SUB-50 NM MULTI-SEGMENT INTERCONNECT DESIGN:
A TREATISE ON SPEED, RELIABILITY AND SIGNAL INTEGRITY

A Dissertation in
Computer Science and Engineering
by
Melvin Ifeanyichukwu Eze

© 2013 Melvin Ifeanyichukwu Eze

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

December 2013
The dissertation of Melvin Ifeanyichukwu Eze was reviewed and approved* by the following:

Vijaykrishnan Narayanan
Professor of Computer Science and Engineering
Dissertation Advisor, Chair of Committee

Yuan Xie
Professor of Computer Science and Engineering

Mary Jane Irwin
Robert E. Noll Professor of Computer Science and Engineering

Jerzy Ruzyllo
Professor of Electrical Engineering, Material Science and Engineering

Raj Acharya
Professor of Computer Science and Engineering
Department Head

*Signatures are on file in the Graduate School.
Abstract

The emergence of System-on-Chip as the dominant chip level architecture in the integrated Circuit industry, has been accompanied by a need to meet the considerable communication requirements of the on-chip processing and I/O blocks. To this end, various commercial vendors have designed and licensed an array of interconnect products such as IBM Core Connect Bus, Altera Avalon, ST Micro ST Bus among others. At the physical level, all these buses are composed of long parallel wire structures that conduct the electrical signals used in communication between on-chip modules. However, with continuing dimension scaling deep below the 0.1 micron threshold, several challenges have emerged to threaten both short term and long term interconnect reliability. In this work, we focus on on-chip buses and explore the effects of basic data transmission on the severity of various phenomena affecting interconnect reliability. Our exploration will span both long and short term interconnect reliability issues such as inter-wire crosstalk, process-induced wire variation and interconnect aging. We will discuss the difficulties inherent in obtaining process-level solutions and propose circuit-level solutions as valid alternatives in many cases. We will propose a tool for profiling bus wire signal transitions in a general purpose processor running benchmark applications, and use the data obtained to generate corresponding bus crosstalk delay profiles,
bus variation delay, and bus aging profiles. We will propose circuit techniques to mitigate the effects of these phenomena and compare our solutions to current approaches in the literature.
# Table of Contents

List of Figures ................................................. ix

List of Tables ................................................. xii

Acknowledgments ................................................. xiii

Chapter 1 Introduction ........................................... 1
  1.1 Problem Statement ........................................... 3
  1.1.1 Process induced variation ................................... 4
  1.1.2 The Crosstalk effect ....................................... 5
  1.1.3 Negative Bias Temperature Instability ....................... 6
  1.2 Methods ................................................... 7
  1.2.1 Variable Cycle Timing with Temporal Redundancy ............... 8
  1.2.2 Multiple Phase Staggered Latching (MPSL) .................... 8
  1.2.3 Duty Cycle Equalization and Optimization .................... 8
  1.3 Contribution ............................................... 9
  1.4 Outline .................................................. 10

Chapter 2 Background and Related Work ......................... 12
Chapter 3  Variable Cycle Timing with Temporal Redundancy  19

3.1 Introduction .................................................. 19
3.2 Background .................................................. 20
   3.2.1 Interconnect Power ................................... 21
   3.2.2 Buses in Deep Sub-Micron Technologies .......... 23
3.3 Analytical Models For Delay and Energy Consumption .... 24
3.4 Related Work on Bus Coding, Static and Dynamic Shielding ... 26
3.5 Our Approach ................................................ 28
3.6 Area and Energy Efficient Crosstalk Analyzer ............... 28
3.7 Variable Cycle Transmission with Temporal Redundancy ...... 31
   3.7.1 Encoder ................................................. 31
   3.7.2 Decoder ................................................. 33
3.8 A Simple Example ........................................... 34
3.9 Correctness of Our Approach ................................ 35
3.10 Experiments Validating VCTR and Results .................. 39

Chapter 4  Multiple Phase Staggered Latching  44

4.1 Introduction .................................................. 44
4.2 Related Work ................................................ 46
4.3 Signal Switching Offset and Line Delay ....................... 48
4.3.1 Signal Response Model ........................................ 48
4.3.2 Nominal Response ............................................. 49
  4.3.2.1 Drive buffer/Repeater .................................. 49
  4.3.2.2 Metal trace .................................................. 50
4.3.3 Noise Response .................................................. 51
  4.3.3.1 Noise Duration ............................................ 53
4.3.4 Delay, Offset and Overlap Threshold ......................... 54
4.4 Multiple Phase Staggered Latching .............................. 57
  4.4.1 Clocking and Latch Control ................................. 58
  4.4.2 Staggered Latch Bus (SLB) .................................. 60
4.5 Experiments and Results ......................................... 61

Chapter 5 Duty Cycle Tunning in Inverting Bit Lines 65
  5.1 Introduction ...................................................... 65
  5.2 Related Work .................................................... 68
  5.3 NBTI stress and Interconnect Lifetime ......................... 69
  5.4 Reducing NBTI effect in on-chip buses ......................... 74
    5.4.1 NBTI induced Bus Lifetime ............................... 74
    5.4.2 Duty Cycle Equalization .................................. 75
    5.4.3 Duty Cycle Optimization .................................. 76
  5.5 Experiments and Results ....................................... 78

Chapter 6 Conclusion 83
  6.1 Summary of Work ............................................... 83
  6.2 Future Work ..................................................... 85
6.2.0.1 Bit Level Flow Control (BLFC) . . . . . . . . . . . 85
6.2.0.2 on-Axis Charge Recycling (oACR) . . . . . . . . . 86

Appendix 87
Correctness of the VCTR Approach . . . . . . . . . . . . . . . . . . . . . . . 87

Bibliography 90
List of Figures

1.1 Potential coupling models for typical on-chip wires both within the metal layer and across metal layers ............................. 6
1.2 Dependence of $V_{th}$ degradation on duty cycle. Source: [1] ........ 7

2.1 Relative delay for different transition patterns on a 3-wire test structure .......................................................... 13
2.2 Pipeline segment of an on chip bus. .............................................. 14

3.1 Interconnect power grows to 65% - 80% within the decade under optimistic scaling [2]. Graph shows Interconnect power for minimum feature size range from 150 nm to 22 nm ............................ 21
3.2 Interconnect power breakdown [2]. The contributions from local and global signal lines dominate even clock signals .......... 21
3.3 Capacitances in DSM interconnect [3] ................................. 22
3.4 Components of total interconnect capacitance as technology scales [3] ................................................................. 22
3.5 A stage in a general purpose, pipelined, n-bit bus ................. 24
3.6 Class 5 and Class 6 transitions can be eliminated by detecting and preventing either of these two conditions. 29

3.7 A one-bit section on the \((n-1)\)-bit crosstalk analyzer. 30

3.8 Block diagram of the Encoder Module. 32

3.9 Block diagram of the Decoder Module. 33

3.10 An example of the 3-step process of crosstalk analysis, encoding, decoding. 35

3.11 Distribution of transition patterns. 40

3.12 Normalized delay and energy. 41

3.13 Benchmark-wise normalized energy. 43

3.14 Benchmark-wise normalized delay. 43

4.1 Delay as a function of inter-signal offset for a 1mm copper line with both neighbors switching in opposite direction. 46

4.2 Segment with maximum overlap to its adjacent neighbors. Transitions at A, will generate a response transition at B composed of the superposition of the direct response and any noise injected from adjacent segments. 47

4.3 Lumped RC model for the general coupled segment. Immediate neighbors are shown, farther segments are assumed grounded. 49

4.5 b-bit Multiple Phase Staggered Latch (MPSL) architecture for systematic offset switching in synchronous data transmission. 58

4.6 Statistical model constants for random, regular or burst data patterns in stacked MPSL[b]. (b) Double-level-sensitive latch with reset. 60
5.1 This shows the percentage change (from nominal) of the end-to-end line delay due to NBTI. Data is plotted for a 1 cm, optimally buffered interconnect line with an input signal 0-probability of 99.9%. .......................................................... 66

5.2 (a) A segmented bit line with inverting buffers. (b) Similar structure with non-inverting buffers. .......................................................... 69

5.3 Delay vs Duty Cycle profile for an inverter segmented bit line ... 70

5.4 Duty Cycle profiles of the I-Cache Address and Data buses for selected programs from the SPECINT2000 Benchmark suite ....... 76

5.5 Average Lifetimes for Raw bit Streams between the \( \mu \)-processor and the L1-Caches .......................................................... 77

5.6 Block schematic of the DCE + DCO scheme. An additional line is needed to transmit the global periodic signal. End-to-end data delivery remains totally unaltered by this scheme. ................. 78

5.7 This shows the lifetime of a typical 1 cm interconnect line implemented in a 45 nm process. The model in equation (5.7) is evaluated and compared to experimentally obtained values for \( 0 < \beta_s < 1 \). The maximum lifetime occurs around \( \beta_s = 0.5 \) for both plots. ... 80

5.8 Comparing Raw Address and Data flows to the DCE and DCE+DCO solutions for programs in the SPECINT2000 Benchmark suite. Average Lifetimes are normalized to AML. Uniformity values close to 1 indicate buses with better aging profiles .......... 82
List of Tables

3.1 Crosstalk classes .................................................. 26
3.2 Wire parameters .................................................... 39
3.3 CPC Codes ............................................................ 41
3.4 Codec overhead ....................................................... 42

4.1 RC Model parameters for segment drivers and metal ............ 49
4.2 Summary of Model Constants as function of $\eta$ ................. 53
4.3 Predicted freq speedup (OS/SS) for a 5-seg OS Bus ............ 56
Acknowledgments

To my family, especially my parents Eziokwu and Nnennaya; my brothers Ejike, Chidiebere and sister Ijeamaka; my inspiration, thank you for your sacrifice, your dedication and love. I also thank Professor Vijaykrishnan Narayanan for his dedicated mentorship, patience and support for the duration of my stay at Penn State. I also acknowledge and thank Professor Yuan Xie, Professor Jerzy Ruzyllo and Professor Mary Jane Irwin for their support and care. Thank you all to friends, students and well-wishers from whom I have learned so much and whose acquaintances I will forever cherish. A special thanks to Dr Madhu Mutyam, without whom this dissertation would not have been possible. Last but not least, Thank you God for the gift of life and wisdom.
Dedication

To my grandparents, parents and my family
Chapter 1

Introduction

Since the early 1970’s, Moore’s law has been used in the Integrated Circuits (IC) industry as a convenient guideline for projecting growth in IC complexity and performance over time. Moore’s law states that the number of transistors on ICs double every two years. This trend has been maintained primarily via the mechanism of dimension scaling where device and interconnect features are sized-down by a scale factor (S) and mapped to a new process technology. This results in ICs with smaller feature sizes and comes with advantages and disadvantages.

Sub-50 nm feature scaling to thinner gate oxides, shorter channel lengths, junction area, etc results in reduced energy, reduced power and faster transistors. Dimension scaling of the copper interconnect reduces wire cross-sectional area such that although ground capacitance is reduced, wire resistance as well as fringing and coupling capacitance increases and this leads to a net degradation of interconnect performance. The ITRS roadmap for Semiconductors, the industry/academic technology forecast envisages introduction of several new techniques, materials and technologies for mitigating the interconnect problem.

The scaling process for metal to smaller nodes leads to a rise in wire resis-
tance due to reduced wire widths. This is mitigated by allowing an increase in metal thickness thereby maintaining the same cross-sectional area. The resulting increase in metal/via aspect ratios exacerbates mutual coupling between signal bearing wires and aggravates signal integrity problems due to crosstalk. In addition, geometrical parameter variations due to sub-wavelength lithography, copper BEOL artifacts like Chemical Mechanical Polishing (CMP) and barrier thickness variation, are also exacerbated by feature scaling. Process methods for reducing the mutual coupling effect involve the introduction of low permittivity inter-metal dielectric materials in place of the traditional silicon dioxide. Novel aero gel materials, and even air gaps have been proposed as potential solutions, however these pose several integration challenges that have delayed their introduction for the foreseeable future barring major breakthroughs.

In addition to logic wiring, power and clock distribution, copper traces are used in multi-bit arrays such as buses, networks which form data pathways between functional blocks in on-chip systems. At an architectural level, we define Interconnect as the high level multi-bit structure which allows data words to move between functional elements. This is different from the typical nomenclature at the process level where interconnect refers strictly to the copper metal between any two vias or contacts, for our purposes we will refer to this as wire or metal trace.

The divergent trends in wire and device performance over process technology nodes has necessitated a shift in long on-chip interconnect design. Multi-segment Interconnect (MSI) also called refers to the insertion of buffers and repeaters in long on-chip wires to improve speed and performance. The introduction of signal drivers into data links allows for higher slew rates which primarily reduces on-current of any loading logic but it also improves end-to-end response time in long
Multi-bit MSI structures such as on-chip buses, router links, other dedicated data paths are composed of an array of minimally spaced, parallel bit lines arranged such that repeated wire segments are adjacent. Below the 50 nm node, the high aspect ratio wires form implicit parallel plate capacitors which creates high signal coupling and leads to noise pulse injections between neighboring segments. These noise pulse injections also referred to as crosstalk results in complex response signals that can cause large delay, jitter and skew variations.

Below the 50 nm node, on-chip interconnect is also susceptible to various aging related issues originating from device or wire degradation. Wire related aging mechanisms include electromigration and stress migration. These mass transport phenomena that can lead to open or shorts in signal bearing interconnect and cause overall circuit failure. In addition, device related aging mechanisms such as Time Dependent Dielectric Breakdown (TDDB), Hot Carrier Injection (HCI), and Negative Bias Temperature Instability (NBTI) can lead to degraded device reliability over time resulting in large interconnect performance drop. At elevated temperatures, due to driver design and typical interconnect data characteristics, Negative Bias Temperature Instability (NBTI) related aging leads to particularly non-negligible effects on MSI performance and reliability.

1.1 Problem Statement

The degree to which an interconnect retains its designed electrical and geometrical characteristics throughout its lifetime, from manufacture through operation to retirement is an indication of both process stability but also strongly dependent on design choices. Feature scaling is increasingly non-trivial due to severe process
related challenges and nano-scale physics constraints. The ever growing demand for consumer applications and the need for improved IC performance to meet this demand forces an equally growing focus on circuit design and architectural level solutions. This dissertation focuses on developing novel design methods and unique circuits with superior performance in the presence of sub-50 nm process, reliability and operational constraints.

1.1.1 Process induced variation

Process variation is a major force affecting IC design in sub-50 nm processes. Both devices and interconnect are victims of this force and must adapt or fail. In on-chip wires, various phenomena contribute to a matrix of variability issues, each with its own level of severity. The persistence of sub-wavelength lithography in sub 50 nm fabs comes at a cost, such that even with modern etching techniques, poor pattern registration is present leading to line edge roughness (LER). In addition, the transition to copper based interconnect technologies has introduced new sources of variation, such as metal erosion, dishing, dielectric/copper loss etc which are caused by the chemical mechanical polishing process. This leads to variation in the metal dimensions and consequent variation in electrical properties such as resistance, capacitance and even line resistivity. Also dynamic effects like clock skew, jitter, and setup time violations inevitably arise. In the end, designers are forced to adopt cautious design methods, and large margins are introduced into the clock period to counter variation, albeit at significant cost to performance. The use of optical proximity correction in DSM masks reduces patterning problems, but it far from eliminates them, besides it increases mask costs and is therefore typically more appropriate for large volume designs. Variations due to the CMP
process are dealt with primarily through dummy insertion, whereby the designer artificially adjusts the wire density around isolated wires. While this technique alleviates the problem, although at the cost of additional metal area, in general, it is not always possible to control material erosion. Therefore, parameter variation models are provided by vendors thereby allowing delay and power margins to be determined accordingly. As mentioned earlier, the large size of the margins in sub-50 nm technologies lead to overly pessimistic designs.

1.1.2 The Crosstalk effect

Crosstalk simply refers to the electrical noise generated by a signal carrying line on another line within close proximity. The key enabler of this effect is the coupling capacitance between the two lines. The magnitude of the effect is also proportional to the size of this capacitance. Although feature scaling in DSM ICs is also applied to the on-chip interconnect, shrinking along all dimensions leads to an exponential growth in resistance, as a result, it is standard practice to reduce wire widths and compensate by increasing the wire thickness. This leads to high aspect ratio metal lines, with neighboring wires forming a virtual parallel plate coupling capacitor. This problem is acute in long intermediate and global metal lines used for cross chip buses leading to delay variations which we now discuss. The delay of an interconnect line depends on its coupling capacitance to neighboring wires. Since the wires are electrically active, the coupling capacitance is not constant and consequently, neither is the line delay. In fact, the line delay depends on the directions of the signal transitions on any two neighboring wires, ranging from fastest to slowest when the transitions are in similar or opposite directions respectively.

Therefore, in on-chip buses where long metal lines are tightly spaced, crosstalk
induced delay is a major problem and depends on the data being transmitted on the bus. To deal with this problem, timing margin insertion was the typical practice but also various forms of shielding and data encoding have been proposed, these will be examined in more detail in chapter 3. Shielding is effective but requires large metal area to implement. the data encoding schemes also are effective but require extra metal lines, are less scalable. We propose three crosstalk aware techniques for on-chip buses, with the goal of minimizing the number of additional metal lines, chip area and maximizing end to end throughput.

1.1.3 Negative Bias Temperature Instability

Negative Bias Temperature Instability refers to the DSM aging phenomenon that affects pMOS devices whereby the device threshold voltage \( V_{th} \) drifts up, when the device junctions are negatively biased and under high temperature. This changing \( V_{th} \) causes variation in the device on current over the life of the chip leading to slow downs that can cause latching errors and circuit failure. NBTI has traditionally been a device related problem, however with the widespread use of repeater insertion and wire pipelining, interconnect delay is now dependent on the repeater
buffers as well as the metal RC factor. Therefore long term $V_{th}$ drift in long wire buses can lead to permanent data skew errors, latching errors.

1.2 Methods

The proposed solutions are presented both in principle and in implementation. The circuit design, performance analysis and results are detailed in the following chapters. First we explore crosstalk an its effect on delay and energy consumption, then focusing on efficient design and signal integrity we detail a novel offset switching scheme to replace traditional simultaneous switching. Focusing on reliability and aging, we explore data dependent interconnect aging and propose optimum solution schema and corresponding implementation circuits.
1.2.1 Variable Cycle Timing with Temporal Redundancy

For this project, we propose an extension to the variable cycle transmission (VCT) scheme presented in [4]. The primary aim is to reduce the overall energy, delay by eliminating the worst kind of crosstalk events, group 5 and 6 (G56). We intend to accomplish this by using a simple code based on temporal redundancy, in essence we will transform every G56 word into 2 lower group word for transmission. Since the delay penalty due to a G56 word is still greater than 2 lower words, we expect significant delay improvements.

1.2.2 Multiple Phase Staggered Latching (MPSL)

Multiple phase staggered Latching is a highly compact solution to the problem of eliminating the effects of G56 event in on-chip data transmissions. In this scheme, we use latch placement to control the occurrence of data transitions on the physical bus lines. We will show that end-to-end data delivery is not compromised at all either in terms of bit value or timing synchronization. This scheme does not require additional wires, area or power in its most basic form. It is therefore highly scalable with improved throughput.

1.2.3 Duty Cycle Equalization and Optimization

The change in pMOS threshold voltage $\Delta V_{th}$ under NBTI is described by models separated into two categories: static and dynamic NBTI models. Under normal operation, on-chip bus lines have non-static electric signals as such they can be appropriately modeled using the dynamic model. $\Delta V_{th}$ for the repeaters in a segmented bus depends on the probability that the signal at the input of a buffer is 0V, thereby reverse biasing its pull up pMOS. The figure 1.2 for the relationship
between $\Delta V_{th}$ and $\alpha$ the input duty cycle also known as, input zero probability. The graph is plotted for a pMOS device at 125 degC in a 65 nm process.

We propose a modified buffer design where the stress probability depends on both the input as well as an additional signal, a modulating signal. We also propose adding explicit redundancy to the pull up network, for instance a simple duplication, with modulating signal controlling their input probabilities in a complementary fashion. We can then modify the buffer delay degradation by our choice of this signal.

### 1.3 Contribution

The main contributions of this work are to introduce circuit and architecture level solutions as a means for achieving high speed, high reliability and high signal integrity in sub-50 nm on-chip interconnect; the introduction of three novel methods relying on VCT with temporal redundancy, Offset Switching, and Duty Cycle Tuning. The primary contribution of this research has been to develop temporal redundancy and offset switching as new ideas and sensibly combine existing ideas of bit-level bus-invert and transition load balancing to provide a systematic solution for sub-50 nm process, performance and signal integrity challenges.

- **Temporal Redundancy**: By eliminating worst-case crosstalk transitions via temporal redundancy, VCT achieves large energy and delay improvements using only minor increases in area and power.

- **Offset Threshold**: We introduce and derive an analytical model for offset threshold and demonstrate it to be the minimum delay necessary for nominal bit line delay in closely coupled long on chip interconnect.
• **Multiple Phase Staggered Latching**: A novel scheme is used to efficiently insert and remove offsets in hardware. No additional area or power or bit lines are necessary.

• **Optimum Aging**: MSI structures composed of bit lines with inverting repeaters are shown to have an optimum aging profile associated with an effective 50% duty cycle input.

• **Duty Cycle Tuning**: Performed using a rotating switch, Duty Cycle Equalization achieves a normalization effect across all the bit lines of an MSI. Duty Cycle Optimization, uses a global balanced signal to shift the effective duty cycle on all the MSI bit lines towards the 50% optimal aging point.

### 1.4 Outline

The dissertation is organized as follows. In chapter 3, the concept Variable Cycle Timing with Temporal Redundancy is introduced, the performance and costs are analyzed and results presented. In chapter 4, Multiple Phase Staggered Latching is introduced. We introduce the concept of offset threshold and a reliable method to obtain it analytically. The general MPSL architecture and the special case MPSL[2] are presented. A comparative analysis of the the MPSL structure vs simultaneous switching including simulation results is presented. In chapter 5, aging of multi-bit bus structures due to Negative Bias Temperature Instability is explored. Using realistic Address bus and Data bus trace data from benchmark programs, we show the aging effect due to NBTI over extended periods of operation. Duty cycle equalization and duty cycle optimization are presented as cost effective means of compensation which result in increased bus lifetimes and aging resilience.
In chapter 6, we present concluding remarks and propose future work and future areas of exploration.
Chapter 2

Background and Related Work

2.1 Introduction

Reliability in on-chip interconnect is not a new area of research. Many studies have explored the various issues involved and have proposed analytical models, research tools, and indeed circuit level solutions. However, current trends in IC feature scaling present new challenges that must be addressed if we must meet future performance demands [5, 6]. Crosstalk, for instance, has been of concern in the subfields of printed circuit boards (PCB) and communication cable design for many years, and is probably the most studied of all wire reliability issues. However, due to the nature of integrated circuits, process chemistries and material integration challenges, existing PCB and large scale solutions are simply inapplicable. Various authors have proposed solutions to on-chip reliability issues, crosstalk to a large extent, interconnect process variation, and interconnect NBTI to a lesser degree. We present a quick review of some pertinent works in this chapter.
Table 2.1. Crosstalk classes

<table>
<thead>
<tr>
<th>Crosstalk Class</th>
<th>Relative Delay on middle wire</th>
<th>Transition Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>- - - - - ↑, ↑ - - - - ↓, - - - - ↑, ↑↑ ↓↑ ↓↓</td>
</tr>
<tr>
<td>2</td>
<td>$C_LR_T$</td>
<td>↑↑↑↑↓↓↓↓</td>
</tr>
<tr>
<td>3</td>
<td>$C_LR_T(1 + \lambda)$</td>
<td>-↑↑, ↑↑↑↑↓↓↓↓</td>
</tr>
<tr>
<td>4</td>
<td>$C_LR_T(1 + 2\lambda)$</td>
<td>-↑↑↑, -↓↓↓↑↑↓↓↓↓</td>
</tr>
<tr>
<td>5</td>
<td>$C_LR_T(1 + 3\lambda)$</td>
<td>-↓↓↓, ↓↓↓↑↑↓↓↓↓</td>
</tr>
<tr>
<td>6</td>
<td>$C_LR_T(1 + 4\lambda)$</td>
<td>↓↓↓↓↑↑↑↑</td>
</tr>
</tbody>
</table>

Figure 2.1: Relative delay for different transition patterns on a 3-wire test structure

2.2 Crosstalk tolerance in on-chip buses

Wire proximity and increased wire aspect ratios are the main reason wire coupling capacitance is expected to continue dominating ground capacitance in sub-100 nm IC technologies [6]. Long wires in buses are particularly susceptible since they are typically minimum spaced and run parallel to each other over long distances. The actual amount of noise generated on a victim due to an aggressor depends on coupling capacitance between the two wires. This capacitance depends on the nature of the electrical signal on both wires, it varies depending on whether they are both constant, switching in the same direction, in opposite directions or only one of them transitions. This means the effective RC delay including buffers and crosstalk will depend on the transitions. In [7], the author presented analytical models for bus delay in terms of the ratio, coupling capacitance vs the ground capacitance. See figure 2.1, if $C_I$ and $C_L$ are the coupling and ground capacitances respectively, then $\lambda = \frac{C_I}{C_L}$ while the arrows indicate the transitions 0 to 1 or 1 to 0.

In figure 2.2 below shows a traditional n-bit bus segment terminated at both ends by flip flops and clocked with a signal of period $T_p$. $T_p$ is chosen such that
during the worst case delay, the edge triggered latches do not fail.

\[ T_p = t_d + t_{flop} + t_{setup} + t_{cycle\_jitter} + t_{dsm\_margin} \]  \hspace{1cm} (2.1)

where \( t_d \) is the metal RC delay plus repeaters, \( t_{flop} \) and \( t_{setup} \) are the flip flop delay and setup time respectively, and \( t_{cycle} \) is the clock cycle jitter. \( t_{dsm\_margin} \) is the additional timing margin included in the period to allow tolerance for crosstalk and other static and dynamic variations. Requiring \( t_{dsm\_margin} \) to be the worst case in DSM ICs leads to vast losses in performance because the gap between the worst case and average case is quite considerable. To reduce the size of the margin, many techniques have been proposed for the elimination of classes 5 and 6 transition patterns. This will eliminate the biggest contributor to crosstalk induced delay and allow for smaller \( T_p \).

Bus encoding techniques to eliminate crosstalk classes 5 and 6 have been proposed in [8, 9]. Note that the technique proposed in [9] can also eliminate part of
the crosstalk class 4. From Table above, it is clear that as a result of eliminating crosstalk classes 5 and 6, the worst-case crosstalk delay becomes $C_L R_T (1 + 2\lambda)$. Though the worst-case delay can be reduced by 50%, these techniques are difficult to implement and have large area overhead (for a 32-bit data, the technique in [9] requires 46 wires without memory and 40 wires with memory and the technique in [8] requires 52 wires). Similar to the ideas of [8, 9], a new coding technique is proposed in [10] to obtain 50% reduction in delay along with 10% reduction in energy, but it also requires 48 wires to encode 32-bit data. A bus encoding technique to minimize power consumption and eliminate crosstalk classes 5 and 6 is proposed in [11], which requires 55 wires to encode 32-bit data. Coupling-driven bus encoding technique [12] reduces power consumption by 30% on an average, but it may not eliminate crosstalk class 4 and above and hence it is not much advantageous from the delay perspective. Another bus encoding technique to minimize both energy and delay is proposed in [13], which can eliminate only crosstalk classes 4 and 6 (but not crosstalk class 5) so that the worst-case delay is still and it requires 55 wires to encode 32-bit data. Odd/even bus invert technique is proposed in [14] to minimize coupling energy. With half cycle delay of one of the adjacent wires, it can eliminate opposite transitions on adjacent wires and achieve 30% power savings. Though this technique is energy efficient, from the delay perspective it is not advantageous because the half cycle delay can result in crosstalk class 4 patterns such as $-\uparrow -$ and $-\downarrow -$. As a result, the net delay for receiving the data becomes $C_L R_T (2 + 4\lambda)$, which is more than that of crosstalk class 6. Another technique to minimize power consumption due to coupling transitions is proposed in [15], but it may not be easy to implement due to its complex codec circuitry. A technique close to the topic of this work is proposed in [4], where instead of considering the worst-case delay of $C_L R_T (1 + 4\lambda)$, the authors considered variable delay which is
determined based on the data present on the bus and next data to be transmitted. Based on the crosstalk class of the next data w.r.t. the present data on the bus, the number of cycles required to transmit the next data is controlled dynamically and hence obtain significant performance benefits. This technique requires one extra wire, which acts as a shield wire between the actual data and the ready\textsuperscript{1} signal. Since the data is transmitted as it is without any coding, this technique does not provide any energy benefits. Contrast to the above mentioned works, here we propose three techniques all of which eliminate crosstalk classes 5 and 6. VCTR accomplishes this with the help of temporal redundancy and uses variable delay for data transmission so that both delay and energy minimization can be achieved. Unlike the technique in [4], where for crosstalk class 6 it takes the delay of \(4C_LR_T(1 + \lambda)\), our technique takes the delay of \(3C_LR_T(1 + \lambda)\) so that we can even get better performance benefits over the technique proposed in [4]. BLFC also accomplishes this while using no extra lines, and reducing the number of dummy bits inserted into the data stream. Finally, MPSL accomplishes this by staggering the input and output latches, using no extra lines, and virtually no increase in area, power, or use of dummy bits. The only existing work similar to the MPSL solution was proposed in [16] and it involved staggered firing of the drive buffers with 5-10% offsets. Process variation levels in both interconnect, device parameters exceed those offsets and therefore renders such a solution ineffective.

### 2.3 Interconnect Variation

Process variation in on-chip bus wires leads to WID and D2D variation in the RC delay characteristics of a the line. In buses containing multiple lines, this exact-

\textsuperscript{1}the authors considered 33-bit data with 32-bit data and one ready signal
erbates data skew problems, setup time violations if not appropriately accounted for. Most existing works such as [17, 18, 19] propose models for estimating the amount of variation margins need to be inserted into the clock. However, in [20] a source synchronous bus is proposed as a variation tolerant interconnect capable of improving interconnect variation susceptibility in the long wire line by up to 90%. This solution requires the use of a single N-flop, which is a complex, multiple clock flip-flop, instead of the traditional wire pipeline latches with an aim of also reducing overall power and energy. This work however relies heavily on multiple delay-skewed clock signals, but assumes no interconnect variation on those clock distribution wires. This may be too optimistic. We propose a dynamic pipeline scheme that uses post-silicon configuration and reconfiguration when necessary to ensure variation tolerance. The clock signal and a configuration bits are the only additional signals needed.

2.4 Negative Bias Temperature Instability

The development of $V_{th}$ drift models for static and dynamic circuit operation, and an analysis of various design compensation schemes were discussed in [1, 21]. These compensation schemes such as gate sizing, Vdd tuning and duty cycle control have been used to improve device lifetime in [22, 23, 24, 25]. Long, segmented interconnect is a unique structure in that it is composed of a cascade of inverting buffers whose input signal characteristics are exactly complementary to those adjacent. This means that a primary input signal with 50% duty, will lead to uniform degradation across the chain of inverters. Real bus lines have a skewed signal probabilities, this in turn results in a skewed degradation. In Kumar et.al [25], bit flipping was used to improve NBTI lifetime in SRAMs. This is of limited utility
in cascaded inverters since it imposes a costly bound on the least achievable level of NBTI degradation. Our proposed scheme modifies the buffer design to allow inverter chains with NBTI degradation levels below those bounds.
Chapter 3

Variable Cycle Timing with Temporal Redundancy

3.1 Introduction

As VLSI fabrication technologies are scaled down to the deep sub-micron region, the inter-wire capacitance \( (C_c) \) becomes significant compared to the wire-to-substrate capacitance \( (C_g) \). As is the dominant capacitance in deep sub-micron era, it has two significant effects: large propagation delay due to opposite transitions on adjacent wires [26, 27, 28] and power dissipation associated with driving on-chip buses [27]. There are techniques to minimize the effects of opposite transitions on adjacent wires and/or power consumption due to capacitive coupling [8, 13, 12, 11, 29, 15, 30, 7, 10, 9, 14]. Many of the techniques use spatial redundancy to minimize power or delay and hence have large area overhead. For example, crosstalk avoidance codes [9] eliminate opposite transitions on adjacent wires completely but require 46 wires to encode 32-bit data and a coding technique proposed in [10] is both area and energy-efficient and eliminates opposite transitions on adjacent wires completely but it requires 48 wires for encoding
32-bit data. In this work, we exploit the *variable cycle transmission* technique [4] to apply *temporal redundancy* in the process of minimizing both delay and energy for on-chip data transmission. Our encoding scheme requires two extra wires to encode *any* bit-width data and achieves 31%(30%) of delay improvement along with 13%(9%) of energy savings over the base case for data transmission on address (data) bus [31].

### 3.2 Background

Power dissipation has emerged as a major consideration in high-performance integrated circuit design. Dynamic Power, the contribution to on-chip power resulting from periodic switching of internal capacitances is by far the primary component [32]. Total dynamic power often approximated by equation 3.1, is composed of around 51% Interconnect power, 34% Gate power, and around 15% Diffusion power—with global clock included.

\[
P = \sum AF_j \cdot C_j \cdot V^2 \cdot f
\]  

(3.1)

Where \( AF_j \) and \( C_j \) in equation 3.1 are the Activity Factor (AF) and Capacitance (C) for the \( j \)-th signal. \( V \) is the Supply Voltage and \( f \) is the Clock Frequency. The contribution of interconnect to total dynamic power is expected to grow with technology scaling to between 65% - 80% within the decade, dominating both gate and diffusion power combined [2].
Interconnect power can be approximated using equation 3.1 with $C_j$ as the lumped capacitance of the wire. For long on-chip metal wires, this capacitance has two components: the metal-to-substrate capacitance $C_g$ and the metal-to-metal capacitance $C_c$ as shown in Figure 3.3. $C_c$ also referred to as the coupling capacitance results from the capacitive effect between neighboring metal lines due to the reduced line pitch in sub-100 nm technologies. In order to control line resistance with scaling, reducing line pitch requires increased line thickness which in turn

Figure 3.2: Interconnect power breakdown [2]. The contributions from local and global signal lines dominate even clock signals

Figure 3.1: Interconnect power grows to 65% - 80% within the decade under optimistic scaling [2]. Graph shows Interconnect power for minimum feature size range from 150 nm to 22 nm

### 3.2.1 Interconnect Power

Interconnect power can be approximated using equation 3.1 with $C_j$ as the lumped capacitance of the wire. For long on-chip metal wires, this capacitance has two components: the metal-to-substrate capacitance $C_g$ and the metal-to-metal capacitance $C_c$ as shown in Figure 3.3. $C_c$ also referred to as the coupling capacitance results from the capacitive effect between neighboring metal lines due to the reduced line pitch in sub-100 nm technologies. In order to control line resistance with scaling, reducing line pitch requires increased line thickness which in turn
leads to increased line aspect ratios and increased coupling capacitance. Therefore as on-chip features scale below the 100 nm node, $C_c$ is expected to rapidly increase and quickly dominate $C_g$. See Figure 3.4.

Primarily, $C_c$ affects interconnect performance through the Miller Effect. The Miller Effect describes the resultant capacitance seen by an interconnect driver due to the signal transitions on neighboring wires. Opposite transitions on neighboring lines result in large values of $C_c$ while similar transitions result in small values of $C_c$. This transition/data dependency of $C_c$ results in a similar dependency of primary interconnect metrics such as power, energy and delay all of which are functions of the line capacitance.

Figure 3.3: Capacitances in DSM interconnect [3]

Figure 3.4: Components of total interconnect capacitance as technology scales [3]
3.2.2 Buses in Deep Sub-Micron Technologies

On-chip buses are architectural level interconnect used to provide a communication link between physically separated points on a chip. A typical $n$-bit bus is composed of $n$ parallel, metal wires of considerable length, minimally spaced, located at either the intermediate or global metal in a multi-level interconnect structure. Each line is often segmented by repeaters interspersed at pre-defined locations to provide sufficient current for switching the line at high frequencies. Very long cross-chip buses may also be partitioned into several stages with latches or flip-flops used to separates the different clock domains akin to a clocked pipeline.

Traditionally, for an $n$-bit bus $B$, with individual bit-lines $b_i$, where $1 \leq i \leq n$, the bus power $P_{bus}$ dissipation is defined as:

$$ P_{bus} = \sum_{i=1}^{n} A F_i \cdot C_i \cdot V^2 \cdot f $$  \hfill (3.2)

and the energy consumption $E_{bus}$ is likewise defined as:

$$ E_{bus} = \sum_{i=1}^{n} \frac{1}{2} C_i \cdot V^2 $$  \hfill (3.3)

and the bus delay which is the inverse of the maximum bus frequency is defined as a function of the individual bit-line delays $d_i$ as

$$ D_{bus} = \text{Max}\{d_i\}_{i=1}^{n} $$  \hfill (3.4)

Where $d_i$ can be taken as the Elmore Delay, $d_i = R_i C_i$ for each bit-line.

These traditional models which rely on fixed values of $C_i$ are too pessimistic in
the DSM regime since the dominant part of $C_i$ moves from $C_g$ to $C_c$ which is data dependent component. The result is $P_{bus}$, $E_{bus}$, and $D_{bus}$ that vary from cycle to cycle depending on the data stream flowing through the bus. New analytical models that accurately account for DSM effects will be key to realizing maximum performance in DSM buses for delay, energy and power.

### 3.3 Analytical Models For Delay and Energy Consumption

In order to measure the actual effects of inter-wire capacitance in deep sub-micron technologies, analytical models for delay and energy consumption in deep sub-micron buses have been proposed in [30, 7, 33]. Throughout the treatise we consider these analytical models. Let $d_t$ be an $n$-bit data present on the bus at time $t$. By
considering $\lambda = (C_I/C_L)^1$, the propagation delay of the RC circuit, for transmitting another $n$-bit data $d_{t+1}$, can be calculated by [7]

$$T(d_t, d_{t+1}) = \max\{T_k(d_t, d_{t+1}) \mid 1 \leq k \leq n\} \quad (3.5)$$

where

$$T_k(d_t, d_{t+1}) = \frac{C_L R_T}{C_I} \begin{cases} ((1 + \lambda)\Delta_1 - \lambda\Delta_2)\Delta_1 & \text{if } k = 1, \\ ((1 + 2\lambda)\Delta_k - \lambda(\Delta_{k-1} + \Delta_{k+1}))\Delta_k & \text{if } 1 < k < n \text{ is even} \\ ((1 + \lambda)\Delta_n - \lambda\Delta_{n-1})\Delta_n & \text{if } k = n \end{cases} \quad (3.6)$$

where $R_T$ is the total resistance and $\Delta_k = d_{t+1}^k - d_t^k$. Similarly, the total energy (due to self and coupling transitions) consumed during the transition from $d_t$ to $d_{t+1}$ is given by [30, 33]

$$E(d_t, d_{t+1}) = \sum_{k=1}^{n} E_k(d_t, d_{t+1}) \quad (3.7)$$

where

$$E_k(d_t, d_{t+1}) = \begin{cases} C_L((1 + \lambda)\Delta_1 - \lambda\Delta_2)d_{t+1}^1, & \text{if } k = 1 \\ C_L((1 + 2\lambda)\Delta_k - \lambda(\Delta_{k-1} + \Delta_{k+1}))d_{t+1}^k, & \text{if } 1 < k < n \\ C_L((1 + \lambda)\Delta_n - \lambda\Delta_{n-1})d_{t+1}^n, & \text{if } k = n \end{cases} \quad (3.8)$$

For example, if $d_t = 010$ and $d_{t+1} = 101$, then $T(d_t, d_{t+1}) = C_L R_T (1 + 4\lambda)$ and $E(d_t, d_{t+1}) = C_L (2 + 4\lambda)$. On the other hand, if $d_t = 000$ and $d_{t+1} = 111$, then

---

$^1$C$_I$ refers to the inter-metal capacitance, C$_c$ was used in chapter 3.1. C$_L$ is the metal-to-ground capacitance called (C$_g$) in chapter 3.1.
Table 3.1: Crosstalk classes

<table>
<thead>
<tr>
<th>Crosstalk Class</th>
<th>Relative Delay on middle wire</th>
<th>Transition Patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>- -,-,- ↑↑↓↓, - -,-,- ↑↑↓↓, ↑↑↓↓</td>
</tr>
<tr>
<td>2</td>
<td>$C_LR_T$</td>
<td>↑↑ ↑↑ ↓↓ ↓↓ ↓↓</td>
</tr>
<tr>
<td>3</td>
<td>$C_LR_T(1 + \lambda)$</td>
<td>↑↑ ↑↑ ↑↑ ↓↓ ↓↓</td>
</tr>
<tr>
<td>4</td>
<td>$C_LR_T(1 + 2\lambda)$</td>
<td>↑↑ ↑↑ ↑↑ ↑↑ ↓↓ ↓↓</td>
</tr>
<tr>
<td>5</td>
<td>$C_LR_T(1 + 3\lambda)$</td>
<td>↑↑ ↑↑ ↑↑ ↑↑ ↓↑ ↓↓</td>
</tr>
<tr>
<td>6</td>
<td>$C_LR_T(1 + 4\lambda)$</td>
<td>↓↓ ↑↑ ↑↑ ↑↑</td>
</tr>
</tbody>
</table>

$T(d_t, d_{t+1}) = C_LR_T$ and $E(d_t, d_{t+1}) = 3C_L$. From these two examples (and hence from the above two equations), it is clear that the transition patterns, such as $0 \rightarrow 1$ (or ↑), $1 \rightarrow 0$ (or ↓), $0 \rightarrow 0$ (or −), and $1 \rightarrow 1$ (or −), of data not only dictate the propagation delay but also the energy consumption.

Since the propagation delay is determined by the transition pattern of the data, transition patterns are classified into six different classes [4, 7] based on the relative delay a wire w.r.t. its adjacent wires, which we call as crosstalk classes. Table 3.1 shows different crosstalk classes. From this table, it is clear that the worst-case crosstalk delay for transmitting data items, without any coding, is $C_LR_T(1 + 4\lambda)$, where $\lambda = \frac{C_I}{C_L}$.

### 3.4 Related Work on Bus Coding, Static and Dynamic Shielding

Bus encoding techniques to eliminate crosstalk classes 5 and 6 have been proposed in [8, 9]. Note that the technique proposed in [9] can also eliminate part of the crosstalk class 4. From Table 1, it is clear that as a result of eliminating crosstalk classes 5 and 6, the worst-case crosstalk delay becomes $C_LR_T(1 + 2\lambda)$. Though the
worst-case delay can be reduced by 50%, these techniques are difficult to implement and have large area overhead (for a 32-bit data, the technique in [9] requires 46 wires without memory and 40 wires with memory and the technique in [8] requires 52 wires). Similar to the ideas of [8, 9], a new coding technique is proposed in [10] to obtain 50% reduction in delay along with 10% reduction in energy, but it also requires 48 wires to encode 32-bit data. A bus encoding technique to minimize power consumption and eliminate crosstalk classes 5 and 6 is proposed in [11], which requires 55 wires to encode 32-bit data. Coupling-driven bus encoding technique [12] reduces power consumption by 30% on an average, but it may not eliminate crosstalk class 4 and above and hence it is not much advantageous from the delay perspective. Another bus encoding technique to minimize both energy and delay is proposed in [13], which can eliminate only crosstalk classes 4 and 6 (but not crosstalk class 5) so that the worst-case delay is still and it requires 55 wires to encode 32-bit data. Odd/even bus invert technique is proposed in [14] to minimize coupling energy. With half cycle delay of one of the adjacent wires, it can eliminate opposite transitions on adjacent wires and achieve 30% power savings. Though this technique is energy efficient, from the delay perspective it is not advantageous because the half cycle delay can result in crosstalk class 4 patterns such as $- \uparrow -$ and $- \downarrow -$. As a result, the net delay for receiving the data becomes $C_L R_T (2 + 4 \lambda)$, which is more than that of crosstalk class 6. Another technique to minimize power consumption due to coupling transitions is proposed in [15], but it may not be easy to implement due to its complex codec circuitry. A technique close to the topic of this work is proposed in [4], where instead of considering the worst-case delay of $C_L R_T (1 + 4 \lambda)$, the authors considered variable delay which is determined based on the data present on the bus and next data to be transmitted. Based on the crosstalk class of the next data w.r.t. the present data on the bus,
the number of cycles required to transmit the next data is controlled dynamically and hence obtain significant performance benefits. This technique requires one extra wire, which acts as a shield wire between the actual data and the ready\textsuperscript{2} signal. Since the data is transmitted as it is without any coding, this technique does not provide any energy benefits. Contrast to the above mentioned works, here we propose a technique which eliminates crosstalk classes 5 and 6 with the help of temporal redundancy and uses variable delay for data transmission so that both delay and energy minimization can be achieved. Unlike the technique in [4], where for crosstalk class 6 it takes the delay of $4C_L R_T (1 + \lambda)$, our technique takes the delay of $3C_L R_T (1 + \lambda)$ so that we can even get better performance benefits over the technique proposed in [4].

3.5 Our Approach

Basic idea behind our approach is to analyze the crosstalk class of next data w.r.t. the data present on the bus and based on the crosstalk class we either transmit the original data or encode the data into two new data items and transmit the encoded data. In either case, we use the necessary delay for data transmission.

3.6 Area and Energy Efficient Crosstalk Analyzer

From Chapter 2, we know that the crosstalk class of next $n$-bit data w.r.t. the present data on the bus is determined by the maximum delay among all the wires $k$, $1 \leq k \leq n$. So, in order to know the crosstalk class, first we have to find the

\textsuperscript{2}the authors considered 33-bit data with 32-bit data and one ready signal
Figure 3.6: Class 5 and Class 6 transitions can be eliminated by detecting and preventing either of these two conditions

delay of each of the $n$ wires. In [4], a crosstalk analyzer is designed in such a way that whenever a next data is received, the analyzer determines the crosstalk class of the next data by comparing it with the present data on the bus. This process incurs extra hardware and consumes more energy as none of the computed data, in the process of finding a crosstalk class, is reused in the next stage. Generally, in the process of finding a crosstalk class, the analyzer checks whether or not there is any opposite transition on the adjacent wires (this is needed for crosstalk classes 4, 5, and 6).

If $d_1$ is the present data and $d_2$ is the next data, then to know whether or not a wire $k$, $1 \leq k \leq n$, has an opposite transition w.r.t. wire $k + 1$, it is enough to take the logical AND of $d_1[k] \oplus d_1[k + 1]$, $d_2[k] \oplus d_2[k + 1]$, and $d_1[k] \oplus d_2[k]$. In this process, we can compute the values of adjacent bits of $d_2$ only once but use them twice (i.e., for finding crosstalk classes of $d_2$ w.r.t. $d_1$ and $d_3$ w.r.t. $d_2$).

As a result, the hardware and the energy consumption for the crosstalk analyzer is significantly reduced compared to the similar circuit given in [4]. We obtain the hardware reduction at the cost of two $(n - 1)$-bit registers, i.e., $XORP$ and $XORN$, to store the $XOR$ values of adjacent bits of present $n$-bit data and next $n$-bit data, respectively. Whenever, a next data $d_2$ is received, we only compute $XORN$ and move the previous $XORN$ (that of $d_1$) to $XORP$ with the possible exception when the next data $d_2$ is in crosstalk class 5 or 6 w.r.t. the present
data $d_1$. Since temporal redundancy is used for crosstalk classes 5 and 6, in such situations, XORP gets the XOR values of adjacent bits of $d''$ (defined in the next subsection). Our design requires 35 wires for transmitting a 33-bit data, which consists of 32-bit actual data and one ready signal. The additional wires are: one temporal_redundancy signal for crosstalk class 5 or 6 and one shield wire between the ready signal and the actual data to prevent opposite transitions. There is no need for a shield wire between the temporal_redundancy signal and the ready signal as the temporal_redundancy signal makes a 0→1 transition only when the ready signal makes 0→1 transition and it makes a 1→0 transition when the ready signal makes a 1→1 transition.
3.7 Variable Cycle Transmission with Temporal Redundancy

The encoding and decoding modules are the key additional components needed to implement the VCTR technique.

3.7.1 Encoder

We now present our bus encoding technique, i.e., variable cycle transmission with temporal redundancy (VCTR technique). In order to facilitate variable cycle transmission, we use two different clocks with periods $C_L R_T(1 + \lambda)$ and $2C_L R_T(1 + \lambda)$.

Whenever a next data $d_{t+1}$ is received, our crosstalk analyzer determines the crosstalk class of $d_{t+1}$ w.r.t. the present data $d_t$ on the bus. If the crosstalk class $d_{t+1}$ of w.r.t. $d_t$ is from the set $\{1,2,3,4\}$, $d_{t+1}$ is transmitted with a delay of $C_L R_T(1 + \lambda)$ (or $2C_L R_T(1 + \lambda)$) if the crosstalk class is $\leq 3$ (or 4) and the ready signal is set and the temporal redundancy signal is reset. If the crosstalk class is either 5 or 6, unlike delaying the transmission by the required number cycles (as proposed in [4]), we encode $d_{t+1}$ into two new data items $d'_{t+1}$ and $d''_{t+1}$ (using the coding technique as given in Algorithm 1) in such a way that the crosstalk class of $d'_{t+1}$ w.r.t. $d_t$ is from the set $1,2,3,4$ and that of $d''_{t+1}$ w.r.t. $d_{t+1}$ is 3 (formal proof is given in the next section).

Note that the encoder (given in Algorithm 1) takes different conditions for the boundary values, i.e., when $i = 1$, it checks only

$$(XORP[2]\&XORN[2]\&(d_t[1] \oplus d_{t+1}[1]))$$

and when $i = n$, it checks $$(XORP[n]\&XORN[n]\&(d_t[n-1] \oplus d_{t+1}[n-1]))$$ . Thus, we transmit $d'_{t+1}$ with a delay of $2C_L R_T(1 + \lambda)$ and $d''_{t+1}$ with a de-
lay of $C_L R_T(1 + \lambda)$. As a result, the net delay for transmitting two new data items becomes $3C_L R_T(1 + \lambda)$. During the transmission of $d'_{t+1}$, we reset both temporal redundancy signal and ready signal, whereas for $d''_{t+1}$ transmission, we set both temporal redundancy signal and ready signal. Clearly, for both crosstalk classes 5 and 6, our approach takes $3C_L R_T(1 + \lambda)$ cycles whereas the technique given in [4] takes $3C_L R_T(1 + \lambda)$ and $4C_L R_T(1 + \lambda)$ cycles, respectively. So, our approach not only provides energy benefits (due to elimination of coupling transitions of crosstalk classes 5 and 6) but also performance benefits.

\begin{algorithm}
\textbf{Require:} present data $d_t$ and $d_{t+1}$
\textbf{Ensure:} encoded data items $d''_{t+1}$ and $d''_{t+1}$
\begin{algorithmic}
\If{crosstalk\_class $\geq 5$}
\State $d'_{t+1} \leftarrow d_{t+1}$; $d''_{t+1} \leftarrow d_{t+1}$
\For{$i = n$ to 0}
\If{$(XORP[i + 1]$ and $XORN[i + 1]$ and $(d_t[i] \oplus d_{t+1}[i])$) \texttt{and} $(XORN[i]$ and $(d_t[i - 1] \oplus d_{t+1}[i - 1]))$}
\State $d'_{t+1}[i] \leftarrow 1$; $d''_{t+1}[i] \leftarrow 0$
\EndIf
\EndFor
\EndIf
\end{algorithmic}
\end{algorithm}
3.7.2 Decoder

Decoding the data at the receiver end is a simple process (as shown in Algorithm 2). At the receiver end, we store the value on the bus in a $n$-bit register $R_1$ when the ready signal is set and the temporal_redundancy signal is reset and the value on the bus is stored in another $n$-bit register when both ready signal and temporal_redundancy signal are reset.

We use the values of these two registers to decode the actual data. When the ready signal is set and the temporal_redundancy signal is reset, there is no need of decoding (because, this situation arises only when the transmitted data is in crosstalk class from the set $\{1,2,3,4\}$), so we consider whatever the data present on the bus as the next data. On the other hand, if both ready signal and temporal_redundancy signal are set, we decode the data using the decoding algorithm given in Algorithm 2. Our decoder is a simple bit-wise comparison circuit so the delay due to the decoder can be ignored.
**Algorithm 2 Decoder pseudo-code**

Require: present data $z_{t+1}$ and previous data $R_1$ and $R_2$

Ensure: decoded data $d_{t+1}$

1: if ready then
2: \hspace{1em} if temporal redundancy then
3: \hspace{2em} for $i = n$ to 0 do
4: \hspace{3em} if $(R_2[i] == z_{t+1}[i])$ then
5: \hspace{4em} $d_{t+1}[i] \leftarrow z_{t+1}[i]$
6: \hspace{3em} else
7: \hspace{4em} $d_{t+1}[i] \leftarrow \sim R_1[i]$
8: \hspace{2em} end if
9: \hspace{1em} end for
10: else
11: $d_{t+1}[i] \leftarrow z_{t+1}$
12: end if
13: end if

### 3.8 A Simple Example

In this section, a simple example is used to demonstrate the functionality of the VCTR technique. This technique can be easily scaled to much wider buses with a linear increase in area and power. In this example, let the current data on an $n$-bit bus be $A$ and let the next data word to be sent be $B$. If $B$ passes through the crosstalk analyzer and no other locations contain class 5 or 6 transitions except those shown in figure 3.10. A flag is raised and the Encoder replaces the new data word $B$ with two new data words $B_1$ and $B_2$, these two words are generated such that all bit positions except at those with class 5 or 6 transitions are equal to corresponding bit locations on the discarded data word $B$. At these positions, a "1" and a "0" are intentionally inserted in the new data words $B_1$ and $B_2$ respectively. These basically serve as markers that allow the decoder to detect those locations. $B_1$ and $B_2$ are then sent over the bus using VCT with a temporal redundancy bit set. Note that the actual values at the marker positions are exact inversions of the
3.9 Correctness of Our Approach

We now prove that our encoding technique works correctly for any data and minimizes the delay and the energy consumption. Let $CC(d_i, d_j)$ denote the crosstalk class of $d_i$ w.r.t $d_j$, $T(d_i, d_j)$ denote the delay for transmitting $d_j$ after $d_i$, and $E(d_i, d_j)$ denote the energy consumption during the transition from $d_i$ to $d_j$. 
Theorem 1. For any pair of n-bit data $d_t$ and $d_{t+1}$, if $CC(d_{t+1}, d_t) \geq 5$, then $d_{t+1}$ can be encoded a $d'_t$ and $d''_t$ such that

- $CC(d'_t, d_t) < 5$ and $CC(d''_t, d'_t) = 3$
- $T(d_t, d'_t) + T(d'_t, d''_t) \leq T(d_t, d^{t+1})$
- $E(d_t, d'_t) + D(d'_t, d''_t) \leq E(d_t, d^{t+1})$

Proof. To prove $CC(d'_t, d_t) < 5$:

Let $tp(a, b)$ be the transition pattern over $\{\uparrow, \downarrow, -\}$ of length $n$ when an $n$-bit data $a$ is transmitted after another $n$-bit data $b$. Assume that $tp(d_{t+1}, d_t) = p_1 \ldots p_n$, where $p_i \in \{\uparrow, \downarrow, -\}$, and $1 \leq i \leq n$. Since $CC(d_{t+1}, d_t) \geq 5$, there must be at least wires which have opposite transitions w.r.t. one or both of their adjacent wires. For simplicity, assume that there are exactly wires, i.e., wire $i$ and wire $i + 1$, which have opposite transitions w.r.t. each other. Note that the proof can be easily extended to any number of wires having opposite transitions w.r.t. one or both of their adjacent wires. So, there is no pattern of type $\uparrow \downarrow$ or $\downarrow \uparrow$ in both $p_1 \ldots p_i$ and $p_{i+1} \ldots p_n$, and $tp(d'_{t+1}, d_t) = p'_1 \ldots p'_n$ and $tp(d''_{t+1}, d'_t) = p''_1 \ldots p''_n$. Since there is no pattern of type $\uparrow \downarrow$ or $\downarrow \uparrow$ in both $p_1 \ldots p_i$ and $p_{i+1} \ldots p_n$, according to the encoding technique, for $1 \leq j \notin \{i, i + 1\}$, $p'_j = p_j$ and $p''_j = -$, and for $j \in \{i, i + 1\}$, $p'_j = \downarrow$, and if $p_j = \uparrow$, $p'_j = p_j$ and if $p_j = \downarrow$, $p'_j = -$. Hence,

$$tp(d'_{t+1}, d_t) = p'_1 \ldots p'_{i-1} p'_i p'_{i+1} p'_{i+2} \ldots p'_n$$

$$= p_1 \ldots p_{i-1} p'_i p'_{i+1} p'_{i+2} \ldots p_n \quad (3.9)$$

We know that either $p_i' = p_i$ or $p'_{i+1} = p_{i+1}$. In either case, $p_1 \ldots p_{i-1} p'_i p'_{i+1} p_{i+2} \ldots p_n$
does not have a pattern of type $\uparrow\downarrow$ or $\downarrow\uparrow$. Hence,

$$CC(d'_{t+1}, d_t) < 5. \quad (3.10)$$

To prove $CC(d''_{t+1}, d'_{t+1}) = 3$: We know that

$$tp(d''_{t+1}, d'_{t+1}) = p''_1 \cdots p''_{i-1}p''_i p''_{i+1}p''_{i+2} \cdots p''_n \quad (3.11)$$

$$= - \cdots - \downarrow\downarrow - \cdots - \quad (3.12)$$

From Table 1, it is clear that

$$CC(d''_{t+1}, d'_{t+1}) = 3. \quad (3.13)$$

To prove $T(d_t, d'_{t+1}) + T(d''_{t+1}, d'_{t+1}) \leq T(d_t, d^{t+1})$:

Since $CC(d'_{t+1}, d_t) < 5$ and $CC(d''_{t+1}, d'_{t+1}) = 3$,

$$T(d_t, d'_{t+1}) = 2CLRT(1 + \lambda)$$

$$T(d''_{t+1}, d'_{t+1}) = CLRT(1 + \lambda)$$

But $CC(d_{t+1}, d_t) \geq 5$. So,

$$T(d_t, d_{t+1}) \geq 3CLRT(1 + \lambda) \quad (3.14)$$

Hence, $T(d_t, d'_{t+1}) + T(d''_{t+1}, d'_{t+1}) \leq T(d_t, d^{t+1})$

To prove $E(d_t, d'_{t+1}) + D(d''_{t+1}, d'_{t+1}) \leq E(d_t, d^{t+1})$: 
For simplicity, as in the case of the first proof, assume that there are exactly 2 wires, i.e., wire 1 and the wire \( i + 1 \), which have opposite transitions w.r.t each other. Let \( d_i^t \) be the value of the \( i^{th} \)-wire at time instant \( t \). Assume that \( d_i^t d_{i+1}^{t+1} = 01 \) and \( d_{i+1}^{t+1} d_{i+2}^{t+1} = 10 \) so that \( d_i^t d_{i+1}^{t+1} = 11 \) and \( d_{i+1}^{t+1} = d_{i+2}^{t+1} = 00 \). Note that the proof can be easily extended to the case where \( d_i^t d_{i+1}^{t+1} = 10 \) and \( d_{i+1}^{t+1} d_{i+2}^{t+1} = 01 \).

Then

\[
E(d_t, d_{t+1}) = C_L(x_1 - \lambda d_{i+1}^{t-1}) + C_L(\lambda d_{i+1}^{t+2} + x_2) \\
+ C_L(1 + 3\lambda - \lambda(d_{i+1}^{t-1} - d_{i}^{t-1}))
\]

(3.15)

Where \( C_L(x_1 - \lambda d_{i+1}^{t-1}) \) is the energy consumption of the first \((i-1)\) wires (negative component is the effect of wire \( i \) on wire \((i-1)\)), \( C_L(1 + 3\lambda - \lambda(d_{i+1}^{t-1} - d_{i}^{t-1})) \) is the energy consumption of wire \( i \) and wire \((i+1)\) including the effects of their other adjacent wires, and \( C_L(\lambda d_{i+1}^{t+2} + x_2) \) is the energy consumption of the last \((n-i-1)\) wires including the effect of wire \((i+1)\) on \((i+2)\).

\[
E(d_t, d_{t+1}) = C_L(x_2) + C_L \left((1 + \lambda - \lambda(d_{i+1}^{t-1} - d_{i}^{t-1}) - \lambda(d_{i+1}^{t+2} - d_{i}^{t+2})) \right) \\
( d_i', d_{i+2}' ) + C_L(\lambda d_{i+1}^{t-1}) + C_L(0) + C_L(\lambda d_{i+1}^{t+2})
\]

(3.16)

If we assume that \( E(d_t, d_{t+1}') + E(d_{t}', d_{t+1}'') > E(d_t, d_{t+1}) \), after simplification, we obtain the following inequality

\[
\lambda(d_{i+1}^{t-1} - d_{i+1}^{t+2} + d_{i}^{t+2}) > 2\lambda
\]

(3.17)

Since \( \lambda > 0 \),

\[
d_{i+1}^{t-1} - d_{i+1}^{t+2} + d_{i}^{t+2} > 2
\]

(3.18)
Table 3.2: Wire parameters

<table>
<thead>
<tr>
<th>Layer</th>
<th>H(µm)</th>
<th>W(µm)</th>
<th>S(µm)</th>
<th>T(µm)</th>
<th>(C_L(\text{fF/mm}))</th>
<th>(C_I(\text{fF/mm}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metal 5</td>
<td>1.04</td>
<td>0.52</td>
<td>0.52</td>
<td>0.7</td>
<td>34.317</td>
<td>80.478</td>
</tr>
</tbody>
</table>

Which is always false. Hence

\[
E(d_t, d'_{t+1}) + D(d''_{t+1}, d''_{t+1}) \leq E(d_t, d'_{t+1}) (3.19)
\]

3.10 Experiments Validating VCTR and Results

We now experimentally validate our approach. We design the crosstalk analyzer, the encoder, and the decoder in Verilog and synthesize it using the Synopsys Design Compiler with 0.16µm oki technology library. We model three different wire lengths, i.e., 2mm, 5mm, and 10mm, to capture short, medium, and long wires, respectively, in the target technology. The Berkeley interconnect model [34] is used to calculate the ground and coupling capacitance of the interconnects. In our experimental results, we use the parameters of metal 5 (shown in Table 3.2).

We use Simplescalar 3.0 [35] and the SPEC2000 CINT [36] benchmark suite to simulate the performance of different on-chip buses between the processor datapath and L1 I-cache/ D-cache. Transition pattern distributions of both address and data buses are given in Figure 3.11.

We compare the variable cycle transmission with temporal redundancy (VCTR) technique with the base case (i.e., without any coding), the variable cycle transmission (VCT) method [4], the crosstalk prevention coding (CPC) technique [9], and the shielding (SHD) method [37]. We consider two delays, \(C_L R_T (1 + \lambda)\) (sin-
CPC code. Table 3.3 gives a set of CPC codes for 3-bit data. In order to prevent
culture to implement the method. As a result,
the output of the crosstalk analyzer, we consider the appropriate delay. Note that
to be a multiple of the single cycle period) for the VCTR technique, and based on
the output of the crosstalk analyzer, we consider the appropriate delay. Note that
the delay for the base case is $C_L R_T (1 + 4\lambda)$, for the CPC technique and shielding
method the delay is $C_L R_T (1 + 2\lambda)$, and for the VCT technique the delays are
multiples of $C_L R_T (1 + \lambda)$ based on the crosstalk class type.

In the VCT method, the delay between the next data item and the present
data item is determined by the crosstalk class of the next data w.r.t. the present
data item. Note that in this method, no encoding is applied on the transmitted
data. Though by using the CPC method, one can get 46-bit crosstalk-free codes
for any 32-bit data, it is very difficult to implement the method. As a result,
here we consider 3-to-4 CPC coding, where every 3-bit data is encoded with 4-bit
CPC code. Table 3.3 gives a set of CPC codes for 3-bit data. In order to prevent
adjacent bits of different 4-bit encoded words to form crosstalk classes 5 and 6,
we use shield wires between every pair of -bit encoded words. Hence, for a 33-

Figure 3.11: Distribution of transition patterns
bit bus we use 55 wires. In the SHD method, one shield, i.e., $V_{dd}$ or $G_{nd}$, wire is inserted between every two wires so that the crosstalk classes and are completely eliminated and hence the worst-case delay becomes $C_L R_T (1 + 2\lambda)$.

Main characteristics of different methods are given in Figure 3.12. The values present in this table are averaged across all the benchmarks and normalized w.r.t. the base case. Compared to the VCT technique, our technique achieves better performance in both address and data buses because of two reasons: 1) in the case of crosstalk class 6, the VCT technique considers a delay of $4C_L R_T (1 + \lambda)$, whereas our technique encodes the data into two data items such that the net delay becomes $3C_L R_T (1 + \lambda)$; 2) after encoding, for each 01 (or 10), $d''_{t+1}$ gets 00 so that this can minimize the chances of getting crosstalk classes 5 or 6 for $d_{t+1}$ w.r.t. $d''_{t+1}$ (though we have not experimentally evaluated). The second reason is especially true, if same adjacent wires are repeatedly getting opposite transitions. This can be seen in address bus, where few LSB bits will change most of the time.

<table>
<thead>
<tr>
<th>Original code</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPC code</td>
<td>0000</td>
<td>0001</td>
<td>0100</td>
<td>0101</td>
<td>1100</td>
<td>1101</td>
<td>1111</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.12: Normalized delay and energy
Table 3.4: Codec overhead

<table>
<thead>
<tr>
<th>Method</th>
<th># of Wires</th>
<th>Normalized Area</th>
<th>Extra Power (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>2 mm 5 mm 10 mm</td>
<td></td>
</tr>
<tr>
<td>Base</td>
<td>33</td>
<td>100 100 100</td>
<td>–</td>
</tr>
<tr>
<td>VCT</td>
<td>34</td>
<td>119 109 106</td>
<td>3.35</td>
</tr>
<tr>
<td>VCTR</td>
<td>35</td>
<td>133 116 111</td>
<td>2.16</td>
</tr>
<tr>
<td>CPC</td>
<td>55</td>
<td>177 171 169</td>
<td>1.06</td>
</tr>
</tbody>
</table>

That is the reason why, though there are less number of crosstalk class transitions (from Figure 1), our technique outperforms the VCT technique. From the delay perspective, our technique achieves 31% (30%) savings for address (data) bus with just two extra wires.

Though the CPC technique and SHD technique give more delay savings, they require 22 and 32 extra wires, respectively. From the energy perspective, our technique achieves 13%(19%) savings for address (data) bus compared to the base case. Since coupling transitions play a major role in the total energy, though self transitions for our technique are more than that of the CPC technique in the address bus case, because of less number of coupling transitions, the energy consumption of our technique is less than that of the CPC technique. Since no encoding is applied in the VCT technique, its energy consumption is same as that of the base case. In the case of data bus, the CPC technique consumes more energy than that of the base case. The SHD technique consumes more energy than the base case because of more coupling transitions. Benchmark-wise normalized delay and energy consumption (w.r.t. the base case) of different techniques are given in Figure 3.13-3.14.

Table 3.4 gives the normalized area (w.r.t. the base case) and the power overhead. One can see from the table that our technique incurs an area overhead of 33%, 16%, and 11% for 2 mm, 5 mm, and 10 mm bus, respectively. This is due to
the codec (i.e., crosstalk analyzer, encoder, and decoder) and two extra wires. The absolute area overhead of our codec is 0.018mm². One can observe that as the length of the interconnects increases the relative area overhead of our technique decreases. Our codec can run at a maximum clock speed of 660 MHz and has power overhead of 2.16mW.
Chapter 4

Multiple Phase Staggered Latching

4.1 Introduction

Feature scaling has been key to sustaining the exponential growth in chip performance over the decades. However, as on-chip dimensions cross the 100 nm threshold, various signal integrity challenges are threatening to limit this trend [38]. The adoption of high aspect ratio metal layers to mitigate the inverse relationship between process scaling and metal resistance, leads to the formation of large implicit coupling capacitances between physically separated metal traces. These large capacitances create non-negligible electrical interference and crosstalk noise that can distort signals on neighboring metal wires. The effect of this crosstalk noise is particularly critical in the design and performance of multi-bit on-chip interconnect structures such as buses, network links, memory bit-lines which provide communication between functional blocks, memory elements, I/O pins etc.

The component bit-lines in multi-bit on-chip interconnect are typically composed of segmented metal lines with CMOS repeaters or buffers driving each seg-
ment. The buffers are sized to provide adequate drive current, and fast slew rates. The conducting material of the segment is a strip of copper metal, with pitch, width, thickness, dielectric constant and other process dependent parameters chosen to minimize the increase in metal RC constant between process generations. Scaling requirements below 100 nm for metal pitch, aspect ratio, inter-metal dielectric constant etc indicate that the coupling capacitance will increasingly dominate ground capacitance. As a result, signal interference phenomena such as crosstalk is expected to increase with process technology leading to crosstalk-induced delay variation, crosstalk-induced latching errors etc.

Crosstalk induced delay refers to the effective switching speed of a coupled metal line due to signal activity on neighboring lines. Line delay will vary depending on the specific direction and the relative temporal overlap of the neighboring transitions. Traditionally, the use of simultaneous latching in synchronous circuits forces a built-in temporal overlap in signal transitions. When transition directions on neighboring lines coincide, line delay is reduced below-nominal, when they diverge, line-delay increases above nominal, otherwise line-delay is nominal. In multi-bit interconnect design, worst-case delay margins are necessary to guarantee transmission reliability with a small performance tradeoff. Future technology nodes promise increasingly higher inter-metal coupling, this will require larger delay margins in order to guarantee reliability but also even larger tradeoffs.

In this work, we analyze the effect of offset switching on line delay in comparison to the traditional simultaneous switching approach. We show that offset switching is superior in a context of highly coupled, segmented bit-lines. We further present a novel staggered latching mechanism for seamless synchronous operation and demonstrate the improved error rates over traditional methods. Our solution requires zero additional wires, and no appreciable increase in power and area. The
rest of this work is organized as follows: Section 2 discusses some selected related work, Section 3 introduces the effect of switching offset in electrically coupled bit-lines. Section 4 introduces staggered latching as a novel clocking scheme. Section 5 presents experiments and results and Section 6 concludes the document.

### 4.2 Related Work

Signal crosstalk in on-chip interconnect due to adjacent-wire capacitive coupling, has received much interest and attention in the literature. Efficient methods for extracting and characterizing wire resistance, ground and coupling capacitance for both local and global wires are well known [39]. Closed form expressions for modeling local interconnect delay in the presence of coupling, have been proposed.
Figure 4.2: Segment with maximum overlap to its adjacent neighbors. Transitions at A, will generate a response transition at B composed of the superposition of the direct response and any noise injected from adjacent segments.

and numerically efficient methods for electronic design automation (EDA) purposes have also been published [40], [41]. The use of miller capacitance models for intermetal coupling capacitance as proposed for fast delay calculations in [42], introduce non-negligible inaccuracies with feature sizes below 50 nm. As a result, complex noise superposition models, which have been shown to offer more reliable delay estimates in the presence of crosstalk, have been developed [41], [43]. In [44], closed form expressions for the total noise waveform due to all active neighbors of a wire was proposed for local wires but is seamlessly extendable to long wire interconnect structures.

Crosstalk induced delay is skew-dependent. Skew-dependent delay fluctuations are due to variations in the temporal overlap between a transitioning signal and the noise waveform from various neighboring aggressors, the larger the overlap, the larger the change in the delay [45]. In [16], a similar bus delay reduction technique is proposed to deliberately introduce transition skew between adjacent wires on a bus. They used the miller capacitance method and assumed a one-way aggressor-
victim model for the key delay analysis. This approach is an over-simplification of the multi-way aggressor-victim reality and as a result, it is insufficient for sub-50 nm processes.

Our proposed approach makes two key contributions: First, using the noise model proposed in [44] for the crosstalk noise waveform, we propose an efficient corresponding delay model as a function of inter-signal input skew for a typical segment in a multi-segment interconnect (MSI). Second, unlike in [16], we propose staggered latching, a novel synchronous clocking strategy that efficiently leverages skewed switching with no additional bus wire overhead. This improves performance in the presence of large coupling capacitance in wide MSI.

4.3 Signal Switching Offset and Line Delay

In this section, we develop analytical models for line delay as a function of the switching offset between closely spaced metal traces.

4.3.1 Signal Response Model

The signal delay of an $n$-bit MSI is limited by the slowest bit-line. The response time on the slowest line depends on the resistance and capacitance measured both to the ground plane and to the adjacent lines.

A generalized coupling structure for a multi-segment interconnect is shown in Figure 4.2. The variable $m$ represents the mis-alignment factor between adjacent segments on neighboring bit-lines such that choosing the value $m = 0$ or $m = 0.5$ allows for the modeling of either a Fully-aligned or Mis-aligned strategy respectively. The test segment of interest is in the middle. The goal is to obtain analytical expressions of the overlap threshold for both arrangement strategies.
Figure 4.3: Lumped RC model for the general coupled segment. Immediate neighbors are shown, farther segments are assumed grounded.

4.3.2 Nominal Response

An RC model for the coupled segments in Figure 4.2 is shown in Figure 4.3. The total response $v_b$ in Figure 4.3 is a superposition of the direct response due to the primary input $v_a$, and the total noise injected by the secondary inputs $v_1$, $v_2$, $v_3$ and $v_4$ through the coupling capacitances. We first obtain the noise-free response and the analytical expression for the corresponding nominal delay.

4.3.2.1 Drive buffer/Repeater

The segment driver is a large inverter with minimum length ($\lambda$) sized transistors. The pull-up PMOS transistor of size $W_p$ is selected to match approximately the

<table>
<thead>
<tr>
<th>$Tech(\lambda)$</th>
<th>$W_n$</th>
<th>$R'_n$</th>
<th>$X_p$</th>
<th>$C'_{ox}$</th>
<th>$p_m$</th>
<th>$ar_m$</th>
<th>$ar_v$</th>
</tr>
</thead>
<tbody>
<tr>
<td>45 nm</td>
<td>8</td>
<td>18.4k</td>
<td>3.3</td>
<td>54.7 aF</td>
<td>102n</td>
<td>1.8</td>
<td>1.6</td>
</tr>
<tr>
<td>32 nm</td>
<td>6</td>
<td>22.8k</td>
<td>3.2</td>
<td>32.4 aF</td>
<td>61n</td>
<td>1.9</td>
<td>1.7</td>
</tr>
<tr>
<td>22 nm</td>
<td>10</td>
<td>28.7k</td>
<td>1.7</td>
<td>20.8 aF</td>
<td>43n</td>
<td>2.0</td>
<td>1.8</td>
</tr>
<tr>
<td>16 nm</td>
<td>9</td>
<td>31.0k</td>
<td>1.5</td>
<td>14.6 aF</td>
<td>30n</td>
<td>2.1</td>
<td>1.9</td>
</tr>
</tbody>
</table>
performance of the pull-down NMOS transistor of size $W_n$. If we define the ratio
of transistor widths $X_p = W_p/W_n$, and the capacitance per square for a mini-
mum length MOSFET, ($C'_{ox}$), we can obtain from ITRS data [38], Figure 4.1 and
HSPICE characterization runs the values shown in Table 4.1. The gate capacitance
($C_{gate}$), diffusion capacitance ($C_{diff}$), and drive resistance ($R_{drv}$) are then modeled
by equations: $C_{gate} = 1.5 \cdot C'_{ox} W_n (1 + X_p)$, $C_{diff} = C'_{ox} W_n (1 + X_p)$, $R_{drv} = R_n/W_n$.
The characteristic driver delay ($tD_{drv}$) is given by the model in equation 4.1.

$$tD_{drv} = 2.5 \cdot R_n' C'_{ox} (1 + X_p)$$  \hspace{1cm} (4.1)$$

4.3.2.2 Metal trace

The direct response is obtained from an $s$-domain analysis of the circuit in Figure
4.3. The secondary inputs are set to zero, and a unit step is applied to the
primary input $v_a$. The product of the total lumped resistance ($R_T$) and the lumped
capacitance to ground ($C_T$) is the intrinsic $rc$-constant ($\tau$) of the line segment.
The coupling capacitance is defined in terms of $C_T$ and a weighting factor ($\eta$),
$C_c = \eta \cdot C_T$.

$$V_{\delta N}(s) = \pm \frac{1 + s(1 + 2\eta)\tau}{s \cdot D(\eta)}$$  \hspace{1cm} (4.2)$$

where $\tau = R_T C_T$ and in the denominator

$$D(\eta) = (1 + 2s(1 + 2\eta)\tau + s^2 (1 + 4\eta + 2\eta^2) \tau^2)$$  \hspace{1cm} (4.3)$$

The total resistance $R_T$, shown in equation 4.4 is the sum of the metal resistance
$R_m$ and the driver switching resistance $R_{drv}$. Likewise, the total capacitance to
ground for a given segment, shown in equation 4.5 is the sum of the contributions
Figure 4.4: General shape and characteristics of the Injected noise pulse for (a) aligned segments and (b) misaligned segments from the metal and the driver.

\[ R_T = R_m + R_n/W_n \]  
\[ C_T = C_m + 2.5 \cdot C'_{ox} W_n (1 + X_p) \]

Applying the Inverse Laplace Transform to \( V_{bN}(s) \) we obtain the general, normalized, time-domain signal form

\[ v_{bN}(t) = \pm (1 - A_0 \cdot e^{-t/\tau G_0}) u[t] \]

where the constants \( G_0 \) and \( A_0 \) are obtained via padé approximation and coefficient matching of the \( s \)-domain polynomials. See table 4.2. The nominal delay is obtained by solving for the \( Vdd/2 \) crossing point of equation 4.6.

\[ tD_{nom} = G_0 \tau \cdot \ln(2A_0) \]

### 4.3.3 Noise Response

The RC model in Figure 4.3 is easily modified for noise signal extraction by grounding the primary input \( v_a \) and driving the secondary inputs \( v_1, v_2 \) and \( v_3, v_4 \) with
unit step signals. Note however, that due to the use of inverting repeaters, compared to \( v_1 \) and \( v_2 \), the transition direction of \( v_3(v_4) \) is opposite and shifted in time by \( T \). The injected noise \( v_b \) is thus comprised of two components, one in-phase, the other counter-phase, see fig. 4.4b. The noise transient parameters depend on the segment RC characteristics, the coupling capacitance \( (C_c) \), and on the actual number of switching neighbors \( (SW) \). For planar 2D layout with a maximum of two closest neighbors (i.e \( SW = 0,1,2 \)), the general noise response in eqn. 4.8 is obtained from an s-domain analysis of the modified fig. 4.3 circuit.

\[
V_b = SW \cdot \tau \left( \frac{\eta_A}{D(\eta_A)} - \frac{e^{-sT\eta_M}}{D(\eta_M)} \right) \quad (4.8)
\]

If we define a generalized noise pulse response for the variable \( t \) and the model constants \( \tau \) and \( G_1(\eta) \) as \( v_\eta(t) \) in eqn 4.9

\[
v_\eta(t) = \pm t \cdot e^{-t/\tau G_1(\eta)} \cdot u[t] \quad (4.9)
\]

Then the Inverse Laplace Transform of eqn. 4.8 yields a corresponding normalized, time-domain noise response \( v_b \) in eqn. 4.10.

\[
v_b(t) = A_1(\eta_A) \cdot v_\eta_A(t) - A_1(\eta_M) \cdot v_\eta_M(t - T) \quad (4.10)
\]

In general, \( \eta_A = (1 - m) \cdot \eta \) and \( \eta_M = m \cdot \eta \). However, focusing on fully-aligned \( (m = 0) \) or mis-aligned \( (m = 0.5) \), the model constants \( A_1 \) and \( G_1 \) are obtained by substituting \( \eta = \eta_A \) or \( \eta = \eta_M \) in the expressions in Table 4.2.
Table 4.2: Summary of Model Constants as function of $\eta$

<table>
<thead>
<tr>
<th>$G_0$</th>
<th>$1 + 2\eta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_1$</td>
<td>$\frac{(1 + \eta)(1 + 3\eta)}{1 + 2\eta}$</td>
</tr>
<tr>
<td>$G_2$</td>
<td>$\frac{1 + 2\eta(2 + 3\eta + SW(1 + 2\eta))}{1 + (2 + SW)\eta}$</td>
</tr>
<tr>
<td>$G_3$</td>
<td>$\frac{SW\eta T + \tau + 4\eta\tau + 6\eta^2\tau}{\tau(1 + 2\eta)}$</td>
</tr>
<tr>
<td>$A_0$</td>
<td>$\frac{1 + 4\eta + 5\eta^2 + 6\eta^2}{1 + 4\eta + 6\eta^2}$</td>
</tr>
<tr>
<td>$A_1$</td>
<td>$\frac{(\eta/\tau)G_1SW}{(1 + 2\eta)(1 + 4\eta + 2\eta^2)}$</td>
</tr>
<tr>
<td>$A_2$</td>
<td>$\frac{1 + (2 + SW)\eta}{G_2}$</td>
</tr>
<tr>
<td>$A_3$</td>
<td>$\frac{(1 + 2\eta)}{G_3}$</td>
</tr>
</tbody>
</table>

4.3.3.1 Noise Duration

The noise pulse $v_b$ in eqn 4.10 has a last crossing time $z_0$, last absolute maximum $(\tilde{n})$ at time $\tilde{t}$. The pulse has a duration ($d_k$) measured in terms of non-zero, integer ($k$) multiples of $G_1\tau$, i.e. for a specified noise limit ($N_{lim}$), and for all integers $k$ larger than $k_0$, $v_b(d_k) \leq N_{lim}$.

$$d_k = z_0 + k \cdot G_1\tau, \text{ where } \tilde{n} \cdot \left(\frac{k_0}{e^{k_0} - 1}\right) \leq N_{lim} \quad (4.11)$$

These parameters can be calculated from $v_b(t)$ for any chosen value of $m$.

$$\tilde{t} = z_0 + G_1\tau, \quad z_0 = \begin{cases} 0 & m = 0 \\ T + T/(e^{T/G_1\tau} - 1) & m = 0.5 \end{cases} \quad (4.12)$$

The constant $T$ is obtained by analyzing the circuit models in fig 4.2 and fig 4.3.

$$T = tD_{drv} + G_0\tau \cdot ln(2A_0) \quad (4.13)$$
Now, if we also express the time shift in terms $T = j \cdot G_1 \tau$, where $j > 0$, then the value of the last maximum value ($\tilde{n}$) of $|v_b(t)|$ can be calculated, see eqn 4.14.

$$\tilde{n} = \begin{cases} 
A_1 G_1 \tau \cdot |1/e| & m = 0 \\
A_1 G_1 \tau \cdot (1 - e^j/e) \cdot e^{-j(m/\sigma)} & m = 0.5
\end{cases}$$  \hspace{1cm} (4.14)

### 4.3.4 Delay, Offset and Overlap Threshold

In general, signal delay on the middle segment in figure 4.2 is defined as the time difference between the last $0.5V_{dd}$ crossing points measured from the signal $v_a$ to $v_b$. Since the signal $v_b$ is a superposition of the direct response and the injected noise, the signal delay is a function of the degree of temporal overlap between them. If we define a variable $\text{alpha} (\alpha)$ as the offset between switching events at the input, of the signal $v_a$ and any adjacent segments, then the signal-to-noise overlap at the output $v_b$, and consequently the signal delay $tD(\alpha)$ can be expressed in terms of $\alpha$. For large enough absolute offset values, the overlap at the output between the transition event of the direct response and the duration of the injected noise is zero. This results in a signal delay that is indistinguishable from a noise free delay. The smallest absolute offset value for which this condition is true is defined as the offset $\text{Overlap Threshold} (\alpha_{OS})$. We can calculate this value by solving for $t$ using the normalized voltage eqn 4.15.

$$v_b(t, \alpha) = r(t) + n(t - \alpha) = 0.5$$  \hspace{1cm} (4.15)

Using an intermediate variable $\text{sigma} (\sigma)$, we can define a parametric relationship $r(t(\sigma)) = 0.5 - n(\sigma))$. The signal $r(t)$ is the noise free response from eqn 4.6. The noise signal $n(t)$ depending on the design, is either the fully aligned or misaligned
noise pulse signal from eqn 4.10. Solving for $t$ and $\alpha$ in terms of the variable $\sigma$, we obtain the parameterized delay and offset eqns 4.16

$$t(\sigma) = t_D^{nom} + G_0 \tau \cdot \ln \left( \frac{1}{1 - 2 \cdot n(\sigma)} \right)$$

$$\alpha(\sigma) = t(\sigma) - \sigma$$  \hspace{1cm} (4.16)

For a given design and a specified noise limit $N_{lim}$, the corresponding $d_k$ can be obtained using equation 4.11. Substituting into eqn 4.16 the following values: $\sigma = d_k$ and $n(d_k) \leq N_{lim}$ we obtain an expression for the overlap threshold for a chosen number $(SW)$ of switching neighbors.

$$\alpha_{OS} \bigg|_{SW} = t(d_k) - d_k \bigg|_{SW}$$  \hspace{1cm} (4.17)

In any MSI, regardless of alignment strategy, $\alpha_{OS}$ represents the minimum, mutual signal-transition offset between any set of coupled segments that assures nominal signal delay on both segments. For comparative analysis, the $0 - 90\%$ segment transition time ($\alpha_{SS}$) is derived for simultaneously switched MSI using eqn 4.6 and the constants from Table 4.2. The constant tuple $(G,A)$ for noise free nominal transition is chosen as $(G_0,A_0)$. For noisy transitions, $(G_2,A_2)$ and $(G_3,A_3)$ are used for aligned and misaligned MSI respectively.

$$\alpha_{SS} \bigg|_{SW} = G \tau \cdot \ln(10A) \bigg|_{SW}$$  \hspace{1cm} (4.18)
The worst case segment delay for simultaneous/offset switching considering all coupling noise is shown in equation 4.19

\[
tD_{\text{max}} \leq \begin{cases} 
G_0 \tau \cdot \ln(2A_0e/(e - 2A_1G_1 \tau)) & \text{SS} \\
G_0 \tau \cdot \ln(2A_0/(1 - 2 \cdot N_{\text{lim}})) & \text{OS}
\end{cases}
\] (4.19)

Using these analytical models, the potential speedup can be estimated for an M5, 0.25mm long, 6 line (4-signal, 2-grounded dummy), 5-segment MSI, using only metal and drive buffer RC parameters from current/predictive BEOL processes, see Table 4.1. Choosing \( N_{\text{lim}} = 0.05 \), a data stream with regular gaussian transition distribution fig. 4.6, setting offset > \( \alpha_M \), the model estimates an average speedup of 2.05X(1.70X) over an SS-MSI for an OS-MSI-aligned(misaligned). Table 4.3 shows the OS speedup results compared to \( F_{SS} = tD_{\text{max}}^{-1} \) across sub-50nm processes.

<table>
<thead>
<tr>
<th>Tech((\lambda))</th>
<th>(F_{SS}(\text{Hz}))</th>
<th>OS/SS</th>
<th>(\alpha_M)</th>
<th>(F_{SS}(\text{Hz}))</th>
<th>OS/SS</th>
<th>(\alpha_M)</th>
</tr>
</thead>
<tbody>
<tr>
<td>45 nm</td>
<td>0.96G 1.8X 92ps</td>
<td></td>
<td></td>
<td>1.44G 1.53X 89ps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32 nm</td>
<td>0.76G 2.0X 97ps</td>
<td></td>
<td></td>
<td>1.14G 1.68X 90ps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22 nm</td>
<td>1.08G 2.1X 64ps</td>
<td></td>
<td></td>
<td>1.63G 1.75X 58ps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 nm</td>
<td>1.07G 2.2X 57ps</td>
<td></td>
<td></td>
<td>1.61G 1.84X 51ps</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.4 Multiple Phase Staggered Latching

In this section, we propose a $b$-bit wide Multiple Phase Staggered Latch ($MPSL[b]$) interconnect architecture that exploits offset switching to achieve improved crosstalk performance.

The top level architecture of an MPSL interconnect, shown in figure 4.5, has two key sets of clocked latches: the Interfacing (IF) latches and the Offset-Tuning (OT) latches. The IF latches are collectively two sets of single-stage latches, $b$-bits wide, each set placed at a boundary to bridge the clock transition points of the enclosed structure with the send-side and the receive-side logic. They are included to provide (where absent) explicit electrical isolation and signal racing avoidance. The OT latches connect the IF latches with the physical bit lines. Using numbered bit positions $[1,2,...,j,j+1,...,b]$, each individually contains exactly $b$ total latch stages. Specifically, the OT latches are arranged on the $j$-th bitline such that $j$ and $(b - j)$ latches are placed at the send-side and receive-side respectively. This results in a staggered configuration and effectively achieves offset insertion at the send-side and resynchronization/offset removal at the receive-side. Note that the total number of latches traversed, end-to-end, for each bit position is exactly equal.

The parallel MSI bit lines that form the physical connection between the send and receive side can be arranged either in an aligned or in a misaligned configuration.

All latches are two-state, sample/hold, clock level-sensitive latches. The latch control signals are periodic with identical period ($T_{clk}$). However, $T_{clk}$ is subdivided into multiple phases and specific clocking signals are generated to operate the MPSL structure. For the IF latches, a two-phase control signal identical to the system clock signal is used to control data ingress and egress. For the OT latches, all stages use a $b$-phase control signal. In order to implement offset switching
however, a stage dependent phase offset is added to the control signals between consecutive OT latch stages, forcing a \( b \)-by-1 bit transmission/reception exclusivity across the \( b \)-bit wide physical bit lines.

### 4.4.1 Clocking and Latch Control

At each bit position, the critical latch stage from a timing perspective is the last latch before the \( X \)-segment MSI bit line. Therefore, the relationship of clock period \( T_{clk} \) to this latch stage, across all bit positions determines the performance of the MPSL interconnect. For a general \( b \)-bit design, with \( i \) consecutive bits-in-flight (\( biF \)), if the MSI has a maximum bit line delay (\( t_{DM} \)) and a bit-to-bit minimum separation (\( \alpha_M \)) at each position we can calculate key parameters. For an \( X \)-segment bit line with \( SW_{max} \) as the maximum possible number of switching neighbors, we use equation 4.19, for the worst case segment delay \( t_{D_{max}} \), and with \( t_{D_{drv}} \) from equation 4.1, we obtain the maximum bit line delay.

\[
(t_{DM} = X(t_{D_{drv}} + t_{D_{max}}))
\]
For the minimum bit-to-bit separation $\alpha_M$, clock period $T_{clk}$ in $b$-phases, if we use $b = 1$ for simultaneous switching, we can write in general a scalar dot product of two vectors $W$ and $\alpha_{SW}$ shown in equation 4.21

$$\alpha_M = W \cdot \alpha_{SW}$$

(4.21)

Where $\alpha_{SW} = [\alpha_{SW_0}, \alpha_{SW_1}, \alpha_{SW_2}]$ is the array of offset threshold values, from eqns 4.17, 4.18, associated with noise injections from neighboring switching activity. The vector $W$ contains the weight of each threshold value derived from the statistical distribution of transitions in a data stream. We also obtain that $T_{clk}$ must satisfy eqn 4.22 at the boundary between $i$ and $(i + 1)$ bits-in-flight.

$$\frac{tD_M - \alpha_M}{i + 1} \leq T \leq \frac{tD_M}{i} \text{ where } T \neq T_{clk} \& i \geq 1$$

(4.22)

The Latch Control Circuit (LCC), generates the actual multiple phase control signals for the IF and OT latch stages. It is composed of $l$ double-level-sensitive (DLS), latches (with configurable reset), where $l$ is the least common multiple $LCM(2, b)$. A DLS latch samples its input and holds its output in every phase of the control clock signals period. $l$ is chosen to guarantee latching synchronization at the clock boundary between the 2-phase IF latches and the $b$-phase OT latches. The $l$ DLS latches are connected in a single loop and controlled by a single clock signal ($\phi$) with a phase time ($p_\phi$) where $p_\phi = T_{clk}/l$. Configuring the reset-mode of the LCC latches, by setting the first $l/2$ (or $l/b$) as reset-to-one and the rest as reset-to-zero, the 2-phase ($b$-phase) signal $\phi_m$ ($\phi_b$) are easily generated. All other latch signals are variants of the primary ($\phi_m$, $\phi_b$) signals with one (or more) added phase offset. They are easily generated via appropriate taps along the length of
Figure 4.6: Statistical model constants for random, regular or burst data patterns in stacked MPSL[b]. (b) Double-level-sensitive latch with reset.

the DLS latch loop of the particular multi-phase LCC. Note that the phase time \( p_b = p_b \cdot (l/b) \geq \alpha_M \) for the b-phase signals is the switching offset inserted between consecutive bit positions in the MPSL[b] structure. For hardware implementations, each output tap of the LCC circuit shown in fig. 4.5 can be distributed to the specific latch stage via a delay equalized buffer tree network (not shown).

### 4.4.2 Staggered Latch Bus (SLB)

The MPSL implementation of an N-bit bus is the stacked-MPSL[b], where N is subdivided into b-bit sections, with each assigned to an MPSL[b]. The simplest form is the stacked-MPSL[2] or Staggered Latch Bus (SLB). In this configuration, the LCC is simplified and the signals \( \bar{\phi}_m \) and \( \phi_b \) are identical, and likewise the

Figure 4.7: Comparing (a) the classical SS n-bit bus and (b) the proposed stacked MPSL[2] or Staggered Latch Bus (SLB)
signals $\phi_m$ and $\phi_{b,1}$. No additional logic area is required and an explicit LCC is therefore not necessary.

4.5 Experiments and Results

In this section, we compare the data transmission error rates of two switching methods: simultaneous switching (SS) and offset switching (OS) over an increasing clock frequency. SS is the traditional strategy widely used in synchronous interconnect design while OS will be based on the MPLS architecture. We present an experimental validation of a 32 bit MPSL, in an SLB-16 configuration and analyze the design cost with the aid of various tools. Although the outputs ($\phi_b$ and $\phi_{b,1}$) of a 2-bit long LCC are identical to the logic clocks ($\phi_m$ and $\bar{\phi}_m$) respectively, an explicit LCC (only needed for $b > 2$) is included in the experiment for completeness. Our approach combines trace data and detailed HSPICE simulations.

For the HSPICE simulations, the MSI setup consists of two planar arrays of 32, 5-segment, closely spaced parallel bit lines, one array with fully-aligned segments the other with misaligned segments. Each bit line segment consists of a strip of M5 copper, 0.25mm long, driven by an optimally sized inverting buffer. Metal sizing, spacing, resistivity and inter-metal dielectric constants, are taken from the ITRS forecast [38]. Device model files for 45 nm Predictive Technology Model (PTM) process [34] are used for the buffer. The electrical model for the wire resistance and ground capacitance were distributed-$\pi$ RC sections, with the coupling capacitances between corresponding sections on adjacent segments similarly modeled.

Bit Error Rates (BER) per word versus data clock frequency ($f_{\text{clk}} = 1/T_{\text{clk}}$) comparisons are performed for a 5 segment, 45 nm MSI and shown in figure 4.8a. Operated in either single or multiple biF mode, an SLB-16 based on the OS scheme...
Figure 4.8: Bit Error Rate (BER) and Eye diagram analysis for a 32-bit, 5 segment per bit line interconnect in 45 nm tech: (a) and (b) BER vs Frequency (1-6 GHz) comparing simultaneously (SS) and offset switching (OS) (c) and (d) Eye opening @ 5GHz for SS and OS respectively using mis-aligned segments

shows a 2.5X improved speed over a similarly sized traditional SS scheme. When MSI-misaligned segments are used, figure 4.8b, we also obtain good results up to 2.1X speedup compared to misaligned SS. Note that multiple biF (2-biF) modes of operation are possible, this allows support for even higher operating frequencies. The eye diagram in figure 4.8d illustrates this, it shows an SLB-16 MSI-misaligned bus operated in 2-biF mode demonstrating a 60% approximate eye opening at
an approximate data clock frequency of 5 GHz. At similar frequencies, the eye
diagram in figure 4.8c shows the inability of the SS MSI-misaligned bus to match
the performance of an OS MSI misaligned bus.

Scaling the SLB-16 design to 32, 22, and 16 nm nodes, similar BER vs frequency
comparative analysis between SS and OS scheme were performed. A summary
of the results shown in figures 4.9a and 4.9b demonstrate similar performance
improvements, for both aligned and misaligned MSI with an average speedup of
2.5X and 2.3X respectively. This result is obtained with average dynamic power
gain of about 0.5 dB, figures 4.9c, 4.9d, a slight deviation from 0 dB primarily due
to the use of (optional) LCC latches.

Although similar as a comparative measure, the difference in nominal val-
ues between the simulated average speedup 2.5X(2.3X) and the predicted values
2.04X(1.70X) presented in section 4.3.4, is attributable to the constraint imposed
by the selection of $N_{lim}$ used in the analytical model. On the contrary, the max-
imum operating frequency reported here in the simulation results indicates the
speed $f_{clk}$ where the BER per word first exceeds zero. Nevertheless, for quick
design space exploration especially across process nodes, the analytical model pro-
vides a realistic, efficient speedup estimate for offset-switched MSI designers and
EDA tool vendors.

In general, MPLS[b] based designs for $b > 2$ require an explicit LCC, careful
control signal distribution planning, additional latch hardware and area. This is
unnecessary for the MPSL[2] based SLB used in the experiment. Note that except
for the latch rearrangements, the total latch count and control signals in the SLB
are identical to the latch count and clock signals respectively in a traditional SS
bus.
Figure 4.9: Plots of SS vs. OS for operating frequency and power on a 32-bit, 5 segment aligned and misaligned MSI in sub-50nm technologies: (a) and (b) max clock frequency and SS-OS speedup (c) and (d) Power and gain.
Chapter 5

Duty Cycle Tuning in Inverting Bit Lines

5.1 Introduction

Feature scaling has been key to sustaining the exponential growth in chip performance over the decades. However, as on-chip dimensions cross the 100 nm threshold, various reliability challenges are threatening to limit this trend [38]. Aging of Integrated Circuits (IC) is one of these challenges and refers to the inability of a chip to retain its electrical and functional design characteristics over long term use. Device aging effects such as Time Dependent Dielectric Breakdown (TDDB), Hot Carrier Injection (HCI), Negative Bias Temperature Instability (NBTI) and Interconnect aging effects such as electromigration have been studied and shown to be critical to deep sub-micron (DSM) IC reliability. However, with the growth of on-chip networks featuring repeater-segmented, multi-bit wire structures, the line between interconnect and device-specific effects is somewhat blurring. With NBTI expected to emerge as the dominant lifetime-limiting effect in nano-CMOS devices, understanding the scope of its effect and managing same will be key to
Figure 5.1: This shows the percentage change (from nominal) of the end-to-end line delay due to NBTI. Data is plotted for a 1 cm, optimally buffered interconnect line with an input signal 0-probability of 99.9%.

achieving robust and reliable DSM designs [3].

Negative Bias Temperature Instability (NBTI), refers to the degradation in functional integrity of nano-PMOS devices over long term use, particularly at elevated temperatures and under negative bias. The NBTI phenomenon was first documented in the early decades of the semiconductor industry but its effect has become particularly troublesome in the current DSM era. It is typically characterized by a long term gentle drift in the threshold voltage of PMOS devices. This effect occurs at high operating temperatures when a PMOS is under sustained negative bias \( V_{gs} = -V_{dd} \). These conditions lead to trap formation at the oxide-semiconductor interface as silicon-hydrogen bonds initially formed by imperfect oxide growth, breakdown generating electric holes that become mobile and migrate to occupy traps in the oxide. These produce positive interface and fixed charges that reduce the gate field effect on the channel. The result is two-fold, a drop in the device on-current \( (I_{on}) \) and an increase in the device threshold voltage \( (V_{th}) \). Although recovery is possible by zero biasing the device, a permanent damage is often left behind and these changes in device characteristics affect
performance and long term device reliability.

The component bit-lines in multi-bit on-chip buses are typically composed of segmented metal lines with CMOS repeaters or buffers driving each segment. These CMOS repeaters with PMOS pull-up paths, improve delay performance in long wires but are one major contributor to interconnect susceptibility to NBTI. The other main contribution stems from the duty cycle skew (or 1-0 data probability) across the component bit lines of the bus. This skew distribution is data dependent, that is it varies for different applications, address/data, and instruction or data memory communications. The NBTI aging profile of individual lines is a strong function of the 1/0 data probability of the signal on the line, over time, corresponding drift in line end-to-end delay is observed. See figure (5.1). The resulting effects are skew growth, reduced noise margins, timing violations, or a need for increased clock margins in the clock period thereby incurring an overall performance penalty.

In this work, we analyze the effect of uneven data duty cycles on the performance and lifetime of long segmented interconnect. A metric for quantifying the lifetime of an interconnect line is presented. The insight gained is extended to a multi-bit bus structure and overall bus lifetime. A two-step duty cycle balancing and optimization scheme is proposed that adapts existing techniques to achieve Uniformity and Maximum Lifetime for any multi-bit bus. Using the data traces from the instruction and data bus transactions of SPECInt2000 benchmark [36] programs, we are able to demonstrate the large lifetime benefits of this scheme with little area and power overhead. The rest of the document is organized as follows: section 5.2 discusses some related work, section 5.3 introduces the NBTI model for PMOS devices and develops metrics for interconnect lifetime analysis. Section 5.4 discusses bus lifetime optimization and the duty cycle equalization schemes.
Section 5.5 presents experiments and results.

5.2 Related Work

In general, the effect of NBTI on circuit performance has received prior research attention. The development of $V_{th}$ drift models for static and dynamic circuit operation, and an analysis of various design compensation schemes were discussed in [21]. These compensation schemes such as gate sizing, Vdd tuning and duty cycle control have been used to improve device lifetime in [22, 23, 24, 25].

Our proposed solution for achieving Maximum Lifetime in an $n$-bit bus is two-fold. First, a Duty Cycle Equalization (DCE) module is used to balance the duty cycle distribution across the bus. The DCE module is similar to the rotating switch proposed in [46] for thermal optimization in on-chip buses. However, beyond uniformity there is optimization, the potential of attaining Maximum Lifetime, a DCE is simply inadequate for this task. Second, a Duty Cycle Optimization (DCO) module is used to dynamically shift individual line duty cycle towards the optimum. The DCO accomplishes this through dynamic inversion of the bit-stream on a line, as long as the inversion control signal is periodic, independent and with equal probability of ’1’ or ’0’. This technique is similar to two previously published works. In [47], the authors proposed a Bus-Invert (BI) code for low-power data transmission. This technique shares structural similarities to our work, however it differs in one key respect: the generation of the inversion signal. For the BI code, the invert signal is data-dependent. Also it requires both additional registers for storing previous data and comparators for the codec. In our work, the invert signal is purely periodic and achieves Maximum Lifetime for general buses. Also similar is the work by Kumar et.al [25] where bit flipping was used to improve
NBTI lifetime in SRAMs. The key difference with their hardware solution is the purely dynamic nature of our work. Dedicated counters or timer interrupts are used to generate the inversion signal and the DCO is dynamic. This implies that lifetime compensation can be done during data transmission without data loss and there is no restriction to processor stand-by mode. To the authors’ knowledge, this work is the first analysis of the peculiarities of NBTI-induced lifetime degradation in Interconnect. In this work, we present a useful metric for interconnect lifetime estimation, analyze multi-bit bus lifetime and propose a solution for achieving Maximum Lifetime.

5.3 NBTI stress and Interconnect Lifetime

In this section, we analyze interconnect delay degradation over long periods of NBTI stress. Using a 1 cm metal line from a modern 45 nanometer process, we analyze the performance of the metal line after 10 years of aging. We show the nature of the relationship between line delay and data transition probabilities measured at the primary input of an inverter segmented line. We extend this insight to an n-bit bus using real traces to show the need for our proposed solution.
In an inverter-segmented wire, an input signal with 50% duty induces equal 0-probabilities at the gates of all the subsequent drive buffers. In general though, an input signal with duty cycle \( \beta \), induces 0-probabilities equal to \( \beta \) on the even-numbered buffers and \( 1 - \beta \) for the odd-numbered ones. As a result, half of the drivers are stress-biased creating an upward pressure on delay the other half recovery-biased and tend to maintain buffer delay. The profile of the combined delay from both the stress and recovery-biased buffers after ten years under NBTI stress is as shown in figure (5.3) for various input signal probabilities \( \beta \).

The plot can be separated into three regions A, B and C. Region B: \( 0.1 \leq \beta \leq 0.9 \), delay varies by less than 5%. In Region A: \( 0 < \beta < 0.1 \) and in Region C: \( 0.9 < \beta < 1 \), the stress induced delay is apparent especially for \( \beta \) values closer to 0 or 1.
Various analytical models exist in the literature for describing the behavior of PMOS devices under negative bias and at elevated temperatures [48, 21]. One widely used model, the Reaction-Diffusion (RD) model, uses the signal characteristics at the gate of a PMOS device, the temperature, and stress time to provide an accurate estimate of the corresponding change in device threshold voltage for either short or long periods of time. For interconnect analysis, we need to examine delay build-up overtime due to NBTI therefore we focus on the dynamic operation model in the RD suite. The R-D Model for dynamic operation is thus:

$$\delta V_{th}(n) = K_v \cdot \beta^{0.25} \cdot T_p^{0.25} \cdot \left( 1 - (1 - \sqrt{\eta(1 - \beta)/(n)})^{2n} \right) + \delta_v$$  \hspace{1cm} (5.1)$$

we note some important model parameters:

$$K_v = A \cdot T_{ox} \cdot \sqrt{C_{ox} \cdot (V_{gs} - V_{th}) \cdot \exp \left( \frac{E_{ox}}{E_a} \right) \cdot \left( \frac{1 - V_{ds}}{\alpha(V_{gs} - V_{th})} \right) \cdot \exp \left( \frac{E_a}{kT} \right) }$$  \hspace{1cm} (5.2)$$

Also

$$E_{ox} = (V_{gs} - V_{th})/T_{ox}$$

$K_v$ is a parameter that couples the exponential dependence on stress temperature into the R-D model. $k$ is the Boltzmann constant in $eV/K$. $T_p$ is the period of the signal at the gate of the PMOS and $\beta$ is the fraction of $T_p$ for which the gate signal is zero, reverse biasing the gate-source junction. The rest of the coefficients are process-dependent standard parameters. For example, if we assume a 45 nanometer
process then, $E_o = 20 \text{ MV/cm}$, $E_a = 0.12 \text{ eV}$, $A = 1.8 \text{ mV/nm/C}^{0.5}$, $\eta = 0.35$, $T$ is the device operating temperature in Kelvin $K$ and $\delta_v = 5 \text{ mV}$. See [21] for further detail on the model coefficients and their descriptions.

In a $k$-segment interconnect line, segment delay build-up after n cycles can be traced directly to the PMOS cell of the CMOS segment buffer and its local change $\delta V_{th}(n)$, all other parameters held constant. As a result, the end-to-end delay build up, which is the sum of all segment contributions, can be described in terms of the sum of local voltage drift $\delta V_{th}(n)$. Over a small range of delay change ($+\Delta 10\%$ of nominal), using curve fitting, a straight-line of slope $C$ can approximate the $\delta V_{th}$ vs $\delta D$ relationship. The value of $C$ is process dependent and is extracted experimentally. Now, for the $k$-segment line, the total change in delay after n cycles $\Delta D_{line}(n)$, due to NBTI can be written as:

$$\Delta D_{line}(n) = \sum_{i=0}^{k-1} \delta D_i(n) = C \cdot \sum_{i=0}^{k-1} \delta V_{th_i}(n)$$ (5.3)

For our derivations, we choose a line with inverting segments numbered 0 through $k-1$, see Figure(5.2a). Let the binary input stream to the line be $s$, with an effective period $T_s$, a 0-probability of $\beta_s$ and a 1-probability of $\mu_s$, where $\mu_s = (1 - \beta_s)$. If we define the Stress Duty Cycle (SDC) at a given buffer as the 0-probability of the signal at its gate, then the SDC at the $i$-th buffer will depend on the characteristics of $s$. Since the segments are inverting, the SDC at consecutive gates are different and in fact alternate between $\beta_s$ and $\mu_s$ across the line.

Even-numbered buffers and odd-numbered buffers will share $SDC = \beta_s$ and $SDC = \mu_s$ respectively. Therefore for the $k$-segment line we can write equation (5.3) as:

$$\Delta D_{line}(n) = C \cdot \frac{k}{2} \cdot (\delta V_{th_{even}}(n) + \delta V_{th_{odd}}(n))$$ (5.4)
If we let: \( \zeta = \left(1 - \sqrt{\eta(1 - \beta_s)/n}\right)^2 \) and \( \gamma = \left(1 - \sqrt{\eta(\beta_s)/n}\right)^2 \) then for the odd and even-numbered buffers the we can write:

\[
\delta V_{\text{th,even}} = K_v \cdot (\beta_s T_s)^{0.25} \cdot \frac{1 - \zeta^n}{1 - \zeta} + \delta_v \tag{5.5}
\]

\[
\delta V_{\text{th,odd}} = K_v \cdot (\mu_s T_s)^{0.25} \cdot \frac{1 - \gamma^n}{1 - \gamma} + \delta_v \tag{5.6}
\]

If \( \beta_s \neq 1 \) or 0, and \( 0 < \beta_s < 1 \) then \( \zeta \) is always less than 1. When \( n \) is very large, i.e. \( n >> 10^6 \) cycles over the device lifetime, \( \zeta^n \) decays exponentially fast to zero compared to \( \zeta \). Similar argument can be made for \( \gamma \). Combining equations (5.4), (5.5), (5.6) and simplifying, we obtain:

\[
\Delta D_{\text{line}}(n) = \left(\frac{C \cdot k}{2}\right) \cdot nK_v \frac{(-(\beta_s - 1)T_s)^{0.25}}{(-\beta_s \eta) + 2n\sqrt{\frac{\beta_s n}{n}}} + \frac{(\beta_s T_s)^{0.25}}{(-1 + \beta_s)\eta + 2n\sqrt{\frac{n-\beta_s n}{n}}} + 2\delta_v \tag{5.7}
\]

If we define the lifetime of a line, \( L_{\text{line}} \) as the maximum number of cycles \( \bar{n} \) for which \( \Delta D_{\text{line}}(n) \leq \Delta D_{\text{max}} \). Where \( \Delta D_{\text{max}} \) is specified for a given process. Then for large values of \( n \) in equation (5.7), we can eliminate the constants in the denominator, then in solving for \( \bar{n} \) (in cycles) we get:

\[
L_{\text{line}} = \bar{n} = \left(\frac{\left(\frac{2}{C \cdot k}\right) \cdot \Delta D_{\text{max}} - 2\delta_v}{K_v (A + B)}\right)^2 \tag{5.8}
\]

where \( A = \frac{(1-\beta_s)T_s)^{0.25}}{2\sqrt{\eta \beta_s}} \), \( B = \frac{(\beta_s T_s)^{0.25}}{2\sqrt{\eta (\beta_s - 1)}} \).
This is a useful metric for comparing the expected lifetimes of various interconnect lines as the input data probabilities vary. The zero of the partial derivative of equation (5.8) w.r.t $\beta_s$ is a maximum point in the range $0 < \beta_s < 1$ and occurs at $\beta_s = 0.5$. The corresponding value of $L_{\text{line}}(0.5)$ is the Absolute Maximum Lifetime (AML) for an interconnect line with inverting segments. For the rest of the document, $L_{\text{line}}(\beta_s)$ for any interconnect line, with 0-probability $\beta_s$, will be assumed as its AML normalized value, that is $L_{\text{line}}(\beta_s)/L_{\text{line}}(0.5)$. The model will be validated experimentally for a 45 nm process and the comparison plotted in section 5.5.

5.4 Reducing NBTI effect in on-chip buses

In chip-level networks composed of routers and multi-bit, point-to-point bus segments, the need for long term NBTI resilience will grow as these networks become an integral part of future multi-core chip design. In this section, we present a description of bus lifetime in terms of its component lines and we generate lifetime profiles for on-chip buses to demonstrate accelerated aging due to NBTI. We also describe the proposed DCE and DCO solutions and their key characteristics.

5.4.1 NBTI induced Bus Lifetime

A general N-bit bus $B$, composed of bit-lines $B = \{b_1, b_2, \ldots, b_N\}$ has an NBTI induced lifetime defined by two key parameters, the Mean Lifetime ($L_{MB}$) and the Lifetime Uniformity ($L_{UB}$). $L_{MB}$ is simply the normalized average, $L_{MB} = \text{mean}(L_{b_1}, \ldots, L_{b_N})$ of the lifetimes $L_{b_i}$ of the component bit-lines. $L_{UB}$ is related to the standard deviation of the lifetimes $L_{b_i}$, from the mean and is defined as: $L_{UB} = \ldots$
\[ 1 - \frac{1}{LM_B} \cdot \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} ((L_{b_i} - LM_B)^2)} \]. The values of \( L_{b_i} \) depend on the 0-probability (or one minus the 1-probability) of the signal on each line. \( LM_B \) and \( LU_B \) can be calculated once the \( L_{b_i} \) values are known, whether determined experimentally or approximated using equation (5.8). For buses with non-uniform lifetime profiles, i.e. low \( LU_B \), \( LM_B \) will not be an effective measure for lifetime. Instead, \( L_{MIN} = \min(L_{b_i}) \) will be more relevant as the few worst performing bit lines will be more dominant. For buses with highly uniform lifetimes, \( L_{MIN} = (LM_B) \), and we can achieve uniform aging and graceful failure. Figure (5.4) shows typical data probability profiles for two standard on-chip buses. Using equation (5.8), we obtain \( LM_B \) values for address/data bus transactions between the \( \mu \)-processor and the L1 Instruction and Data Caches. The results are shown in figure (5.5)

### 5.4.2 Duty Cycle Equalization

A high Lifetime Uniformity, means that the bus lines will age at a similar rate and as such overall lifetime is not limited by one or a few bit lines. To that end, Duty Cycle Equalization (DCE) is a load balancing technique for maximizing bus Lifetime Uniformity \( LU_B \). Two \( n \) by \( n \) switches are each placed at one end of the bus allowing multiple data streams with varying duty cycle characteristics to share one physical line. The switch control is implemented as shift-register link, configured for a simple rotation operation. The rotation travels clockwise at the sender end and anti-clockwise at the receiver end eliminating any need for complex decoders. This scheme is an adaptation of similar work by Wang et.al [46] to the duty cycle equalization problem. Depending on the rotation frequency, \( LU_B \) can either be maximized at the cost of increased power consumption or power savings.
maximized at the cost of low $LU_B$. Between these two extremes, the designer has wide options for a given application.

5.4.3 Duty Cycle Optimization

A bus with high Lifetime Uniformity but a low Mean Lifetime relative to the AML point can be improved by shifting the mean duty towards 0.5. Duty Cycle Optimization (DCO) uses the concept of an Effective Duty Cycle (EDC) to shift
Comparing Lifetimes for Level–1 Cache Address/Data Buses

Figure 5.5: Average Lifetimes for Raw bit Streams between the \( \mu \)-processor and the L1-Caches

The lifetime of a line towards the AML point. If a global periodic signal (G) with a 0-probability \( \beta_{gbl} \), is used to periodically invert the data transmitted on a bit line then the EDC of the line is described by equation (5.9).

\[
EDC = \beta_{gbl} \cdot (\beta_s) + (1 - \beta_{gbl}) \cdot (1 - \beta_s)
\]  

(5.9)

The proposed optimization scheme can be implemented by selectively placing data or its data-complement on the bus for transmission. Signal inversion is repeated at the end of transmission if data-complement is sent. For the basic interconnect stage terminated by flip-flops at both ends, the data signal and its complement are multiplexed onto the bit line. At the edge of the terminating flip flop, the bus signal and its complement are multiplexed once more. The select signal, shared by the multiplexors at both ends, is the global equalization signal and
its duty cycle determines the effective line duty cycle. For an inverting repeater bit line, simply setting $\beta_{gbl} = 0.5$ in equation (5.9) will result in an $EDC = 0.5$ for the given buffer input, irrespective of the value $\beta_s$. To generate the signal $G$, dedicated counters can be used for a given application or alternatively, the on-chip timer interrupts can be adapted as a source for $G$.

### 5.5 Experiments and Results

In this section, the experimental validation and an analysis of design costs is performed with the aid of various tools. Due to computational efficiency, we use SPICE to validate the model proposed in section 3 and then apply the metric to evaluate the proposed DCE+DCO solution.

For the SPICE simulations, our experimental platform was an optimally segmented, 10 mm long, metal 5, intermediate interconnect wire from a 45 nm process. To determine model constant $k$, optimal buffer insertion was performed on the line and the nominal end-to-end delay without the NBTI effect obtained. For the failure threshold, $\Delta D_{max}$, we use $|\Delta Nominal \ Delay| = 10\%$. We then extracted the model constant $C$ by plotting $\delta V_{th}$ vs $\delta D$ for the line and calculating the slope of a straight-line-fit through the small range of points where, nominal delay $\leq \delta D \leq$
90% nominal. To determine the AML point, we first used equation (5.1) with $\beta_s = 0.5$ to determine $\Delta V_{th}$ after an operating period of $n$ cycles has elapsed. Then second, we modified the SPICE device model files using these set of $\Delta V_{th}$ values and performed transient analysis at each point. The value of $n$ where $\Delta D$ first exceeds $\Delta D_{max}$ for $\beta_s=0.5$ is the AML point. Using a similar procedure we obtained the corresponding lifetimes for other values of $\beta_s : 0 < \beta_s < 1$. The resulting values were normalized to the AML point and plotted in Figure (5.7) for comparison to our proposed model, equation (5.8). The graph shows the obtained difference is within a constant factor thereby ascertaining the validity of the proposed metric.

Models for both device and RC interconnect in a 45 nm technology were obtained from the Predictive Technology Model (PTM) online resource [49]. The metal segments were modeled as equal RC "T" networks.

For the hardware implementation, three 32-bit structures were programmed for synthesis using the Verilog HDL. First, a simple 32-bit bus segment (RAW) with input and out registers, second, another 32-bit structure with the DCE module (DCE) and third, a 32-bit bus with both DCE and DCO (DCE+DCO). These were synthesized and mapped to a 90 nm TSMC standard cell library for functional verification, area and power comparison. 32 by 32 crossbar switches and 5-bit shift registers were used to construct the rotating switch and a dedicated 10-bit counter was used to generate the global periodic inversion signal for the DCE + DCO system. This signal also doubled as the rotation-enable signal for the DCE module, therefore rotation and inversion occurred at the same time every $2^{10}$ transmissions. Compared to the DCE, which is structurally similar to the Spreading Module proposed in [46], the DCE+DCO area increases by approximately 5%. The power consumption was profiled at 79 $\mu$W, less than reported for the Spreading Module. We believe our lower rotation frequency of one rotation per 1000 compared to their
Figure 5.7: This shows the lifetime of a typical 1 cm interconnect line implemented in a 45 nm process. The model in equation (5.7) is evaluated and compared to experimentally obtained values for $0 < \beta_s < 1$. The maximum lifetime occurs around $\beta_s = 0.5$ for both plots.

1 per 100 cycles is the main source of the difference. The DCE and DCE+DC0 modules operated at 1.02GHz.

Using the SPEC Benchmark traces already profiled earlier in the document, we obtain new duty cycle distributions for the component bit lines of the Address/Data L1-Cache buses using our RAW, DCE, and DCE+DCO implementations. In MATLAB, we apply our earlier validated lifetime metric and compare the new values for Average Lifetime $L_{MB}$ and Lifetime Uniformity $L_{UB}$ obtained. Figure (5.8) is a summary of the results. The results in Figure (5.8) for I-Cache transactions illustrate the effectiveness of the DCE+DCO technique. In Figure (5.8a) and (5.8b), the Lifetime Uniformity $L_{UB}$ for Raw data flow averages below 40% across all benchmarks. The address bar for the benchmark mesa shows close to zero uniformity, this indicates that the trace contains two sets of bit lo-
cations, one set with 0-probability much higher than mean and the other much lower. DCE with a rotation period of 1024 clock cycles allows us to improve $LU_B$ to values above 95% for all benchmarks. In Figures (5.8b and d), the graph shows that even with an improved $LU_B$, the Average Lifetimes improve only slightly for the data bus (12%) and although more significantly for the Address bus (50-60%). DCE+DCO offers a better to combat NBTI degradation solution. The graphs for both $LU_B$ and $LM_B$ show over 95% recovery for address buses and 100% recovery on the I-Cache data buses. Similar trends were observed for the D-Cache but due to space constraints, only I-Cache plots were included here.
Figure 5.8: Comparing Raw Address and Data flows to the DCE and DCE+DCO solutions for programs in the SPECINT2000 Benchmark suite. Average Lifetimes are normalized to AML. Uniformity values close to 1 indicate buses with better aging profiles.
Chapter 6

Conclusion

6.1 Summary of Work

In this dissertation, we point out the new challenges in terms of energy, delay and power for Deep Sub-Micron bus and other interconnect structures. We focus on the contribution of inter-metal, coupling capacitance on line delay and energy consumption and show the inadequacies of traditional models in accounting for such effects. New DSM models are presented which create new opportunities for improving bus performance by reducing the amount of guard-banding and margin insertion necessary to achieve timing and power closure. Current techniques such as shielding, spacing, variable cycle transmission which have been proposed as solutions are effective but leave significant room for improvement.

Therefore, we show in this work that by combining the ideas of variable cycle transmission and temporal redundancy, we obtain a novel technique for delay and energy efficient data transmission for on-chip buses. To demonstrate the improvement resulting from using our technique, we analyzed data transactions on the data and address buses connecting a Level 1 (L1) cache with a microprocessor core. Trace data from a series of benchmark programs from the SPEC Integer
benchmark suite (SPEC2000CINT) were used for the experiment and the results presented. We show that when compared to the base case for data transmission, to the Shielding, Crosstalk Prevention Coding (CPC), Variable Cycle Transmission (VCT) techniques, our method, Variable Cycle Transmission with Temporal Redundancy (VCTR), achieves 31% (30%) of delay improvement along with 13% (9%) energy savings for general communication over the address (data) bus.

As interconnect variation becomes a significant component of overall variation, it may be worthwhile to explore techniques that employ bus segmentation with VCTR i.e. grouping lines with similar delays in to sub-buses with a dedicated ready signal for each sub-bus. This will allow not just crosstalk tolerance but also variation tolerance in wide on-chip buses. The main challenge in realizing such a structure will be in achieving the optimal segmentation post-silicon and designing the hardware to achieve this.

Ignoring Interconnect and its contributions to overall power, energy and delay is no longer an option in DSM chip design. In fact, achieving high throughput communication and low power chip designs will depend significantly on the ability of the designer to optimize interconnect performance. As power, delay and energy contributions of on-chip functional blocks scale with technology, a shift from function centric design to an interconnect centric approach is inevitable.

We explored offset-switched interconnect, its performance, power and area characteristics. We proposed a staggered latch bus as a simple implementation of a more general multi-phase staggered latch interconnect architecture. We performed a comparative analysis with the classical simultaneously switched interconnect. The results show that offset switching in the form of the simple SLB can achieve over 2X improvement in line delay for a given line length, segment size with no appreciable increase power, or need for extra wires.
Finally, we explored the effect of Negative Bias Temperature Instability on the performance and overall lifetime of interconnect lines. We show that threshold voltage degradation due to duty cycle skew can lead to failure in deep sub-micron buses. To combat this problem and improve lifetime in multi-bit buses, we proposed a two step solution, Duty Cycle Equalization based on load-balancing and followed by a Duty Cycle Optimization scheme based on a periodic line inversion. This approach allowed for over 98% recovery in line lifetime for typical data streams and 63% - 95% for address streams. Modest area and power increases result however, power consumption can be traded for lifetime in applications were AML lifetime levels are not necessary.

6.2 Future Work

A large part of this dissertation has been concerned with the description and illustration of novel circuit techniques for sub-50 nanometer process nodes. Whilst some progress has been made in describing the theoretical and practical aspects of the VCTR, MPSL and DCT methods, there is much scope for further investigation. In this section some aspects of the current work which could form the basis for additional investigation are examined.

6.2.0.1 Bit Level Flow Control (BLFC)

In Bit level flow control, an alternate solution to the VCTR technique is proposed. Using fine-grain control of bit placement on the bus, we can achieve similar elimination of G56 events with no extra wires, and with better overall throughput. The VCTR solution eliminates G56 events by introducing large number of dummy bits into the data stream. Although it is effective, data streams with high G56
transitions will cause throughput deterioration. We propose BLFC as an effective way to minimize the dummy bit penalty, improve throughput and eliminate extra wires used in VCTR.

The BFLC uses two kinds of bit lines: a thru bit line and a stall bit line. Each stall(thru) bit line is arranged such that both neighbors must be thru(stall) bit lines. Additionally, each stall bit line is preceded at the send-side by a fifo and followed at the recv-side by a fifo. The thru bit line transmits data every cycle and the stall line transmits conditionally in every cycle, otherwise the bit is stalled for exactly one cycle. The condition for transmission on the stall line is that a G56 transition will not occur. In the event of a stall, a dummy bit transmitted and is easily detected and eliminated at the recv-side.

BFLC is a bit-level solution as opposed to VCTR a word-level solution.

6.2.0.2 on-Axis Charge Recycling (oACR)

In on-Axis Charge Recycling, each bit-line segment is pre-charged to Vdd/2 via a charge sharing between consecutive segments in a multi-segment bit line. This is in-effect a low swing bus solution that reduces both power dissipation on transitions but also energy consumption. The reduced current density of this design leads to a dramatic improvement in Electromigration related failures. Special three mode drivers are used to short each metal segment to power, ground or to its successor segment on the same bit line. A timer cell is used switch between the two power rails and the default mode: oACR. This solution is unique in its approach and uses normal repeaters with only slight modifications. There is no need for non-zero static current sense amplifiers, specialized detector circuits etc.


Appendix

Correctness of the VCTR Approach

Let $CC(d_i, d_j)$ denote the crosstalk class of $d_i$ w.r.t $d_j$, $T(d_i, d_j)$ denote the delay for transmitting $d_j$ after $d_i$, and $E(d_i, d_j)$ denote the energy consumption during the transition from $d_i$ to $d_j$.

**Theorem.** For any pair of $n$-bit data $d_t$ and $d_{t+1}$, if $CC(d_{t+1}, d_t) \geq 5$, then $d_{t+1}$ can be encoded as $d'_{t+1}$ and $d''_{t+1}$ such that

- $CC(d'_{t+1}, d_t) < 5$ and $CC(d''_{t+1}, d'_{t+1}) = 3$
- $T(d_t, d'_{t+1}) + T(d'_{t+1}, d''_{t+1}) \leq T(d_t, d^{t+1})$
- $E(d_t, d'_{t+1}) + D(d'_{t+1}, d''_{t+1}) \leq E(d_t, d^{t+1})$

**Proof.** To prove $CC(d'_{t+1}, d_t) < 5$:

Let $tp(a, b)$ be the transition pattern over $\{\uparrow, \downarrow, -\}$ of length $n$ when an $n$-bit data $a$ is transmitted after another $n$-bit data $b$. Assume that $tp(d_{t+1}, d_t) = p_1 \ldots p_n$, where $p_i \in \{\uparrow, \downarrow, -\}$, and $1 \leq i \leq n$. Since $CC(d_{t+1}, d_t) \geq 5$, there must be at least wires which have opposite transitions w.r.t. one or both of their adjacent wires. For simplicity, assume that there are exactly two wires, i.e., wire $i$ and wire $i+1$, which have opposite transitions w.r.t. each other. Note that the proof can be easily extended to any number of wires having opposite transitions w.r.t. one or both of their adjacent wires. So, there is no pattern of type $\uparrow \downarrow$ or $\downarrow \uparrow$ in both $p_1 \ldots p_i$ and $p_{i+1} \ldots p_n$ and $tp(d'_{t+1}, d_t) = p'_1 \ldots p'_n$ and $tp(d''_{t+1}, d'_t) = p''_1 \ldots p''_n$. Since there is no pattern of type $\uparrow \downarrow$ or $\downarrow \uparrow$ in both $p_1 \ldots p_i$ and $p_{i+1} \ldots p_n$, according to the encoding technique, for $1 \leq j \notin \{i, i+1\}$, let $n$, $p'_j = p_j$ and $p''_j = -$.
\( j \in \{i, i + 1\}, p_j' \neq j = \downarrow \), and if \( p_j = \uparrow, p_j' = p_j \) and if \( p_j = \downarrow, p_j' = - \). Hence,

\[
tp(d_{t+1}', d_t) = p'_1 \ldots p'_{i-1}p'_{i+1}p'_{i+2} \ldots p'_n
\]
\[
= p_1 \ldots p_{i-1}p_i p_{i+1}p_{i+2} \ldots p_n
\] 

(1)

We know that either \( p_i' = p_i \) or \( p_{i+1}' = p_{i+1} \). In either case, \( p_1 \ldots p_{i-1}p_i' p_{i+1}p_{i+2} \ldots p_n \) does not have a pattern of type \( \uparrow\downarrow \) or \( \downarrow\uparrow \). Hence,

\[
CC(d_{t+1}', d_t) < 5.
\] 

(2)

To prove \( CC(d_{t+1}', d_t') = 3 \): We know that

\[
tp(d_{t+1}', d_{t+1}') = p''_1 \ldots p''_{i-1}p''_{i+1}p''_{i+2} \ldots p''_n
\]
\[
= - \ldots \downarrow\downarrow \ldots - \ldots -
\] 

(3)

(4)

From Table 1, it is clear that

\[
CC(d_{t+1}', d_{t+1}') = 3.
\] 

(5)

To prove \( T(d_t, d_{t+1}') + T(d_{t+1}', d_{t+1}'') \leq T(d_t, d^{t+1}) \):

Since \( CC(d_{t+1}', d_t) < 5 \) and \( CC(d_{t+1}', d_{t+1}') = 3 \),

\[
T(d_t, d_{t+1}') = 2C_LRT(1 + \lambda)
\]
\[
T(d_{t+1}', d_{t+1}') = C_LRT(1 + \lambda)
\]

But \( CC(d_{t+1}, d_t) \geq 5 \). So,

\[
T(d_t, d_{t+1}) \geq 3C_LRT(1 + \lambda)
\] 

(6)

Hence, \( T(d_t, d_{t+1}') + T(d_{t+1}', d_{t+1}'') \leq T(d_t, d^{t+1}) \)

To prove \( E(d_t, d_{t+1}') + D(d_{t+1}', d_{t+1}'') \leq E(d_t, d^{t+1}) \):

For simplicity, as in the case of the first proof, assume that there are exactly 2 wires, i.e., wire 1 and the wire \( i + 1 \), which have opposite transitions w.r.t each
other. Let $d'_i$ be the value of the $i^{th}$-wire at time instant $t$. Assume that $d'_id'_{i+1} = 01$ and $d''_td''_{t+1} = 10$ so that $d''_td''_{t+1} = 11$ and $d''_td''_{t+1} = 00$. Note that the proof can be easily extended to the case where $d'_id'_{i+1} = 10$ and $d''_td''_{t+1} = 01$.

Then

$$E(d_t, d_{t+1}) = C_L(x_1 - \lambda d'_{t+1}^{-1}) + C_L(\lambda d'_{t+1}^{i+2} + x_2)
+ C_L(1 + 3\lambda - \lambda(d'_{t+1}^{i-1} - d'_{t}^{i-1})),$$

(7)

Where $C_L(x_1 - \lambda d'_{t+1}^{-1})$ is the energy consumption of the first $(i-1)$ wires (negative component is the effect of wire $i$ on wire $(i-1)$), $C_L(1 + 3\lambda - \lambda(d'_{t+1}^{i-1} - d'_{t}^{i-1}))$ is the energy consumption of wire $i$ and wire $(i+1)$ including the effects of their other adjacent wires, and $C_L(\lambda d'_{t+1}^{i+2} + x_2)$ is the energy consumption of the last $(n-i-1)$ wires including the effect of wire $(i+1)$ on $(i+2)$.

$$E(d'_t, d'_{t+1}) = C_L(x_2) + C_L((1 + \lambda - \lambda(d'_{t+1}^{i-1} - d'_{t}^{i-1}) - \lambda(d'_{t+1}^{i+2} - d'_{t}^{i+2})))
\quad + C_L(\lambda d'_{t+1}^{i-1}) + C_L(0) + C_L(\lambda d'_{t+1}^{i+2})$$

(8)

If we assume that $E(d_t, d'_{t+1}) = E(d'_t, d''_{t+1}) > E(d_t, d_{t+1})$, after simplification, we obtain the following inequality

$$\lambda(d'_{t+1}^{i-1} - d'_{t+1}^{i+2} + d'_{t}^{i+2}) > 2\lambda
$$

(9)

Since $\lambda > 0$,

$$d'_{t+1}^{i-1} - d'_{t+1}^{i+2} + d'_{t}^{i+2} > 2$$

(10)

Which is always false. Hence

$$E(d_t, d'_{t+1}) + D(d'_t, d''_{t+1}) \leq E(d_t, d^{t+1})$$

(11)
Bibliography


Melvin I. Eze

Contact Information
1134 South Pugh Unit 5 814-777-1724
State College, PA 16801 eze@cse.psu.edu

Research Interests
Low-power IC Design, high speed on-chip interconnect, TFET circuit design, mixed-signal RF circuits

Education
Pennsylvania State University, State College, PA
Ph.D., Computer Science and Engineering, Expected: Fall 2013
- Dissertation Topic: Sub-50nm Multi-Segment Interconnect: A treatise on speed, reliability and signal integrity
- Advisors: V. Narayanan, Ph.D and Y. Xie, Ph.D and M.J. Irwin, Ph.D

M.S., Computer Science and Engineering, Aug 2007
- Topic: Delay and Energy Efficient Data Transmission in On-Chip Interconnect
- Advisor: V. Narayanan, Ph.D and Y. Xie, Ph.D

City University of New York - City College, New York, NY
B.S., Electrical Engineering, May 2004
- Cum Laude

Research Experience
Research Assistant
Microelectronics Design Lab, Pennsylvania State University
Supervisor: Vijay Narayanan, Ph.D
Research Assistant
VLSI Research Lab, City College of New York
Supervisor: Srinivasa Vemuru, Ph.D
Research Intern
Materials and Processes Lab, NASA Marshall Space Flight Center, Huntsville AL
Supervisors: Cydale Smith, Ph.D

Teaching Experience
Instructor
CSE 270 - Digital Circuits Design Laboratory
Computer Science and Engineering Department, Pennsylvania State University
Fall 2004 - Fall 2010

Co-Instructor
CSE 271 - Introduction to Digital Circuits and Systems
Instructor: Kysun Choi, Ph.D
Computer Science and Engineering Department, Pennsylvania State University
Spring 2010

Selected Awards
Fellowships and Scholarships
- GE Future Faculty Award Oct 2004
- Bunton-Waller Graduate Fellowship Oct 2005

Selected Publications