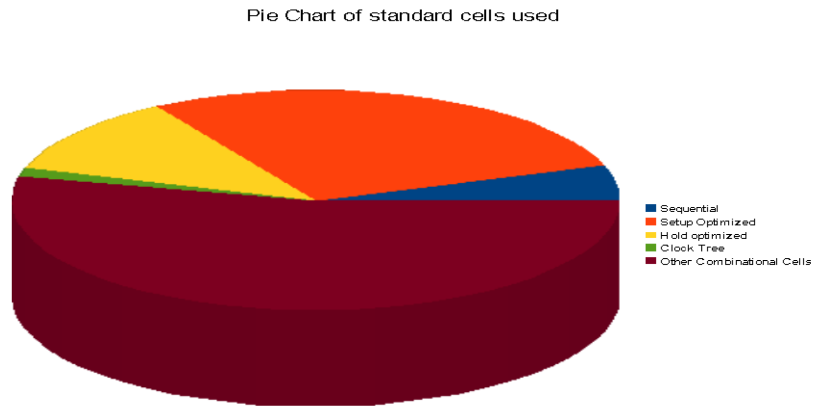

Fig. 6.1 Pie chart showing cell distribution

for routing owing to high metal 1 pin density in standard cells and decoupling capacitors.

The pie chart in Fig. 6.2 gives the standard cell distribution in the design based on their design timing functionality. About 40% of the cells were added for optimizations.

Total Standard cells	168573
Sequential	8739
Setup Optimized	48210
Hold optimized	19396
Clock Tree	1957
Other Combinational Cells	90271

Table. 6.2. Standard cell distribution

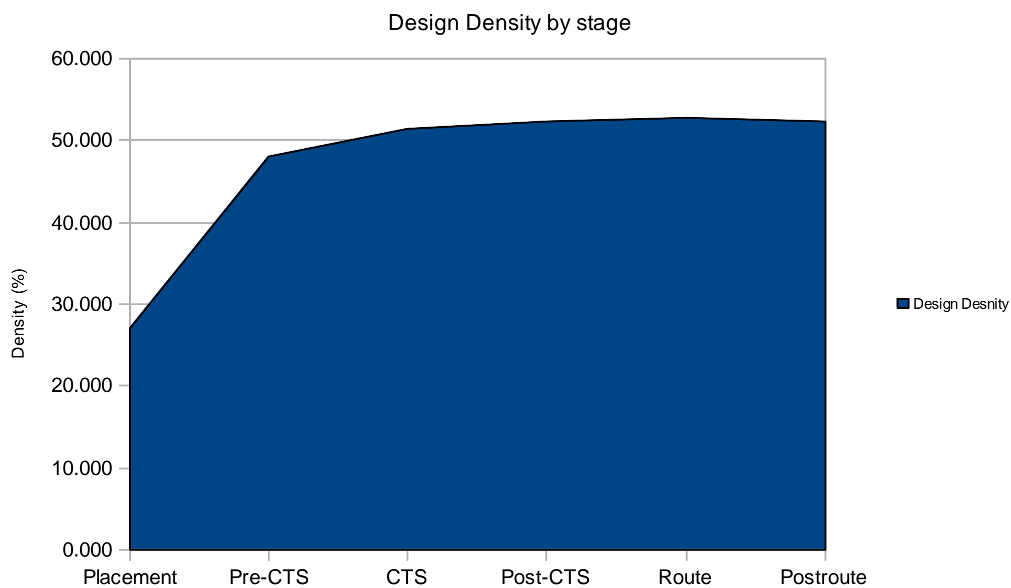

Fig. 6.3 Design overall density by stage.

Area comparison is shown in Fig. 6.3 and 6.4. Design density grows gradually and saturates around 54% after routing. The A and the B DMR logic show slightly more

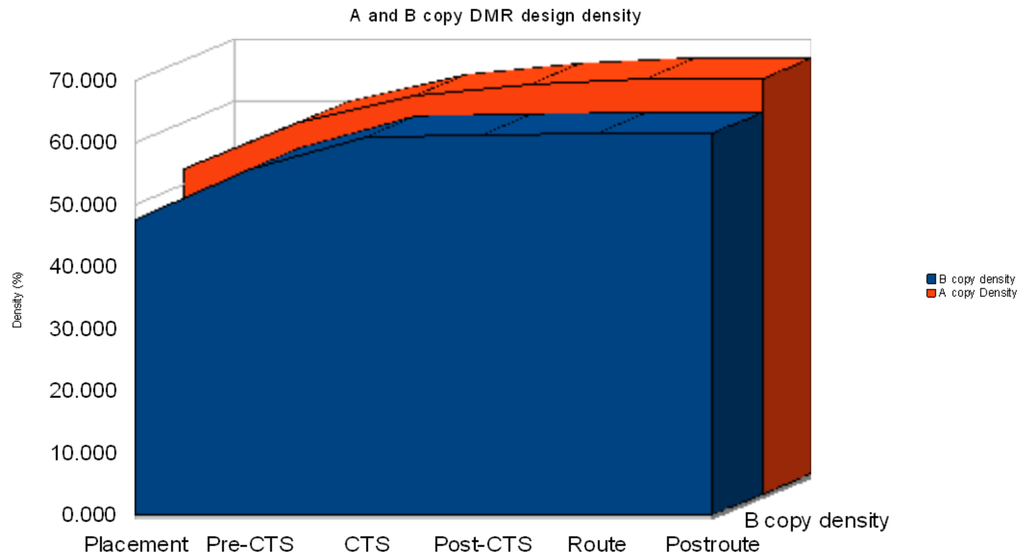

Fig. 6.4 DMR A and B copy density growth by stage.

abrupt increase since they have the largest number of timing critical paths that need to be optimized.

A and B DMR logic density is shown below in Fig. 6.4. They grow at a different rate in spite of having the same redundant logic because of the non-symmetrical logic around them that interacts with both copies.

Worst negative slack (WNS), Total negative slack (TNS) and failing endpoints per implementation stage have been plotted in Fig. 6.5-6.10. Fig 6.6 shows that the synthesis timing achieves a much higher frequency than the physical implementation.

Physical implementation is typically slower but the magnitude of this difference shows that there was a drastic speed-down owing to placement, routing and TMR complexity.

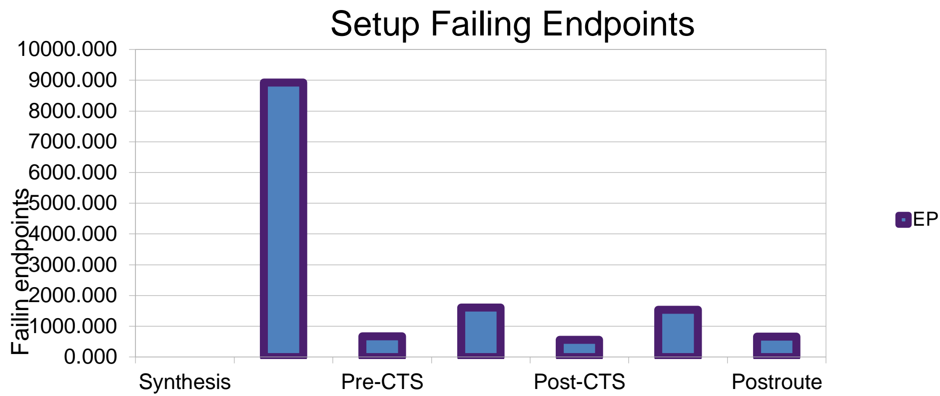

Fig. 6.6 . Failing endpoints for setup violations.

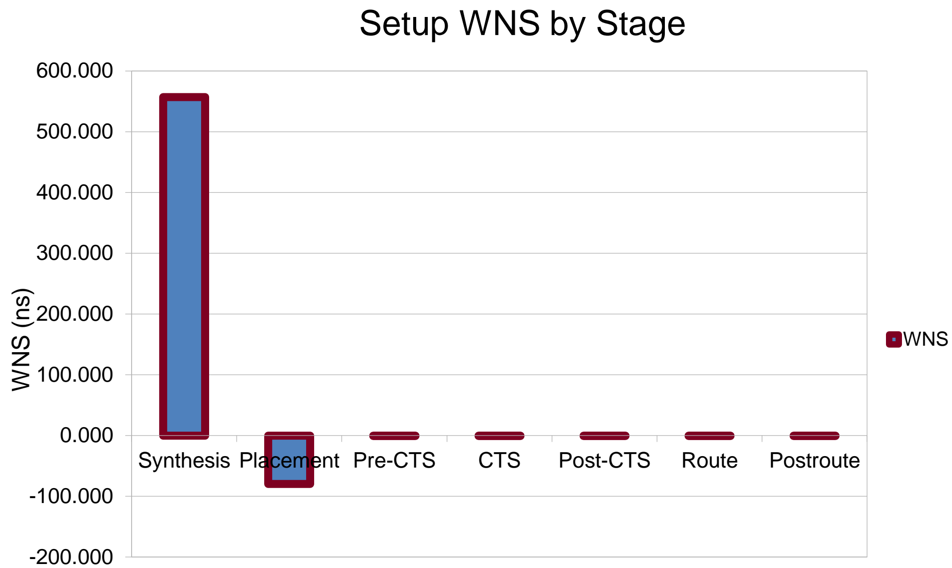

Fig. 6.5 Setup WNS for each stage.

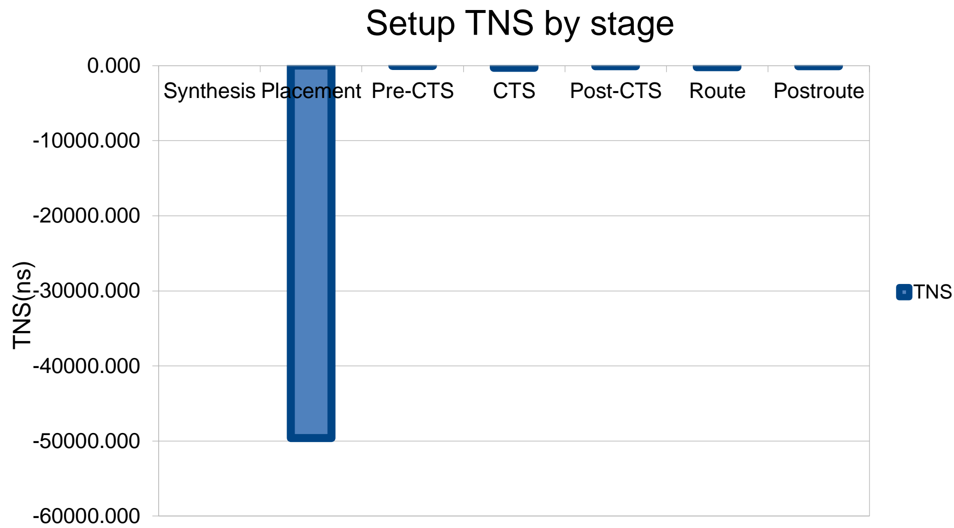

Fig. 6.8 Setup TNS by stage.

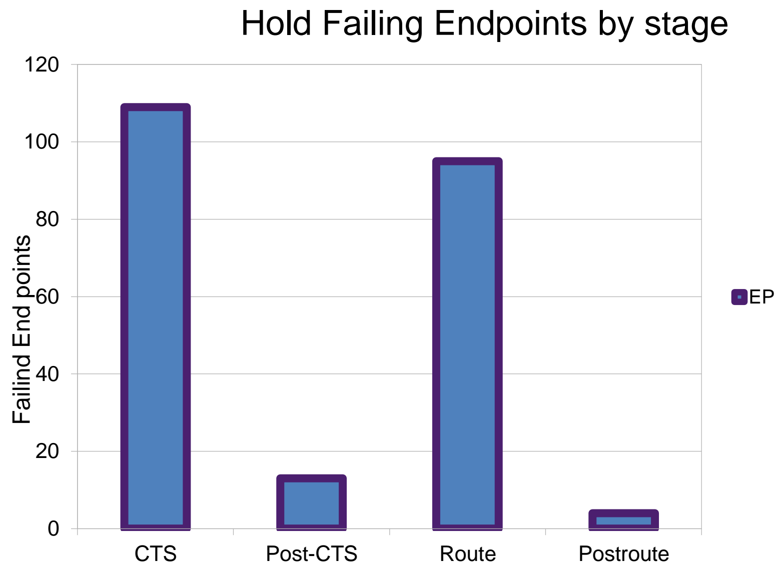

Fig. 6.7 Hold endpoints by stage.

Hold timing violations were successfully fixed with an extra margin of 150ps. Quality of the clock tree is the reason for low numbers of clock buffers required to fix existing post-CTS violations. Placement TNS is disproportionately large because of the

fact that no design rule violations are fixed, thereby resulting in larger slews and delays that manifest inside the design. Hold timing statistics are shown in Fig 6.8- 6.10.

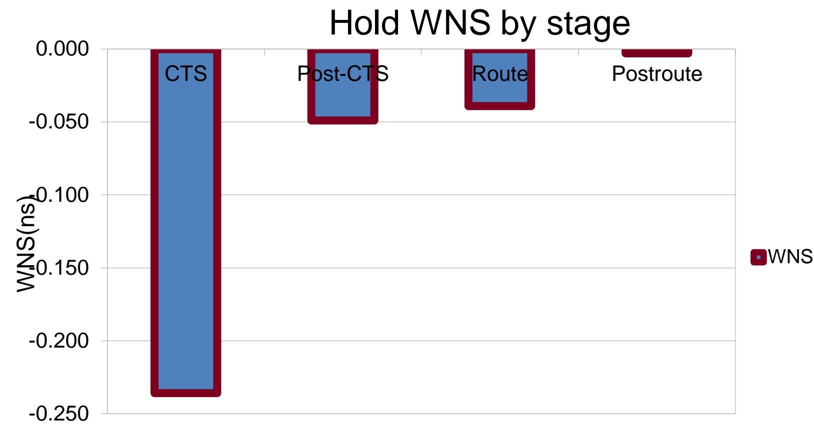

Fig. 6.10 Hold WNS by stage

Relatively few hold buffers (~20K) were introduced owing to the low skew clock tree built. Hold failing endpoints and WNS increase after routing because of extraction and timing difference between global and detail routing.

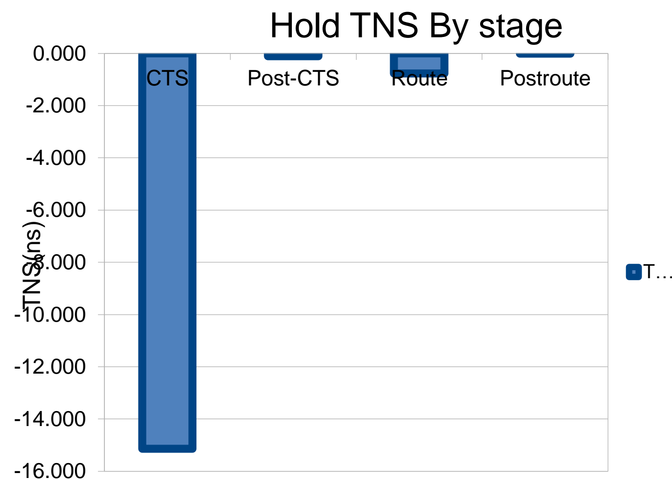

Fig. 6.9 Hold TNS by stage.

Distinct A and B copy separation was one of the main design goals and it has been achieved as explained in chapter 5. It can also be shown that same paths from A and B logic copies are spatially separated by more than 500u. This shows that path shown in Fig. 6.11 has 0 probability of getting upset by a single particle hit. Path from

IFUInst/IFITLBSDInstA/B/InstrVPNSEL_reg[18]/q to

IFUInst/IFITLBSDInstA/B/CacheableSIH_reg/d is shown.

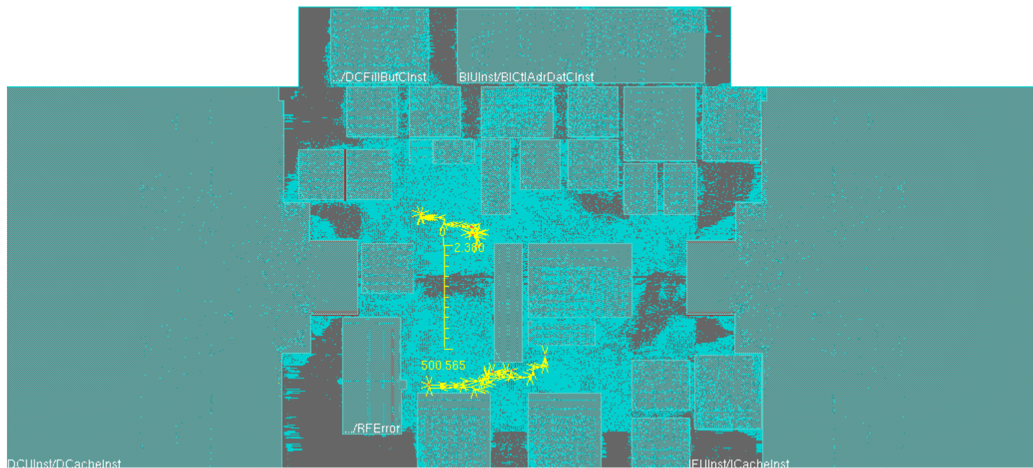

Fig. 6.11 A and B copy of a same critical path highlighted separated by 500um.

Power was measured using Primetime-PX and the numbers are summarized in table 6.3. Power was calculated using Dhrystone generated vectors for anticipated worst-case instructions. Power budget of the chip was 1W and results achieved are well within that estimate. The cache power is estimated from [Yao2010] at 400 MHz to be ~75mW. This means that the total power is less than 500mW, which is half the estimated number.

	Average Power(Watts)	Peak Power(Watts)
Total	0.3692	0.5236
Switching	0.1395	
Internal	0.1946	
Leakage	0.0351	

Table. 6.3. *Power statistics at 400MHz using Dhrystone vectors.*

Total power numbers show that IEU (Instruction Execution Unit), as expected, dissipated the most power. IEU contains the highly dense IEArchitechture block that has logic which interacts the most with other blocks in the design. It can also be inferred that hierarchical clock-gating has been a very effective strategy and the total switching power is reduced as a consequence.

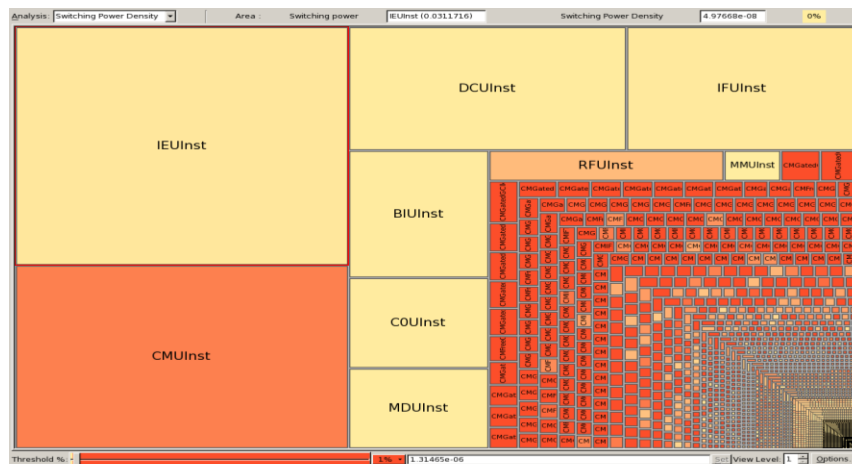

Fig. 6.12 Switching power density map from primetime showing module contributions.

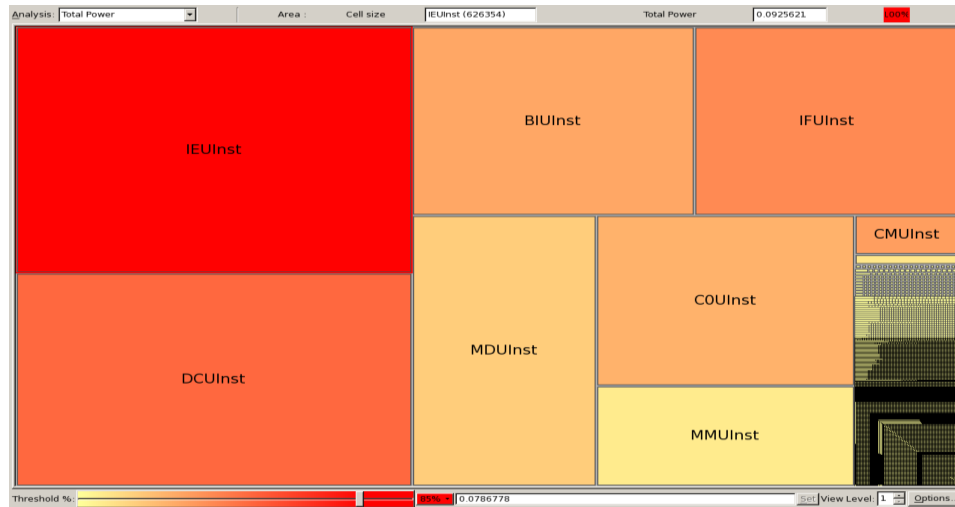

Fig. 6.14 Total Power map from primetime showing module contributions.

Switching power density has CMU module as the major contributor since this block controls the clocking to the whole design and has high capacitance, high activity factor nodes. Snapshots from Primetime are shown in Fig. 6.12 to Fig. 6.14.

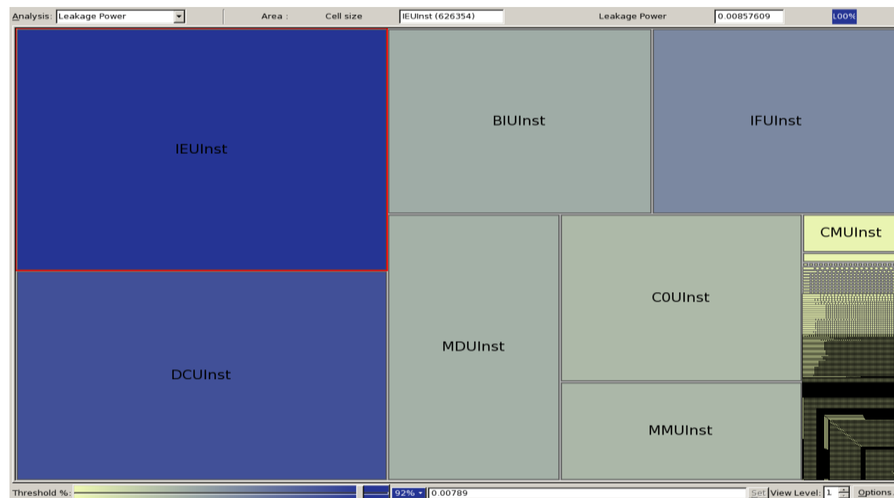

Fig. 6.13 Leakage Power map from primetime showing module contributions.

6.1. ***Summary***

We may conclude that most of the design targets in terms of speed, power, area and radiation hardness have been met with slight degradations in speed. A cursory look at the top level path in Fig. 6.1 proves that an improvement in speed can be achieved if the quality of placement can be improved. To pursue this, future work comprising of improved placement algorithms and techniques are already being developed and tested. Improvements to the current implementation in terms of the ASIC RHBD flow and design techniques to speed-up the design, while making it lower power (pulsed clock latches), are also being analyzed and will be subsequently implemented. Additionally, work on temporal hardening based flows with special placement algorithms and pipeline based TMR design strategies are also being done to provide complete RHBD ASIC design methodology.

REFERENCES

[1] B. E. Pritchard, G. M. Swift, and A. H. Johnston, “Radiation effects predicted, observed, and compared for spacecraft systems,” Proc. IEEE NSREC Radiation Effects Data Workshop Record, pp. 7–17, 2002. [Pro2002].

[2] Narayanan. V, Xie. Y, "Reliability concerns in embedded system designs " Computer , vol.39, no.1, pp. 118- 120, Jan. 2006. [Nar2006].

[3] Hughes, H.L, Benedetto, J.M, "Radiation effects and hardening of MOS technology: devices and circuits," Nuclear Science, IEEE Transactions on, vol.50, no.3, pp. 500- 521, June 2003. [Hug2003].

[4] Barth, J.L.; Dyer, C.S.; Stassinopoulos, E.G.; , "Space, atmospheric, and terrestrial radiation environments," Nuclear Science, IEEE Transactions on , vol.50, no.3, pp. 466- 482, June 2003. [Bar2003].

[5] Stassinopoulos, E.G., Raymond, J.P., "The space radiation environment for electronics," Proceedings of the IEEE, vol.76, no.11, pp.1423-1442, Nov 1988." [Sta1988].

[6] Kerns, S.E.; Shafer, B.D.; Rockett, L.R., Jr.; Pridmore, J.S.; Berndt, D.F.; van Vonno, N.; Barber, F.E.; , "The design of radiation-hardened ICs for space : a compendium of approaches," Proceedings of the IEEE , vol.76, no.11, pp. 1470-1509, Nov 1988." [Ker1988].

[7] Hsieh, C. M.; Murley, P. C.; O'Brien, R. R.;, "Dynamics of Charge Collection from Alpha-Particle Tracks in Integrated Circuits," Reliability Physics Symposium, 1981. 19th Annual, vol., no., pp.38-42, April 1981." [Hse1981].

[8] "Mavis, D.G.; Eaton, P.H.; ,"Soft error rate mitigation techniques for modern microcircuits, "" Reliability Physics Symposium Proceedings, 2002. 40th Annual , vol., no., pp. 216- 225, 2002" [Mav2002].

[9] Barnaby, H. J.; , "Total-Ionizing-Dose Effects in Modern CMOS Technologies," Nuclear Science, IEEE Transactions on , vol.53, no.6, pp.3103-3121, Dec. 2006 [Bar2006].

[10] Saggese, G.P.; Wang, N.J.; Kalbarczyk, Z.T.; Patel, S.J.; Iyer, R.K.; , "An experimental study of soft errors in microprocessors," Micro, IEEE , vol.25, no.6, pp. 30- 39, Nov.-Dec. 2005 [Sag2005].

[11] Dodd, P.E.; Massengill, L.W.; , "Basic mechanisms and modeling of single-event upset in digital microelectronics," Nuclear Science, IEEE Transactions on , vol.50, no.3, pp. 583- 602, June 2003 [Dod2003].

[12] "Hsieh, C.M.; Murley, P.C.; O'Brien, R.R.; , ""A field-funneling effect on the collection of alpha-particle-generated carriers in silicon devices,"" Electron Device Letters, IEEE , vol.2, no.4, pp.103-105, April 1981." [Hsi1981].

[13] "Karnik, T.; Hazucha, P.; , ""Characterization of soft errors caused by single event upsets in CMOS processes,"" Dependable and Secure Computing, IEEE Transactions on , vol.1, no.2, pp. 128- 143, April-June 2004" [Kar2004].

[14] "S. J. Heileman, W. R. Eisenstadt, R. M. Fox, R. S. Wagner, N. Bordes, and J. M. Bradley, "CMOS VLSI single event transient characterization," IEEE Trans. Nucl. Sci., vol. 36, no. 6, pp. 2287-2291, 1989." [Hei1989].

[15] "Benedetto, J.; Eaton, P.; Avery, K.; Mavis, D.; Gadlage, M.; Turflinger, T.; Dodd , P.E.; Vizkelethyd, G.; , ""Heavy ion-induced digital single-event transients in deep submicron Processes,"" Nuclear Science, IEEE Transactions on , vol.51, no.6, pp. 3480- 3485, Dec. 2004" [Ben2004].

[16] "Gadlage, M.J.; Schrimpf, R.D.; Benedetto, J.M.; Eaton, P.H.; Mavis, D.G.; Sibley, M.; Avery, K.; Turflinger, T.L.; , ""Single event transient pulse widths in digital microcircuits,"" Nuclear Science, IEEE Transactions on , vol.51, no.6, pp. 3285- 3290, Dec. 2004" [Gad2004]

[17] "Axness, C. L.; Weaver, H. T.; Fu, J. S.; Koga, R.; Kolasinski, W. A.; , ""Mechanisms Leading to Single Event Upset,"" Nuclear Science, IEEE Transactions on , vol.33, no.6, pp.1577-1580, Dec. 1986." [Axn1986].

[18] "Musseau, O.; Gardic, F.; Roche, P.; Corbiere, T.; Reed, R.A.; Buchner, S.; McDonald, P.; Melinger, J.; Tran, L.; Campbell, A.B.; , ""Analysis of multiple bit upsets (MBU) in CMOS SRAM,"" Nuclear Science, IEEE Transactions on , vol.43, no.6, pp.2879-2888, Dec 1996." [Mus1996].

[19] Koga, R.; Penzin, S.H.; Crawford, K.B.; Crain, W.R.; , "Single event functional interrupt (SEFI) sensitivity in microcircuits,", RADECS 97. Fourth European Conference on , vol., no., pp.311-318, 15-19 Sep 1997. [Kog1997].

[20] H. J. Barnaby, M. L. McLain, I. S. Esqueda, and X. J. Chen, "Modeling ionizing radiation effects in solid state materials and CMOS devices," IEEE Trans. Circuits Syst. I, vol. 56, pp. 1870–1883, Aug. 2009. [Bar2009].

[21] Lacoe, R.C.; Osborn, J.V.; Koga, R.; Brown, S.; Mayer, D.C.; , "Application of hardness-by-design methodology to radiation-tolerant ASIC

technologies," Nuclear Science, IEEE Transactions on , vol.47, no.6, pp.2334-2341, Dec 2000 [Lac2000].

[22] P. Dodd and F. Sexton, “Critical charge concepts for CMOS SRAMs,” IEEE Trans. Nucl. Sci., vol. 42, no. 6, pp. 1764–1771, Dec. 1995. [Dod1995].

[23] Hindman, N.D.; Clark, L.T.; Patterson, D.W.; Holbert, K.E.; , "Fully Automated, Testable Design of Fine-Grained Triple Mode Redundant Logic," Nuclear Science, IEEE Transactions on , vol.58, no.6, pp.3046-3052, Dec. 2011 [Hin2011].

[24] Hindman, N.D.; Pettit, D.E.; Patterson, D.W.; Nielsen, K.E.; Xiaoyin Yao; Holbert, K.E.; Clark, L.T.; , "High speed redundant self-correcting circuits for radiation hardened by design logic," Radiation and Its Effects on Components and Systems (RADECS), 2009 European Conference on , vol., no., pp.465-472, 14-18 Sept. 2009 [Hin2009].

[25] C. Weaver, J. Emer, S. Mukherjee, and S. Reinhardt, “Techniques to reduce the soft error rate of a high-performance microprocessor,” Proc. ISCA, 2004, pp. 264-27 [Wea2004].

[26] Fujiwara, E.; Pradhan, D.K.; , "Error-control coding in computers," Computer , vol.23, no.7, pp.63-72, July 1990. [Fuj1990].

[27] Chen, C. L.; Hsiao, M. Y.; , "Error-Correcting Codes for Semiconductor Memory Applications: A State-of-the-Art Review," IBM Journal of Research and Development , vol.28, no.2, pp.124-134, March 1984 [Che1984].

[28] Yao. X; Clark, L.T.; Chellappa, S.; Holbert, K.E.; Hindman, N.D., "Design and Experimental Validation of Radiation Hardened by Design SRAM Cells," Nuclear Science, IEEE Transactions on , vol.57, no.1, pp.258-265, Feb. 2010. [Yao2010].

[29] Knudsen, J. E.; Clark, L. T.; , "An Area and Power Efficient Radiation Hardened by Design Flip-Flop," Nuclear Science, IEEE Transactions on , vol.53, no.6, pp.3392-3399, Dec. 2006. [Knu2006].

[30] LaBel, K.A.; Gates, M.M.; , "Single-event-effect mitigation from a system perspective," Nuclear Science, IEEE Transactions on , vol.43, no.2, pp.654-660, Apr 1996. [Lab1996].

[31] R. Lacoe, “CMOS scaling, design principles and hardening-by-design methodologies,” presented at the IEEE Nuclear and Space Radiation Effects Conf., Monterey, CA, Jul. 2003, short course. [Lac2003].

[32] D. R. Alexander, "Design issues for radiation tolerant microcircuits for space", Short Course Nuclear and Space Radiation Effects Conf., 1996. [Ale1996].

[33] W. Beauvais, P. McNulty, W. A. Kader, and R. Reed, "SEU parameters and proton-induced upsets," in Proc. Sec. European Conf. on Radiation and its Effects on Components and Systems, pp. 540-545, Sept. 1993. [Bea1993].

[34] W. Massengill, M. Alles, and S. Kerns, "SEU error rates in advanced digital CMOS," in Proc. Sec. European Conf. on Radiation and its Effects on Components and Systems, pp. 546-553, Sept. 1993. [Mas1993].

[35] N. Seifert, et al., "Radiation-induced soft error rates of advanced CMOS bulk devices," Proc. Int. Phys. Reliab. Symp, March 2006, pp. 217-225. [Sei2006].

[36] D. Kobayashi, T. Makino, and K. Hirose, "Analytical expression for temporal width characterization of radiation-induced pulse noises in SOI CMOS logic gates," Proc. IRPS, pp. 165-169, 2009. [Kob2009].

[37] Varghese George; et.al. , "Penryn: 45-nm next generation Intel® core™ 2 processor," Solid-State Circuits Conference, 2007. ASSCC '07. IEEE Asian , vol., no., pp.14-17, 12-14 Nov. 2007 [Var2007].

[38] K. Hass and J. Gambles, "Single event transients in deep submicron CMOS," in Proc. IEEE 42nd Midwest Symp. on Circuits and Systems, vol. 1, Aug. 1999, pp. 122-125. [Has1999].

[39] N. Seifert, et al., "Radiation induced clock jitter and race," Proc. Int. Phys. Rel. Symp., April 2005, pp. 215-222. [Sei2005].

[40] R. Dash, R. Garg, S. P. Khatri and G. Choi, "SEU hardened clock regeneration circuits," Intl Symp on Quality of Elec. Design, 2009, pp. 806-813. [Das2009].

[41] X. Yao, L. Clark, D. Patterson, K. Holbert, "A 90 nm bulk CMOS radiation hardened by design cache memory," IEEE Trans. Nucl. Sci, vol. 57, no. 4, pp. 2089-2097, Aug. 2010. [Yao2010].

[42] T. Calin, M. Nicolaidis and R.Velazco, "Upset hardened memory design for submicron CMOS technology," IEEE Trans. Nucl. Sci., vol. 43, no. 6, pp. 2874-2878, Dec. 1996. [Cal1996].

[43] "K. Warren, et al., “Heavy ion testing and single event upset rate prediction considerations for a DICE flip-flop,” IEEE Trans. Nucl. Sci., vol. 56, no. 6, pp. 3130-3137, Dec. 2009." [War2009].

[44] J. Knudsen and L. Clark, “An area and power efficient radiation hardened by design flip-flop,” IEEE Trans. Nucl. Sci., vol. 53, no. 6, pp. 3392-3399, Dec. 2006. [Knu2006].

[45] . Mavis and P. Eaton, “Soft error mitigation techniques for modern microcircuits,” Proc. IEEE IRPS, Aug. 2002 pp. 216-225, 2002. [Mav2002]

[46] "B. Matush, T. Mozdzen, L. Clark and J. Knudsen, “Area efficient temporally hardened by design flip flop circuits,” IEEE Trans. Nucl. Sci., vol. 57, no. 6, pp. 3588-3595, Dec. 2010." [Mat2010].

[47] K. Mohr and L. Clark, “Experimental characterization and application of circuit architecture level single event transient mitigation,” Proc. IRPS, pp. 312-317, 2007. [Moh2007].

[48] "H. Cha and J. H. Patel, “A logic-level model for particle hits in CMOS circuits,” Intl Conf. on Computer Design, pp. 538-542, Oct. 1993." [Cha1993].

[49] D. Fulkerson, D. Nelson, R. Carlson, and E. Vogt, “Modeling ion-induced pulses in radiation-hard SOI integrated circuits,” IEEE Trans. Nucl. Sci., vol. 54, no. 4, pp. 1406-1415, Aug. 2007.[Ful2007].

[50] L. Clark, D. Patterson, N. Hindman, K. Holbert, and S. Guertin, “A dual mode redundant approach for microprocessor soft error hardness,” IEEE Trans. Nucl. Sci., vol. 58, no. 6, Dec. 2011, pp. 3018-3025. [Cla2011].

[51] V. Horvat, Seuss, Radiation Effects Testing Facility, Cyclotron Institute, Texas A&M University, Online: cyclotron.tamu.edu/ref/beams.php, updated Dec. 21, 2011. [Hor2011].

[52] A. Chandrakasan, W. J. Bowhill, and F. Fox, Design of High-Performance Microprocessor Circuits, 2001 :IEEE Press [Cha2001].

[53] "TaikiUemura, Yoshiharu Tosaka, Hideya Matsuyama and Ken Shono. ""SEILA: Soft Error Immune Latch for Mitigating Multi-node-SEU and Local-clock-SET "" IRPS 2010, IRPS10-223" [Uem2010].

[54] "LaBel, K.A; Johnston, A.H. ; Barth, J.L. ; Reed, R.A. ; Barnes, C.E."" Emerging radiation hardness assurance (RHA) issues: a NASA approach for space flight programs"" Nuclear Science, IEEE Transactions on (Volume:45 , Issue: 6)" [Lab1998].

[55] "Nguyen, H.T. ; Intel Corp., Hillsboro, OR, USA ; Yagil, Yoad ; Seifert, N. ; Reitsma, M. ""Chip-level soft error estimation method"", Device and Materials Reliability, IEEE Transactions on (Volume:5 , Issue: 3)" [Ngu2005].

[56] Hass, K.J. ; Sandia Nat. Lab., Albuquerque, NM, USA ; Treece, R.K.; Giddings, A.E. ""A radiation-hardened 16/32-bit microprocessor"" Nuclear Science, IEEE Transactions on (Volume:36 , Issue: 6)" [Has1989].

[57] Yoshioka, S. Semicond. & Integrated Circuits Div., Hitachi Ltd., Tokyo, Japan Kamimura, H. ; Akiyama, M. ; Nakamura, M. ; Tamura, T. ; Kuboyama, S""A radiation-hardened 32-bit microprocessor based on the commercial CMOS process"" Nuclear Science, IEEE Transactions on (Volume:41 , Issue: 6)" [Yos1994].

[58] Haddad, N. ; Lockheed Martin Space Electron. &Commun., Manassas, VA, USA ; Brown, R. ; Cronauer, T. ; Phan, H. ""Radiation hardened COTS-based 32-bit microprocessor"" Radiation and Its Effects on Components and Systems, 1999. RADECS 99. 1999 Fifth European Conference on" [Had2000].

[59] "The RAD750TM-a radiation hardened PowerPCTM processor for high performance spaceborne applications"", Berger, R.W. BAE Syst., Manassas, VA, USA Bayles, D. ; Brown, R. ; Doyle, S. ; Kazemzadeh, A. ; Knowles, K. ; Moser, D. ; Rodgers, J. ; Saari, B. ; Stanley, D. ; Grant, B." [Ber2001].

[60] www.Synova.com

[61] www.Honeywell.com

[62] www.baesystems.com [Rad750]

[63] www.atmel.com.

[64] "D.A. Patterson, John. L. Hennessy, Computer Architecture: A quantitative Approach. 2nd ed., Morgan Kaufmann Publishers, Inc., 1996, pp. 375-390. [Pat1996].

[65] "J. Handy, The Cache Memory Book, San Diego: Academic Press, Inc, 1998. [Han1998]".

[66] "D. Lambert, J. Baggio, V. Ferlet-Cavrois, O. Flament, F. Signe, B. Sgnes, N. Buard, T. Carriere, ""Neutron-Induced SEU in Bulk SRAMs in Terrestrial Environment: Simulations and Experiments,"" IEEE Trans. Nuc. Sci., vol. 51, no. 6, pp. 3435-3441, Dec. 2004. [Lam2004]"

[67] S. S. Mukherjee, J. Emer, T. Fossum, and S. K. Reinhardt, "Cache Scrubbing in Microprocessors: Myth or Necessity?," 10th International Symposium on Pacific Rim Dependable Computing (PRDC), pp. 37-42, March 3-5, 2004. [Muk2004].

[68] K. Mohr and L. Clark, "A radiation hardened by design register file with lightweight error detection and correction," IEEE Trans. Nucl. Sci., Vol. 54, pp. 1335-1342, August 2007. [Moh2007]"

[69] "Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations",Slayman, C.W.; Sun Microsystems Inc., Santa Clara, CA, USA, Device and Materials Reliability, IEEE Transactions (Volume:5 , Issue: 3) Sept. 2005. [Sla2005]

[70] "D. Pettit, X. Yao, L.T. Clark, K.E. Holbert, N. Hinaman, and S. Mhambrey, "Design and validiaiton of a 90 nm radiation hardened by design standard cell library," presented at IEEE Nuclear and Space Radiation Effects Conference, 2009, July 20-24. [Pet2009].

[71] "Delay and Area Efficient First-level Cache Soft Error Detection and Correction" Mohr, K.C. Arizona State Univ., Tempe Clark, L.T. Computer Design, 2006. ICCD 2006. 1-4 Oct. 2007. [Moh2006].

[72] "A 90-nm Radiation Hardened Clock Spine" Nuclear Science, IEEE Transactions (Volume:59 , Issue: 4), Chellappa, Clark, L.T. ; Holbert, K.E. S. School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ, USA . [Che2012].