University of Cincinnati

Date: 3/28/2011

I, Ajaay Ravi, hereby submit this original work as part of the requirements for the degree of Master of Science in Computer Engineering.

It is entitled:
Run-Time Active Leakage Control Mechanism based on a Light Threshold Voltage Hopping Technique (LITHE).

Student's name: Ajaay Ravi

This work and its defense approved by:

Committee chair: Ranganadha Vemuri, PhD
Committee member: Wen Ben Jone, PhD
Committee member: Carla Purdy, C, PhD
Run-Time Active Leakage Control Mechanism based on a Light Threshold Voltage Hopping Technique (LITHE)

A Thesis submitted to the
Graduate School
of the University of Cincinnati
in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

In the Department of
Electrical and Computer Engineering
of the College of Engineering and Applied Sciences

Spring 2011

by

AJAAY RAVI

B.Tech (Electronics and Communication Engineering)
SRM University, Chennai, India.

June 2007

Thesis Co-Advisors: Dr. Ranga R Vemuri, Ph.D and Dr. Wen-Ben Jone, Ph.D
ABSTRACT

Leakage aware designs are an indispensable part of the design and manufacturing process in today’s deep sub-micron technologies. Technology scaling continues to be a constant factor in CMOS designs, with the feature sizes of devices manufactured being scaled down below 28nm.

Starting from the 45nm technology, it has been shown that the leakage power consumption in a circuit catches up with the dynamic power consumption and continuing this trend, it has been projected that for future technologies, the leakage power consumption will even dominate the dynamic power consumption. This increasing leakage power consumption in the deep sub-micron CMOS technologies has manifested the need for more aggressive control mechanisms. The leakage control mechanisms in use today can be widely categorized into 2 categories namely, Design-Time control mechanisms and Run-Time control mechanisms. As the names suggest, Design-Time control mechanisms are incorporated into the circuit during the design phase and are not capable of dynamic control. This limits the extent of effectiveness in the leakage power reduction capability of this technique. Alternatively, Run-Time leakage control mechanisms monitor the circuit and dynamically flip it into a low power mode of working, depending upon the circuit’s workload. These techniques yield a significant power saving and a significant amount of research in low power designs today, is directed towards this technique.
The research presented by means of this thesis, is based upon a prominent run-time control mechanism known as Reverse Body Biasing. The workload of any circuit can be defined under 2 broad classifications, viz. Active mode and Standby mode. There are many robust leakage power reducing techniques that are in use today to tackle the issue during the standby mode of a circuit. It is the active mode that presents an interesting view to the problem as a whole. Scrutinizing the workload of a circuit in its active mode of working showcases that there are copious opportunities of slackness that a designer can take advantage of and utilize to construct a better leakage aware design. This is classified as Run-Time Active leakage control (RALC).

Key issues to using RALC are the optimum granularity level on which it can be applied and deciding on an efficient leakage reduction mechanism to be used with it. These issues are addressed by the technique presented as the central idea of this research, known as LITHE (Light Threshold Voltage (V\text{TH}) Hopping Technique). The idea behind LITHE is based off of a popular technique known as Threshold Voltage hopping and this is achieved by means of Adaptive Substrate Biasing. Together, this forms the core of this research. This research aims to convincingly address all the issues of the RALC as a viable solution to designing robust leakage aware designs. Aggressive exploitation of idleness during the active mode working of a circuit, fused together with the idea of LITHE, is the solution proposed towards tackling leakage power issues in deep sub-micron technologies, by means of this research. Extensive experimentation has been performed on benchmark circuits to support and verify its accuracy.
To my dearest parents,

for providing constant support and limitless encouragement throughout my education.
ACKNOWLEDGEMENTS

First and foremost, I would like to thank the almighty for his blessings and never ending guidance in all my endeavors.

I profoundly thank my parents, without whom it would not have been possible for me to have reached where I am today. They showed exceptional levels of patience and had constant words of encouragement throughout my time as a graduate student. I am forever grateful to their unconditional love and limitless efforts. My brother Akshay, has been a moral source of strength during testing times and has always helped keep me in my best spirits. I thank him for being there when it mattered most.

I am very thankful to my advisor Dr. Ranga Vemuri, for all his guidance and valuable support. I thank him for the opportunity to work in his research group and his assistance with my research. I am also grateful to him for making me his Teaching Assistant and the confidence he showed in me helped me gain more knowledge and invaluable experience. It helped me win the coveted excellence in teaching award for graduate students, which serves as a highlight to my achievements at the University of Cincinnati.

I am also thankful to Dr. Wen-Ben Jone for his support as my co-advisor and his wonderful recommendations. He was a constant source of encouragement and I would also like to thank him for his valuable time, which he shared with me quite a lot, during my time as his TA.

I also thank Dr. Carla Purdy for taking time to serve on my thesis defense committee.

The time spent at the Digital Design Environments lab would not have been half as wonderful as what it was without the presence of my good friends and fellow lab-mates, Vijay, Shubhankar, Angan, Almitra, Surya, Pritesh, Shikhanshu, Romana, Aditi, Annie, Mike, Hao, Arun, Lakshmi Narasimhan, Jon and Natwar. Thanks to them for all the good times and also lending me their ears and a helping hand as well, whenever I needed them. A very special word of thanks to my dear friends Annie and Arun for having endured the pain of reviewing my thesis report and also to Hao without whose guidance this research would not have been possible.
Table of Contents

Acknowledgements.......................................................................................................................... vi

List of Figures ................................................................................................................................... xi

List of Tables .................................................................................................................................... xiv

1. INTRODUCTION

1.1 Introduction ................................................................................................................................... 1

1.2 Technology Scaling and its impact on Power .................................................................................. 3

1.3 Need for low power designs ......................................................................................................... 5

1.4 Leakage Power ............................................................................................................................... 7

1.4.1 Sub-Threshold Conduction ..................................................................................................... 9

1.4.2 Gate Leakage ............................................................................................................................ 9

1.4.3 Reverse-Biased Junction BTBT Leakage ................................................................................ 10

1.5 Leakage Mitigation Techniques .................................................................................................... 11
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5.1</td>
<td>Lowering the Operating Voltage</td>
<td>11</td>
</tr>
<tr>
<td>1.5.2</td>
<td>Power Gating</td>
<td>12</td>
</tr>
<tr>
<td>1.5.3</td>
<td>Multi – $V_T$ Transistors</td>
<td>13</td>
</tr>
<tr>
<td>1.6</td>
<td>Aim of this Research</td>
<td>14</td>
</tr>
<tr>
<td>1.7</td>
<td>Substrate Biasing</td>
<td>15</td>
</tr>
<tr>
<td>1.8</td>
<td>Dynamic Substrate Biasing</td>
<td>18</td>
</tr>
<tr>
<td>1.8.1</td>
<td>Mathematical Analysis</td>
<td>21</td>
</tr>
<tr>
<td>1.8.2</td>
<td>Advantages of Dynamic Substrate Biasing</td>
<td>23</td>
</tr>
<tr>
<td>1.9</td>
<td>Thesis Overview</td>
<td>25</td>
</tr>
<tr>
<td>2.1</td>
<td>Introduction</td>
<td>30</td>
</tr>
<tr>
<td>2.2</td>
<td>Run-Time Active Leakage Control (RALC)</td>
<td>31</td>
</tr>
<tr>
<td>2.3</td>
<td>Energy Overhead Compensation</td>
<td>35</td>
</tr>
<tr>
<td>2.4</td>
<td>Analytical Observations</td>
<td>39</td>
</tr>
<tr>
<td>2.5</td>
<td>Conclusions</td>
<td>40</td>
</tr>
</tbody>
</table>
3. IMPLEMENTATION AND ANALYSIS OF THE LITHE TECHNIQUE

3.1 Introduction ........................................................................................................ 41

3.2 LITHE .................................................................................................................. 42

3.2.1 Optimum Granularity ....................................................................................... 44

3.2.2 Light Threshold Hopping .................................................................................. 47

3.3 Automating the LITHE Mechanism ................................................................. 51

3.3.1 LITHE Selection Algorithm .............................................................................. 52

3.4 Analytical Observations .................................................................................... 54

3.4.1 Results analysis for a D-Flip Flop .................................................................. 54

3.4.2 Benchmark Results .......................................................................................... 57

3.5 LITHE on Dynamic CMOS designs ............................................................... 57

3.6 Conclusions ...................................................................................................... 62

4. IMPROVEMENTS AND AUTOMATION OF THE LITHE CONTROL

4.1 Introduction ....................................................................................................... 63

4.2 Improvements to the LITHE control ............................................................... 64

4.2.1 Analytical Observations ................................................................................ 68
4.3 Automating the LITHE control ................................................................. 70

4.3.1 Analytical Observations......................................................................... 72

4.4 Conclusions .............................................................................................. 74

5. USING THE LITHE ON RTL MODULES

5.1 Introduction ............................................................................................. 75

5.2 Drawback to the Built-In LITHE Control .................................................. 76

5.2.1 Analytical Observations ...................................................................... 77

5.3 A Heuristic Approach to automating the LITHE ...................................... 80

5.3.1 Heuristic LITHE signal generation ....................................................... 82

5.3.2 Analytical Observations ..................................................................... 83

5.5 Conclusion ............................................................................................... 84

6. CONCLUSION AND FUTURE ENHANCEMENTS

6.1 Future Enhancements ............................................................................... 85

6.1.1 Fine Tuning ......................................................................................... 86

6.1.2 Automatic Place and Route ................................................................. 86
6.1.3 Need for a Predictor Circuit ................................................................. 87

6.2 Summary of Research Accomplishments.................................................. 88

APPENDIX A - EXPERIMENTAL SETUP

A.1 Using Synopsys HSPICE ............................................................................ 91

A.2 Using Synopsys NANOSIM ....................................................................... 93

BIBLIOGRAPHY  96

List of Figures

Fig. 1.1 – Impact of Temperature on Leakage ...................................................... 4

Fig. 1.2 – Factors Contributing to Power Consumption with developing technology sizes ... 7

Fig. 1.3 – Contribution of Leakage Power to Total Power Consumption ................... 8

Fig. 1.4 – Major Leakage Current Components in a MOS Transistor ....................... 8

Fig. 1.5 – Lowering the Operating Voltage (V_{DD}) .......................................... 11

Fig. 1.6 – Power Gating using Sleep Transistors ................................................. 12
Fig. 1.7 – Using Multi-$V_T$ Cells in Design .................................................................13

Fig. 1.8 – Exponential Characteristic of Sub-threshold Conduction (Log Scale) ......................16

Fig. 1.9 – Conventional NMOS transistor vs. Variable Threshold NMOS transistor ................17

Fig. 1.10 – Dynamic Substrate Bias .....................................................................................18

Fig. 1.11 – Dynamic Substrate Bias for $V_{TH}$ Control ........................................................20

Fig. 1.12 – Threshold Voltage Variation for Varying Substrate Biasing in a NMOS transistor.21

Fig. 1.13 – Leakage Saving Comparison ..............................................................................24

Fig. 2.1 – Run-Time Active Leakage Control (RALC) ..........................................................32

Fig. 2.2 – Working of the RALC ........................................................................................33

Fig. 2.3 – PMOS Substrate Biasing Model ..........................................................................36

Fig. 3.1 – A Light $V_{TH}$ Hopping Leakage Control Mechanism .......................................43

Fig. 3.2 – A 16 chain inverter with conventional substrate bias voltages ..............................47

Fig. 3.3 – A Lightly biased 16 chain inverter .....................................................................48

Fig. 3.4 – Optimal Bias Voltage Selection .........................................................................50

Fig. 3.5 – LITHE CAD process ..........................................................................................51

Fig. 3.6 – Gate Level Implementation of a D-Flip Flop .......................................................55

Fig. 3.7 – Layout of a LITHE enabled positive edge D-Flip Flop standard cell ..................55
Fig. 3.8 – Substrate Bias Voltages $V_P$ and $V_N$ .................................................................56

Fig. 3.9 – Leakage Current in the D-Flip Flop ($\mu$A) ...............................................................56

Fig. 3.10 – A glitch free D-CMOS D-Flip Flop [21] .................................................................59

Fig. 3.11 – Layout of Domino logic D-CMOS D-Flip Flop .......................................................59

Fig. 3.12 – Layout of a D-CMOS North-East route checker (Single bit slice) .........................61

Fig. 4.1 – A Light $V_{TH}$ Hopping Leakage Control Mechanism .......................................65

Fig. 4.2a – Leakage savings comparison between NMOS only and traditional LITHE control..69

Fig. 4.2b – reduction comparison between the NMOS only and traditional LITHE control……69

Fig. 4.3 – Automated LITHE Control .....................................................................................70

Fig. 4.4 – A fully automated LITHE enabled Full Adder Circuit ...........................................72

Fig. 4.5 – 8-bit Linear Carry Select Adder ..............................................................................73

Fig. 5.1 – Gate Level Schematic of the NERC – Single Bit Slice ...........................................77

Fig. 5.2 – Layout of a single bit slice of the NERC .................................................................78

Fig. 5.3 – Top level architecture of the 8-bit RISC processor .................................................81

Fig. 6.1 – LITHE enabled cell locations during Place and Route ...........................................86

Fig. 6.2 – LITHE with a Predictor .........................................................................................87
List of Tables

Table 2.1 – Computed EBP for ISCAS-85/89 Benchmark Circuits .................................39

Table 3.1 – Conventional and Light Biasing comparison on a 16 chain Inverter ...............49

Table 3.2 – Leakage Savings on a standalone D-Flip Flop ........................................55

Table 3.3 – 32nm Technology LITHE simulations ......................................................57

Table 3.4 – Static CMOS LITHE vs D-CMOS LITHE implementations .........................60

Table 4.1 – Leakage saving and area overhead using the NMOS only control circuitry ......68

Table 4.2 – Leakage saving and area overhead using conventional NMOS & PMOS control...68

Table 4.3 – Results of Automating the LITHE on the test benchmarks ..........................73

Table 5.1 – Comparison of NERC results based on the primitive and heuristic approach.....79

Table 5.2 – 8-Bit RISC microprocessor benchmark results ............................................83
CHAPTER 1

1.1 Introduction

Integrated Circuits (ICs) were first designed in the late 1940s and have been developing rapidly, ever since. According to reports, a German engineer named Werner Jacobi, first applied for a patent for his semiconductor amplifier, in as early as 1949. This was one of the most basic designs which had 5 transistors arranged on a common substrate, resembling a 2-stage amplifier [1]. From this point onwards, technology advanced rapidly and the semiconductor industry kept expanding its horizons by exponentially increasing the number of transistors on a chip, while simultaneously decreasing device sizes from generation to generation.
The first digital integrated circuits contained only tens of transistors and this generation was referred to as Small Scale Integration (SSI). Technology developed further in the late 1960s into the next generation, known as Medium Scale Integration (MSI). ICs that belonged to this generation contained hundreds of transistors in each fabricated chip. More complex systems were started to be designed and driven by other advantages too, the mid-1970s saw the introduction of Large Scale Integration (LSI). This generation had tens of thousands of transistors on each fabricated chip.

LSI is seen as the starting point of today’s complex systems as the first 1Kilo Byte (KB) Random Access Memory (RAM), early calculator chips, first and second generation microprocessors were all manufactured at this time. The beginning of the 1980s saw technology develop into the next generation, commonly referred to as Very Large Scale Integration (VLSI). Starting from hundreds of thousands of transistors on every chip, today devices with more than tens of billions of transistors on a chip are being manufactured.

Technology has further advanced into domains such as System-On-Chip (SOC) and 3-Dimensional Integrated Circuits (3-D ICs), having more than 2 layers of active components integrated both in the horizontal and vertical orientations, into a single circuit [1]. Starting from SSIs, all the way down to the SOCs of today, technology scaling has been one of the most constant factors in the technological development.
1.2 Technology Scaling and its impact on Power

Making much more with the very limited resources available, is a question that the industry has continually answered by reducing the minimum feature size of the transistors while simultaneously balancing the increase in circuit integration capacities. Manufacturers keep on moving to using smaller design rules and smaller device sizes.

The rapid development on the technology scaling front has been considerably slowed down of late in technologies with minimum feature size, lesser than 90nm. Reasons such as parasitic dependencies, heat dissipation due to the tighter circuit integrations, high amounts of leakage current and parameter uncertainties of the smaller devices are the major setbacks to the rapid development of lower feature sizes. Especially in the deep sub-micron technologies of today, leakage power has been rapidly increasing and has manifested itself into one of the most important design aspects to be considered before fabricating any chip.

Roughly 40 years ago, Dr. Gordon E. Moore, the co-founder of Intel® Corporation stated that, “The number of transistors that can be placed inexpensively on an integrated circuit, doubles with every new manufacturing generation (which takes place approximately every 18 months).” This is widely known as Moore’s Law.

The development trend of ICs has very closely followed Moore’s law since its inception and is not expected to stop until at least a few years from now. In order to keep up with Moore’s Law, research is continually performed on tackling the previously stated factors that hamper the speed of technology scaling. Fault tolerant computing is one of the recent design paradigms that
exhibits the potential to alleviate the questions put forth by the parameter uncertainties of the smaller devices that belong to the newer generations.

The heat dissipation and leakage current issues are considered to be two sides of the same coin that go hand in hand in influencing each other.

This is a problem which boasts of extensive research backing but still throws new questions with every step, thereby increasing its horizon further with every new generation.

![Temperature impact on Leakage current](image)

**Fig. 1.1 –** Temperature impact on Leakage current [3]

It can be seen from the above Fig. 1.1, that temperature influences leakage, which in turn influences temperature.
This is a vicious cycle that designers need to be aware of and break by means of robust low power designs. For an increase in temperature from 30°C to 90°C, the leakage current increases 10-fold.

1.3 Need for low power designs

Design for low power consumption and lower energy dissipation is being carried out since the start of this decade. This alone has shown to be insufficient as the heat dissipation poses problems on a new dimension to the circuit run-time temperatures and integrity. Excessive hot spots and varying increase in circuit runtime temperatures can even inflict physical damages to the IC. Due to the limits set by heat dissipation issues and concerns such as Electro-migration, it has become improbable to increase the operation frequency of the circuit much further. Given this reason, development switched tracks to multi-cores bringing in more parallelism thereby improving circuit performance with limited clock frequency availability.

Low power design has become an indispensable part of the normal design flow. Until about the 0.35 micron technology, the major source of power consumption was the dynamic power. Dynamic power is the power that is consumed during the switching of the transistors in the circuit and is represented as:

\[
P_{\text{dyn}} = C_L \cdot V_{\text{DD}}^2 \cdot P_{0,1} \cdot f \]

\[
\text{................................................. (1.1)}
\]
The product $C_L \cdot P_{0,4}$ can be termed as the Effective Capacitance or $C_{\text{EFF}}$. The early low power designs were thus obtained by reducing either the effective capacitance ($C_{\text{EFF}}$) or the operating voltage ($V_{\text{DD}}$) or the frequency of operation ($f$) of the chip.

But as technology scaled down further, leakage (static) power and direct-path power components also added to the total power consumption in a circuit. Leakage power is the power that is consumed even when the circuit is not switching i.e. it is in a steady state.

This power is represented as:

$$P_{\text{leak}} = I_{\text{leak}} \cdot V_{\text{DD}} \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldot
The complexity of the power problem faced by designers today has considerably increased. Both temperature as well as process variations also need to be accounted for when performing power estimations. The addition of temperature variations has made the issue even more intricate due to its inter-dependencies with leakage current.

1.4 Leakage Power

Leakage power is the power that is consumed in a circuit, even in the absence of any switching activity. It is the steady state or static power consumption, due to the leakage current that flows between the power rails in the circuit. According to the ITRS reports, in technologies as recent as the 45nm technology, the total leakage power accounts to approximately 55% of the entire power consumption in a circuit [26].
Fig. 1.3 – Contribution of Leakage Power to Total Power Consumption [26]

The graphical data represented in the above Fig. 1.3, further implies that the dynamic power and leakage power components are roughly equal and is only going to worsen in subsequent technologies. The three major contributors to leakage power in a circuit are Sub-Threshold conduction, Gate Leakage and Junction Band-To-Band-Tunneling (BTBT) Leakage.

Fig. 1.4 – Major Leakage Current Components in a MOS Transistor
1.4.1 Sub-Threshold Conduction

The drain current $I_D$, does not drop down to zero as expected when the value of gate-to-source voltage $V_{GS} = V_T$ (Threshold Voltage). The transistor conducts partially, even for voltages that are below the threshold value and this is termed as Sub-Threshold conduction.

The current, does not drop to zero for a voltage $V_{GS} < V_T$, instead decays in an exponential fashion. This conduction occurs before the onset of strong inversion and is also known as weak inversion conduction. The NPN/PNP junctions actually form a parasitic Bi-Polar Junction transistor (BJT) in the absence of a channel to conduct.

The current in this region can be expressed as:

$$I_D = I_S e^{\frac{V_{GS}}{nkT/q}} \left(1 - e^{\frac{-V_{DS}}{kT/q}}\right)(1 + \lambda V_{DS})$$  \hspace{1cm}  (1.5)

Where, $I_S$ and $n$ are empirical parameters and typically, $n \geq 1$. [2]

1.4.2 Gate Leakage

While using thin Gate oxides such as SiO$_2$ to design fast circuits, electrons tunnel into the channel due to presence of high electric field intensity. The direct tunneling of electrons causes a large gate leakage current to flow. This phenomenon is more common in the deep sub-micron technologies used today [4].
The current density due to the gate oxide tunneling, can be expressed as:

\[ J_{DT} = A_g \left( \frac{V_{\alpha}}{T_{ox}} \right)^2 e^{-B_g \left( 1 - \left( \frac{1}{\sqrt{1 - \frac{V_{\alpha}}{\phi_{\alpha}}} \right)^3 \right)} \]

where, \( A_g \) and \( B_g \) are physical parameters. For the purposes of our research, presented in this thesis we neglect the gate leakage as it is very insignificant when high-k materials are employed to fabricate the gate [5].

### 1.4.3 Reverse-Biased Junction BTBT Leakage

The reverse-biased leakage current is caused mainly due to the thermally generated charge carriers. With a rise in junction temperature, an exponential increase of leakage current is observed [2]. The tunneling of charge carriers into the substrate from the drain, results in a leakage current, which is directly proportional to the operating voltage and the junction doping. The current through the junction in a MOS transistor can be expressed as:

\[ I_{JN} = W A \frac{\xi_{JN}}{E_g^{\frac{3}{2}}} V_{db} e^{\exp\left( \frac{BE_g^{\frac{3}{2}}}{\xi_{JN}} \right)} \]

where, \( E_g \) is the energy band gap and \( A \) and \( B \) are physical parameters. It is evident from the previous equation that increase in the Drain-to-Base voltage \( V_{DB} \), would help in decreasing the BTBT leakage current [4].
1.5 Leakage Mitigation Techniques

There are many leakage mitigation techniques that are used in today’s deep sub-micron technology designs. Some of the most common methods being, Cell sizing, Stacking transistors, lowering the Operating Voltage (\(V_{\text{DD}}\)) of the circuit, Power Gating using Sleep Transistors, using Multi-\(V_T\) transistors and Substrate Biasing. While techniques such as changing the operating voltage, substrate biasing and power gating can be qualified as run time leakage control mechanisms, the others are primarily design time techniques. Certain design time techniques can also be used along with the run time leakage control techniques in order to achieve a higher amount of leakage current reduction. All the leakage mitigation techniques discussed in this chapter aim at reducing the sub-threshold conduction and reverse-biased junction leakage components.

1.5.1 Lowering the Operating Voltage

![Fig. 1.5 – Lowering the Operating Voltage (\(V_{\text{DD}}\)) [6]](image-url)
As can be seen from the above Fig. 1.5, the operating voltage of the circuit under consideration is reduced to a value lesser than $V_{DD}$. The reduced voltage on the power rail charges the capacitances in the circuit to a lesser value than when it was $V_{DD}$. Since the total charge stored in the capacitors is much lesser than normal, amount of leakage is also proportionally smaller. The drawback in utilizing this technique is that this increases the delay of the gate/block due to the requirement of more time to charge the capacitance, given the smaller voltage. This technique can be employed on gates/circuit blocks that are not a part of the critical path.

Summarizing, wherever a timing tradeoff can be accommodated in a circuit, the option of lowering the operating voltage is employed, thereby saving on both the dynamic as well as leakage power consumptions.

1.5.2 Power Gating

![Power Gating using Sleep Transistors](image)

Fig. 1.6 – Power Gating using Sleep Transistors [6]
Low leakage PMOS and NMOS sleep transistors are used as header and footer sleep transistors, to achieve power gating. The insertion of these sleep transistors actually helps to split the entire power network into 2 main sub-divisions. One is the permanent power network that is connected to the supply power rails and the other is the temporary virtual power network that is used to drive the cells/blocks. The virtual power network can be turned on or off depending upon circuit idleness/workload and helps achieve leakage power reduction.

Designers can choose between using either a header or a footer (advantageous as area is lesser) sleep transistor or using both, but care has to be taken to ensure that the size of the sleep transistors are large enough to handle the high switching current involved. This technique of temporarily shutting down circuit blocks, by means of power gating has proven to be a very effective method of leakage reduction [7].

1.5.3 Multi – V<sub>T</sub> Transistors

Fig. 1.7 – Using Multi-V<sub>T</sub> Cells in Design [6]
The deep sub-micron technology designs today, employ the use of Multi-$V_T$ standard cell libraries. The standard cell libraries consist mainly of 4 types of cells viz. Standard $V_T$ (SV$_T$ Cells), High $V_T$ (HV$_T$ Cells), Low VT (LV$_T$ Cells) and Ultra Low VT (UV$_T$ Cells).

All the transistors in the HV$_T$ cells are designed with a higher threshold voltage and are aimed at reducing the sub-threshold leakage current. The transistors in the LV$_T$ cells are designed with a lower threshold, to achieve high speeds. If the optimization target to be achieved is low power, then the designs are synthesized using the HV$_T$ cells which help to achieve least leakage power and after this, the cells in the critical path are replaced with LV$_T$ cells, in order to satisfy the timing closure requirements. Similarly, if the target was performance as opposed to lower power consumption, then the designs are first synthesized using LV$_T$ cells to meet the timing requirements and then HV$_T$ cells are used wherever possible to reduce leakage power, confirming to timing closure [8].

1.6 Aim of this Research

The research presented by means of this thesis is aimed at reducing the leakage current and designing a robust control mechanism to achieve low power design. The term ‘deep sub-micron technology’, used extensively in this thesis refers mainly to technologies that are smaller than 65nm. All experimentation was performed for the 45 and 32nm technologies.

The research in low power electronics today can be classified widely into 2 categories, viz. Design Time leakage control and Run Time leakage control.
As the name design time leakage control suggests, the circuit is modified during design time and cannot be changed once confirmed. The scope of total leakage power minimized is limited. Run time leakage control looks to adaptively switch the circuit into a low leakage mode or continue normally, depending upon the workload to the circuit in consideration. There is a lot of scope for extensive leakage power reduction when run time leakage control techniques are employed.

The study presented in this thesis focuses on one type of run-time leakage control, achieved by utilizing the substrate biasing approach, thereby aiming to reduce both the sub-threshold conduction and junction leakage. It is also simultaneously tuned in the presence of temperature variations.

1.7 Substrate Biasing

The MOS transistor, experiences a drain to source current, even when the conducting channel is not yet fully formed i.e. even when the gate to source voltage, $V_{GS}$ is less than the threshold voltage $V_T$. As the value of $V_T$ reduces, the amount of leakage current increases. To counter this negative effect of increasing leakage current for threshold voltages closer to zero, the $V_T$ of the transistors should ideally be high.

But, the tradeoff here is that devices with a higher $V_T$ are slower compared to those with low $V_T$ and hence this is not a permanent solution while designing circuits for high performance.
The choice of an apt threshold voltage for the MOS devices, presents an interesting picture, where the designer is required to decide on a tradeoff between performance and low leakage power. The exponential characteristic of Sub-threshold conduction is shown by the following Fig.

![Fig. 1.8 - Exponential Characteristic of Sub-threshold Conduction (Log Scale) [2]](image)

It can be seen from the above Fig. 1.8 that the drain current $I_D$ does not drop down to zero for value of $V_{GS} < V_T$ as expected. Instead, it decays in an exponential fashion. Therefore, in order to prevent this sub-threshold leakage current, the threshold voltage value needs to be increased.

$V_T$ can be represented by the following equation.

$$V_T = V_{TO} + \gamma \left( \sqrt{|-2\varphi_F + V_{SB}|} - \sqrt{|2\varphi_F|} \right) \ldots \ldots \ldots \ldots \ldots (1.8)$$
It can be observed from the above equation that the value of threshold voltage $V_T$ can be changed by changing the Substrate Voltage, $V_{SB}$. The surface potential required for strong inversion to occur and a conducting channel to be formed is equal to $|2\phi_F|$, where $\phi_F$ is the Fermi Potential.

But this fails to hold when a substrate bias voltage ($V_{SB}$) is applied between the substrate and source terminals of the MOS transistor. The surface potential required to cause strong inversion and form a channel is now increased to $|-2\phi_F + V_{SB}|$. In equation 1.8 above, $V_{TO}$ is an empirical parameter that depicts the threshold voltage of the device under zero substrate bias conditions and the parameter $\gamma$ is the Body effect co-efficient that expresses the impact of the changes due to the substrate bias voltage.

Fig. 1.9 – Conventional NMOS transistor vs. Variable Threshold NMOS transistor [10]

It can be seen from the above Fig. 1.9 that the body terminal of the Variable Threshold MOS transistor is connected to a Substrate Bias Voltage other than ground, as it would have been if designed as per the conventional norms.
Summarizing, the amount of sub-threshold leakage current can be reduced by increasing the threshold voltage $V_T$ of the MOS transistor, by varying the substrate bias voltage $V_{BS}$, which in turn also helps reduce the reverse-biased junction leakage. Thus, Substrate biasing presents itself as a very promising option to use for low power designs.

### 1.8 Dynamic Substrate Biasing

![Diagram of Dynamic Substrate Biasing](image)

Fig. 1.10 – Dynamic Substrate Bias [10]

The research in this thesis focuses on employing Substrate Biasing as a Run-Time leakage power reduction technique. This is achieved by dynamically varying the Source to Substrate Voltage, $V_{SB}$ of the devices based upon the circuit workload and hence this approach is termed as Adaptive or Dynamic Substrate Biasing.
It can be observed from the above Fig. 1.10 [10], that the substrate voltages of the Variable Threshold MOS transistors are connected to power rails $V_{DD}$ and GND of the circuit during the active mode (ON period) of the circuit and to a different set of substrate biasing voltages $V_P$ and $V_N$ when the circuit is in the standby mode or idle.

This helps reduce the leakage when the circuit is idle or is in standby by increasing the threshold voltage $V_T$ of the devices and also does not hamper the performance of the circuit during the active period, as the bias voltages are switched back to the normal zero bias condition. Since this is done dynamically during the active mode working of the circuit, it is one of the most viable methods of runtime active leakage control.

Conceptually, dynamic substrate biasing is very similar to dynamic voltage scaling. The Energy Overhead that is encountered in this method is the energy penalty due to the charging and discharging of the substrate capacitances. Since reaction speed is also a concern while designing, generally the bias voltages are not chosen to be very high.

This limits the amount of effective leakage reduction that is obtained as a result of this method. Thus, only 10x the reduction in leakage power is obtained by employing this technique.
As seen from the above Fig. 1.11, the threshold voltage $V_{TH}$ of the transistors can be dynamically controlled for efficient low power design. It has been proven by experimentation that compared to the normal substrate biasing where the body terminal of the NMOS transistors are connected to ground and that of the PMOS transistors are connected to the power supply $V_{DD}$, when the body voltages are biased to a value of -0.5V for the NMOS and $V_{DD} + 0.5V$ for the PMOS, the threshold voltages $V_T$ of the transistors, increase to 0.15V and when the substrate terminal are further biased to a higher voltage of -3.3V for the NMOS and $V_{DD} + 3.3V$ for the PMOS, the threshold voltages $V_T$, increase to 0.55V.

It is proven that dynamically varying the substrate biasing depending on the circuit load, the leakage power in a circuit can be considerably lowered.
The variation in the threshold voltage $V_T$ for a NMOS transistor fabricated using the 0.3$\mu$m technology, is shown in the following Fig. 1.12 [10].

![Graph showing threshold voltage variation for varying substrate bias in a NMOS transistor](image)

**Fig. 1.12 – Threshold Voltage Variation for Varying Substrate Bias in a NMOS transistor [10]**

### 1.8.1 Mathematical Analysis

Due to this increase in threshold voltage $V_T$, there is a significant reduction in the sub-threshold leakage. But on the other hand, this also increases the tunneling (BTBT) leakage slightly. For any gate in the logic circuit block under consideration, this effect can be modeled as follows:

The reduced sub-threshold leakage upon applying substrate basing is given as,

$$I_s = e^{K_s\Delta V}I_s$$  \hspace{1cm} (1.9)
and, the increased BTBT leakage is given as,

\[ I_t = e^{-K_t \Delta V} \bar{I}_t \]  

(1.10)

where, \( \Delta V \) is the potential difference, \( (V_{DD} - V_p) \). \( \bar{I}_t \) and \( \bar{I}_s \) are the tunneling and sub-threshold leakage current values before applying the substrate bias potential, and \( K_t \) and \( K_s \) are the technology dependant exponential reduction factors for the tunneling and sub-threshold leakages [11, 13].

The Energy Overhead in charging the PMOS substrate capacitance \( C_b \), from \( V_{DD} \) to a different substrate bias voltage \( V_p \), is given as,

\[ E_{overhead} = C_b(\Delta V)^2 \]  

(1.11)

and, the energy saving over an idleness time period \( t \), is represented as,

\[ E_{saving} = V_{DD} \{ \bar{I}_s(1 - e^{K_s \Delta V}) + \bar{I}_t(1 - e^{-K_t \Delta V}) \} \cdot t \]  

(1.12)

Therefore, the overall leakage saving is now,

\[ E_{overall} = E_{saving} - E_{overhead} \]  

(1.13)

The above mathematical analysis has been performed based on the PMOS substrate biasing models. The NMOS models follow a very similar pattern.
1.8.2 Advantages of Dynamic Substrate Biasing

The advantages of using dynamic substrate biasing over the other methods for leakage power reduction are:

a) There are no additional costs involved, such as addition of decaps or Isolation cells.

b) It is more suited to be used in the active mode working of a circuit, given its low energy over head and quick wake-up times.

c) One of the most important advantages is that dynamic substrate biasing possesses state retention capabilities, which gives this method a phenomenal lead over the others since we require this functionality while using it during the active mode of a circuit block.

The following Fig 1.13 [16], shows the comparison of the total power saved between 3 of the most prominent leakage control mechanisms. It can be seen that substrate biasing is not as powerful as stacking transistors or employing the concept of power gating using sleep transistors, in terms of leakage power saving. But, one major advantage of this technique is that it can be used effectively to reduce leakage during the active mode of any logic circuit while the others render themselves unusable due to the very high energy overheads caused.

This concept of dynamically altering the threshold voltage by way of employing varying substrate bias voltages forms the central idea behind this research.
Extensive experimentation has been performed using various test circuits, to fine tune the exact biasing voltages used for the different benchmark circuits.

Dynamic substrate biasing has been used as an active mode run–time leakage control technique based upon the circuit workloads. The following chapters discuss about the concept and implementation of LITHE (A light $V_{TH}$ hopping technique), the solution proposed in this thesis to designing robust low power circuits. The concept of LITHE is derived from dynamic substrate biasing and this when used as a run-time leakage control technique, has proven to be very effective in reducing the leakage power.

Fig. 1.13 – Leakage Saving Comparison [16]
1.9 Thesis Overview

Leakage aware designs are an indispensible part of the design and manufacturing process in today’s deep sub-micron technologies. Technology scaling continues to be a constant factor in CMOS designs, with the feature sizes of devices manufactured being scaled down below 28nm. Chapter 1 in this thesis presents an introduction to power and the different kinds of power that are present in transistors today. Leakage power in particular is discussed in detail along with the concept of technology scaling and its effects on leakage power. Starting from the 45nm technology, it has been shown that the leakage power consumption in a circuit catches up with the dynamic power consumption and continuing this trend, it has been projected that for future technologies, the leakage power consumption will even dominate the dynamic power consumption.

This increasing leakage power consumption in the deep sub-micron CMOS technologies has manifested the need for more aggressive control mechanisms. A section of Chapter 1 is dedicated to discussing the leakage power control mechanisms that are in use today. Further, the leakage control mechanisms are widely categorized into 2 categories namely, Design-Time control mechanisms and Run-Time control mechanisms. As the names suggest, Design-Time control mechanisms are incorporated into the circuit during the design phase and are not capable of dynamic control. This limits the extent of effectiveness in the leakage power reduction capability of this technique.

Alternatively, Run-Time leakage control mechanisms monitor the circuit and dynamically flip it into a low power mode of working, depending upon the circuit’s workload. These techniques yield a significant power saving and a significant amount of research in low power designs
today, is directed towards this technique. The final section of Chapter 1, focuses on a prominent run-time control mechanism known as Dynamic Substrate Biasing. The workload of any circuit can be defined under 2 broad classifications, viz. Active mode and Standby mode.

There are many robust leakage power reducing techniques that are in use today to tackle the issue during the standby mode of a circuit. It is the active mode that presents an interesting view to the problem as a whole. Scrutinizing the workload of a circuit in its active mode of working showcases that there are copious opportunities of slackness that a designer can take advantage of and utilize to construct a better leakage aware design. This is classified as Run-Time Active leakage control (RALC).

Chapter 2 of this thesis elaborately discusses the concept of RALC and its effectiveness in exploiting the short periods of idleness encountered in circuit blocks, during the active working state of a circuit. The concepts of Energy Breakeven Period (EBP) and Wakeup Time Period (WTP) provide the metrics based upon which RALC can be effectively deployed, in order to maximize power savings and not cause any terminal errors.

Experimental observations show that the WTP for any logic circuit block is not much of a concern given the fact that it is generally much faster compared to EBP. The results for EBP computations obtained, show that the EBP values are pretty large for aggressive leakage saving purposes and some tradeoff needs to be made in order to reduce this.

Also, the EBP varies significantly for different circuits and there needs to be a check such that the expected EBPs are all within a specific time range, so that a better generic control mechanism can be designed.
Key issues to using RALC are the optimum granularity level on which it can be applied and deciding on an efficient leakage reduction mechanism to be used with it. These issues are addressed by the technique presented as the central idea of this research, known as LITHE (Light Threshold Voltage ($V_{TH}$) Hopping Technique).

Chapter 3 of this thesis discusses the concept of the LITHE extensively. The idea behind LITHE is based off of a popular technique known as Threshold Voltage hopping and this is achieved by means of Adaptive Substrate Biasing. Together, this forms the core of this research. This research aims to convincingly address all the issues of the RALC as a viable solution to designing robust leakage aware designs.

Aggressive exploitation of idleness during the active mode working of a circuit, fused together with the idea of LITHE, is the solution proposed towards tackling leakage power issues in deep sub-micron technologies, by means of this research. The LITHE is applied at a block level granularity given the advantages of easiness in automation and less contribution to area and power overhead.

At the logic circuit level the RALC control is also simple and straightforward. Light substrate biasing voltages are chosen as opposed to the high conventional ones in order to reduce the EBP of the circuit block and also help quicken the WTP. The tradeoff made by choosing lighter bias voltages is a low penalty in overall leakage reduction. This is acceptable as the leakage power saved is during the active mode working of a circuit, where functional integrity is more important.
Chapter 3 also presents a script that was developed in order to aid the automation of the LITHE process and this has been used to prove that the LITHE mechanism can be easily integrated within the power aware CAD design tools available today. The LITHE mechanism was also tested out on dynamic CMOS based designs and the results obtained through simulations showed that this concept does not yield successful results when integrated into D-CMOS designs.

Continuing on, chapter 4 deals with a salient improvement to the already existent LITHE control circuitry. The improvement proposed to the existing LITHE controller consists of using only NMOS switches for both the PUN as well as the PDN of a circuit. This has a significant performance improvement over the conventional approach in terms of total leakage power saved and also has a reduced area penalty.

The impediment of this method is that it requires a total of 3 extra voltages to be generated viz. \( V_{DD} + V_T, \) Light \( V_P + V_T \) and –Light \( V_N, \) as opposed to only the 2 extra substrate biasing voltages (Light \( V_P \) and –Light \( V_N \)) required as per the conventional approach. This drawback is offset by the total area reduction achieved.

Next, Chapter 4 discusses about automating the generation of the LITHE control signal. A primitive solution is first analyzed, whose concept is based off of the look ahead concept used in the forwarding technique while designing instruction pipelines. This is a more deterministic approach and involves actually looking at the inputs before they are propagated to the logic circuit block. This method has shown to work well on bigger circuits without any signal interdependencies.
Finally, chapter 5 deals with using the LITHE successfully on RTL modules of high performance circuits, where timing is of utmost importance. The primitive deterministic approach proposed earlier is further analyzed for the timing and reliability issues coupled with throughput reduction. This is unacceptable in the case of circuits with a large number of critical signals that are inter-dependent between circuit blocks. Thus, the need for pursuing a more heuristically directed approach was necessitated.

An 8-bit RISC microprocessor was designed to be used as a benchmark for this approach and a pseudo-deterministic approach was designed with the help of the instruction pipeline present. Experiments proved to be conclusive and help validate the method’s accuracy.
CHAPTER 2

RUN-TIME LEAKAGE CONTROL

2.1 Introduction

As discussed earlier, leakage control techniques can be broadly classified as either design time leakage control techniques or run-time leakage control techniques. Run-time leakage control techniques use a dynamic approach towards reducing the leakage power. More often than not, there is a control circuitry that works in tandem with the circuit block under test, which dynamically asserts control signals thereby switching the circuit between a low power mode and the normal mode of working.

There is bound to be an energy overhead that is caused due to the addition of the extra control circuitry. This is an important factor to be considered while designing for low power using run-
time leakage control techniques. The most important factors that need to be accounted for while utilizing run-time leakage control techniques are, a good workload monitor that does not contribute very heavily to the energy overhead in switching the circuit between the low leakage and normal states and a robust leakage control mechanism that can help deliver greater power savings when compared to the energy overhead incurred.

Categorizing further, the run-time leakage control mechanisms can either be active mode control or standby mode control mechanisms. All the ideas discussed in prior chapters about Adaptive Substrate Biasing have all been based upon research conducted, using the method as a standby mode leakage control mechanism. There are many robust leakage power reducing techniques that are in use today to tackle the issue during the standby mode of a circuit. It is the active mode that presents an interesting view to the problem as a whole. The research presented in this thesis, aims at utilizing Adaptive Substrate Biasing as an active mode leakage power control mechanism.

2.2 Run-Time Active Leakage Control (RALC)

In the technologies of today, that are lesser than 45nm, it is observed that the leakage during the active mode working of a circuit is much greater than that during its standby mode. This necessitates the use of leakage control mechanisms such as Power Gating, Adaptive Substrate Biasing and Input Vector control. A majority of the run-time leakage power control techniques present today only help in aggressive leakage power reduction during the standby mode of any circuit. But, as technology scales down further, the need for much a more intense leakage power
reduction mechanism arises. The concept of Run-Time Active Leakage Control (RALC) promises to provide a viable solution to this issue.

![Image](image.png)

**Fig. 2.1 – Run-Time Active Leakage Control (RALC) [11]**

The concept of RALC is depicted using the above Fig. 2.1. It can be seen that the there is a workload monitor present, which keeps track of the input vectors to the logic circuit block. Once there is a sufficient idleness that is detected in the circuit, the RALC automatically asserts a control signal thereby changing the state of the circuit to a low leakage state. This provides an opportunity to significantly reduce leakage power in a circuit, since it helps to exploit the idleness in the active mode working of the circuit.

Other than reducing the leakage power of the circuit, the RALC also provides for other added advantages that make the RALC not only a viable solution but also a more powerful one.
The working of the RALC is further explained using the following graphical representation:

![Diagram](image)

**Fig. 2.2 – Working of the RALC**

Let us assume the workload shown above in Fig. 2.2, to be the actual workload of any logic circuit block. It has an active time period when all the component blocks of the circuit are active and working and also a standby time period. As discussed earlier, there are many aggressive leakage reduction techniques that are present to tackle the leakage issue during the standby time period of a circuit and therefore, we will not concentrate on this for the purposes of the research presented in this thesis.

Observing the active time period of the circuit more closely, we can see that there are inherent periods of idleness that are present when not all the component blocks of the circuit are actually working. These intermittent periods of idleness are depicted as grayed blocks in the above Fig. 2.2. It can be observed that in all the traditional control mechanisms that are present today, the circuit is maintained in its normal working state during its entire active time period and is changed to a low leakage state for the entire duration of the standby time period.
The RALC helps to exploit the temporary periods of idleness in the active mode of the circuit by keeping it in its normal working state and switching it to a low leakage state upon detection of sufficient idleness.

There are 2 important issues that need to be addressed while using RALC as the solution to designing robust leakage resistant circuits. They are:

a) The control mechanism that is to be used with the RALC concept.

b) The granularity of the logic circuit block on which this control is to be applied.

These issues are addressed by the technique presented as the central idea of this research, known as LITHE (Light Threshold Voltage ($V_{TH}$) Hopping Technique). The idea behind LITHE is based off of a popular technique known as Threshold Voltage hopping and this is achieved by means of Adaptive Substrate Biasing. Together, this forms the core of this research.

As stated earlier, the concept of RALC also offers added advantages other than just leakage power reduction. It is already known that sub-threshold leakage is tightly coupled with temperature. During the active mode time period, the die temperatures are high due to the higher switching activity in the circuit, which in turn increases the sub-threshold leakage. Therefore, it can be stated that the leakage power during the active mode is generally very high due to the higher die temperatures. Employing the RALC effectively, helps to reduce the leakage which helps reduce the die temperature thereby paving means for a higher reduction of leakage current. Thus, the RALC presents itself as a powerful solution to designing robust low leakage circuits.
2.3 Energy Overhead Compensation

Until now, we have only been discussing about the benefits offered by the concept of RALC. It also imposes a few constraints while using to design low power circuits. One of the major issues being, the addition of an extra leakage control circuitry and workload monitor will definitely cause extra energy overhead. This is inevitable as the above mentioned circuits need to be integrated into the design, without which dynamic leakage power reduction cannot be achieved. Therefore it is important to design the RALC in such a manner by which, the power savings not only compensate for the energy overhead but also provide a greater power saving than the overhead incurred.

The other issue of the RALC is area penalty. Addition of extra circuitry and an extra set of power rails to achieve adaptive substrate biasing all add up to the penalty incurred on area. It is known that designs need to be made as compact as possible given the fact that silicon costs money. But there is a tradeoff that needs to be made here as there is no workaround for the penalty on area. The tradeoff that needs to be made is between the total power saved, in terms of granularity of the RALC and different voltage levels for substrate biasing and the maximum area permitted for the chip.

Two key parameters that help provide a good estimation of the effectiveness of the RALC are, Energy Brekeven Period (EBP) and Wakeup Time Period (WTP). The EBP and WTP for any logic circuit block can be computed as follows:
The above Fig. 2.3 depicts the PMOS substrate biasing model derived by Xu et al. Here, the voltage $V_P$ is the substrate bias voltage, which is actually $(V_{DD} + \Delta V)$. $V_S$ is the substrate voltage of the PMOS device and $I_S$ is the sub-threshold current. $C_{VS}$ and $C_{GS}$ are the $V_{DD}$ to Substrate and Gate to substrate capacitances respectively. $I_{TV}$ and $I_{TG}$ are the BTBT leakage currents and $R_1$, $R_2$ and $C_1$, $C_2$ (Not depicted in above Fig. 2.3) are the equivalent resistances and capacitances of the control transistors $T_1$ and $T_2$[23].

When substrate biasing is turned ON, the resistance $R_1$ is turned ON and the substrate capacitances, $C_{VS}$ and $C_{GS}$ begin to get charged by the substrate bias voltage, $V_P$.

The Energy Overhead encountered, is caused primarily due to 2 factors, viz. the Switching Energy to continually switch the control transistors $T_1$ and $T_2$ and the Body Charging Energy to charge and discharge the substrate capacitances.
The Switching Energy can be represented as,

\[ E_{\text{switch}} = V_P^2 (C_1 + C_2) \] ……………………………………… (2.1)

And, The PMOS Body Charging Energy is represented as,

\[ E_{\text{charge}} = C_{GS} \cdot V_{DD} \cdot \Delta V + \Delta V^2 \cdot (C_{VS} + C_{GS}) \] …………………… (2.2)

When the PMOS substrate is fully charged, the leakage current in the circuit reduces to,

\[ I_{\text{leak}} = e^{-Bt \Delta V} I_t + e^{Bs \Delta V} I_s \] ……………………………………… (2.3)

where, \( I_t \) and \( I_s \) are the tunneling (BTBT) and sub-threshold leakage current values before applying the substrate bias potential, and \( Bt \) and \( Bs \) are the exponential reduction factors for the tunneling and sub-threshold leakages [11, 13].

The leakage power savings per unit time is given as:

\[ P_{\text{save}} = \{(I_t + I_s) - I\} \cdot V_{DD} \] ……………………………………… (2.4)

Now, the Energy Breakeven Period (EBP) can be computed as follows:

\[ \frac{E_{\text{EBP}} = E_{\text{switch}} + E_{\text{charge}}}{P_{\text{save}}} \] ……………………………………… (2.5)

This EBP time value that is obtained, helps provide the designer with an accurate estimate when the power savings and energy overhead would break even for the logic circuit block under consideration, in terms of the number of clock cycles.
Therefore, the EBP is made the lower bound for applying the adaptive substrate biasing mechanism. The leakage mode must be turned ON for at least the duration of the EBP for effective leakage power reduction and substrate biasing is not to be applied, if the period of idleness is definitely lesser than that.

The Wakeup Time Period (WTP) is calculated as, [14]

\[
WTP = \frac{(C_{VS} + C_{GS}) \Delta V}{I_{R2}} \\
\]

This WTP gives an accurate estimate of the time it would take for the logic circuit block under consideration to wake up from the low leakage state into the normal working state [14]. \( I_{R2} \) in the above equation is the ON current that flows through the control transistor \( T_2 \), when the substrate potential is discharged back to \( V_{DD} \).

Now, the control policy for the duration of the adaptive substrate biasing to be turned ON can be set as follows:

\[
T_{RALC} > T_{EBP} + T_{WTP} \\
\]

Summarizing, the time period for which the run-time active leakage control has to be turned ON must be greater than the sum of the Energy Breakeven Period and the Wakeup Time Period. If this condition is not met and adaptive substrate biasing is still turned ON no matter what, then it paves way for drastic consequences that will impair the basic functionality of the logic circuit block.
Therefore, extra care needs to be taken to ensure that the above condition is met every single time. Generally, WTP is much faster when compared to EBP and hence more importance is given to EBP in this research.

2.4 Analytical Observations

Experiments were conducted to compute the expected EBP, on a few select circuits from the ISCAS – 85 and 89 benchmark models. The RALC was setup using adaptive substrate biasing as the leakage control mechanism. A 1ns clock was used, so that the frequency of operation could be set to 1 GHz and the substrate bias voltages were chosen for maximum leakage reduction. All the benchmark circuits listed below, were simulated using the Arizona State University’s Predictive Technology Model (PTM) [15] for both the 32nm as well as the 45nm technology simulations. The results obtained are shown in the following Table 2.1.

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th># of Gates</th>
<th>45nm Tech(ns)</th>
<th>32nm Tech(ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S349</td>
<td>176</td>
<td>2.34</td>
<td>1.16</td>
</tr>
<tr>
<td>C880</td>
<td>383</td>
<td>2.91</td>
<td>1.89</td>
</tr>
<tr>
<td>C2670</td>
<td>1193</td>
<td>1.87</td>
<td>2.07</td>
</tr>
<tr>
<td>C3540</td>
<td>1669</td>
<td>3.48</td>
<td>2.38</td>
</tr>
<tr>
<td>C5315</td>
<td>2406</td>
<td>4.02</td>
<td>2.43</td>
</tr>
<tr>
<td>C6288</td>
<td>2406</td>
<td>6.13</td>
<td>4.52</td>
</tr>
<tr>
<td>C7552</td>
<td>3512</td>
<td>5.28</td>
<td>3.49</td>
</tr>
</tbody>
</table>

Table 2.1 – Computed EBP for ISCAS-85/89 Benchmark Circuits
2.5 Conclusions

Summarizing, this chapter covers the concept of Run-Time Active Leakage Reduction (RALC) and its effectiveness in exploiting the short periods of idleness encountered in circuit blocks, during the active working state of a circuit. The concepts of Energy Breakeven Period (EBP) and Wakeup Time Period (WTP) provide the metrics based upon which RALC can be effectively deployed, in order to maximize power savings and not cause any terminal errors. Experimental observations have shown that the WTP for any logic circuit block is not much of a concern given the fact that it is generally much faster compared to EBP.

The results for EBP computations show that the EBP values are pretty large for aggressive leakage saving purposes and some tradeoff needs to be made in order to reduce this. Also, the EBP varies significantly for different circuits and there needs to be a check such that the expected EBPs are all within a specific time range, so that a better generic control mechanism can be designed.
CHAPTER 3

IMPLEMENTATION AND ANALYSIS
OF THE LITHE TECHNIQUE

3.1 Introduction

As discussed earlier, key issues to using RALC are the optimum granularity level on which it can be applied and deciding on an efficient leakage reduction mechanism to be used with it. These issues are addressed by the technique presented as the central idea of this research, known as LITHE, which stands for a Light $V_{TH}$ (Threshold Voltage) Hopping Technique.

The idea behind LITHE is based off of a popular technique known as Threshold Voltage hopping and this is achieved by means of Adaptive Substrate Biasing. The evident gap that exists between the dynamic power reduction techniques and the leakage power reduction techniques of today, while using the conventional power saving mechanisms is mainly because of the very large EBP
and WTP times. The main focus of the LITHE is to reduce EBP and achieve a total EBP and WTP of within one clock cycle, while still maintaining the effectiveness of the leakage power reduction so that, both the energy overhead problems and the wake-up delay issues encountered, can be better resolved.

For all experimentation purposes in this research, the operating clock frequency of all test circuits and benchmark models is set to be 1 GHz universally. This helps provide a uniform platform to compare results obtained.

### 3.2 LITHE

As mentioned earlier, LITHE is the acronym for a light $V_{TH}$ (threshold voltage) hopping technique. Aggressive exploitation of idleness during the active mode working of a circuit, fused together with the idea of LITHE, is the solution proposed towards tackling leakage power issues in deep sub-micron technologies, by means of this research. This forms the core of the research presented in this thesis and is covered in a detailed fashion in this chapter. Dynamic substrate biasing is used as the leakage control mechanism, in order to realize the RALC for a circuit. This is achieved by means of a method popularly known as VTH Hopping. It is conceptually very similar to that of adaptive substrate biasing.

This when combined along with a good circuit workload monitoring mechanism, forms the backbone of the LITHE. The basic architecture of the LITHE is depicted as follows:
In the above Fig. 3.1, the Light $V_P$ and Light $-V_N$ are the biasing voltages for the respective PMOS and NMOS substrates in the design. As is evident from the above figure, this requires the usage of 2 sets of power rails for all the gates in the circuit as there needs to be one dedicated set of power rails for carrying the substrate bias alone.

There are two ways, in which the substrate biasing voltages can be provided to the circuitry and that is either through the usage of off-chip voltage sources or via using an on-chip voltage level converter. Both the methods have their inherent pros and cons. When chip area is not much of an issue, the usage of an on-chip voltage converter is better justified as it helps to save on pins and
extra voltage sources. But, in designs where the overall chip area is a tightly bound constraint, it is better to use off-chip voltage sources. The cons in employing this approach are dedicating extra pins to carry the bias voltages and also the usage of multiple voltage sources. To some extent, the designer can be excused of settling in for this tradeoff, in the case of multiple voltage island based designs. The required substrate bias voltages can be easily supplied from the power rails of the voltage islands that use the same voltage levels.

The transistors S1 thru S4, shown in the above Fig. 3.1 are the control transistors that are used in order to realize $V_{TH}$ hopping. The differential widths of the transistors seen in the figure are due to the WTP. The control transistors S2 and S4 are sized to have a greater width in order to enable a faster WTP, so that the switch from the low leakage mode to the normal mode of working is as quick as possible.

The application of the LITHE control signal implies that the source to substrate voltage, $V_{SB}$ of the PMOS transistors switches from $V_{DD}$ to the N-type substrate bias voltage, $V_P$ and that of the NMOS transistors in the design switches from GND to the P-type substrate bias voltage –$V_N$.

3.2.1 Optimum Granularity

One of the most important concerns of implementing the LITHE is to decide on the optimum granularity of the logic circuit block. The LITHE can either be applied to a specifically selected logic block of the circuit, which is also known as the coarse grained approach or, it can also be applied at the gate level for each individual gate within the circuit and this is the fine grained approach.
A more aggressive fine grained approach can also be pursued, where the LITHE is applied individually to every transistor that is present in the design.

Most of the established researches tend to adopt the coarse grained approach [17]. Recent studies are however being conducted into employing a more fine grained approach [18, 19] to help enable more aggressive leakage power savings. The fine grained approach still needs to be better refined as issues such as sizing, area overhead contribution and wakeup delay issues still exist. It still remains to be confirmed if adopting a fine grained approach can help optimize the design on hand.

In order to employ this approach, each gate in the design needs to be individually controlled, i.e. each gate will need its own individual RALC control signals. This not only increases the area overhead, but also contributes to an unacceptable power overhead. The extra power required to drive all the control signals will increase to such an extent where it eclipses the total power saved and in the process renders the whole LITHE concept unusable.

Thus, it can be stated that even though adopting a fine grained approach definitely helps increasing the total leakage power savings, it does not yield the most optimum solution to the problem on hand given its high area and power overhead and complexity. Therefore, the coarse grained approach is used for all purposes of the research presented in this thesis. Assuming total chip area to be a very important cost factor, LITHE is applied at block level in all the experimentations performed.
When $V_{TH}$ hopping is applied, it can be observed that both the leakage energy saving, the substrate charging/discharging overhead and the switching energy overhead, for each gate in a circuit vary significantly. Some gates in the circuit may have a higher power saving and a low energy overhead, while certain gates have lower power saving capabilities and high energy overheads.

The gates that display higher power saving for a lesser energy overhead are chosen as the candidates on whom the LITHE is applied, as the gates with this quality can be effectively used both in the active mode as well as standby mode power reduction mechanisms. The aim of this approach is to select gates with lower EBPs, in order to reduce the overall EBP of the entire circuit.

Therefore, it is sufficient, if the LITHE was applied on a select subset of the circuit containing the low EBP blocks/gates. This ensures greater power savings and lesser area and power overhead. Further, it can also be said that gates with a smaller EBP have smaller overheads and thus smaller substrate capacitances. Since the capacities of the capacitors that need to be discharged are low, this aids in reducing the wakeup delay.

Therefore, the gates/blocks with lower EBPs also have very fast WTPs. This also helps attain the goal of reducing total EBP and WTP to within one clock cycle. By means of experimentation, gates with lower EBPs have also shown to possess a high amount of leakage. Thus, applying the LITHE on this subset of blocks, helps reduce a significant amount of leakage power. Summarizing, the LITHE is applied coarsely at a logic block level within the circuit and it is specifically applied to those select subset that possess a low EBP and quick WTP.
3.2.2 Light Threshold Hopping

Light in the acronym LITHE, refers to a light substrate biasing voltage. Conventionally, the voltage at which both the sub-threshold leakage as well as the tunneling (BTBT) leakage reduction is maximum, is defined as the optimal substrate biasing voltage [13]. But, this optimal substrate bias voltage is generally too high to be used in active mode leakage reduction mechanisms. This is mainly because of the reason that the substrate biasing overhead that occurs for the high level of bias voltage is not negligible. In plain terms, this just means that the charging and discharging of the substrate capacitances will take a very long time given the high amount of charge. Therefore, choosing smaller or a lighter substrate biasing voltage, helps in significantly reducing this substrate charging overhead.

![Diagram](image)

Fig. 3.2 – A 16 chain inverter with conventional substrate bias voltages
The above Fig. 3.2 depicts a 16 chain inverter that has a LITHE control connected to it. The substrate biasing voltages applied are the conventional bias voltages at which there is maximum reduction in sub-threshold as well as tunneling leakage.

Fig. 3.3 – A Lightly biased 16 chain inverter

Fig. 3.3 depicts a 16 chain LITHE enabled inverter that is lightly biased. Compared to the previous conventional setup, the substrate bias voltages \( V_P \) and \( -V_N \) here are much smaller.
The measured EBP and WTP values are summarized in the following Table 3.1:

<table>
<thead>
<tr>
<th></th>
<th>VP</th>
<th>VN</th>
<th>EBP (ns)</th>
<th>WTP (ns)</th>
<th>Leak Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional</td>
<td>3.0 V</td>
<td>-2.0V</td>
<td>9.7</td>
<td>0.52</td>
<td>- 99 %</td>
</tr>
<tr>
<td>Light</td>
<td>1.2V</td>
<td>-0.3V</td>
<td>0.37</td>
<td>0.1</td>
<td>- 88 %</td>
</tr>
</tbody>
</table>

Table 3.1 – Conventional and Light Biasing comparison on a 16 chain Inverter

It is evident from the above Table 3.1 that when the substrate bias voltages $V_P$ and $-V_N$ are changed from the conventional values of 3.0V and -2.0V to a light bias value of 1.2V and -0.3V, there is a 26-time reduction that is seen in the EBP and the WTP also reduces by approximately 5 times.

The negative aspect is that there is a tradeoff of approximately 11% that needs to be made in terms of total power saved. However, 88% in itself is a good amount of power saving that is obtained during the active working mode of the circuit. The EBP of 0.37ns and WTP of 0.1ns obtained also fall within the important target of reducing the overall EBP and WTP to within 1 clock cycle (Assuming a 1GHz clock).

For a low penalty in leakage power reduction, there is a significant reduction in EBP and WTP that is obtained [11]. Since the biasing voltages are lower, the time required to charge and discharge the substrate capacitances is very less.
The above Fig. 3.4, shows a graph that is plotted for substrate biasing voltage against Energy. It can be seen that as the value of the substrate biasing voltages chosen goes on increasing, the leakage reduction that is obtained increases initially and settles to become a constant after a certain voltage. The substrate charging overhead on the other hand, rises exponentially from the break-even point when the body charging overhead and leakage reduction are equal.

Therefore, if a higher conventional bias voltage is chosen, even though the reduction in leakage power obtained is marginally more than that obtained in the case of a lighter bias voltage, the substrate charging overhead experienced is very great. Thus, choosing a light substrate bias voltage helps in significantly reducing the energy overhead and delivers a better percentage of overall power savings compared to the conventional technique.

LITHE has been used at a block level granularity using light biasing voltages in all experimentation performed to support the claim of this research.
3.3 Automating the LITHE Mechanism

One other major advantage of resorting to applying the LITHE mechanism at a block level granularity is the ease in automating. If a finer grained approach was pursued, it would only make it more complex to automate the integration of the LITHE. Excessive fine grained approaches not only increase the overheads caused as stated earlier, they also make it improbable to automate the LITHE enabled designs.

![Fig. 3.5 – LITHE CAD process](image)

51
The above Fig. 3.5 shows the top level CAD flow used in the implementation of the LITHE. The LITHE cell library is created using HSPICE netlists of LITHE enabled cells. The layouts of all the LITHE standard cells were first done using the Magic Layout tool and later extracted to HSPICE. These were then analyzed for various input slews and output loads before the LITHE cell library was successfully created.

Once the leakage power savings and the energy overheads were computed, along with the individual EBPs and WTPs of the logic circuit blocks, circuit blocks with low EBPs and high power savings were selected and replaced with the LITHE enabled versions of the same and the final netlist was synthesized. These netlists were then simulated by applying various test vectors and the overall power savings were measured.

### 3.3.1 LITHE Selection Algorithm

The function of the LITHE selection algorithm is to select the set of logic circuit blocks with a low EBP and high power saving yield and replace them with their LITHE enabled counterparts for using the RALC mechanism.

First, the energy overhead and leakage power saving for every logic block is computed. Next, the corresponding EBPs and WTPs are also calculated and stored. Then, a specific set of circuit blocks with low EBP are chosen for replacement. Next, a list of these selected blocks is constructed and the corresponding LITHE enabled blocks are marked for replacement. The most important constraint in executing this algorithm is that the above steps are followed as long as 2 vital criteria are satisfied at all instances of the process.
They are:

a) Maximize the power savings, $P_{\text{save}}$ (from eqn 3.4)

b) $T_{\text{EBP}+\text{WTP}} \leq T_{\text{CLK}}$

The LITHE selection algorithm is as follows:

Step 1: Obtain values of $\bar{I}_t$ and $\bar{I}_s$ (The tunneling (BTBT) and sub-threshold leakage current values before applying the substrate bias potential) and $B_t$ and $B_s$ (The exponential reduction factors for the tunneling and sub-threshold leakages), from the technology specified.

Step 2: Compute $E_{\text{switch}}$, $E_{\text{charge}}$ and $P_{\text{save}}$ for all logic circuit blocks.

Step 3: Compute EBPs and WTPs of all logic circuit blocks.

DO

Step 4: Select logic circuit blocks with high power savings and low EBP.

Step 5: Queue select blocks to be replaced with LITHE enabled versions.

Step 6: Mark the blocks for replacements and replace.

WHILE

$P_{\text{save}}$ is maximized at all times.

$T_{\text{EBP}+\text{WTP}} \leq T_{\text{CLK}}$

Step 7: Re-synthesize design to obtain final netlist.
Utilizing the aforementioned CAD process flow, with the LITHE selection algorithm integrated within, the LITHE based circuit designs are successfully implemented.

3.4 Analytical Observations

Experiments were conducted on various benchmark circuits to test the functioning of the LITHE and the results were recorded for analysis. Basic circuits such as the 16 stage inverter chain and positive edge D-flip flop were laid out using the Magic layout editor and then extracted to HSPICE and simulated. The other ISCAS 85/89 benchmark circuits were simulated only in HSPICE for verification purposes. All the simulations were done using the Arizona state University’s low power (LP) predictive technology model (PTM) for the 32nm technology [15]. Please refer to the appendix-A section, for the detailed description of the experimental setup.

3.4.1 Results analysis for a D-Flip Flop

The following figures, 3.6 and 3.7 are the gate level implementation and layout representation for a static D-flip flop cell. The positive edge D-flip flop used for this experimentation was obtained from the IIT standard cell library (now rechristened the Oklahoma state university (OSU) standard cell library) [20].
Fig. 3.6 – Gate Level Implementation of a D-Flip Flop

Fig. 3.7 – Layout of a LITHE enabled positive edge D-Flip Flop standard cell

<table>
<thead>
<tr>
<th>Technology</th>
<th>ΔVP (V)</th>
<th>-ΔVN (V)</th>
<th>EBP (ns)</th>
<th>WTP (ns)</th>
<th>Leakage I (µA)</th>
<th>LITHE ON Leakage (µA)</th>
<th>Leakage Saving</th>
<th>Area Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>45nm</td>
<td>0.26</td>
<td>0.35</td>
<td>0.59</td>
<td>0.2</td>
<td>47.6</td>
<td>19.5</td>
<td>- 59 %</td>
<td>+ 4 %</td>
</tr>
<tr>
<td>32nm</td>
<td>0.3</td>
<td>0.36</td>
<td>0.43</td>
<td>0.1</td>
<td>58.1</td>
<td>28.3</td>
<td>- 51 %</td>
<td>+ 4 %</td>
</tr>
</tbody>
</table>

Table 3.2 – Leakage Savings on a standalone D-Flip Flop
The results obtained from applying the LITHE to a standalone D-Flip Flop are documented in Table 3.2. It can be seen that for the 32nm technology, the substrate of the pull-up network (PUN) is lightly biased to 1.2V and the pull-down network (PDN) substrate is lightly biased to a voltage of -0.36V. Under these circumstances, the leakage current which was normally 58.1µA has reduced to a value of 28.3µA upon application of the LITHE. This constitutes an approximate reduction in leakage of 51%. The area penalty paid in this case is an increase in layout area by +4%.

Figures 3.8 and 3.9 are graphical representations of the above mentioned results. The reduction in the leakage current reported can be observed in Fig. 3.9.
3.4.2 Benchmark Results

The following Table 3.3 summarizes the results of the leakage power savings measured for the different benchmark circuits on which LITHE was employed. Please refer to the appendix-A section, for the detailed description of the experimental setup.

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th>∆VP</th>
<th>-∆VN</th>
<th>Leakage Reduction</th>
<th>Area Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>16 Chain Inverter</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 63 %</td>
<td>+ 3.5 %</td>
</tr>
<tr>
<td>D-Flip Flop</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 51 %</td>
<td>+ 4 %</td>
</tr>
<tr>
<td>S349</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 46 %</td>
<td>+ 2 %</td>
</tr>
<tr>
<td>C880</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 57 %</td>
<td>+ 3 %</td>
</tr>
<tr>
<td>C2670</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 52 %</td>
<td>+ 2 %</td>
</tr>
<tr>
<td>C3540</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 52 %</td>
<td>+ 2 %</td>
</tr>
<tr>
<td>C5315</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 50 %</td>
<td>+ 1.5 %</td>
</tr>
<tr>
<td>C6288</td>
<td>0.3V</td>
<td>0.35V</td>
<td>- 56 %</td>
<td>+ 1.5 %</td>
</tr>
</tbody>
</table>

Table 3.3 – 32nm Technology LITHE simulations

It can be observed that there is a leakage power saving of 50% to 60% overall when the LITHE is used, for a marginal penalty in area of 1.5% to around 3.5%. Thus, it can be seen that the LITHE is indeed a viable solution to solve the problem of active mode leakage in a circuit.

3.5 LITHE on Dynamic CMOS designs

A 7 cell dynamic CMOS (D-CMOS) cell library was created with the entire layouts done using the Magic layout editor, to test the application of LITHE on D-CMOS based designs.
D-CMOS designs inherently are low on area and power consumption as well.

The main reasons for these being the logic circuitry is constructed entirely using only the PUN or the PDN along with 2 more extra transistors, one for Pre-Charge and one for Evaluation. By property of design, D-CMOS designs have only N+2 transistors compared to the same implementation done using Static CMOS where, the number of transistors is 2N and hence are said to be area efficient. Also, the evaluate transistor acts as a footer, functionally similar to the one in the power gating mechanism, during the pre-charge phase of the clock and hence by virtue of design again, there is an inherent reduction of leakage power as well.

The designs were done with the intent of exploiting this available low power advantage and LITHE was applied on the cells to study the power patterns. All the D-CMOS designs were done, using the Domino logic design style.

This required the addition of 2 more transistors, in order to realize the inverter the output of every cell, thus increasing the total number of transistors in a cell to N+4. This is still a significant reduction in total design area compared to the equivalent static CMOS implementation.

The following figures, 3.10 and 3.11 shows the transistor level implementation of a glitch free Domino logic based D-CMOS D-Flip flop [21] and its layout level implementation.
The dynamic CMOS design based D-flip flops require 2 clocks for the normal operation, namely the CLK and DCLK signals as seen in the above Fig. 3.10. CLK is the normal clock input to the
D-flop and DCLK is the dynamic logic clock that toggles between the Pre-Charge phase and the Evaluation phase. Given this complexity in the D-flip flop design, the flop is prone to glitches. Therefore, in order to design a robust glitch free D-flip flop, the one designed by Y. You et al. [21] was used for our purposes.

The above Fig. 3.10 is a transistor level implementation of the same, as designed by Y. You et al. The transistor sizing was done based on the requirements of our research and the layout was generated using the Magic layout editor tool, as shown in Fig. 3.11.

All the D-CMOS cell designs were simulated using HSPICE and the dynamic as well as leakage powers were measured after the application of the LITHE. Please refer to the appendix-A section, for the detailed description of the experimental setup. The results obtained are summarized in the following Table 3.4:

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th>Leakage Power (D-CMOS)</th>
<th>Leakage Power (Static)</th>
<th>Dynamic Power (D-CMOS over Static)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inverter</td>
<td>- 96 %</td>
<td>- 87 %</td>
<td>+ 121 %</td>
</tr>
<tr>
<td>AND Gate</td>
<td>- 92 %</td>
<td>- 76 %</td>
<td>+ 106 %</td>
</tr>
<tr>
<td>OR Gate</td>
<td>- 89 %</td>
<td>- 74 %</td>
<td>+ 112 %</td>
</tr>
<tr>
<td>NAND Gate</td>
<td>- 92 %</td>
<td>- 78 %</td>
<td>+ 107 %</td>
</tr>
<tr>
<td>D - Flip Flop</td>
<td>- 63 %</td>
<td>- 51 %</td>
<td>+ 97 %</td>
</tr>
</tbody>
</table>

Table 3.4 – Static CMOS LITHE vs D-CMOS LITHE implementations

As is evident from the above Table 3.4, the leakage power savings obtained while using the LITHE on D-CMOS based designs over their static counterparts is of the order of 9% to 16% for the basic standard cell designs. While, on the other hand, the dynamic power consumption of the D-CMOS designs compared to the equivalent static CMOS designs is at least 2 times.
This overhead incurred in terms of the dynamic power consumption makes it next to impossible to break-even with the power savings obtained and defeats the very purpose of using the LITHE. This is mainly due to the very high switching activity present in the circuits. By virtue of design, the D-CMOS cells switch very frequently between pre-charge and evaluate states and this switching happens even under steady state input conditions. This high frequency switching, causes a large amount of power to be consumed. To top this, there is also a switching activity that happens in the substrates because of the LITHE turning on or off. All of these put together, make the dynamic power consumption unmanageable.

Other than using basic cells as a benchmark, more advanced circuits such as decoders and a single bit north-east route checker were also laid out and tested upon. There was no improvement in the leakage power reduction observed and the dynamic power overhead was phenomenal.
Based on all the results gathered from sufficient experimentation of Dynamic CMOS circuits, it can be ascertained that the LITHE is unsuitable for D-CMOS based designs.

3.6 Conclusions

Summarizing, this chapter covers the basics of using LITHE as a successful and robust RALC mechanism. The LITHE is applied at a block level granularity given the advantages of easiness in automation and less contribution to area and power overhead.

Further, at the logic circuit level, the RALC control is also simple and straightforward. Light substrate biasing voltages are chosen as opposed to the high conventional ones in order to reduce the EBP of the circuit block and also help quicken the WTP. The tradeoff made by choosing lighter bias voltages is a low penalty in overall leakage reduction. This is acceptable as the leakage power saved is during the active mode working of a circuit, where functional integrity is more important.

A simple script was developed in order to aid the automation of the LITHE process and this shows that the LITHE mechanism can be easily integrated within the power aware CAD design tools available today. The LITHE mechanism was also tested out on dynamic CMOS based designs and the results obtained through simulations showed that this concept cannot be used successfully while using D-CMOS designs.
CHAPTER 4

IMPROVEMENTS AND AUTOMATION

OF THE LITHE CONTROL

4.1 Introduction

The LITHE control this far, was enabled manually in order to execute the HSPICE simulations for testing the benchmark circuits. This chapter looks at taking it further and automating the LITHE control mechanism based upon circuit activity. Also, the LITHE is tested on complex circuits with signal inter-dependencies between logic blocks and the results of using the LITHE under such conditions are analyzed.
All along, it was simpler to analyze the results from using the LITHE as the control was done manually. While trying to automate this control, this brings along with it the need to have a dedicated control circuitry. Deciding on a control mechanism and designing this extra control circuitry is the easy part. But, this control circuitry will also add to the energy and area overhead already present. The LITHE now needs to compensate for the overall energy overhead with a better overall power saving. Also, providing LITHE controls manually off-chip meant the requirement of dedicated pins for this purpose. Therefore, a control method needs to be designed that can generate all the required LITHE control signals internally without the need for any extra pins.

4.2 Improvements to the LITHE Control

Before automating the generation of the LITHE control signals, an improvement was made to the existing LITHE control, designed by Xu, et al. [11]. This change to the conventional LITHE control, involved replacing the traditional PMOS transistors that control the pull-up network (PUN) substrate biasing voltages with NMOS transistors. Using an NMOS transistors only based LITHE control circuit showcased significant performance improvement, in terms of achieving increased leakage reduction and reduction in the overall area increase. The performance enhancements observed by incorporating this change into the existing control and the penalties obtained thereby are further discussed in the following analytical observations section.
In the above Fig. 4.1, the Light $V_{P} + V_{T}$ and Light $-V_{N}$ are the biasing voltages for the respective PMOS and NMOS substrates in the design. As is evident from the above figure, this requires the usage of 2 sets of power rails for all the gates in the circuit as there needs to be one dedicated set of power rails for carrying the substrate bias alone.

It is a known fact that the NMOS switches cannot pass logic 1, without a voltage drop of $V_{T}$. Thus, they are said to be weak conductors of logic high. This is due to the fact that the NMOS transistors stop conducting when the gate to source voltage, $V_{GS} < V_{T}$. Therefore, in order to counter this inherent drawback of the NMOS transistor, we pass voltages of $V_{DD} + V_{T}$ and Light $V_{P} + V_{T}$, to the NMOS switches in the PUN body bias control circuitry.
This ensures that, after accounting for a voltage drop of $V_T$, the correct voltage levels of $V_{DD}$ and Light $V_P$ get latched on to the substrate bias power rails, as required.

As stated earlier, there are 2 significant performance improvements observed by using only NMOS transistors for controlling the PUN substrate bias. These enhancements being:

a) The control transistors being all NMOS switches, are now much faster and switch in much lesser time compared to the conventional PMOS switches. This implies that the LITHE control can be turned ON for a greater period of time, on the select blocks. This approach also helps greatly with bigger designs because a lot more blocks now, have a lesser EBP and are thus selected to be utilized for power saving. Thus, both the individual power saving of the blocks as well as net power saving of the circuit as a whole, both increase profitably.

b) Increase in area which is an inherent drawback of both the conventional method as well as this new NMOS only based approach, is now reduced significantly. The NMOS transistors need not be sized as large as the PMOS, given the fact that the mobility of electrons is much faster than that of holes. Both the minimum-sized control transistor as well as the larger sized (for quicker wake-up WTP) NMOS transistor are now designed such that they are much smaller, compared to their PMOS counterparts. Therefore, the overall increase in area is reduced considerably.

As in the conventional method, there are two ways in which the substrate biasing voltages and the voltage $V_{DD} + V_T$, can be provided to the control circuitry. This is achieved either through the usage of off chip voltage sources or by using an on chip voltage level converter.
Both the methods have their respective pros and cons. When chip area is not much of an issue, the usage of an on-chip voltage converter is better justified as it helps to save on pins and extra voltage sources. The primary difference of this approach being, there will be 3 voltage levels required, as opposed to the requirement of only 2 voltages levels while using the conventional method. It can be argued that this would lead to an increase in area. But, the total area that is saved in the LITHE control circuits is now large that even after the addition of this extra voltage line, there is still a net area saving observed. This is the only penalty encountered, while using this NMOS only approach.

In designs where the overall chip area is a tightly bound constraint, it is better to use off-chip voltage sources. The cons in employing this approach are dedicating extra pins to carry the bias voltages and also the usage of multiple voltage sources, due to the addition of the one extra voltage required. To some extent, the designer can be excused of having to decide on this tradeoff, in the case of multiple voltage island based designs. The required substrate bias voltages can be easily supplied from the power rails of the voltage islands that use the same voltage levels, as proposed earlier.

The transistors S1 thru S4, shown in the above Fig. 4.1 are the control transistors that are used in order to realize $V_{\text{TH}}$ hopping. The differential widths of the transistors seen in the figure are due to the WTP. The control transistors S1 and S4 are sized to have a greater width in order to enable a faster WTP, so that the switch from the low leakage mode to the normal mode of working is as quick as possible.
The application of the LITHE control signal implies that the source to substrate voltage, $V_{SB}$ of the PMOS transistors switches from $V_{DD}$ to the N-type substrate bias voltage, $V_p$ and that of the NMOS transistors in the design switches from GND to the P-type substrate bias voltage $-V_N$.

### 4.2.1 Analytical Observations

This new and improved approach of the LITHE was simulated for a set of test cases and the results obtained were compared with those obtained while using the control circuitry setup as suggested by Xu, et al [11] in their research. Please refer to the appendix-A section, for the detailed description of the experimental setup. The results obtained are documented in the following Tables, 4.1 and 4.2:

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th>$\Delta V_P$</th>
<th>$\Delta V_N$</th>
<th>Leakage Reduction</th>
<th>Area Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>16 Chain Inverter</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 86 %</td>
<td>+ 2 %</td>
</tr>
<tr>
<td>D-Flip Flop</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 77 %</td>
<td>+ 2 %</td>
</tr>
<tr>
<td>S349</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 67 %</td>
<td>+ 1 %</td>
</tr>
<tr>
<td>C880</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 65 %</td>
<td>+ 2.1 %</td>
</tr>
<tr>
<td>C3540</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 74 %</td>
<td>+ 0.8 %</td>
</tr>
<tr>
<td>C5315</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 63 %</td>
<td>+ 0.76 %</td>
</tr>
</tbody>
</table>

Table 4.1 – Leakage saving and area overhead using the NMOS only control circuitry

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th>$\Delta V_P$</th>
<th>$\Delta V_N$</th>
<th>Leakage Reduction</th>
<th>Area Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>16 Chain Inverter</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 63 %</td>
<td>+ 3.5 %</td>
</tr>
<tr>
<td>D-Flip Flop</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 51 %</td>
<td>+ 4 %</td>
</tr>
<tr>
<td>S349</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 48 %</td>
<td>+ 2 %</td>
</tr>
<tr>
<td>C880</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 57 %</td>
<td>+ 3 %</td>
</tr>
<tr>
<td>C3540</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 52 %</td>
<td>+ 2 %</td>
</tr>
<tr>
<td>C5315</td>
<td>0.3 V</td>
<td>-0.35 V</td>
<td>- 50 %</td>
<td>+ 1.5 %</td>
</tr>
</tbody>
</table>

Table 4.2 – Leakage saving and area overhead using the conventional NMOS & PMOS control
The graphs represented in figures 4.2a and 4.2b above, summarizes the performance superiority of the NMOS switches only based LITHE control in terms of power saving achieved as well as the reduction in area overhead, compared to the conventional method of using both the PMOS as well as the NMOS switches. The x-axis corresponds to the benchmark number, as listed in the above tables 4.1 and 4.2. Thus, given these prominent factors, the LITHE control is better designed using only NMOS transistors.
4.3 Automating the LITHE Control

The method proposed in this section, to enable the automation of the LITHE control is very similar to the concept of Forwarding, as is used in an Instruction Pipeline. Conceptually, it is based on a look-ahead method. In this concept, the control signals to the LITHE are generated based upon the current state inputs and the next state inputs that come in to the logic circuit block. If there is an idleness that is detected in the inputs, the LITHE control to the logic block is turned on. Once a change in input has been detected while the LITHE is on, the control switches the LITHE off, before the inputs actually propagate in to the circuit block. Since the substrates are lightly biased and the blocks are chosen with a total EBP and WTP less than 1 clock period, the circuit block can be easily switched between the normal mode where LITHE is disabled and a low power mode with LITHE enabled.

Fig. 4.3 – Automated LITHE Control
The above Fig. 4.3 shows the top level setup for the automated LITHE control described. It consists of a set of registers that are operating at a clock frequency 3 times that of the circuit clock frequency. The CLK port observed in the above Fig. is not the circuit clock but the clock to the input registers. The inputs are passed on to the logic circuit block after this small delay. While the current state inputs are available at the output of this forwarding register, the next state inputs are also available at the input of the register.

Now, utilizing both the current state inputs and the next state inputs, a convenient control mechanism can be designed which effectively controls the switching on and off of the LITHE mechanism. The tradeoff while using this method is that there is a slight alternation in the throughput of the circuit block. This is evident only during the first output. Instead of arriving at the expected moment, it is delayed by the frequency of the forwarding register. After this initial glitch, the circuit block behaves as expected.

In case the output from this logic block is designed to drive the input of any other time sensitive circuit block, then it requires the addition of a synchronizer that sees to it that the signal integrities are maintained and the circuit is still functionally valid.

The EBP equation is now modified as follows:

\[
\text{EBP} = \frac{E_{\text{switch}} + E_{\text{charge}} + E_{\text{control}}}{P_{\text{save}}} \\
\text{.......................... (4.1)}
\]

where, \(E_{\text{control}}\) is the energy overhead caused due to the addition of the control circuitry. Also, the total power saved is now
\[ P_{\text{total(saved)}} = P_{\text{save}} - P_{\text{control}} \]  \hspace{1cm} (4.2)

Where, \( P_{\text{control}} \) is the power utilized by the control circuitry.

### 4.3.1 Analytical Observations

The following Fig. 4.4 shows the layout of a 4-Bit Full Carry Adder circuit done using the Magic layout editor tool. The LITHE control mechanism described above has been integrated within the design. The LITHE control mechanism employed in this case is a simple Exclusive-OR operation of the inputs. This detects any idleness in input and enables the LITHE as required.

![Fig. 4.4 – A fully automated LITHE enabled Full Carry Adder Circuit](image)
An 8-bit linear carry select adder circuit was also designed and constructed to be used as a benchmark along with the traditional full adder, to aid in the testing of the LITHE automation mechanism proposed.

![8-bit Linear Carry Select Adder](image)

**Fig. 4.5 – 8-bit Linear Carry Select Adder [2]**

The above Fig. 4.5, depicts the top level schematic of an 8-bit Linear Carry select adder. The circuit was constructed as per the specifications of J.Rabaey, et al. [2].

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th>$\Delta V_P$</th>
<th>$\Delta V_N$</th>
<th>Leakage Reduction</th>
<th>Leakage Reduction (without interconnect parasitics)</th>
<th>Area Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full Carry Adder</td>
<td>0.36V</td>
<td>-0.4V</td>
<td>- 39 %</td>
<td>- 67 %</td>
<td>+ 6.5 %</td>
</tr>
<tr>
<td>Linear carry select adder</td>
<td>0.36V</td>
<td>-0.4V</td>
<td>- 43 %</td>
<td>- 74 %</td>
<td>+ 5 %</td>
</tr>
</tbody>
</table>

**Table 4.3 – Results of Automating the LITHE on the test benchmarks**
Table 4.3 shows the results obtained when the LITHE control mechanism was built-in with the circuit. It can be observed that while the total power saving obtained is approximately 40%, the area penalty has increased to approximately 6%. The addition of the extra control circuitry is the primary cause to these observations. When the extracted interconnect parasitics are completely ignored, then the leakage reduction obtained increases to approximately 70%. This presents a good measure to compare with the ISCAS benchmark circuits experimented on, in the earlier chapters for a better evaluation of the predictive technology model.

### 4.4 Conclusions

Summarizing, this chapter covers a salient improvement to the LITHE control circuitry. The improvement proposed to the existing LITHE controller consists of using only NMOS switches for both the PUN as well as the PDN of a circuit. This has a significant performance improvement over the conventional approach in terms of total leakage power saved and also has a reduced area penalty. The impediment of this method is that it requires a total of 3 extra voltages to be generated, as opposed to only the 2 extra substrate biasing voltages required as per the conventional approach. This drawback is offset by the total area reduction achieved.

A solution to automating the generation of the LITHE control signal has also been proposed in this chapter. The idea is based upon the look-ahead concept used in the forwarding technique while designing instruction pipelines. This is a more deterministic approach and involves actually looking at the inputs before they are propagated to the logic circuit block. This method has shown to work well on bigger circuits without any signal interdependencies.
CHAPTER 5

USING THE LITHE ON RTL MODULES

5.1 Introduction

All experimentation this far was done manually by either laying out the said test circuits or using the SPICE netlists of the same. The LITHE control was integrated into them and tested for leakage power reductions. This chapter focuses on using the LITHE in the high level RTL modules. Continuing on from the previous chapter, the LITHE is tested on complex circuits with signal inter-dependencies between logic blocks and the results of using the LITHE under such conditions are analyzed. There were a few niggles with not being able to close timing, that was evident while simulating the 8-bit linear carry select adder circuit.
5.2 Drawback to the Built-In LITHE Control

Adding to the above results, there was also a negative impact observed on the linear carry select adder benchmark. Since the proposed technique to automate the LITHE control within the chip, causes an inherent small delay, it has a major impact on circuits having signal dependencies between logic blocks. As seen in the previous Fig. 4.5, the circuit is highly dependent on timing and the outputs of the 0-carry and 1-carry block have to be present at the input of the multiplexer before the arrival of the carry bit select input. The proposed LITHE control method, delays the availability of these critical inputs thereby latching incorrect values.

In order to avoid this terminal error, it requires the addition of extra registers at the carry inputs thereby delaying the arrival of the select input to the multiplexer in order to synchronize with the circuit working speed. But, this defeats the whole purpose of the linear carry select adder’s working concept. This type of adder was so designed with the sole purpose of being able to perform quick computations and the idea of delaying the inputs in order to save the active mode power does not fit well into the circuit setup. Instead, more aggressive standby mode power saving mechanisms may be pursued to save more on leakage power, while at the same time not compromising the functional benefits of the linear carry select adder.

Thus, this approach is not a viable solution in the case of complex circuits that are designed for high performance with critical signal inter-dependencies.
5.2.1 Analytical Observations

This primitive look ahead based automation approach was also tested on a North East Route Checker (NERC) circuit, for an exact understanding of its impact on the performance of a circuit with a high percentage of signal inter-dependencies.

The basic operation of the NERC circuit designed is to determine the existence of a North or an East route between a specified source cell and a destination cell. The slices i.e. the cells form an N x N array and a 7 x 7 matrix was laid out for this purpose, using 49 bit-sliced cells.

Fig. 5.1 – Gate Level Schematic of the NERC – Single Bit Slice
The above figures 5.1 and 5.2 show the gate level schematic and the layout of a single bit slice used in the NERC circuit. 49 such slices were connected together to form the entire circuit. A LITHE controller along with the previously proposed automating mechanism was connected to each slice in the circuit, to reduce the overall leakage power.

As is evident from the above gate level schematic in Fig. 5.1, the output of the D flip flop Q connects to every next flop input D. Further, the outputs N and E are connected to the next slice inputs W and S and finally, the output OUT is also connected to every next input G.
Given this high signal interdependency, this circuit presents itself as an excellent candidate to
test the short comings of the proposed automatic look ahead based LITHE signal generation.

This circuit was designed for a clock period of 2ns. Therefore, the frequency of operation was set
to be 500 MHz. Even though the clock in the automating mechanism was operated at 1500 MHz,
the issue with the latency showed up at every single cell, thereby latching incorrect values into
the flops deeper in the design. Also, the output that had to show up in the 50\textsuperscript{th} clock pulse was
available only on the 52\textsuperscript{nd} clock pulse. This delay in throughput coupled with the reliability issue
of the output being incorrect, poses a great risk in using this approach in high-performance
circuits as well as circuits with a large number of interdependent signals that need to talk
between blocks.

In the pursuit of a more heuristic approach, an RTL module of the NERC was written and tested
upon. The RTL was synthesized directly with LITHE enabled blocks to determine if
incorporating the LITHE into the RTL stages was purposeful.

The following table, summarizes the comparisons of the using the primitive approach and the
direct RTL synthesis approach on the NERC circuit designed.

<table>
<thead>
<tr>
<th></th>
<th>ΔVP</th>
<th>ΔVN</th>
<th>Leakage Reduction</th>
<th>Output</th>
<th>Validity of Output</th>
<th>Area Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>NERC Layout</td>
<td>0.26V</td>
<td>-0.3V</td>
<td>- 43 %</td>
<td>52\textsuperscript{nd} clock edge</td>
<td>Not Reliable</td>
<td>+ 1.5 %</td>
</tr>
<tr>
<td>(Primitive)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NERC RTL</td>
<td>0.26V</td>
<td>-0.3V</td>
<td>- 37 %</td>
<td>50\textsuperscript{th} clock edge</td>
<td>Good</td>
<td>+ 1.5 %</td>
</tr>
</tbody>
</table>

Table 5.1 – Comparison of NERC results based on the primitive and heuristic approach
Therefore, in order to better optimize the LITHE, a good heuristic approach was pursued to enable the automation of the LITHE control signal generation.

5.3 A Heuristic Approach to automating the LITHE

This drawback experienced in automating the generation of the LITHE control signal internally, brings in the need to adopt a more heuristic approach to solving the problem. As the name heuristic suggests, the method will provide an intuitive guess that is based solely on probabilistic determination. The heuristic approach cannot always guarantee an optimal solution.

For the purpose of generating heuristics to enable the LITHE in the presence of critical signal interdependencies in high performance circuits, an 8-bit reduced instruction set (RISC) processor was developed to be used as a benchmark to aid in evaluating the heuristics.

The processor synthesis as well as all the other synthesis work for this thesis was done using the Synopsys® Design Compiler tool. All the power measurements were done using the Synopsys® NanoSim EDA tool as well as HSPICE, as described in the appendix-A section of this thesis. The memory and ALU were developed as separate RTL modules. This processor was developed entirely using the hardware descriptive language, VHDL. This was then synthesized using both the OSU standard as well as LITHE enabled cell libraries and then simulated for power measurements.
The above Fig. 5.3 shows a top-level architectural view of the 8-bit microprocessor that was designed to be used as a benchmark for the heuristic approach. The ALU along with the Control unit, form the core of this processor.

The main memory is divided into 2 segments, viz. the code segment (CS) and the data segment (DS). The Instruction register (IR) fetches each instruction that is pointed by the Program Counter (PC), in the code segment and forwards it to the Control unit. The control unit instructs the Arithmetic and Logic Unit (ALU) to execute a particular instruction by verifying the condition code register. It also controls the memory read and write operations.
There are a set of general purpose registers associated with the ALU. One input to the ALU comes from the Register bank, through the Accumulator and the other input coming in from the temporary register, may be from the memory or a register. The PC stores the address of the next instruction to be executed.

The 8-bit ALU is capable of performing binary operations as well as logic and shift operations. The logic operations performed by the ALU are mixed with its arithmetic capabilities. The arithmetic operations that this ALU is capable of performing are addition, subtraction and multiplication. Only the ALU of the 8-bit processor was used for experimentation, as the adder, multiplier and shifter in the synthesized netlist could easily be replaced by their LITHE enabled versions and re-synthesized for the subsequent power calculations. Incorporating the LITHE into memories and other modules of the processor and utilizing it at a system level, is discussed under the future scope of this research.

5.3.1 Heuristic LITHE signal generation

LITHE is applied to the core of the microprocessor in order to save on leakage power. The ALU in the core primarily consists of an 8-bit adder, multiplier and shifter units. Since all these blocks are designed for high-performance and consist of a large number of critical signal inter-dependencies, which are highly time sensitive in order to keep up to the overall operating frequency, this setup presents itself as an excellent candidate to test the heuristic auto generation of the control and application of the LITHE.
A basic 3-level instruction pipelining was incorporated within the processor design. The 3 levels in the pipeline were Fetch, Execute and Store. This pipelining concept greatly aided the LITHE control generation process.

When a new instruction is in the Fetch stage a control signal is generated to enable the LITHE, on all logic blocks within the circuit that will not be used by that particular instruction, starting from the next clock edge. Only logic blocks with a total EBP and WTP of less than 1 clock period are chosen for application of the LITHE. This enables easy turning on or off the LITHE every clock cycle, without losing on any of the critical data executions. The substrate biasing voltages chosen are very light, in order to enable quick transitions thereby maximizing the time of power saving.

### 5.3.2 Analytical Observations

This approach is still pseudo-deterministic, as the decision to enable or disable the LITHE is done after the arrival of the instruction but before it is executed. This processor was synthesized using the LITHE enabled cell library and simulated for 2 benchmark programs, viz. 8-bit multiplication of 2 numbers and sorting of eight 8-bit numbers.

<table>
<thead>
<tr>
<th>Benchmarks</th>
<th>ΔVP</th>
<th>ΔVN</th>
<th>Leakage Reduction</th>
<th>Area Penalty</th>
</tr>
</thead>
<tbody>
<tr>
<td>8-Bit Sorting</td>
<td>0.26V</td>
<td>-0.25V</td>
<td>-27 %</td>
<td>+1 %</td>
</tr>
<tr>
<td>8-Bit Multiplication</td>
<td>0.26V</td>
<td>-0.25V</td>
<td>-36 %</td>
<td>+1 %</td>
</tr>
</tbody>
</table>

Table 5.2 – 8-Bit RISC microprocessor benchmark results
The results obtained while simulating the 8-bit microprocessor designed for the 2 benchmark programs are documented in the above Table 5.2. The area penalty is very marginal and is well under acceptable limits. But, the leakage reduction seen is only in the order of 25% to 35%.

5.5 Conclusions

This chapter covered using the LITHE successfully on RTL modules of high performance circuits, where timing is of utmost importance. The deterministic approach proposed in the earlier chapter was further analyzed for the timing and reliability issues coupled with throughput reduction. This is unacceptable in the case of circuits with a large number of critical signals that are inter-dependent between circuit blocks. Thus, the need for pursuing a more heuristically directed approach was necessitated. An 8-bit RISC microprocessor was designed to be used as a benchmark for this approach and a pseudo-deterministic approach was designed with the help of the instruction pipeline present. Experiments proved to be conclusive and sufficient.
CONCLUSION AND FUTURE ENHANCEMENTS

6.1 Future Enhancements

The research on the LITHE that was presented in this thesis still has scope for further enhancements. There is enough room for improvement given the fact that 28nm designs have already started rolling out and this has created the need for more efficient leakage saving mechanisms. The LITHE can be designed to be more robust easily adaptable to future technologies.
6.1.1 Fine Tuning

The LITHE still needs to be fine tuned in the presence of circuit parasitics, using a more stable and robust technology model. All simulations in this research were performed using the Arizona state university’s PTM for 32nm.

This model does not possess all the required parasitics information. This constitutes the need to develop an efficient scaling model, which returns actual the parasitic values, so that the LITHE can be fine tuned.

6.1.2 Automatic Place and Route

Fig. 6.1 – LITHE enabled cell locations during Place and Route

One major consideration while performing automatic place and route is that the transistors placed in a single row are generally manufactured on the same substrate. Having a continuous substrate, helps save on total chip area.
Else, if the substrate was broken then there needs to be a minimum distance between the substrates in order to not violate any design rule checks (DRC). But, substrate biasing cannot be done without breaking the substrates as individual blocks need to be individually biased. Therefore, this necessitates the need for designing a better place and route tool that can also take substrate biasing into consideration and try to place the LITHE enabled blocks closer to each other so that a continuous substrate can still be maintained.

As is evident from Fig. 6.1, since substrate biasing requires the usage of additional sets of power rails, it also places a constraint on the channel area required for routing. Thus, it would help if there existed a uniform methodology to tackle this issue.

### 6.1.3 Need for a Predictor Circuit

The heuristic approach to designing the LITHE presented in Chapter 5, still needs to be refined further. A fully heuristic solution to designing the LITHE would prove very beneficial in future technologies.

Fig. 6.2 – LITHE with a Predictor
In order to be able to achieve this, a robust and efficient predictor circuit needs to be designed that can predict the inputs before they actually arrive and the margin of error should not exceed 5%. Else, using a predictor with a low probability of successful prediction only misfires the LITHE and hampers the functionality of the circuit.

The development of good high probability predictor circuits, to use along with the LITHE will prove to be very effective as the LITHE can then be applied at level of the design. It does not need to be only at a circuit level as what has been presented in this thesis. Prior knowledge about future circuit activity can help increase the level of abstraction to an architecture and even system level, where the employment of the LITHE is bound to return a very high percentage of power savings.

6.2 Summary of Research Accomplishments

The research presented in this thesis convincingly addresses all the issues of the RALC as a viable solution to designing robust leakage aware designs. The proposed solution towards an Aggressive exploitation of idleness during the active mode working of a circuit, based on the idea of LITHE helps to successfully tackle leakage power issues in deep sub-micron technologies.

Key issues to using RALC are the optimum granularity level on which it can be applied and deciding on an efficient leakage reduction mechanism to be used with it.
The decision to apply the LITHE at a block level granularity has ceded advantages in terms of easiness in automation and less contribution to area and power overhead. Light substrate biasing voltages are chosen as opposed to the high conventional ones in order to reduce the EBP of the circuit block and also help quicken the WTP.

A script was developed in order to aid the automation of the LITHE process and this has been used to prove that the LITHE mechanism can be easily integrated within the power aware CAD design tools available today. The LITHE mechanism was also tested out on dynamic CMOS based designs and by means of the results obtained through simulations it has been proven that this concept does not yield successful results when integrated into D-CMOS designs.

A salient improvement to the already existent LITHE control circuitry is also proposed. The improvement proposed to the existing LITHE controller consists of using only NMOS switches for both the PUN as well as the PDN of a circuit. This has a significant performance improvement over the conventional approach in terms of total leakage power saved and also has a reduced area penalty. The impediment of this method is that it requires a total of 3 extra voltages to be generated viz. This drawback is offset by the total area reduction achieved.

2 methods were presented as solutions to automating the generation of the LITHE control signal. A primitive solution was first analyzed, whose concept is based off of the look ahead concept used in the forwarding technique while designing instruction pipelines. This method has shown to work well on bigger circuits without any signal interdependencies.
Finally, the LITHE was also used successfully on RTL modules of high performance circuits, where timing is of utmost importance. The primitive deterministic approach proposed earlier is further analyzed for the timing and reliability issues coupled with throughput reduction.

This necessitated the pursuit of a more heuristically directed approach. A pseudo deterministic method was developed. An 8-bit RISC microprocessor was designed to be used as a benchmark for this approach and the method was designed with the help of the instruction pipeline present. The research on the LITHE that was presented in this thesis, dispensed interesting issues that can be probed further and this research has a potential to be developed further and integrated into today’s semiconductor manufacturing industry.

Extensive experimentation has been performed on benchmark circuits to support and verify the accuracy in the claim of this thesis.
APPENDIX A

EXPERIMENTAL SETUP

A.1 Using Synopsys HSPICE:

All the power computations for small and medium sized benchmark circuits in this thesis, was carried out using the Synopsys HSPICE tool. The circuit netlist was first generated in the spice file format (.sp), to which the input vectors were applied for analysis. Since idleness during the active mode working of the circuit is what is exploited, the inputs to the test circuits are specified such that the circuit block is first kept active by providing the required inputs, after which the circuit is left to idle to a steady state. Once in the idle state, a transient analysis is performed on the circuit, for the required period of time.
For the smallest of benchmark circuits such as a standalone flip flop and basic gates the conventional differential equation solver model was used. This method involved the insertion of a dummy voltage source on the required nodes and modeling an equivalent current controlled current source, across a dummy node. This method returns accurate results but is not feasible to be used on larger benchmarks, due to the requirement of inserting dummy voltage sources manually.

Therefore, in order to measure the leakage currents and power in more complex and comparatively bigger circuits, the inputs to the test circuit/block was first maintained at steady state followed by which, the .measure tran and .measure param options available in the hspice tool deck were used.

Usage:

<table>
<thead>
<tr>
<th>FUNCTIONALITY</th>
<th>HSPICE Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>To measure the average and max current</td>
<td>.MEASURE &lt;TRAN&gt; user_var AVG/MAX node FROM = start_time TO = end_time</td>
</tr>
<tr>
<td>To perform Equation Evaluations</td>
<td>.MEASURE &lt;TRAN&gt; user_var PARAM='Equation to be Evaluated’</td>
</tr>
</tbody>
</table>
The cell libraries were all created using the Cadence SignalStorm tool. This tool creates the liberty file (.lib) for the standard cells and it automatically computes the cell_leakage_info. This is done using the inputs namely, the spice netlist, the input slew, output load, technology, process and cell functionality that are given to the tool, to create the cell library.

Since the larger benchmarks are synthesized using a standard cell library, the current between the nodes in a subcircuit can be probed, using the .PROBE option in-built in HSPICE. This provides for easy current computations in large circuits.

For small circuits that had 16 input combinations or less, the circuit/block was simulated for all the input combinations and the power was measured. For the larger blocks, a total of 15 different input patterns were used, to compute the leakage current and power information. The input vectors were chosen such that it included the worst and best case input combinations along with other input patterns chosen at random.

As the test benchmarks get bigger, this approach of using HSPICE for the computations become difficult, time consuming and inflexible. Therefore, we use the Synopsys NanoSim tool to model the leakage of the ASIC libraries.

**A.2 Using Synopsys NANOSIM:**

Synopsys Nanosim is another power analysis tool that has been used to measure the leakage currents on the bigger benchmark circuits used in this thesis.
The tool defines leakage power under “wasted power” [27]. The wasted power, modeled by the tool comprises of short-circuit current that flows during the input transitions as well as the steady-state leakage current that flows when the circuit or block is inactive. In order to differentiate between the two, the tool deck has been devised in such a way that wasted current is sub-divided into 2 categories namely, Dynamic wasted current and Static wasted current [27].

Short-circuit currents that flow in the circuit or block makes up the dynamic wasted current while steady state leakage is exclusive reported under static wasted current.

Block level power analysis using NanoSim, is done by using the report_block_powr configuration command. Using this command, along with the track_wasted option of the command enabled, returns the wasted power information for the specified circuit block.

Usage: report_block_powr user_var track_wasted=1 sub_circuit_info

In order to enable measurement of wasted power, the track_wasted switch needs to be set to 1. This reports the average wasted current, RMS wasted current as well as the wasted current percentage.

To only measure the steady state leakage component, ie. the static wasted power alone, the split_wasted option also needs to be enabled.

Usage: report_block_powr user_var track_wasted=1 split_wasted=1 sub_circuit_info
When the split_wasted switch is set to 1, the tool now reports the average and RMS static wasted current of the specified circuit block.

There are other commands such as report_ckt_leak available in the tool, which can detect the presence of static conducting paths. Also the currents at the specific nodes required, can be probed by using commands such as report_probe_i and print_probe_i.

The input vectors to the test circuits were chosen as discussed previously and provided as input along with the configuration and the netlist to the tool. The Synopsys NanoSim tool had an execution time that was faster than what was achieved by HSPICE and was also easy to use on large designs. The tool also has an in-built waveform viewer called nWave/turboWave which can be used to analyze the outputs obtained and also make necessary graphical measurements.


[8] Online Discussion Forum [Online] Available,


[10] ISSCC’96 pp 166 – 167


[16] Intel – P.Gelsinger (DAC04)


