OPEN SOURCE RADIATION HARDENED STANDARD CELL LIBRARY AND EDA FLOW

By

RYAN RIDLEY

Bachelor of Science in Electrical Engineering
Oklahoma State University
Stillwater, OK
2020

Bachelor of Science in Computer Engineering
Oklahoma State University
Stillwater, OK
2020

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTERS OF SCIENCE
May, 2022

OPEN SOURCE RADIATION HARDENED STANDARD CELL LIBRARY AND EDA FLOW

Dissertation Approved:

Dr. James E. Stine, Jr.

Dissertation Advisor

Dr. Gary Yen

Dr. John Hu

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. James Stine for his constant support along with everyone in the VLSI Computer Architecture Research Group.

Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Name: RYAN RIDLEY

Date of Degree: MAY, 2022

Title of Study: OPEN SOURCE RAD-HARD LIBRARY AND EDA FLOW

Major Field: ELECTRICAL ENGINEERING

Abstract: The development of integrated circuits (ICs) has historically had a high cost of entry due to the cost of licensing industry standard Electronic Design Automation (EDA) tools and the Intellectual Property (IP) of semiconductor manufacturers, essential to IC design. With the release of Skywater Technologies open source 130nm PDK, a robust and functional EDA IC design flow using open source tools can be developed, allowing for a zero cost of entry into IC design. With the created open source flow, it can be used to fill the gaps where open source software has yet to fill. More importantly, there are currently no open source radiation hardened standard cell libraries available today. Using the open source EDA flow developed here, a VLSI compatible radiation hardened standard cell library is created, implemented, and assembled.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The System on Chip (SoC) and IC industry have been augmented by the vast infrastructure that Electronic Design Automation (EDA) tools create. These tools enhance the semiconductor industry by producing a methodology for implementing fast and efficient ICs with a faster time-to-market [24]. However, this innovation has largely been locked behind expensive EDA tools and the Intellectual Property (IP) of Semiconductor Manufacturers (SM) that hinder hobbyists and designers who lack the backing of a larger group with the means to provide these tools. Cadence Design Systems has historically produced the industry standard tools used by most IC designers. For layout design, Virtuoso is the primary software used, which can also include a schematic editor and circuit simulation software.

The cost of licensing this software is not publicly disclosed, with a final price being determined between Cadence Design Systems and the party wanting to license the software. An article from 2003 by the EETimes wrote that "U.S. pricing for a one-year license starts at $140,000 for Virtuoso Multi-mode Simulation, $15,000 for Virtuoso XL and $100,000 for Virtuoso Silicon Analysis." [9]. With this article being close to two decades old, it's sensible to assume that the cost of licensing these products have remained constant or likely increased. This demonstrates that using industry standard IC design tools is realistically only available to larger groups. Fortunately, open source options for EDA tools are available for smaller companies and hobbyists, most of which have maintained constant support and updates over the years, allowing for a design flow that is free of commercial tools. Although, these open source tools do not have the robust functionality and features that products from Cadence

or Synopsys can provide in one software package. But, with the design of a custom EDA flow built around these open source tools, they can stand to compete with the functionality and features of industry standard tools.

Open source software (sometimes referred to as "Freeware") is the free and open distribution of source code for software applications. Open source software promotes a collaborative development process, often resulting in higher quality, faster upgraded, and more secure software. Some of the open source EDA tools available today have been in development for decades, receiving constant support and improvements, reaching a point where they are comparable in functionality with industry standard tools. The primary components needed for a standard SoC/IC design flow comprise of a Process Design Kit (PDK) provided by a foundry, a standard cell library to develop ICs at scale, layout/schematic/circuit simulator tool, a Register Transfer Level (RTL) synthesis tool for generating gate level netlists, and a Place and Route (PnR) tool that digitally models the physical representation of the design, which is then given to the foundry for fabrication. Arguably, the most essential part of IC design is the standard cell library. These libraries provide the capability of making ICs with hundreds of thousands of on chip transistors. Most designers utilize standard cell libraries designed by other parties, requiring a license to use. Currently, only Skywater Technologies provides a small set of open source standard cell libraries. There is a large gap in open source IC design when it comes to standard cell libraries available for different applications.

Standard-cell libraries are robust, in that they can be use to develop ICs for different applications with ease. Several SoC-based standard-cell libraries have been created that users can create or purchase through specific Intellectual Property (e.g., ARM). More importantly, the availability of EDA-based tools allows many standard cells to be created for radiation-hardened environments [14, 18]. Although these applications are robust and innovative, they tend to use specialized PDKs [5]. Comparable radiation hardness can also be achieved without a specialized PDK by using Radiation Hardened by Design (RHBD) techniques. These techniques are mainly used when doing layout design of standard cells.

These techniques are relatively easy to implement for an experienced layout designer, and can transform a traditional standard cell library into a library capable of almost any heavy radiation environment application. Considerations do have to be made for the increased area and power consumption that follows implementing these techniques. Further explanation of RHBD will be covered in chapter III.

The motivation behind the development of a radiation hardened standard cell library using an open source flow is that there are currently no open source options for radiation hardened libraries. Recently, the space industry has seen a boom in popularity and market share with the advancements in rocket technology brought forth by companies like SpaceX, Blue Origin, and United Launch Alliance [15]. In 2020, the global space economy rose from 428 billion to 447 billion, a 4.4% increase from 2019 and a 55% increase from over a decade ago [2]. With this renewed interest in the space industry, radiation hardiness of Systems on a Chip (SoCs) must keep up with the increasing radiation tolerance requirements. Having an open source option available can allow smaller companies to enter into the space industry and provide reliable SoCs that are marketable to the growing industry, with a zero cost to entry. Innovations and improvements to open source tools and standard cell libraries could also produce a boom to the space industry, adding more collaborators who produce better and higher quality SoCs that ultimately progress the industry further.

Outside the protection of the earth's atmosphere, CMOS devices cannot perform reliably without some form of radiation protection. High radiation environments can also cause irreversible damage to CMOS devices, ultimately leading to total device failure. Radiation hardening techniques have been developed over time that allow ICs to perform reliably for long periods of time. These techniques ultimately come at the cost of increased power and area, along with a full redesign of the standard cell library. For companies wanting to enter into the space industry, the time and capital investment needed for the complete redesign of the standard cell library could prove to be not worthwhile. In certain applications, a specialized PDK may need to be utilized to provide additional radiation hardness. This

would require the added cost of obtaining a license for the usage of a SM's PDK. Again, the barrier of entry into the industry continues to grow higher. This shows a severe need for an open source solution to be available to everyone. Even if the library is not used directly, it can be used as a reference for other custom libraries to be made, and a source of comparison for power/timing characteristics. The radiation hardening techniques discussed in this thesis and implemented into a custom standard cell library are not exhaustive, but have allowed for radiation hardened designs to be made that meet the current demands of space applications.

This thesis is organized as follows. Section II will provide a background on what comprises a standard IC design flow, radiation hardened libraries, and why they are important. Section III will detail how heavy radiation environments affect CMOS devices and techniques to harden the design. Section IV will go into more depth on what comprises an IC design flow and what open source tools and methods were used in the creation of this radiation hardened standard cell library. Section VI will present metrics of the radiation hardened standard cell library and the results of a placed and routed design using this library. Section VII will present the conclusions of using a mainly open source EDA flow to design SoC/ICs for fabrication.

CHAPTER II

BACKGROUND

SoC architectures designed for use in space, nuclear plants, and certain aeronautical/military applications face unique design challenges that conventional hardware does not. High radiation environments can cause irreversible damage to SoCs resulting in total failure of non hardened chips [16, 20, 25]. Radiation hardening techniques must be employed for any reliable SoC/IC device to operate reliably in these environments.

Depending on the intensity and duration of radiation exposure, there are three main effects that SoC/IC designs can experience. Total Ionizing Dose (TID), Single Event Upsets (SEU) which is sometimes referred to as Single Event Transient (SET), and Single-Event Latchup (SEL) [3]. TID effects come from the cumulative dose of radiation the design receives over a period of time that irreversibly change the characteristics of devices, specifically the on-chip N and P type Metal Oxide Semiconductor Field Effect Transistors (MOSFET). Characteristics like off-state drain current and a reduction in threshold voltage for N type devices, ultimately leading to a massive increase in leakage current, and eventually, complete device failure. Over time, the threshold voltage for N and P type MOSFETS change due to high energy particles, such as protons, striking the oxide and creating electron-hole pairs [6]. For N type MOSFETS in a design, they become "leaky" due to the decreased threshold voltage and parasitic channels formed between source and drain caused by TID effects [13].

SEUs create metastability in digital logic, specifically for memory devices such as latches and flip flops. These upsets (also called soft errors) are not permanent, but cause an increased error rate requiring more robust error detection and error correction schemes to be

utilized [3]. It is essential to implement techniques to combat these effects for SoCs to have a usable life span with high reliability. Radiation Hardening by Design (RHBD) is a technique to harden standard cell libraries without utilizing a PDK specific for high radiation environments [21]. This allows for non radiation hardened standard cell libraries to be converted over to radiation hardened libraries. RHBD comes in two forms, hardening through layout design and hardening through architecture design.

An example of an already existing radiation hardened standard cell library that employs RHBD techniques is a commercial 0.18μm CMOS technology designed by J. Liu et al. [15]. This library employed radiation hardened techniques such as temporal filtering structures and P+/N+ guard rings, which are the same techniques used for this open source library. Through SPICE simulation verification, this library was able to realize a TID tolerance of $>$ 100 Krad(Si), SEL tolerance of $>$ 100 MeV$\cdot$cm^2/mg, and an SEU tolerance of $>$ 37 MeV$\cdot$cm^2/mg. Given that a 0.18μm technology was used, total cell height was 8.86μm and an integral multiple width of 0.66μm. Another radiation hardened library example includes a 0.15μm bulk CMOS process technology standard cell digital ASIC design library made by BAE Systems [18]. The only radiation hardening techniques mentioned in [18] are on-chip decoupling capacitors for noise and transient upset mitigation, and temporal filtering registers for SET mitigation. For library validation, a test chip was developed that contains at least one of every cell in the library, and was subjected to "extensive electrical and radiation evaluations" [18]. The radiation hardness realized was a TID tolerance of $>$ 1 Mrad(Si), SEU tolerance of $< 1^{-10}$ errors/bit-day, and a SEL tolerance of $>$ 120 MeV-cm^2/mg. Its important to mention that this technology also featured shallow trench isolation for leakage current mitigation between devices. Both libraries mention above are proprietary and not available in an open source fashion. Through searching both online and through common electrical/computer engineering literature databases (e.g. IEEExplore), no free and open source radiation hardened standard cell library is available. This is also not including an associated open source EDA design flow that would allow designers an easy and efficient way

to contribute and iterate on the current state of the library.

Expanding further on standard cell libraries, these libraries consist of pre-made cell layouts, usually designed by hand in a layout editor (e.g. Cadence Virtuoso). These cell layouts are typically hand-crafted to a specific cell height and integral multiple width (also referred to as pitch) that give routing layers dedicated space to create connecting nets between instances or cells. Also, these cells conform to this pre-made pitch definitions to allow a Place and Route (PnR) tool to place and route cells together in a cohesive manner. After a cell has been designed in layout, it is good practice to the perform Design Rule Checks (DRC) and Layout vs. Schematic (LVS) on the cell to validate proper functionality. Performing DRC on the cell layout verifies that the physical design conforms with the rules specified in the PDK. This insures that the cell can actually be fabricated by the SM. LVS verifies that the given layout matches an equivalent schematic diagram made in a schematic capture tool (e.g. Cadence Virtuoso Schematic Editor, Xschem). This insures that the operation of the layout is equivalent to what is described in the schematic. This includes matching input/output ports, device width and lengths, and net connections between devices and ports.

After verifying that the layout conforms to all design rules and is equivalent to the schematic, a Library Exchange Format (LEF) file and GDSII (also referred to as calma or just GDS, named after the company that developed the format: Graphic Design Systems) can be generated from the layout. A LEF file is an abstract view of the physical layout described in ASCII format. This mainly contains positional data on the pins present in the layout as well as the associated layer types. Refer to II.1 for an example of a cell macro and pin definition in a LEF file. An associated technology LEF provided by the manufacturer is also used alongside the standard cell LEF which describes the positional design rules of metal layers and vias specific to the technology being used. LEF files can be generated using either a layout tool (e.g. Magic Layout Editor) or a dedicated LEF generation tool (e.g. Cadence Abstract).

Listing II.1: Section of LEF file containing MACRO definition of INVX1 cell and output pin Y definition.

```
MACRO INVX1
  CLASS BLOCK ;
  FOREIGN INVX1 ;
  ORIGIN 0.420 0.075 ;
  SIZE 3.060 BY 7.950 ;
  PIN Y
    ANTENNADIFFAREA 0.771900 ;
    PORT
      LAYER li1 ;
        RECT 1.025 4.895 1.195 7.250 ;
        RECT 1.025 4.725 1.565 4.895 ;
        RECT 1.395 2.305 1.565 4.725 ;
        RECT 1.025 2.135 1.565 2.305 ;
        RECT 1.025 0.975 1.195 2.135 ;
      LAYER mcon ;
        RECT 1.395 3.245 1.565 3.415 ;
      LAYER met1 ;
        RECT 1.365 3.415 1.595 3.445 ;
        RECT 1.335 3.245 1.745 3.415 ;
        RECT 1.365 3.215 1.595 3.245 ;
    END
  END Y
```

GDSII is a binary database file format that describes the entire physical geometry of the cell in a hierarchical format. The GDS files of the standard cells are primarily used by the PnR tool for cell placement and is the file format for the final output of the PnR tool after placing and routing a design. The final GDS file can then be given to the foundry for fabrication. Typically, after generating the LEF and GDS of all the cells, they then need to be characterized by a tool (e.g. Cadence Liberate) that utilizes some form of SPICE simulator (e.g. Cadence SPECTRE, Synopsys HSPICE) to simulate each cell under different input and load conditions and gather timing, power, and capacitance metrics. The standard output of the characterization tool is a Liberty file containing the previous mentioned metrics for each cell. Liberty files can be represented in both ASCII (.lib) or binary database (.db)

format depending on what the synthesis or PnR tool requires. At this point, all relevant files pertaining to the standard cell library and underlying technology have been created, and the creation of an SoC or ASIC design can be performed. This starts with first synthesizing the Register Transfer Level (RTL) code of the design using a synthesis tool (e.g. Synopsys DC Shell, Yosys). Using the LEF and Liberty files, synthesis will produces a gate level netlist that is a complete representation of the RTL code using the cells available in the library. Refer to II.2 for an example snippet of a gate level netlist.

Listing II.2: Section of a gate level netlist using cells available in the standard cell library to describe an RTL code design.

```
NOR2X1 U5528 ( .A(n5250), .B(n5249), .Y(n6650) );
INVX1 U5529 ( .A(n6860), .Y(n5249) );
INVX1 U5530 ( .A(n6654), .Y(n5250) );
INVX1 U5531 ( .A(n11072), .Y(n5251) );
AND3X1 U5532 ( .A(n11074), .B(n11073), .C(n5251), .Y(n11075) );
INVX1 U5533 ( .A(n11318), .Y(n5252) );
INVX1 U5534 ( .A(n8838), .Y(n5253) );
NOR2X1 U5535 ( .A(n8852), .B(n5253), .Y(n5893) );
INVX1 U5536 ( .A(n5254), .Y(n11499) );
INVX1 U5537 ( .A(n11274), .Y(n5254) );
NAND2X1 U5538 ( .A(n12011), .B(n5255), .Y(n7239) );
AOI3X1 U5539 ( .A(n9201), .B(instr[2]), .C(n12004), .YN(n5255) );
```

This gate level netlist can then be passed to the PnR tool (e.g. Cadence Innovus, OpenROAD), where using the netlist along with the LEF and GDS files, will place down cells and then route them together. This process is usually done in multiple steps as shown below. Figure 1 gives a simplified example of how global routing is performed, followed by detailed routing.

1. Floorplanning: Initialize the pad area.

2. Place: Place down cells using the abstract view derived from the LEF file

3. Clock Tree Synthesis (CTS): Balances the clock signals between sequential devices to

reduce gate delay and skew.

4. Global Route: Divides the design into tiles, then determines a general direction for routing between the tiles to optimize wire length and timing.

5. Detailed Route: Routes gates together, assigns metal types for nets, and places vias.

6. Postroute Optimization: Optimizes the routing of entire design after routing has been placed.

7. Signoff: Perform a series of verification steps to verify timing paths and generate reports for metrics.

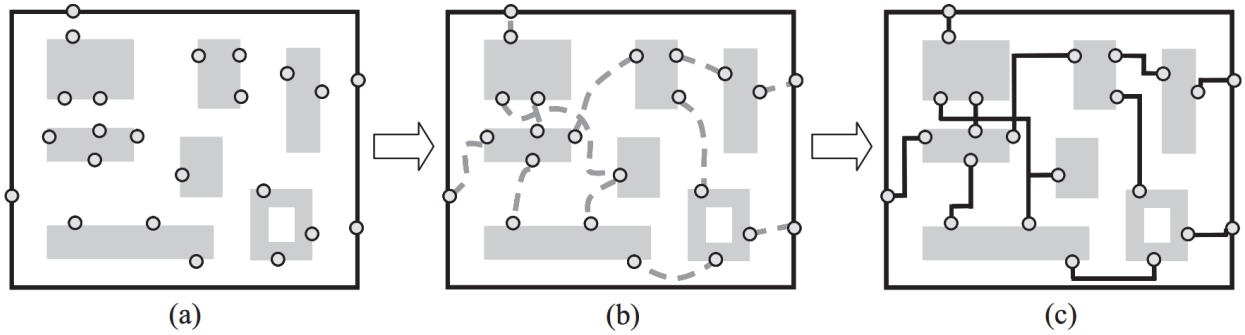


Figure 1: (a) A given placement result with fixed locations of blocks and pins. (b) Global routing. (c) Detailed routing. [23]

CHAPTER III

RADIATION HARDENED BY DESIGN

Radiation hardened standard cell libraries are essential to designing any IC that has to perform reliably in heavy radiation environments Without hardening, operational lifetime is cut drastically short. In space applications, ICs need to survive for long periods of time since repairs and replacement of parts could mean sending another rocket into orbit, costing millions. Also, the health and safety of humans in space rely on these ICs to operate correctly, with failure possibly costing lives. Layout techniques have been developed over time that can increase radiation hardness in standard cells. These techniques do not required a radiation application specific PDK to be utilized and can be implemented into libraries using standard bulk CMOS processes.

3.1 Annular Gates

Total ionizing dose effects inevitably lead to leaky N type MOSFETS that get incrementally worse over time. The reason for this effect is due to the increase in positively charged particles embedded in the oxide. This is due to high energy particles striking the oxide, creating electron-hole pairs. After electron-hole pair creation, some pairs recombine while other become permanently separated. The electrons that do not recombine are attracted to the positive gate voltage of NMOS devices or the positive power rail. The holes are repelled from the positive gate voltage and move towards the SiO_2-Si interface. Some of these positively charged holes become trapped near the SiO_2-Si interface, increasing the charge of the oxide. Figure 2 shows deep hole trappings where positive charges become

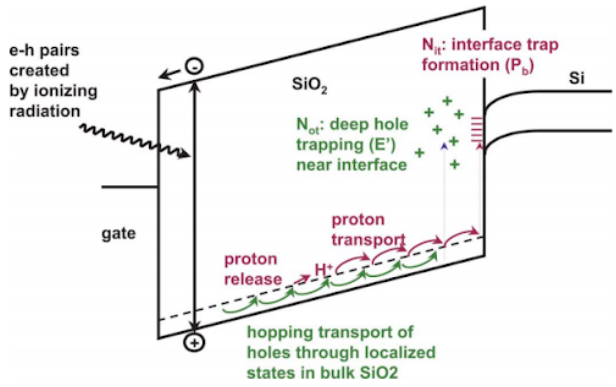


Figure 2: Band diagram of a MOS capacitor with a positive gate bias. Illustrated are the main processes for radiation-induced charge generation [20].

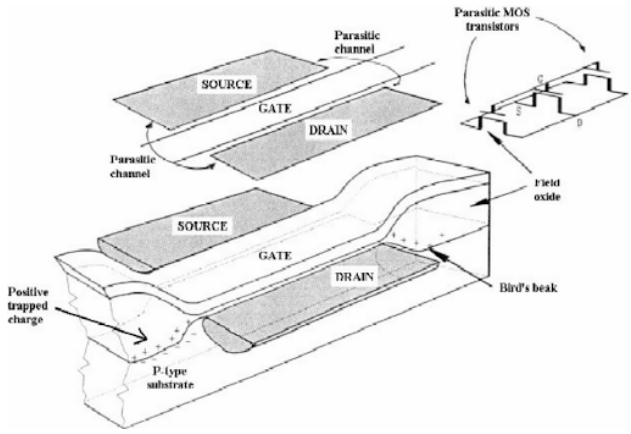


Figure 3: Illustration of trapped holes collecting around the Bird's Beak region.

trapped at the interface causing a drop in the threshold voltage of NMOS devices.

Leakage current is produced by the formation of parasitic channels developing between source and drain, caused by the increase in positively charged particles that manifests near the edges of traditional rectangular gates. Figure 3 shows the "bird's beak" region, where the SiO_2 of the MOSFET gate terminal comes up and over the field oxide [20]. These edges cause parasitic channels to be formed between source to drain, increasing overall power consumption of NMOS devices.

Annular gates provide a solution by completely eliminating edges from the gate oxide, eliminating the parasitic channel between the source and drain. In order to visualize how the source and drain terminals in annular gate transistors are completely enclosed in the

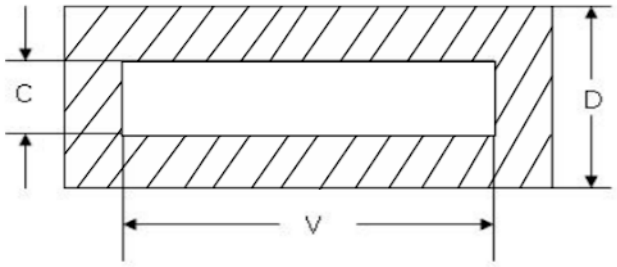


Figure 4: Relevant dimensions of a rectangular gate annular transistor.

gate oxide, refer to Figure 10a.

Annular gate transistors can increase the area of a single transistor by up to ten times that of a traditional rectangular transistor. Furthermore, the effective width and length of annular gate transistors cannot be directly measured like rectangular gate transistors. Figure 4 shows the three relevant dimensions used in the calculation for effective width and length of rectangular annular transistors.

Using these dimension, an effective width and length can be calculated for a rectangular gate annular transistor.

$$W_{eff} = 2 \cdot V + C + D \tag{3.1.1}$$

$$L_{eff} = (D - C)/2 \tag{3.1.2}$$

These equations estimate the effective width and length for annular gates that are perfectly rectangular. The design of the gates designed in this thesis are not perfectly rectangular and incorporate 45° mitered edges. Also, the poly on one side the gate extends away from the main body, referred to as the "tail". The tail still lies in diffusion, therefore it must be accounted for in the equation for effective width and length. An example of a rectangular annular gate with defined dimensions can be found in Figure 5. Using the dimensions shown there, we can modify the equations to account for the added geometries and compute the

resulting effective width and length of an example NMOS. It is important to note that the dimensions shown in Figure 5 are in microns.

$$\begin{aligned} W_{eff} &= 2 \cdot V + D + C + E \\ &= 2 \cdot (0.240) + 0.240 + 0.160 + 0.120 \end{aligned} \tag{3.1.3}$$

$$\begin{aligned} L_{eff} &= \frac{D - C}{2} + \sqrt{S_1^2 + S_2^2} \\ &= \frac{0.240 - 0.160}{2} + \sqrt{0.40^2 + 0.40^2} \end{aligned} \tag{3.1.4}$$

The above calculation is just on of many ways to calculate the effective width and length of annular gate transistors. Extensive research has gone into different methods of dividing annular gate transistors into separate devices that can be added together to find the effective width and length. An article by Bezhenova et al. showcase many different methods of calculating the aspect ratio of annular gate transistors [7]. The method used above was given as an example since the calculated values match exactly with extraction using Calibre PEX. This is however different than the effective width and length produced by Magic extraction. An example of how magic views transistors can be seen in Figure 6. The algorithm for determining the width of these devices incorporates a "boundary scan", although Magic still has to deal with tiles being divided in irregular ways to accommodate 45° angles.

All annular devices created in this library were sized to extract an effective width and length of $W = 3\mu$m and $L = 0.15\mu$m using Magic's extraction algorithm. This choice was made to be consistent in using only open source tooling where possible. The width used for the annular NMOS devices was the smallest width allowed by the DRC rules.

3.2 Diffusion Rings

In high radiation environments it is unavoidable that high incident particles will become embedded into the oxide, altering device characteristics. One method to mitigate this is to

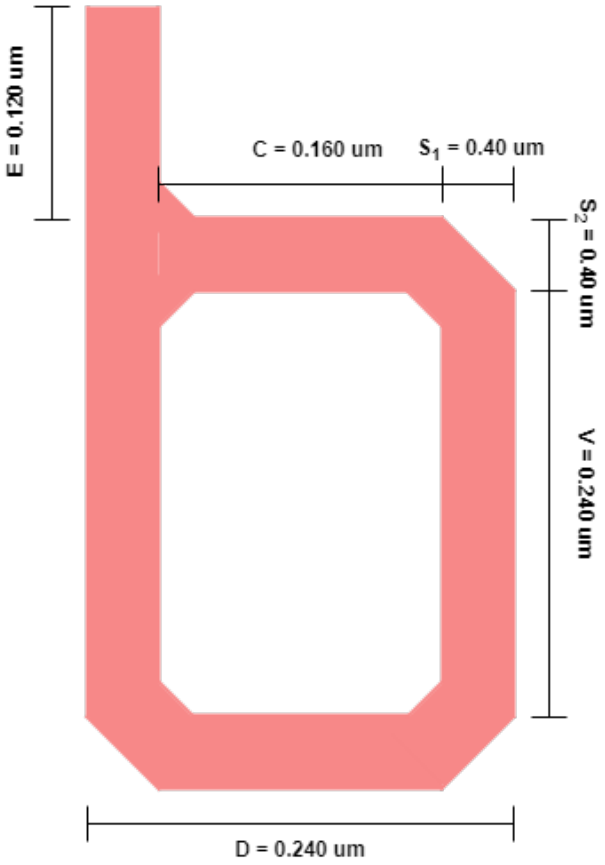


Figure 5: Example of an annular gate with labeled dimensions.

create N+/P+ diffusion rings around same type transistors to draw leakage current away from devices and send them back to the power rails. The substrate diffusion type used is always opposite the device type it is surrounding. For example, N type transistors are surrounded by P type substrate diffusion to draw in stray charges and sends them to the ground rail through the body contacts. Refer to figure 7 for an example of P type diffusion rings surrounding N type transistors. This method reduces inter-device leakage as well as inter-cell leakage throughout the design [12]. Like annular gates, diffusion rings also bring a significant area penalty to the design. Further research needs to be done to find the optimal way to enclose same type devices in a placed design to reduce the overall area penalty.

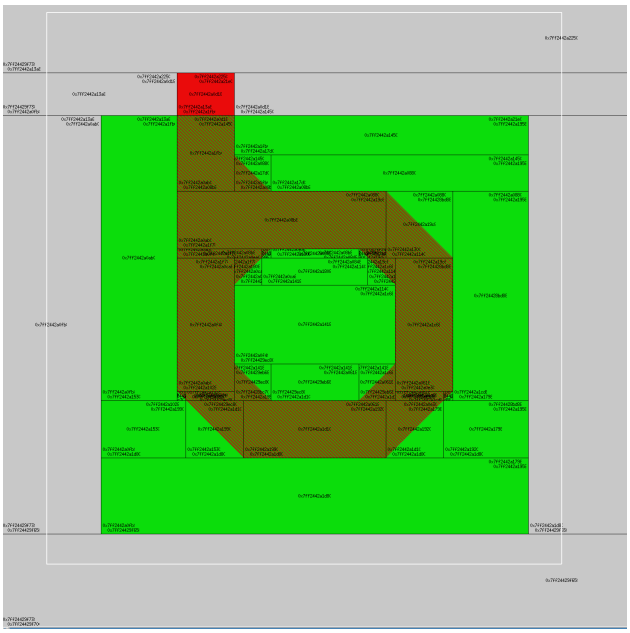


Figure 6: Annular device from the perspective of Magic.

3.3 Triple Modular Redundancy

The use of Triple Modular Redundancy Flip Flops (TMRDFF) is an architecture technique that can mitigate SEUs using a fault tolerant redundant design [11]. TMRDFFs are a form of *spatial redundancy* in digital circuits. Memory storing devices such as D flip flops are particularly vulnerable to SEU since high-incident particles have the chance of causing metastability when latching data. TMRDFFs use three identical flip flops that work in tandem and use a majority voter to determine the output final output [11, 4]. The majority voter will output a 0 if at least two of the flip flops have 0 as the output and vice versa. Again, this comes with a significant area penalty due to the added flip flops and voter cell that replace a regular flip flop. Refer to table 1 and 2 for the truth table and associated schematic of both the VOTER3 cell and TMRDFFQ cell.

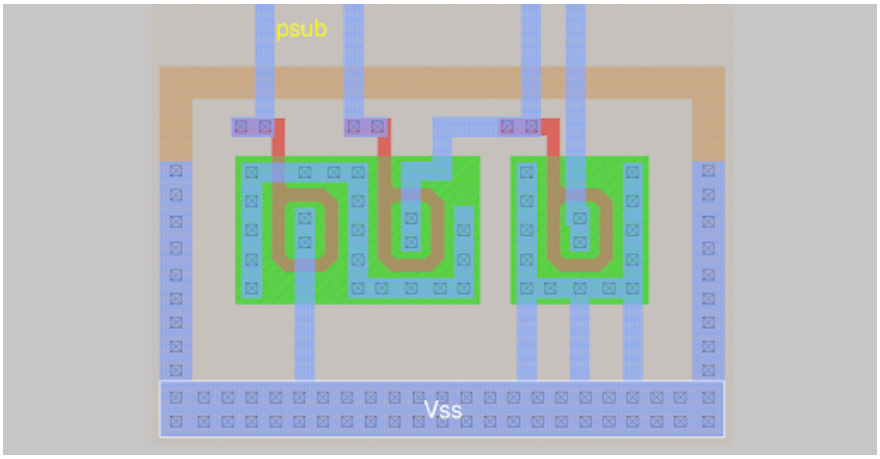


Figure 7: p-type diffusion rings.

A	B	C	YN
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Table 1: Truth Table of VOTER3

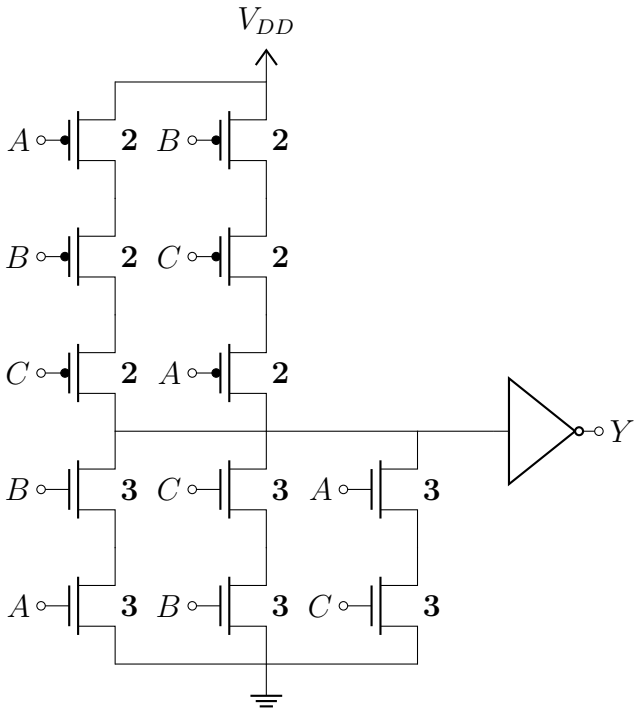


Figure 8: Schematic of VOTER3

X1.Q	X2.Q	X3.Q	Q
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Table 2: Truth Table of TMRDFFQ

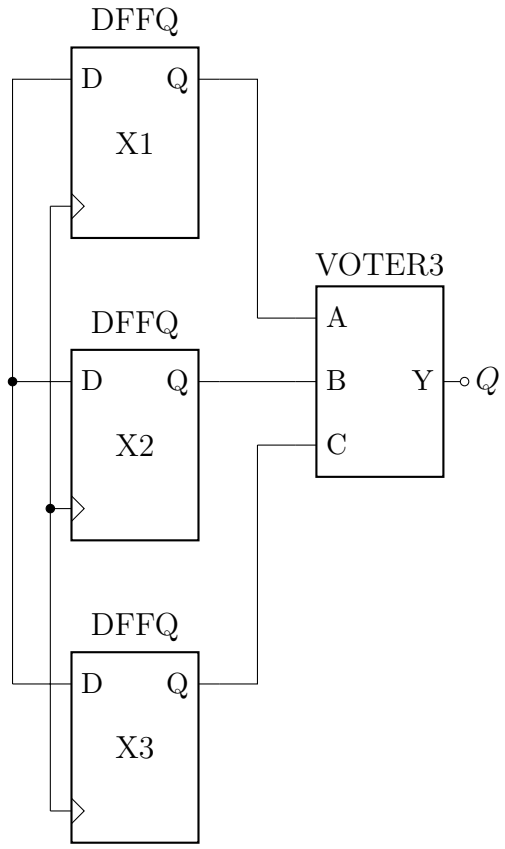


Figure 9: Schematic of TMRDFFQ

CHAPTER IV

OPEN-SOURCE EDA TOOLS AND FLOW

In the previous chapter, a general description was given of a typical IC design flow starting from cell layout creation to a final placed and routed design. In this chapter, a similar description will be made, but specific to this flow, and how open source EDA tools are utilized to perform the required functions needed for IC design. First, an overview of each open source tool and it's features, with a detailed description of the custom EDA flow and the role each tool plays in the process.

4.1 Open Source EDA Tools

4.1.1 Process Design Kit

A PDK is a collection of files and documents given to designers by a foundry. These files are required for designers to develop designs that the foundry can fabricate. In essence, these files describe the function and operation of a specific technology node. This includes technology data like layers, process constraints, electrical rules, etc. PDKs also contains design rules which physical designers must maintain when doing layout. These rules contain things like the minimum width of a metal layer, how close two of the same metal layers can be to each other, the minimum amount of poly overhang when making a transistor. All of these rules must be followed for the foundry to actually fabricate the design. These PDKs are given from a SM to designers who wish to design something using their specific technology and process. Most PDKs do not come free, and require a license to be purchased. This can cost upwards of hundreds of thousands of dollars. They also come attached with

a Non-Disclosure Agreement that must be signed by anyone working with the files, barring outside collaborators from contributing. This high cost of entry is often to much for hobbyists and designers who are not part of a larger organization. There exists today an open source PDK developed by North Carolina State University in collaboration with Mentor Graphics called FreePDK. FreePDK offers everything a proprietary PDK offers, except that anything designed using it cannot be fabricated in reality. Since FreePDK was not developed by an actual foundry, the files and rules included are an abstraction of what an actual PDK would include. This PDK is useful for learning purposes, but lacks the ability to bring designs into the real world. Recently, Skywater Technologies in collaboration with Google, has released an open source 130nm PDK called SKY130. Unlike FreePDK, Skywater Technologies is a foundry that can fabricate designs built with SKY130. Before the release of SKY130, an open source IC design flow was impossible. Along with the PDK, Skywater also released seven basic open source standard cell libraries for digital design including high density, low voltage, high voltage, etc. This is the PDK that is used for the development of the radiation hardened standard cell library in this thesis.

4.1.2 Magic Layout Editor

Magic VLSI Layout Editor from Open Circuit Designs [8] is an open source layout editor that rivals industry standard tools in its features and capabilities. Originally, Magic was written in the 1980's by John Ousterhout. It then went through a dormant period before being publicly released in September of 2000. Since then it has had continued support and development, remaining popular with hobbyists and universities. Magic is a tool that works well with Tool Scripting Language (Tcl) being one of the reasons why it has kept it's relevance to this day. Along with its layout editing capabilities, Magic is also able to produce LEF and GDS files from layout.

4.1.3 Xschem Schematic Capture

Xschem [19] is an open source schematic capture tool that again, has functionality and capabilities that rival industry standard tools. Xschem can be configured to use device models from the SKY130 PDK, which can be directly added into the schematic. This allows Xschem to accurately extract the SPICE netlists of the schematic. Xschem also has the capability to perform SPICE simulations directly in the schematic editor, allowing for easy testing.

4.1.4 Ngspice Netlisting and Simulation

Ngspice [22] is an open source circuit simulator that can perform linear and nonlinear analyses. It includes support for all primitive electrical devices such as resistors, capacitors, inductors, voltage and current sources, etc. It also supports the simulation of nonlinear devices such as MOSFETS, BJTs, MESFETS, JFETS, diodes, switches, etc. Ngspice inherits from the XSPICE framework and supports both digital and analog simulation algorithms. It supports user defined nodes which is useful in digital simulations and is the primary circuit simulator used in this thesis. Like Xschem, there is support for the SKY130 models which allows for accurate testing of cells.

4.1.5 Liberty Characterization

Currently, no reliable open source option is available for Liberty characterization. Due to this, the proprietary tool Calibre Liberate was used to generate the Liberty file for this radiation hardened standard cell library.

4.1.6 RTL Synthesis and Place and Route

Open source options for RTL synthesis and PnR are available, such as Yosys [26] and OpenROAD [17]. OpenROAD is a complete RTL to GDS flow that incorporates Yosys as the RTL synthesis tool. The integration of OpenROAD into this EDA flow is still in progress,

but full integration is estimated to be complete in the near future. For the purposes of this thesis, Synopsys Design Compiler was used to generate the gate level netlist, and Cadence Innovus was used as the PnR tool.

4.2 EDA Flow

As was mentioned before, the following will be a more detailed explanation of the EDA flow described in the previous chapter, specific to this open source flow.

4.2.1 Planning

This first step in this process began with determining what PDK would be used and creating a list of what cells along with their associated drive strength will be included in the final library. With Skywater Technologies offering an open source PDK along with opportunities to be included in fabrication runs, it was the obvious choice.

As for the included cells, Table 3 lists the initial collection of basic logic gates of times 1 drive strength currently implemented in the library. This collection of gates is adequate to perform syntheses on most RTL designs. There are future plans to add more drive strengths, as well as more niche and complex logic gates.

4.2.2 Standard Cells

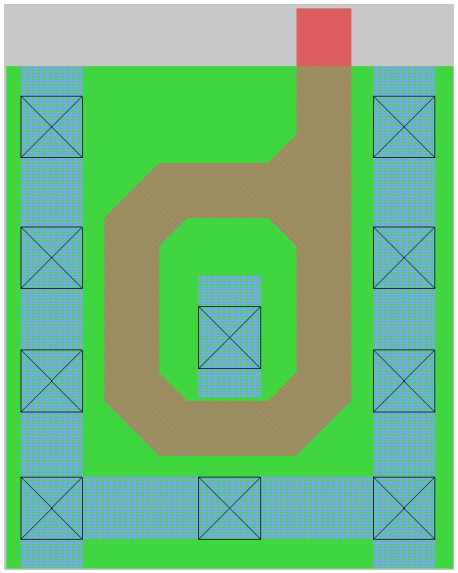
The primary focus of this thesis is to detail the design and operation of a radiation hardened standard cell library. A standard cell library is a collection of "standard cells", essentially boolean logic gates, that have been fully designed in layout and can be given to a tool to perform place and route. These primitive cells include INV, NAND, NOR, AND, OR, DFF, BUF, AO, AOA, TMRDFF, VOTER, and XOR gates, along with other more specialized cells. Depending on the library, they can include over 200 individual cells. Many come with the same cells of different sizing's which allow for higher drive strengths. There are many design considerations to make when designing standard cells. One important consideration

is deciding what track height to use. Track height refers to the distance between metal wires going horizontally through the layout. This will ultimately control how high the standard cells must be, since there must be room in the cell to connect nets together. The height of cells designed here are 7.77μm. This is larger than most non-radiation hardened libraries, due mainly to the amount of RHBD techniques used in each cell. A self imposed design rule used in this library was to set port via connections directly on the grid. A 0.37μm by 0.37μm grid was used to place not only via connections but also the side diffusion rings. This makes each standard cell width a multiple of the grid spacing. This is done to help the PnR tool be more effective in placing cells as well as routing. Each metal used for routing was also only used in a specific direction, either horizontal or vertical. This again is to help the PnR tool effectively route between cells. With concrete directions for routing layers, strips of metal could be routed horizontally or vertically through each cell without fear of accidentally connecting two of the same metal nets together.

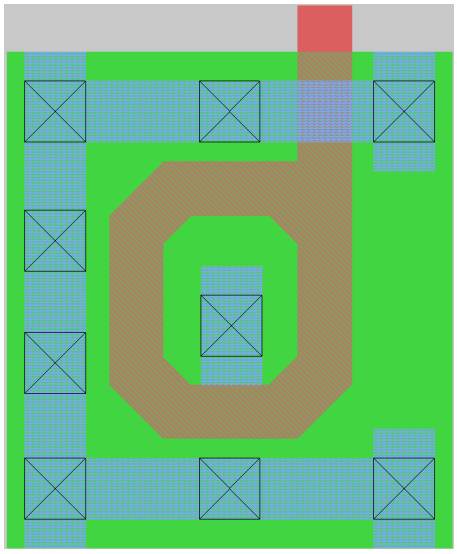
4.2.3 P Cells

A design consideration to take special notice of is the use of p cells in this library. P cell, short for "parameterized cell", is a smaller, more primitive element that goes into each standard cell. For example, both NMOS and PMOS devices were made into p cells. For the NMOS devices, specific widths and lengths of each side are needed to produce a device with the correct effective width and length parameters. Figure 10a shows the p cell of an NMOS device in the top connection configuration. Figure 10b shows a similar NMOS device, but in the side configuration instead. These p cells could be imported into each cell design, assuring consistent sizing of devices between cells.

Due to the height of the NMOS devices used in this library, folding of the PMOS devices was required to keep the height of standard cells being very large while also keeping a PMOS/NMOS width ratio above 1. Figure 11 shows a folded PMOS device with two "fingers". Fingers typically refer to the individual lengths of gate oxide in the PMOS device.



(a) P cell of annular NMOS device in top connection configuration.



(b) P cell of annular NMOS device in side connection configuration.

Folding essentially multiplies the width of the device depending on how many fingers are present, adding width to the cell without increasing the overall height. Folding of a single device can be done as many times as necessary to achieve the required W/L ratio.

Other instances of p cells created for this library include multiple variations of NMOS devices, poly to local interconnect vias, local interconnect to metal 1 vias, and multiple iterations of diffusion rings. The use of p cells ultimately increases productivity while keeping device characteristics consistent between separate standard cells.

4.2.4 Layout Design

Next, layout development is started using Magic. To optimize productivity, p cells of the annular gate NMOS device and its multiple configurations were created first, followed by the folded PMOS devices, diffusion ring sides, poly to local interconnect and local interconnect to metal 1 vias. Multiple iterations of the NMOS device configurations had to take place, making small changes to the lengths of the poly on each side so that magic extraction would produce the same device parameters for each configuration. This ultimately sped up layout

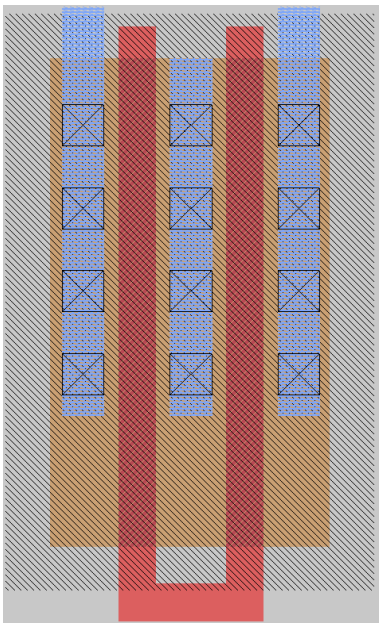


Figure 11: P cell of folded PMOS device.

design, leading to future plans of adding p cells for power rails and PMOS devices with added fingers. In parallel with layout development, schematics of each cell were created in Xschem for future LVS testing.

4.2.5 DRC, LVS, and File Generation

After layouts and schematics had been created for every cell, Ngspice netlists were generated from the schematics using Xschem, while GDSII and Ngspice files of the layout were generated using Magic. Using Magic's built in DRC functionality, design rule violations could be fixed immediately while doing layout. LVS was then performed on the netlists using Netgen. A custom GNU Makefile was created to automate file generation, performing LVS, and generating reports identifying cell mismatches. This again increased the productivity of the flow since property errors and net mismatches had to be address multiple times requiring the layout netlist to be regenerated. After verifying that the cells pass DRC and LVS, a LEF file containing a macro definition for each cell was generated using Magic through a custom Tcl script. This again was all handled by the Makefile that would run Magic along with a

custom Tcl script.

4.2.6 Liberty File Generation

Next was to generate the Liberty file. For this, the commercial tool Cadence Liberate had to be used since there are currently no open source tools that can generate accurate and reliable Liberty files. Cadence provides a formatting template to aid in the characterization process, but multiple additions and modifications had to be made to accommodate the unconventional layout techniques present in the layout. With the Liberty file generated, timing and power metrics could be gathered, and are tabulated in chapter V.

4.2.7 Synthesis and PnR

With all relevant files generated, a design to test functionality and metrics of the library could then be synthesized and put through PnR. The design chosen was a 32-bit single cycle processor using the open standard RISC-V instruction set architecture (ISA) with the base integer set extension. The RTL was developed by the Oklahoma State University VLSI Architecture Research Group. It follows the architecture proposed in *Digital Design and Computer Architecture, RISC-V Edition: RISC-V Edition* [10]. Further explanation of RISC-V and it's implementation are discussed in chapter VI. Synthesis was performed on the RTL using Synopsys DC Shell followed by PnR using Cadence Innovus. Again, explanation and results of Synthesis and PnR are discussed in chapter VI. To reiterate, both synthesis and PnR were performed using proprietary tools, but plans to incorporate open source options are currently in progress.

Gate	Drive Level
ADDF	X1
ADDH	X1
AND2	X1
AND3	X1
AO3	X1
AOA4	X1
AOAI4	X1
AOI3	X1
BUF	X1
DFFQN	X1
DFFQ	X1
DFFRNQN	X1
DFFRNQ	X1
DFFRN	X1
DFFSNQN	X1
DFFSNQ	X1
DFFSNRNQN	X1
DFFSNRNQ	X1
DFFSNRN	X1
DFFSN	X1
DFF	X1
DLATCH	X1

Gate	Drive Level
DLATCHN	X1
MUX2	X1
INV	X1
NAND2	X1
NAND3	X1
NOR2	X1
OR2	X1
TIEHI	X1
TIELO	X1
TMRDFFQN	X1
TMRDFFQ	X1
TMRDFFRNQN	X1
TMRDFFRNQ	X1
TMRDFFSNQN	X1
TMRDFFSNQ	X1
TMRDFFSNRNQN	X1
TMRDFFSNRNQ	X1
VOTER3	X1
VOTERN3	X1
XNOR2	X1
XOR2	X1

Table 3: Included cells in sky130 radiation hardened standard cell library.

CHAPTER V

CELL IMPLEMENTATION

5.1 Standard Cell Layout Characteristics

Following are characteristics of each standard cell layout gathered from the Liberty file. The construction of the Liberty file was briefly touched on in chapter II, but further explanation is needed to understand the data presented in table 4. The Liberty file contains timing, leakage power, total power, and pin capacitance metrics for each cell and pin in the library under different load and input conditions. These values are then tabulated into look up tables that are NxN dimensions, depending on how many different load and input conditions are simulated. For example, the code in V.1 shows two index variable being defined, with *index_1* defining a list of input waveform rise times, which is measured from 20% of VDD to 80% of VDD in nanoseconds. The *index_2* variable defines the output load condition of the pin. There are six values in each index, resulting in a 6x6 table called *values* being created with *values*[x][y] being the cell rise time when simulated with an input rise time of *index_1*[x], and an output load condition of *index_2*[y]. Liberate can independently determine what the optimum index values should be, but they can also be directly assigned by the user.

Listing V.1: Beginning of Liberty cell definition for the output pin Y for the AND2X1 cell.

```
pin (Y) {
  direction : output;
  function : "(A * B)";
  power_down_function : "(!VDD) + (VSS)";
  related_ground_pin : VSS;
  related_power_pin : VDD;
  max_capacitance : 5.40378;
  timing () {
    related_pin : "A";
    timing_sense : positive_unate;
    timing_type : combinational;
    cell_rise (delay_template) {
      index_1 ("0.00997761, 0.0397394, 0.158276, 0.630392, 2.51076, 10");
      index_2 ("0.00864206, 0.0313208, 0.113514, 0.411401, 1.49101, 5.40378");
      values ( \
        "0.0522372, 0.0823179, 0.18595, 0.558322, 1.9071, 6.79535", \
        "0.0607922, 0.0906772, 0.194341, 0.566817, 1.91561, 6.80387", \
        "0.0839701, 0.114189, 0.218106, 0.590889, 1.93979, 6.82809", \
        "0.110947, 0.144615, 0.248472, 0.621684, 1.97083, 6.85906", \
        "0.078709, 0.131275, 0.246922, 0.625769, 1.97665, 6.86493", \
        "-0.291388, -0.202547, -0.0145038, 0.413244, 1.81406, 6.71147" \
      );
    }
```

The look up tables included in a pin timing definition are cell rise, cell fall, rise transition, fall transition, output current rise, output current fall, and receiver capacitance. This is not to mention other parameters included in the pin definition pertaining to power and capacitance. When performing synthesis, the tool will reference these tables to determine the propagation delay of a certain path. This is how preliminary timing reports can be generated just from the gate level netlist. With this in mind, table 4 lists relevant characteristics derived from the Liberty file. Since there are a large amount of values associated with each cell, table 4 will list the timing metrics associated with the minimum and maximum index values. Since the minimum and maximum index values vary between cells, they are included in the table for each cell.

Since this library utilizes TMRDFFs, the design trade-off between the increased area of

Cell	Area(μm)	Leakage Power(nW)	Min Fall(ns)	Max Fall(ns)	Min Rise(ns)	Max Rise(ns)	Min:Max Input Rise(ns)	Min:Max Load(pF)
AND2X1	50.801	1.265	0.054	4.460	0.052	6.711	0.010:10.0	0.009:5.404
AND3X1	62.567	1.264	0.015	4.229	0.070	6.779	0.010:10.0	0.009:5.388
AO3X1	77.274	5.000	0.018	4.233	0.084	7.78	0.010:10.0	0.009:5.401
AOA4X1	103.748	5.000	0.014	4.231	0.149	6.570	0.010:10.0	0.009:5.400
AOAI4X1	86.099	5.000	0.218	6.376	0.927	7.556	0.010:10.0	0.009:5.212
AOI3X1	59.625	5.000	0.017	2.084	0.830	6.634	0.010:10.0	0.009:2.625
BUFX1	41.980	5.425	0.123	4.260	0.361	6.601	0.010:10.0	0.009:5.425
DFFQNX1	177.285	7.691	0.135	7.245	0.116	9.719	0.010:10	0.009:7.691
DFFQX1	177.285	5.056	0.148	7.262	0.110	9.726	0.010:10	0.009:7.681
DFFRNQNX1	212.583	5.980	0.180	7.700	0.130	7.535	0.010:10	0.009:5.861
DFFRNQX1	212.583	5.980	0.165	7.268	0.146	9.876	0.010:10	0.009:7.691
DFFRNX1	212.583	5.980	0.182	21.678	0.146	9.965	0.010:10	0.009:7.691
DFFSNQNX1	200.817	5.392	0.122	9.741	0.129	9.741	0.010:10	0.009:7.691
DFFSNQX1	200.817	5.392	0.175	7.630	0.093	7.449	0.010:10	0.009:5.861
DFFSNRNQNX1	236.115	5.914	0.164	7.554	0.132	7.467	0.010:10	0.009:5.861
DFFSNRNQX1	236.115	5.914	0.181	7.508	0.127	7.546	0.010:10	0.009:5.861
DFFSNRNX1	236.115	5.914	0.197	17.693	0.127	7.599	0.010:10	0.009:5.861
DFFSNX1	200.817	5.392	0.193	18.005	0.093	7.548	0.010:10	0.009:5.861
DFFX1	177.285	5.056	0.100	0.200	0.051	14.096	0.010:10	0.009:7.691
DLATCH	165.500	2.721	0.100	0.200	0.083	0.400	0.010:10	0.009:2.722
DLATCHN	183.200	5.425	0.100	0.200	0.300	0.400	0.010:10	0.009:2.722
ADDF	315.536	6.932	0.100	0.200	0.300	0.400	0.010:10	0.009:5.861
ADDH	139.046	3.024	0.100	0.200	0.300	0.400	0.010:10	0.009:5.861
INVX1	24.327	0.506	0.100	0.200	0.300	0.400	0.010:10	0.009:5.861
MUX2X1	103.748	2.531	0.100	0.200	0.300	0.400	0.010:10	0.009:5.381
NAND2X1	33.152	0.506	0.100	0.200	0.300	0.400	0.010:10	0.009:5.174
NAND3X1	44.918	0.001	0.100	0.200	0.300	0.400	0.010:10	0.009:3.057
NOR2X1	33.152	0.378	0.100	0.200	0.300	0.400	0.010:10	0.009:2.625
OR2X1	50.801	0.633	0.100	0.200	0.300	0.400	0.010:10	0.009:5.403
TMRDFFQNX1	597.920	14.903	0.168	1.336	0.253	7.373	0.010:10	0.009:3.634
TMRDFFQX1	615.569	15.409	0.285	3.514	0.192	6.931	0.010:10	0.009:5.391
TMRDFFRNQNX1	703.814	17.939	0.222	1.529	0.279	7.400	0.010:10	0.009:3.616
TMRDFFRNQX1	721.463	18.613	0.302	3.578	0.248	7.133	0.010:10	0.009:5.401
TMRDFFSNQNX1	668.516	15.928	0.157	1.287	0.305	7.451	0.010:10	0.009:3.631
TMRDFFSNQX1	686.165	16.267	0.335	3.602	0.181	6.889	0.010:10	0.009:5.403
TMRDFFSNRNQNX1	774.410	17.647	0.219	1.498	0.319	7.442	0.010:10	0.009:3.624
TMRDFFSNRNQX1	792.059	18.053	0.343	3.629	0.244	7.139	0.010:10	0.009:5.435
VOTER3X1	103.748	0.979	0.025	4.245	0.028	9.988	0.010:10	0.009:5.395
VOTERN3X1	86.099	0.473	0.038	0.060	0.099	9.911	0.010:10	0.009:2.081
XNOR2X1	94.923	1.761	0.100	0.200	0.300	0.400	0.010:10	0.009:2.650
XOR2X1	94.923	1.761	0.100	0.200	0.300	0.400	0.010:10	0.009:2.658

Table 4: Cell layout characteristics of SKY130 Radiation Hardened Standard Cell Library. Minimum and maximum values are associated with the minimum and maximum index values for input waveform rise time and output load capacitance specified in the Liberty file.

the cell vs. the added radiation hardness it provides needs to be considered when synthesising and performing PnR on a design. A comparison between area shows that TMRDFF cells are on average 3.36x larger than standard DFF cells. This increase in area is coherent since TMRDFF cells are comprised of three standard DFFs. The clock to Q time of the TMRDFF shows only a 1.73x increase from the standard DFFQ cell. This comes from the added propagation delay produced by the VOTER cell. The average increase in clock to Q delay of the TMRDFF cells is 2.05x that of the standard DFFs. These values also cohere with what was directly simulated in Ngspice. The Ngspice netlist shown in V.2 was used to measure the cell rise time of the TMRDFFQX1 cell. A cell rise time of 0.180ns matches almost exactly to what is found in the Liberty file, having a percentage difference of only 6.45%.

Listing V.2: Ngspice netlist to measure cell rise.

```
** TMRDFFQX1.sp

**** Voltage Sources ****
V1 VDD GND 1.8
V2 CLK GND pulse 0 1.8 0 16.667p 16.667p 20n 40n
V3 D GND PWL 0p 1.8
************************

**** Capacitive Load ****
C1 Q GND 0.009p
************************

**** Cell Instance ****
x1 Q D CLK VDD GND TMRDFFQX1
**********************

**** Include Netlist ****
.inc TMRDFFQX1.spice
************************

**** Being Simulation ****
.control
tran 0.1p 1n
meas tran cell_rise TRIG v(CLK) VAL=0.9 RISE=LAST TARG v(Q) VAL=0.9 RISE=LAST
.endc
************************
.end

**** OUTPUT ****
No. of Data Rows : 10011
cell_rise=1.801318e-10  targ=1.884651e-10  trig=8.333333e-12
***************
```

CHAPTER VI

DESIGN RESULTS

The next step in the design flow is to synthesis and PnR an actual design to gain metrics on the standard cell libraries performance. Up to this point, only open source tools were used to create the library and all relevant files. Where this streak must be broken is in the generation of the Liberty characteristics file needed for synthesis and PnR. Currently, there are no feasible open source Liberty characterization tools that provide accurate and reliable results. With that said, Cadence Liberate was used to perform timing analysis on the library. For synthesis and PnR, there are reliable open source tools that can perform these steps, specifically OpenROAD. Since performance metrics and feasibility of the radiation hardened standard cell library is the goal of this thesis, Synopsys DC Shell will be used for synthesis and Cadence Innovus used for PnR. It is a future goal of this project to integrate this radiation hardened library into the OpenROAD RTL to GDS flow.

6.1 32-bit RISC-V Single Cycle Processor

6.1.1 RISC-V

RISC-V stands for "Reduced Instruction Set Computer" and is an instruction set architecture [1] that has gained in popularity recently due to its open source documentation and licensing. Version 2.0 of the RISC-V spec supports 32, 64, and 128 bit instructions along with multiple extensions including:

- I : Base integer.

- E : Reduced base integer instructions for embedded systems.
- M : Integer, multiplication, and division.
- A : Atomic instructions.
- F : Single-precision floating-point.
- D : Double-precision floating-point.
- Q : Quad-precision floating-point.
- L : Decimal floating-point.
- C : Compressed instructions.

An attractive feature of the RISC-V ISA is the use of instruction types with dedicated bit field locations that simplify the usage of multiplexers as well as sign extension, refer to figure 12.

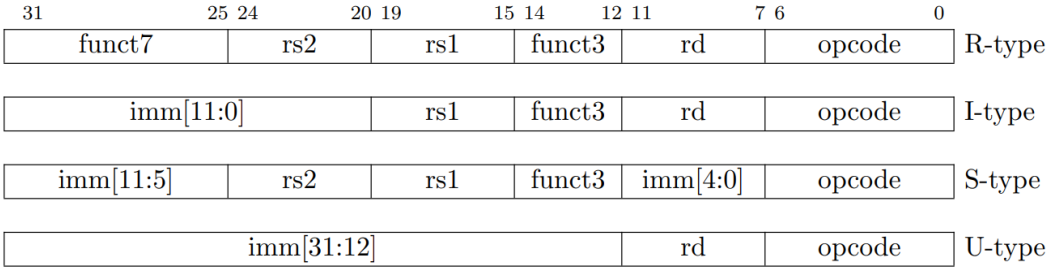


Figure 12: RISC-V version 2.0 instruction types.

For the complete specification of the RISC-V ISA, refer to the RISC-V instruction set manual [1].

6.1.2 Implementation

Using the files generated from the open source tools along with the Liberty characterization file, a 32-bit RISC-V Single Cycle processor was synthesized and then placed and routed using

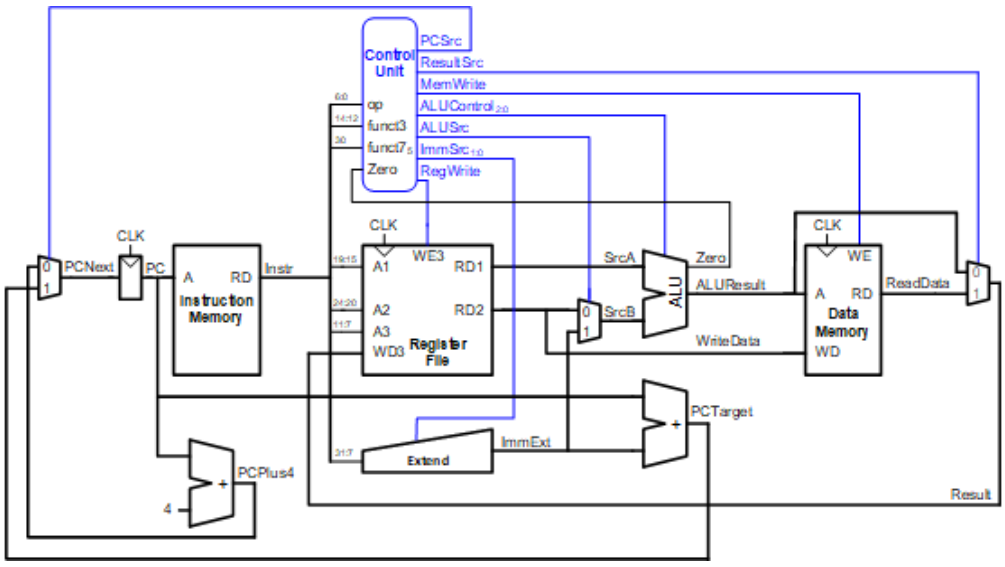


Figure 13: Block diagram of single cycle RV32I implementation [10]

the SKY130 Radiation Hardened library. This single cycle RISC-V processor was designed using System Verilog and includes the the base integer set, RV32I. A block diagram of the architecture can be seen in figure 13.

The organization of the design can be split into 5 main stages.

- Instruction fetch: The instruction is fetched from memory, determined by the value of PC, and given to the decoder.

- Decode: Once the instruction has been fetched, the decoder module assigns the specific control signals unique to that instruction. These control signals consist of what type of operation to perform (load/store, arithmetic, branch), which register to pull values from, and where to write the output.

- Execute: The instruction is then executed, with values pulled from register file or immediate values being used by the Arithmetic Logic Unit (ALU) to perform the required operation. The output of the ALU could be a value that's going to be written back into the register file, or a memory address for a branch instruction.

- Memory: If values are required to be accessed from the data memory, this stage will

retrieve them using the address calculated by the ALU in the previous stage.

- Writeback: Values are then written back into either the register file or data memory, and the process repeats.

Using this design, synthesis was performed using Synopsys DC Shell. Table 5 list all the metrics produced from synthesis. The total cell area that synthesis estimates is 598,785.399 μm. This estimate, however, does not take into account actual cell placement and routing. DC Shell just knows the area of each cell, how many cells are in the design, and makes an estimate off of that.

From table 5 it can be shown that in synthesis, the radiation hardened library was able to reach clock frequencies higher than that of the non radiation hardened libraries. Power consumption for the radiation hardened library is much higher than what is seen in standard libraries. which is to be expected due to the larger overall size of the cells, along with spatial redundancies techniques being employed. Notice the discrepancy between leakage power for the radiation hardened library between synthesis and PnR. In synthesis, it was calculated to have 15.53 mW of leakage power, while when placed and routed, only 0.0127 mW of leakage power was reported. Since PnR is more accurate when reporting power, the 0.0127 mW of leakage power is considered to be the more accurate metric. This leakage power is considerably lower than what is found in other non-radiation hardened libraries, proving the effectiveness of the RHBD techniques used. In terms of area, the radiation hardened library can be shown to have an increase of 6x the area of it's non-radiation hardened counterparts. This is also to be expected, due to the extra area the RHBD techniques add on.

Figure 14 shows the fully placed and routed design complete with routing. Since routing obscures the standard cells below, figure 15 shows the place and routed design without routing between the standard cells. It can be shown that much of the core area is not utilized. This was done to give ample room for the router to place the cells, since the goal was to obtain power and timing metrics only. If a fabrication run were to be planned in

the future for this specific design, improvements would be made to the overall routing and standard cell placement to better utilize the core area. Something important to note for this PnR run is that Innovus struggled placing the standard cells in an efficient way. On closer inspection, there are many instances of irregular placement of cells leading to even more area being wasted. The most probable cause for this is the added complexity of placing cells with diffusion rings. Modifications to the configuration of the Innovus flow need to be made to better help the tool place cells intelligently. This is also likely the cause of increased power usage.

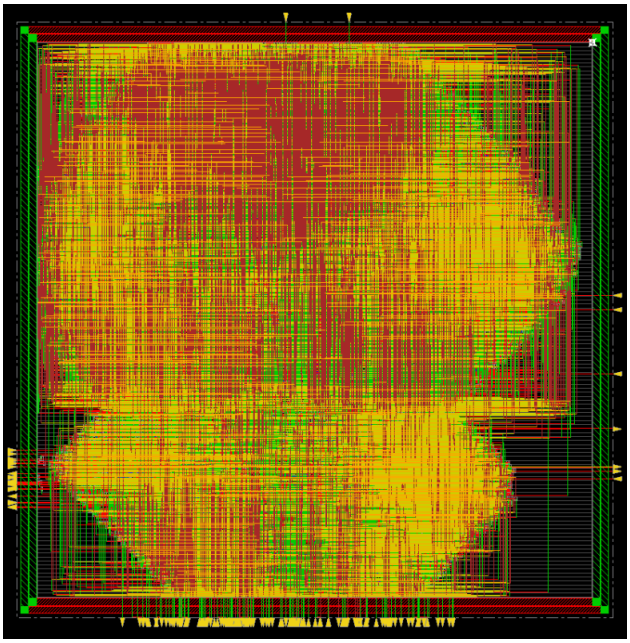


Figure 14: Placed and routed 32-bit RISC-V processor with routing.

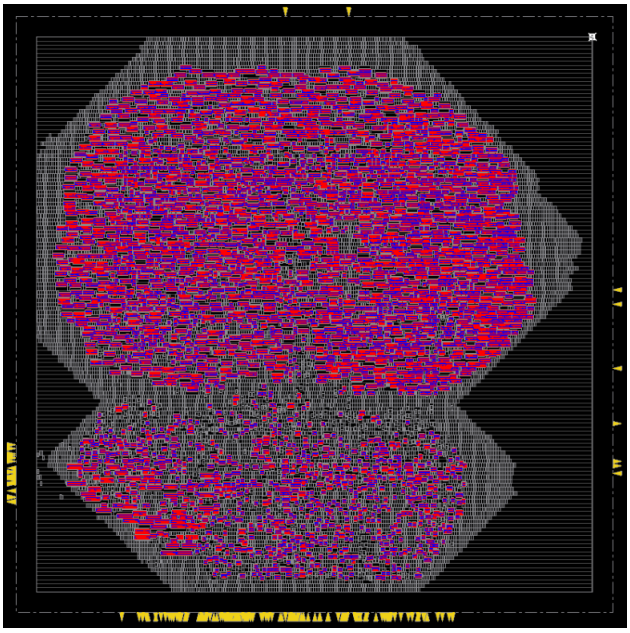


Figure 15: Placed and routed 32-bit RISC-V processor with no routing shown.

Table 5: Cell quantity, area, power, and timing metrics of synthesized RISC-V design using radiation hardened library, with a clock frequency of 500MHz.

RISC-V Synthesis using Radiation Hardened SKY130 Library	
Number of ports	165
Number of nets	11702
Number of cells	11157
Number of combinational cells	10133
Number of sequential cells	1024
Number of BUF/INV	1957
Number of references	16
Combinational area (μm^2)	416,116.020
BUF/INV area (μm^2)	50,679.560
Noncombinational area (μm^2)	182,669.379
Total cell area (μm^2)	598,785.399
Internal power (mW)	59.645
Switching power (mW)	11.960
Leakage power (μW)	13.492
Total power (mW)	71.618
Timing slack (ns)	-0.026

Table 6: Area and maximum clock frequency metrics of synthesis RISC-V design using non radiation hardened SKY130 libraries.

RISC-V Synthesis using Non Radiation Hardened SKY130 Library		
Library	Area (μm^2)	Max Clk Freq. (MHz)
scs8hs	87.560	431.000
scs8ms	87,931	371.000
scs8ls	88,663	321.000
scs8hd	68,490	359.000

Table 7: Results from Innovus PnR on RISC-V design using the Radiation Hardened SKY130 library.

Internal Power (mW)	76.460
Switching Power (mW)	77.560
Leakage Power (mW)	0.011
Total Power (mW)	154.031
Max Clk Freq. (MHz)	130
Instances	11,157
Area (μm^2)	598,784.699

Table 8: Area and max clock frequency obtained from Innovus PNR on RISC-V design using non Radiation Hardened SKY130 library.

Library	Area (μm^2)	Max Clk Freq. (MHz)
scs8hs	102,119	307
scs8ms	102,079	371
scs8ls	108,555	321
scs8hd	83,057	359

CHAPTER VII

CONCLUSION

In conclusion, there is a complete lack of open source options for radiation hardened standard cell libraries, as well as comprehensive EDA flows for SoC and digital ASIC design development. With open source EDA tools reaching comparable functionality of proprietary EDA tools produced by companies like Cadence and Synopsys, the ability for smaller companies and hobbyists who lack the capital to buy the licenses for theses proprietary tools, to create standard cell libraries and ICs using open source tools is now worthwhile, which has been demonstrated in this thesis. The cost of entry into the IC design market is nearing zero, with the only area open source tooling is lacking being the Liberty file generation. Current open source software available for Liberty file generation does not provide the reliability and usability of commercial tools such as Calibre Liberate. But, constant support and development of the software by interested and competent parties is intrinsic to the concept of open source. Also, with the increasing popularity of open source EDA tools, it is only a matter of time until the tools reach reliable results and adequate usability. To further prove the capabilities of open source tools, the radiation hardened standard cell library created here shows the practicality and usability of these tools by designing a more niche standard cell library that utilizes more complex layout techniques. the open source tools were able to keep up and provided an efficient design environment for development.

With the space industry continuing to grow at it's current pace, it's reasonable to assume that more parties will interested in entering into the field, bringing fresh ideas and innovations along with them. This influx of new parties has a major bottleneck that only the

larger and cash rich companies will be able to pass through, due to the cost of licensing the IPs of a proprietary radiation hardened standard cell library. It's been shown that without specialized standard cell libraries designed to mitigate the effects of high radiation environments, ICs cannot survive outside of the earths protective atmosphere. Currently, there are no radiation hardened libraries, aside from the one presented here, that are completely free and open source, capable of large scale integration, and have an associated EDA flow that utilizes completely open source tools, with the exception of one. Even with Liberty file generation being locked behind proprietary tools, the already existing Liberty file present in the repository is available for anyone to use, allowing synthesis and PnR to take place using the current state of the standard cell library.

7.1 Future Work and Collaboration

Further iteration and improvements to the library and EDA flow are planned and currently in development. The current plans are as follows:

- Add large drive strengths of existing cells: It's common to have 6 different drive strengths of a single cell, with 12 to 16 being not uncommon in large commercial libraries. A minimum of 4 drive strengths for each cell is the goal.

- Adding more specialized/niche cells: Cells like comparators, JK Flip Flops, Select DFF, and more complex cells for the purpose of optimizing the synthesis of gate level netlists. The synthesis tool can use a small collection of primitive logic gates to represent almost any combinational/sequential logic, but is inefficient in terms of both timing and area. Custom layouts of the logic gates are smaller and more efficient.

- Additions and improvements to EDA flow: Currently, layouts can be design in magic along with the equivalent schematic capture, and the process to extract the relevant files, perform LVS, generate library LEF, and configure scripts to perform characterization, is completely automated. The only step for a complete SoC design flow that is

currently not included (aside from Liberty file generation) is PnR. This is currently in progress, and will soon include the open source RTL to GDS flow *OpenROAD*.

- Implement better and more optimized radiation hardening technqiues: The science of radiation hardening SoCs is still in its infancy, and new emerging technologies are being developed. The plan is to continue to stay knowledgeable in the field and employ new RHBD techniques as they are discovered.

- Fabricate an SoC using the library: This is a long term goal of actually fabricating a chip using this radiation hardened technology. Skywater, in partnership with Google, allow individuals to send in custom ASIC designs for fabrication runs. Before that, improvements to the library and flow are still needed. The timeline for fabricating a chip is to be determined.

In the spirit of open source, all files, scripts, and documentation are available on GitHub. Suggestions for improvements, additions, and bug reports are welcome, while an open invitation for collaboration stands to anyone who wants to participate.

REFERENCES

[1] *The RISC-V instruction set manual, volume i: User-level isa, document version 2.2*, May 2017.

[2] *Global space economy rose to $447b in 2020, continuing five-year growth*, Jul 2021.

[3] Jacob A. Abraham, Edward S. Davidson, and Janak H. Patel, *Memory system design for tolerating single event upsets*, IEEE Transactions on Nuclear Science **30** (1983), no. 6, 4339–4344.

[4] P. Balasubramanian and Nikos E. Mastorakis, *Power, delay and area comparisons of majority voters relevant to TMR architectures*, CoRR **abs/1603.07964** (2016).

[5] Anton O. Balbekov, Maxim S. Gorbunov, and Artur M. Galimov, *Design-stage hardening of 65-nm cmos standard cells against multiple events*, IEEE Transactions on Nuclear Science **68** (2021), no. 8, 1712–1718.

[6] H. J. Barnaby, *Total-ionizing-dose effects in modern cmos technologies*, IEEE Transactions on Nuclear Science **53** (2006), no. 6, 3103–3121.

[7] Varvara Bezhenova and Alicja Michalowska-Forsyth, *Aspect ratio of radiation-hardened mos transistors: Modelling of the equivalent channel dimensions of integrated mos transistors in radiation-hardened enclosed layout*, e&i Elektrotechnik und Informationstechnik **135** (2018).

[8] R. Timothy Edwards, *Magic vlsi layout tool*, 2022, Last accessed April 2022. http://opencircuitdesign.com/magic/.

[9] EETimes, *Cadence rolls custom-ic tools into one platform*, Sep 2003.

[10] D. Harris, S. Harris, and D. Money, *Digital design and computer architecture: Risc-v edition*, Morgan Kaufmann Publishers, 2022.

[11] F.L. Kastensmidt, L. Sterpone, L. Carro, and M.S. Reorda, *On the optimal design of triple modular redundancy logic for sram-based fpgas*, Design, Automation and Test in Europe, 2005, pp. 1290–1295 Vol. 2.

[12] Kostas Kloukinas, Federico Faccio, Alessandro Marchioro, and Paulo Moreira, *Development of a radiation tolerant 2.0 v standard cell library using a commercial deep submicron cmos technology for the lhc experiments*, (1998).

[13] Minwoong Lee, Seongik Cho, Namho Lee, and Jongyeol Kim, *New radiation-hardened design of a cmos instrumentation amplifier and its tolerant characteristic analysis*, Electronics **9** (2020), no. 3.

[14] Dave Y.-W. Lin and Charles H.-P. Wen, *Dad-ff: Hardening designs by delay-adjustable d-flip-flop for soft-error-rate reduction*, IEEE Transactions on Very Large Scale Integration (VLSI) Systems **28** (2020), no. 4, 1030–1042.

[15] Jia Liu, Weidong Yang, Yao Li, Ruitao Zhang, Jing Yang, Xiaogang Feng, Yuxin Wang, Dongbing Fu, Guangbing Chen, and Ruzhang Li, *A radiation-hardened standard cell library for commercial 0.18 μm cmos technology*, 2014 12th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), 2014, pp. 1–3.

[16] He Lv, Hongwei Zhang, Bo Mei, Yi Sun, and Qingkui Yu, *Simulation and calculation method of total ionizing dose radiation resistance of aerospace components based on in-orbit data*, 2019 3rd International Conference on Circuits, System and Simulation (ICCSS) (2019), 146–150.

[17] The OpenROAD Project, *The OpenROAD Project*, 2022, Last accessed April 2022. https://github.com/The-OpenROAD-Project.

[18] Leonard R Rockett and Daniel J Kouba, *Radiation hardened 150nm standard cell asic design library for space applications*, 2008 IEEE Aerospace Conference, 2008, pp. 1–8.

[19] Stefan Schippers, *XSCHEM : Schematic captureand netlisting eda tool*, 2022, Last accessed April 2022. https://xschem.sourceforge.io/stefan/index.html.

[20] Swati Shah, *Study of soi annular mosfet*, Master's thesis, 2009.

[21] Rakesh Trivedi and Usha S Mehta, *A survey of radiation hardening by design (rhbd) techniques for electronic systems for space application*, International Journal of Electronics and Communication Engineering & Technology (IJECET) **7** (2016), no. 1, 75.

[22] Holger Vogt, *Ngspice, the open source spice circuit simulator - intro*, 2022, Last accessed April 2022.http://ngspice.sourceforge.net/.

[23] Laung-Terng Wang, Yao-Wen Chang, and Kwang-Ting Cheng, *Chapter 12: Global and detailed routing*, p. 688–688, Morgan Kaufmann/Elsevier, 2009.

[24] Neil Weste and David Harris, *Cmos vlsi design: A circuits and systems perspective*, 4th ed., Addison-Wesley Publishing Company, USA, 2010.

[25] Austin James Womac, *The characterization of a cmos radiation hardened-by-design circuit technique.*

[26] YosysHQ, *Yosyshq/yosys: Yosys open synthesis suite*, 2022, Last accessed April 2022. https://github.com/YosysHQ/yosys.

VITA

Ryan Ridley

Candidate for the Degree of

Master of Science

Thesis: OPEN SOURCE RADIATION HARDENED STANDARD CELL LIBRARY AND EDA FLOW

Major Field: Electrical Engineering

Biographical:

Education:

Completed the requirements for the Master of Science degree with a major in Electrical Engineering at Oklahoma State University in May, 2022.

Completed the requirements for the Bachelor of Science in Electrical Engineering at Oklahoma State University, Stillwater, Oklahoma in May, 2020.

Completed the requirements for the Bachelor of Science in Computer Engineering at Oklahoma State University, Stillwater, Oklahoma in May, 2020.