Embedding Logic and Non-volatile Devices in CMOS Digital Circuits for Improving Energy Efficiency

by

Jinghua Yang

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2018 by the
Graduate Supervisory Committee:

Sarma Vrudhula, Chair
Hugh Barnaby
Yu Cao
Jae-sun Seo

ARIZONA STATE UNIVERSITY

May 2018

ABSTRACT

Static CMOS logic has remained the dominant design style of digital systems for more than four decades due to its robustness and near zero standby current. Static CMOS logic circuits consist of a network of combinational logic cells and clocked sequential elements, such as latches and flip-flops that are used for sequencing computations over time. The majority of the digital design techniques to reduce power, area, and leakage over the past four decades have focused almost entirely on optimizing the combinational logic. This work explores alternate architectures for the flip-flops for improving the overall circuit performance, power and area. It consists of three main sections.

First, is the design of a multi-input configurable flip-flop structure with embedded logic. A conventional D-type flip-flop may be viewed as realizing an *identity function*, in which the output is simply the value of the input sampled at the clock edge. In contrast, the proposed multi-input flip-flop, named PNAND, can be configured to realize one of a family of Boolean functions called *threshold functions*. In essence, the PNAND is a circuit implementation of the well-known *binary perceptron*. Unlike other reconfigurable circuits, a PNAND can be configured by simply changing the assignment of signals to its inputs. Using a standard cell library of such gates, a technology mapping algorithm can be applied to transform a given netlist into one with an optimal mixture of conventional logic gates and threshold gates. This approach was used to fabricate a 32-bit Wallace Tree multiplier and a 32-bit booth multiplier in 65nm LP technology. Simulation and chip measurements show more than 30% improvement in dynamic power and more than 20% reduction in core area.

The functional yield of the PNAND reduces with geometry and voltage scaling. The second part of this research investigates the use of two mechanisms to improve the robustness of the PNAND circuit architecture. One is the use of forward and

reverse body biases to change the device threshold and the other is the use of RRAM devices for low voltage operation.

The third part of this research focused on the design of flip-flops with non-volatile storage. Spin-transfer torque magnetic tunnel junctions (STT-MTJ) are integrated with both conventional D-flipflop and the PNAND circuits to implement non-volatile logic (NVL). These non-volatile storage enhanced flip-flops are able to save the state of system locally when a power interruption occurs. However, manufacturing variations in the STT-MTJs and in the CMOS transistors significantly reduce the yield, leading to an overly pessimistic design and consequently, higher energy consumption. A detailed analysis of the design trade-offs in the driver circuitry for performing backup and restore, and a novel method to design the energy optimal driver for a given yield is presented. Efficient designs of two nonvolatile flip-flop (NVFF) circuits are presented, in which the backup time is determined on a per-chip basis, resulting in minimizing the energy wastage and satisfying the yield constraint. To achieve a yield of 98%, the conventional approach would have to expend nearly 5X more energy than the minimum required, whereas the proposed tunable approach expends only 26% more energy than the minimum. A non-volatile threshold gate architecture NV-TLFF are designed with the same backup and restore circuitry in 65nm technology. The embedded logic in NV-TLFF compensates performance overhead of NVL. This leads to the possibility of zero-overhead non-volatile datapath circuits. An 8-bit *multiply-and-accumulate* (MAC) unit is designed to demonstrate the performance benefits of the proposed architecture. Based on the results of HSPICE simulations, the MAC circuit with the proposed NV-TLFF cells is shown to consume at least 20% less power and area as compared to the circuit designed with conventional DFFs, without sacrificing any performance.

To my parents

for their consistent encouragement and love

To my husband Desai

for his understanding and support through all the hard times

To my cat Athena

for sleeping on my laptop

ACKNOWLEDGMENTS

I would like to thank my adviser, Prof. Sarma Vrudhula, for his faith in me, and for all his guidance and advice through the years. Especially during the frustrating time, he helped me to keep moving forward by providing both intellectual and moral support. This work would not have been possible without him. My sincere thanks to my committee members as well, for taking the time to review my work, attending my presentations, and offering many helpful suggestions.

I am fortunate to have collaborated with Niranjan Kulkarni, whose work laid the foundation of my work, without which, this work would not have progressed to the current state. I am thankful to share working space with many other colleagues who became my friends. My thanks to them for being there for me when I needed them and bring a lot of fun to the working place. My special thanks to Ankit Wagle, without his collaboration, this work would be far from complete.

My family has always been a constant source of encouragement and support through these long years. I am grateful for their trust in me, and for everything they have done for me.

I gratefully acknowledge the support I received from the following agencies: The National Science Foundation for grants #1230401, #1237856, #1701241, #0702831, the NSF IUCRC Center for Embedded Systems, and The Stardust Foundation through a Science Foundation Arizona grant SRG 0211-01.

I would also like to thank the School of Computing, Informatics and Decision Systems Engineering and School of Electrical, Computer and Energy Engineering for granting me a research assistantship, and providing necessary resources in a timely manner.

TABLE OF CONTENTS

Page

LIST OF TABLES viii

LIST OF FIGURES xii

CHAPTER

1 INTRODUCTION 1

1.1 The Challenges of Power Reduction 3

1.2 Intermittently Powered Systems 5

1.3 Emerging Non-volatile Memory Devices 9

1.4 Static CMOS Logic vs Threshold Logic 12

1.5 Research Contribution and Dissertation Outline 13

2 LITERATURE REVIEW 17

2.1 Threshold logic gate design 17

2.2 Threshold logic based synthesis 20

2.3 Non-volatile memory and non-volatile logic 22

2.4 Other non-volatile logic gate 25

3 THRESHOLD LOGIC GATE IMPLEMENTATION 26

3.1 Multi-input TLG in Digital Circuit 26

3.1.1 Circuit Architectures 27

3.1.2 Implementing Threshold Function 32

3.1.3 Delay Modeling 38

3.1.4 Energy Consumption 42

3.1.5 Robust Operation 46

3.1.6 Layout Technique 47

3.1.7 Extension: Preset, Clear and Scan 49

3.2 Single Input Threshold Gate – Differential D-flipflop 51

CHAPTER Page

3.2.1 Circuit Designs 52
3.2.2 Experiment Results 55
3.3 Circuit Implementation and Silicon Verification 63
3.3.1 Automated Design Flow with Threshold Gates 64
3.3.2 Post P&R Simulation Results 64
3.3.3 Silicon Verification 66
4 TECHNOLOGY SCALING OF THRESHOLD LOGIC GATE 80
4.1 PNAND Performance on 40nm GP Technology 80
4.2 PNAND Performance on 28nm FD-SOI Technology 83
4.2.1 28nm FD-SOI Technology 83
4.2.2 Performance and Design Challenges 86
4.3 65nm Low Voltage Operation 90
4.3.1 Threshold Logic Gate Architecture 91
4.3.2 Low Voltage Operation 92
4.3.3 Resistor Network 97
4.3.4 Cell Comparison of Energy, Delay and Area 101
4.3.5 Circuit Implementations and Comparison 106
5 ENERGY-EFFICIENT NON-VOLATILE LOGIC 109
5.1 NVL System Powered by Harvested Energy 109
5.2 NVL Design Trade-offs 111
5.2.1 The STT-MTJ Cell 111
5.2.2 Driver Sizes Ignoring Process Variations 113
5.2.3 Driver Sizes Considering Process Variations 119
5.3 Non-Volatile Flipflops with Scan 125

CHAPTER Page

5.3.1 Yield versus Energy Consumption 125
5.3.2 NVSFF Basic Structure 127
5.3.3 Non-Volatile Storage Unit (NVSU) 127
5.3.4 Non-Volatile Scan Differential Flipflop (NVSFF-DM) 131
5.3.5 Non-Volatile Scan Master-Slave Flipflop (NVSFF-MS) 133
5.3.6 Extension of Scan for Non-volatile Test 135
5.3.7 Robustness of the Restore Operation 137
5.4 Experimental Results 138
5.4.1 STT-MTJ Cell 138
5.4.2 Performance Evaluation of Proposed NVSFFs 140
5.4.3 Performance Evaluation of Circuits 142
5.5 Non-Volatile Majority Flipflop (NV-MJFF) 144
5.5.1 NV-MJFF Structure 146
5.5.2 Performance Evaluation 146
6 CONCLUSIONS AND FUTURE WORK 151
REFERENCES 154

LIST OF TABLES

Table Page

3.1 Truth table of threshold function $f = [3, 1, 1, 1 : 3]$. LIN and RIN ON transistor comparison for two signal assignments: (1)$\{\overline{a}, \overline{b}, \overline{b}, \overline{c}, \overline{c}, \overline{d}, \overline{d}|a, a, a, a, a, 1, 1\}$ and (2)$\{\overline{a}, \overline{a}, \overline{a}, \overline{a}, \overline{b}, \overline{c}, \overline{d}|a, a, b, c, d, 1, 1\}$ 35

3.2 Truth table of threshold function $f = [2, 1, 1 : 2]$. LIN and RIN ON transistor comparison for OSA and CSA. OSA: $\{\overline{a}, \overline{a}, \overline{a}, \overline{b}, \overline{c}|a, b, c, 1, 1\}$. CSA: $\{\overline{a}, \overline{a}, \overline{b}, \overline{c}, 0, 0, 1|a, a, b, c, 1, 1, 0\}$ 38

3.3 PNAND-3 clock to Q delay (C2Q)for all possible input cases. C2Q delay is split into three parts: a) input network delay (IND); b) sense amplifier delay (SAD); c) latch set delay (LSD). 42

3.4 PNAND-5 clock to Q delay split for all possible input cases. 43

3.5 PNAND-7 clock to Q delay split for all possible input cases. 44

3.6 PNAND-9 clock to Q delay split for all possible input cases. 45

3.7 PNAND cells yield with process variation. The yield is evaluated by 100,000 Monte Carlo simulation. Statistical corner provided by foundry is used in Monte Carlo simulation. Supply voltage is 1.2V and operation temperature is $25°C$. Output load is set to $20fF$. 47

3.8 65nm technology design comparison (schematic). The simulation is done on slow/slow corner, 1.1V VDD and $105°C$. The load cap is $20fF$. Signal transition time is $70ps$. 57

3.9 65nm technology design comparison (layout). The simulation corner is slow/slow, 1.1V VDD and $105°C$. The load cap is $20fF$. Signal transition time is $70ps$. 57

Table Page

3.10 28nmRVT technology design comparison (schematic). The simulation corner is slow/slow, 0.9V VDD and 125°C. The load cap is 7.5fF. Transition time of all signals is 65ps. 58

3.11 28nmRVT technology design comparison (layout). The simulation corner slow/slow, 0.9V VDD and 125°C. The load cap is 7.5fF. Transition time of all signals is 65ps. 58

3.12 40nmGP technology design comparison (layout). The simulation corner typical/typical, 0.9V VDD and 25°C. The load cap is 7.3fF. CLK transition time is 110ps and input transition time is 4.64ps. 59

3.13 40nmGP technology design comparison (layout). The simulation corner typical/typical, 0.9V VDD and 25°C. The load cap is 7.3fF. CLK transition time is 110ps and input transition time is 216.6ps. 60

3.14 40nm technology design comparison (layout). The simulation corner typical/typical, 0.9V VDD and 25°C. The load cap is 7.3fF. CLK transition time is 110ps and input transition time is 806.7ps. 60

3.15 Breaking capacitor experiment(schematic and layout). The simulation corner is the same as other tables. All four circuits have zero out of 100,000 MonteCarlo functional failures on foundry provided statistical corner. 63

3.16 Results of transformation in Fig. 3.18($C = \Phi$) 65

3.17 65nm LP technique mapping improvements of hybrid over conventional for various circuits 67

3.18 Test results of conventional and hybrid multipliers 73

3.19 Test results of booth multipliers 77

Table Page

4.1 Setup time vs clock transition time of PNAND-3, 5, 7 and 9 in 40nmGP technology. Output load is set to $7.3fF$. The simulation corner typical/typical, 0.9V VDD and $25°C$. 82

4.2 C2Q delay vs clock transition time and output load of PNAND-3, 5, 7 and 9 in 40nmGP technology. The simulation corner typical/typical, 0.9V VDD and $25°C$. 83

4.3 TLL failures in 100K MC simulations without resistor network 93

4.4 Comparison of functions 103

4.5 Area Comparison(μm^2) 108

5.1 STT-MTJ parameters. 139

5.2 Mean and standard deviations of STT-MTJ resistances versus t_{ox}. The mean of random variable t_{ox} is set to two values, $.85nm$ and $.8nm$, with sigma equal to 3%, 5% and 10% of $\bar{t}_{ox}$. 139

5.3 Performance of NVSFF-MS, NVSFF-DM and SFF-MS. The average energy is based on 30% input switching activity. Simulation conditions are: $25°C$, 0.9V, TT corner, and output load of $3fF$. 140

5.4 Delay and energy consumption of restore from NVSU to output Q. 141

5.5 Comparison of non-volatile flipflop with prior reported data. a) Ref. Cai *et al.* (2015), b) Ref. Ryu *et al.* (2012). 141

5.6 Comparison of backup schemes. (a) and (b) use single backup time for all dice, and (c) refers to chip-specific backup time. (b) and (c) include variations in both CMOS and MTJ. 142

5.7 Comparison of logic cell count and area using different flipflops in MAC and adder. 145

5.8 The performance comparison between NVFF-MS and NVFF-DMs. The average energy is based on 30% input switching activity. The simulations is done under $105°C$, 1.1V, SS corner. Output load is set to $20fF$. 148

5.9 MAC combinational cell count and area comparison. 32 flipflops are not included in cell count. 8 NV-MJFF are included in the third circuit. 149

LIST OF FIGURES

Figure Page

1.1 Trends of major sources of power dissipation in nano-CMOS transistor Abbas and Olivieri (2014) 6

1.2 Threshold logic gate 14

3.1 Threshold gate circuit block 27

3.2 Schematic of a TLL circuit. 28

3.3 Basic PNAND architecture 30

3.4 Transistor M_9 and M_10 would prevent unexpected state flips in the same clock period 31

3.5 PNAND-n C2Q and partial delays for 2:1 input case 40

3.6 Partial and total delay of PNAND-7 for multiple input cases 41

3.7 Partial and total delay of PNAND-9 for multiple input cases 42

3.8 Range of energy consumption for PNAND-3, 5, 7 and 9. 44

3.9 Layout of PNAND-9 in double height standard cell format. 48

3.10 PNAND-9 layout with M2 shield on the top of N5 and N6. 50

3.11 PNAND cell with scan, asynchronous preset and clear function. 51

3.12 Edge triggered flipflop design: master-slave D-flipflop (DFF) 52

3.13 Differential Sense-Amplifier flipflop(SAFF) with SR latch 53

3.14 Single-input TLG (TLG-1) with SR latch 54

3.15 The schematic of improved single-input threshold gate (KVFF) without latch 55

Figure Page

3.16 Setup time and hold time distribution comparison on 65nm designs. 100 MonteCarlo simulations are applied on foundry set statistic corner, 1.2V VDD and $-40°C$. The data point center is the mean value (hold time, setup time) and the vertical/horizontal bar is the standard deviation. 61

3.17 The breaking capacitor experiment to determine the node reliability against radiation and coupling noise: (a) The test circuit; (b) Signals applied in the test. 63

3.18 Hybridization example: A threshold function replaced by PNAND 65

3.19 Synthesis and hybridization steps 66

3.20 Test chip architecture with Wallace tree multiplier and original PNAND cell array 69

3.21 PNAND cell array structure 70

3.22 Die photo of prototype chip:(1) Conventional multiplier; (2) Hybrid multiplier; (3) Clock generator 71

3.23 Input sequences for single test round 71

3.24 Test chip with Booth multiplier and improved PNAND cell array 72

3.25 EnClk signal is included in test sequence to remove glitches during clock transition 72

3.26 Hybridization of multiplier 73

3.27 Multiplier dynamic power 74

3.28 Measured energy delay product (EDP) vs frequency over 19 dies. Mean and $\pm 3\sigma$ boundary are shown for EDP, mean and $\pm\sigma$ boundary are shown on improvements below. 75

3.29 Multiplier dynamic power for booth multiplier 78
3.30 Measured energy delay product (EDP) vs frequency over 24 dies. 79
4.1 The total delay of DFF and PNANDs vs input transition 84
4.2 Bulk vs FD-SOI technology STMicroelectronics (2018). 85
4.3 Use ground plane implant adjustment for RVT and LVT transistors in FDSOI technology. 86
4.4 DFF performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C. 87
4.5 PNAND-3 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C. . 88
4.6 PNAND-5 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C. . 89
4.7 PNAND-7 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C. . 89

Figure Page

4.8 PNAND-9 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C. . 90

4.9 Overall performance comparison between CMOS and threshold cell. a) Overall delay comparison between DFF and PNAND-3. b) Energy delay product comparison between a hybrid circuit and its CMOS equivalence for 5-input function y=[22111;4]. The hybrid circuits consists of inverters and a PNAND-7 while the CMOS equivalent circuit consists of standard logic gates and a DFF. 91

4.10 Average leakage power for PNAND cells and DFF with same drive strength. 92

4.11 Schematic of a TLL circuit with resistor network. 93

4.12 Simplified input network of TLL 94

4.13 TLL functionality vs R 95

4.14 RRAM lifetime vs stress voltage 99

4.15 TLL-7 yield with RRAM 100

4.16 (a) Programming RRAM (b) 3-D structure 100

4.17 A 16 inputs 1-bit sorter 106

4.18 Energy × Delay(fJ·ns) of Sorter 107

4.19 Energy × Delay(fJ·ns) of Comparator 108

5.1 (a) Simplified driver circuit providing bidirectional current to switch STT-MTJ cell; (b)The structure of STT-MTJ 111

5.2 $I_{d,01}$ (blue line) and $I_{d,10}$ (red line) vs normalized width. $I_{c,01}$ and $I_{c,10}$ are also included. $\bar{t}_{ox} = 0.8nm$ 114
5.3 Driver current versus transistor width Case I. 116
5.4 Driver current versus transistor width Case II and Case III. 117
5.5 Driver current versus transistor width Case IV and Case V. 118
5.6 Frequency histograms of R_L and R_H using 10K MonteCarlo samples. Data for CMOS is from a 40nm commercial library with foundry supplied parameters and HSPICE models. Data for MTJ variations was generated assuming $\bar{t}_{ox} = 0.8nm$ and $\sigma_{tox} = 0.1\bar{t}_{ox}$, and using models in (Wang *et al.* (2014); Zhang *et al.* (2015)). 121
5.7 Frequency histogram of I_d using 10K MonteCarlo samples. MonteCarlo configuration is the same as 5.6. 122
5.8 I_d current vs driver width. Blue line (dots) is $I_{d,10}$, Red line (dots) is $I_{d,01}$. Lines are with no process variation, dots are currents with t_{ox} and CMOS (local and global) variation. Note: To avoid clutter, only a subset of widths are plotted. 123
5.9 Average total energy versus driver width, for different yields, accounting for process variations. Minimum energy is achieved with $W_{2,Emin} = W_{2,ub}$ when no variation is included. In the presence of process variations, yield constrained minimum energy can be achieved with smaller $W_{2,Emin}$, whose value depends on the target yield. 124
5.10 The basic structure of non-volatile scan flipflop (NVSFF). 127
5.11 The schematic of NVSU. The NVSU includes a write buffer, two STT-MTJ devices and a state sense amplifier. 128

5.12 The control signal sequence during non-volatile test mode. 130
5.13 Schematic of the NVSFF-DM. The tri-state buffers between NVSU and SR latch are not shown. 132
5.14 Schematic of NVSFF-MS. 134
5.15 Non-volatile scan test procedure. N is number of flipflops in design. 136
5.16 GBT: Single, global backup time, PFT: post-fabrication tuning. Core energy is the same for GBT and PFT. For achieving high yield, the energy wastage with PFT is much less than GBT. 137
5.17 8-bit MAC unit. It includes input and output flipflops, a synchronous reset and FMA (fused multiply-add) unit. 143
5.18 MAC total energy vs input switching activity under normal operation. The simulation is done by PTPX under 25°C typical corner. 144
5.19 32-bit adder total energy vs input switching activity under normal operation. The simulation is done by PTPX under 25°C typical corner. 145
5.20 Structure of NV-MJFF 147
5.21 MAC energy breakdown vs input switching activity, the simulations is done under 25°C, 1.2V, TT corner. Numbers in *italics* denote totals. . . 150
5.22 MAC total energy vs input switching activity under normal operation. The simulation is done by HSPICE under 25°C typical corner 150

Chapter 1

INTRODUCTION

Over the past fifty years, CMOS technology has been miniaturized by nearly four orders of magnitude, spread over twenty process generations. This went hand-in-hand with efforts to reduce power consumption at the device, circuit, and architecture levels. One of the far reaching consequences of scaling below 32 nanometer has been the inability to increase clock frequency due to the exponential increase in power consumption. When this situation was reached nearly two decades ago, the microelectronics industry undertook a paradigm shift in computing – moving from single core processors to multi-core processors. This allowed operating each core with lower clock frequencies, while improving performance (system throughput) by increasing the amount of concurrency or parallel computation. Since its adoption in early 2000, the multi-core strategy was supremely successful, enabling the rise of *cloud computing* with massive numbers of high performance server farms, and the proliferation of high performance mobile systems with the ubiquitous *smartphone*.

For all practical purposes, device scaling has almost stopped, and is expected to end within the next three to five years. Furthermore, the multi-core strategy has also reached its end, and no significant additional performance gains can be expected by simply increasing the number of cores. Yet, at the same time, there are two new paradigm shifts in computing are beginning to take place that will have a disruptive influence on the microelectronics industry. One of them is the explosive rise of applications based on new computation paradigms such as *machine learning* that require the processing of massive amounts of data in real-time, and the other is the natural evolution of embedded systems referred to as the "*Internet of Things*"

(IoT), in which sensing, computing, actuation and communication functions are all to be integrated into one or a few chips that can be embedded in almost any object on the planet (home appliances, buildings, roads, jet engines, automobiles, etc.). These two new developments will require either further exponential improvements in performance without the concomitant increase in power consumption, or ultra energy-efficient devices that can function with harvested energy and remain operational for many decades.

The vast majority of digital circuits employ a tried and tested *style* of logic – commonly referred to as static CMOS logic (SCMOS). Its dominance is due primarily to its high robustness and (ideally) near zero static power dissipation. A digital circuit implemented using SCMOS logic is a multi-level circuit comprised of network of combinational logic cells and sequencing elements (latches or flip-flops). Each logic cell is a complementary structure that computes a scalar (i.e., single output) Boolean function of its inputs, by establishing a conducting path between one of the two supply rails to its output. In spite of all the changes that have taken place in digital microelectronics, SCMOS logic has been the dominant design style for the past four decades. As a result, techniques for reducing the dynamic and standby power in such circuits have been thoroughly investigated, and have been incorporated into modern design practices and design tools. Examples of techniques to reduce dynamic power include logic synthesis and restructuring to reduce switching activity, gate sizing, technology mapping, retiming, and voltage scaling. The use of dual supply and device threshold voltages, dynamic control of body bias, clock and power gating, etc., are some of the well known ways to reduce standby power. Thus it appears that there is little or no additional opportunities left for improving the performance and power of CMOS digital design. This is indeed the case for CMOS combinational logic. However one aspect of digital circuits that has not changed is the sequential

components in a design, i.e., the flipflops, which serve as sequencing elements in datapaths and control logic. It is this aspect that this dissertation explores.

1.1 The Challenges of Power Reduction

It is known that the power consumption of a digital design consists of dynamic power and static power. Dynamic or switching power is due to charging and discharging of capacitive loads. It's power consumption P can be represented by the product of energy per transition and transition rate, as shown in Equation 1.1. $C_L \cdot V_{dd}^2$ is the energy stored in the output load C_L, and the transition rate $f_{0\to1}$ is the frequency of output transitions between 0 to 1. f_{CK} is the clock frequency and SA is the switching activity.

$$\begin{aligned} P &= \text{Energy/transistion} \cdot \text{transition rate} \\ &= C_L \cdot V_{dd}^2 \cdot f_{0\to1} \\ &= C_L \cdot (V_{dd}^2/2) \cdot f_{CK} \cdot SA \end{aligned} \tag{1.1}$$

Reducing the dynamic power is accomplished by reducing the impact of each term in Equation 1.1. Due to the quadratic dependence on voltage, voltage scaling has the greatest impact on reducing the dynamic power. However, this strategy is not sustainable with technology scaling. The reason is that transistor threshold voltage V_t can not be scaled by the same factor as a transistor's physical dimensions. First, lowering V_t can result in an exponential increase in leakage current. However, maintaining a V_t while reducing the supply voltage V_{dd}, reduces the gate overdrive $V_{dd} - V_t$, which reduces the circuit's speed. An alternative is to use multiple supply voltages to minimize the performance degradation. This results in several voltage islands cross the system. Some voltage islands which lie on the critical path are powered by a high voltage to boost performance while other non-critical parts are powered by a lower voltage in order to save on power consumption. The amount of

power saving depends on the ratio between critical and non-critical paths and the voltage partitioning granularity. Furthermore, the complexity of the design increases substantially due the the need for routing multiple supply lines and voltage shifting between islands.

Another way to reduce the dynamic power is to reduce the output load C_L, which includes the parasitic capacitances associated with the transistors and the interconnect. Parasitic capacitances of devices are reduced with scaling, reducing their switching delay. In the meantime, thinner and closer wiring at lower geometries results in higher parasitic resistances and coupling capacitances. At 40nm and below, wire delay dominates the total delay between logic gates. Parasitics of logic gate is determined by standard cell layout. A careful layout can minimize parasitics and improve gate performance. DRC rules become more complicated, and major innovation on layout is less likely on lower geometry. Synthesis and P&R tools can also help to reduce C_L by optimizing mapping and routing algorithm.

Yet another way of reducing the dynamic power of a SCMOS circuit is to reduce the swiching activity (SA) on gate outputs, which in turn reduces the frequency of charging and discharging interconnect and gate capacitances. There is no general method to reduce the SA of a circuit, as it is mostly data and structure dependent. One source of switching that can be minimized is due to glitching – spurious transistions that are caused by unequal delays on paths terminating at inputs of logic gates. In SCMOS, this can be minmized by balancing signal paths.

Dynamic power is linearly proportional to the clock frequency f_{CK}. Frequency scaling and clock gating are applied to eliminate unnecessary switching in circuit. At the system level, clock frequency can be reduced for non-critical tasks, and in the extreme case, reduced to zero by gating the clock signal in a sleep mode. Dynamic voltage and frequency scaling is a power management technique that combines V_{dd}

and f_{CK} scaling to optimize trade-offs between power and performance in computer system.

Static or sub-threshold leakage power is the power that is dissipated when there is no signal activity. It is the result of current flowing through a transistor even when its gate voltage is below the threshold voltage and the transistor is supposed to be comptelely turned off. In bulck CMOS, it increases exponentially with scaling and in small geometry devices, equals or even exceeds the dynamic power as shown in Fig. /reffig:powerscaling. Sub-threshold leakage also depends on design and fabrication parameters such as channel length, doping, gate oxide thickness, etc. Abbas and Olivieri (2014). Equation 1.2 shows how drain leakage current is related to threshold voltage and transistor voltage bias. I_{s0} is the sub-threshold saturation current, n is the sub-threshold slope factor whose value is around 1, q/kT is thermal voltage. Lowering V_t would cause exponential grows on leakage current. FinFet and UTBSOI are implemented to suppress leakage path in body. UTBSOI has back-gate bias which can be used for V_t tuning.Logic gates with multiple V_t can also be mixed to boost critical path performance while suppress leakage and power using slack on non-critical path. In systems where leakage dominates, power gating can also be applied on partially or on the whole system to cut-off leakage in deep sleep mode. In such situations, all data in gated units would be lost when power gating is applied. Therefore additional steps to restore the data are needed before computation can be resumed.

$$I_D = I_{s0} \cdot e^{(V_{gs}-V_t)q/nKT}(1 - e^{-V_{ds}q/kT}) \tag{1.2}$$

1.2 Intermittently Powered Systems

All the methods mentioned above have already been implemented in digital system design. However, the rapidly growing transistor count in microprocessor and the

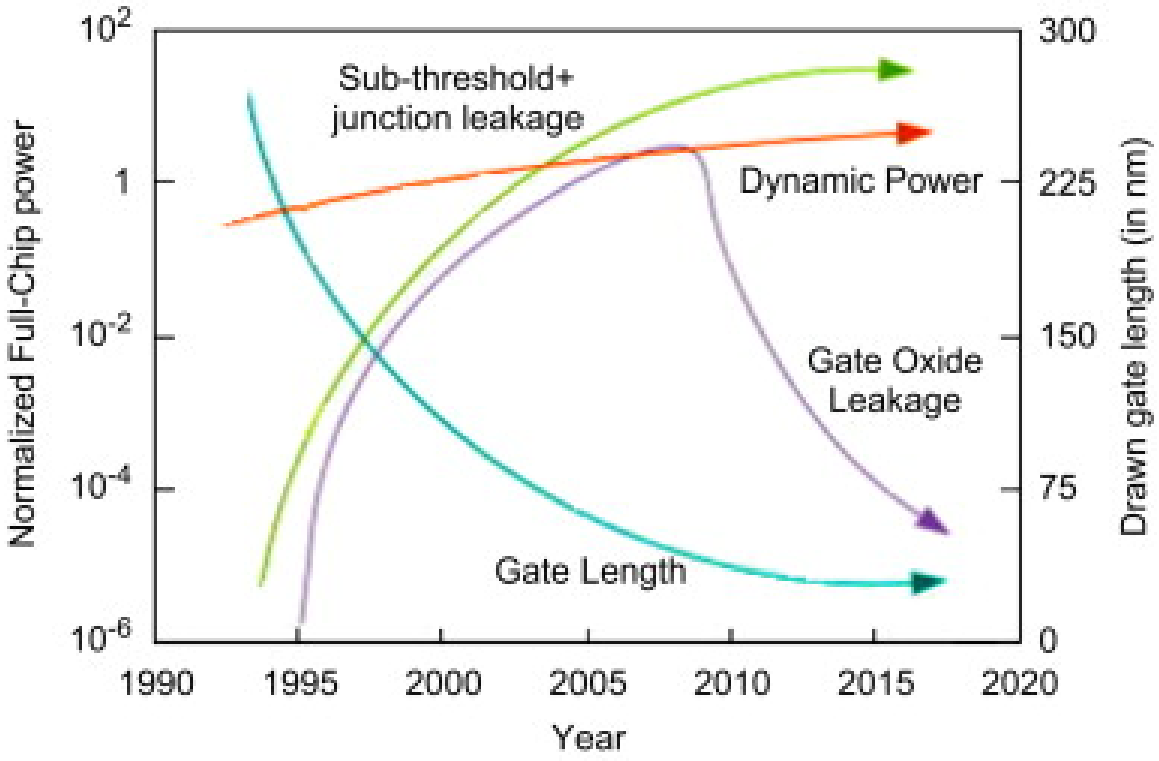


Figure 1.1: Trends of major sources of power dissipation in nano-CMOS transistor Abbas and Olivieri (2014).

demanding of increasing battery-life as well as keeping cost low drives chip design towards a new challenge. In some implementation like large scale IoT network, providing computing energy to each object is very costly, especially for devices located on remote source. In this case, prolong the computing time by providing alternate energy source is necessary. Circuit that obtain their energy from ambient energy sources are proposed to release this problem. Some of the more common ambient energy sources (AES) include solar, piezoelectric, vibration, airflow, and thermoelectric Priya and Inman (2008). However, the energy that can be harvested is highly unreliable in terms of magnitude, magnitude variation and variation in timeMa *et al.* (2015). The energy that can be tapped is usually only a very small fraction of total transmitted energy. System powered by these energy resources have to be designed to tolerant intermittent power lose.

Current digital systems are generally architected for continuous operation. The logic circuits and memory such as SRAM or DRAM is volatile. i.e., the information (state of the computation and state of memory) is lost when the power supply is disrupted. Batteries or super-capacitors can be applied to smooth out the rapid changing of power supply. However, they usually require long charging time and would increase mass, area, and cost of system Rodriguez Arreola *et al.* (2015). Recent approaches remove the middle energy storage part and directly powered the system by AES. Therefore, accurately predicting an impending power disruption, and saving the state to non-volatile memory (NVM) is critical for all but the simplest devices. Three state-of-art systems are proposed to address this new challenge.

1. **Mementos Ransford *et al.* (2011)** assumes implementation of RFID-scale device which has two segments of flash memory. It inserts trigger point at compile time on three mode: loop-latch mode that trigger point is placed at each loop latch; function-return mode that it is placed after function call; timer-aided mode, the trigger point is only placed when timer flag is up. At each trigger point, Mementos estimates remaining energy by sample voltage from on-chip ADC and compare it with checkpoint threshold voltage. Voltage lower than threshold would trigger a checkpointing procedure. Checkpointing procedure carefully backed up registers, stack pointer, program counter with proper header and tailer, then mark other checkpoint for erasure. When energy is plentiful, state would be recovered from active checkpoint, then the whole segment would be marked erasable.

2. **QUICKRECALL Jayakumar *et al.* (2014)** is based on FRAM instead of flash memory, which significantly reduced backup and restore time. FRAM is used as unified memory in QUICKRECALL. During operation, it acts as

the conventional RAM as well as ROM. Unlike Mementos which insert trigger point during compile time, QUICKRECALL checkpointing is triggered by ISR. A comparator monitors vdd by comparing its value with trigger voltage. When vdd is lower than trigger voltage, it would generate a digital signal and issue ISR. When low voltage interrupt occurs, the system only need to backup general purpose registers (GPRs), status register (SR), stack pointer (SP) and program counter (PC). compare with Mementos, checkpointing overhead is significantly reduce.

3. **Hibernus Balsamo *et al.* (2015)**: Similar as QUICKRECALL , checkpointing is triggered by comparator in Hibernus. Instead of one trigger threshold voltage, Hibernus has two threshold voltage. V_H is the threshold voltage for hibernate, and V_R is the threshold voltage for restore. V_R is set higher to add hysteresis, allowing the system to restore without taking the VDD below V_H. Hibernus use normal RAM and registers for active operation, and use FRAM as backup memory only. Hibernate process would push registers and entire RAM to FRAM first, then general registers, and finally the SP and PC.

In all these systems, a center non-volatile memory is assumed in the system. During backup and restore, data would be transferred between non-volatile memory and the place they are actually used. The trigger voltage should be set such that the energy stored in parasitic and decoupling capacitors are higher than the energy required to complete the whole backup process. In small device that total capacitance is not enough, extra capacitor like supper capacitor would be added to the system.

Another potential circuit architecture for intermittent power system is non-volatile logic (NVL). With the help of emerging non-volatile technology, it is possible to integrate logic circuit with non-volatile device to form non-volatile logic. NVM is

close to the logic circuit that do computing, is would be able to avoid data movement. Compare with conventional solution, NVL has potential to consume less energy and delay during backup and restore. Instantaneous backup and restore with low power operation would be very important for energy harvesting IoTs.

1.3 Emerging Non-volatile Memory Devices

Normal digital systems are designed to operate with continuous power supply. During the operation, memory like SRAM or DRAM would receive computation result and feed input data to processor. The stored data would be lost when power-off. All data that need to be stored during power-off would be moved into non-volatile memory. Non-volatile memory (NVM) is a type of memory that can sustain its storage information even when power is turned off. The traditional NVMs include hard disk drive, optical disk, read-only memory, flash memory, etc. This NVMs have disadvantages like limited write endurance (number of program cycles), long access time and high write/read energy cost. Recently, more emerging NVM technologies become popular in both industrial and academia. Comparing with traditional ones, the emerging NVMs provide much more advantages such as zero leakage power, high density and better technology scaling. In this section, several emerging NVM techniques are listed.

1. **Floating gate transistor or flash memory**: Floating gate transistor in flash memory is the most mature technology for flash drive and solid state drive that is already under volume production. The information is stored as amount of trapped charge in the floating gate. The density of flash memory is increased through 3D array processing and multi-bit storage. However, other merits such as performance, endurance, data retention time and energy efficiency decline substantially in technology scaling Grupp *et al.* (2012), which leaves doubt if

flash memory can be adapted to fit requirements of future technology.

2. **RRAM (Resistive random-access memory)**: RRAM is two terminal device with dielectric material as insulating film. Information is stored as resistance values, which can be changed by providing current flow through opposite direction. Comparing with SRAM, RRAM has higher density, similar read latency and lower leakage power Mittal *et al.* (2015).

3. **CBRAM (Conductive bridging memory) or PMC(Programmable metallization cell)**: CBRAM is a registered trademark for PMC technology. PMC is also a two terminal resistive memory technology, including an active metal (Ag, Cu, etc) terminal, inert metal (Ni, Pt, W, etc) terminal and a solid electrolyte thin film (Ag-doped chalcogenide $Ag-Ge_{30}Se_{70}$) sandwiched in between Mahalanabis *et al.* (2014); Valov *et al.* (2011). The PMC can be back-end integrated with standard CMOS technology. It has many attractive properties such fast write speed, low write energy and high reliability.

   PCM is a two terminal resistive memory that use the crystallization property of chalcogenide glass to storing data. The resistance value is depends on the material state, and can be changed between low resistance state (crystalline) and high resistance state (amorphous)by current pulses. PCM cell typically has 1T1R structure, which is much smaller than SRAM and DRAM cell. However, the high write current density, long write latency and limited endurance ($10^9 - 10^{12}$ write cycles) still make big challenges for PCM to be extensively applied on mobile devices .

4. **PCM (Phase change memory)**: PCM is a two terminal resistive memory that use the crystallization property of chalcogenide glass to store data. The resistance value depends on the material state, which can be changed between

low resistance state (crystalline) and high resistance state (amorphous)by applying current pulses. PCM cell typically has 1T1R structure, which is much smaller than SRAM and DRAM cell. However, the high write current density, long write latency and limited endurance ($10^9 - 10^{12}$ write cycles) still make big challenges for PCM to be extensively applied on mobile devices .

5. **FeRAM (Ferroelectric memory)**: FeRAM cell has similar 1T1C structure as DRAM except that it uses a ferroelectric layer in its capacitor. In ferroelectric material, applied electric field doesn't change linearly with stored charge. The information is stored in the polarization of ferroelectric material. Even though FeRAM has high endurance ($\sim 1^{16}$), its low density and high manufacturing cost make it less attractive on next generation technology Endoh *et al.* (2016).

6. **MRAM (Magnetoresistive memory)**: MRAM uses magnetic tunnel junction (MTJ) as non-volatile bit storage. The endurance can be as high as FeRAM. Couple of methods can be applied on flipping magnetization. Among them, spin transfer torque(STT) is widely explored recently due to its high write endurance, high density, high speed and relative low write energy Mittal *et al.* (2015); Fong *et al.* (2012). STT base MRAM is also referred as STT-MRAM, the memory cell is referred as STT-MTJ.

7. **DWM (Domain wall memory)**: DWM device is a three terminal devices with two parts: ferromagnetic nanowire containing domain wall and MTJ for state reading Currivan-Incorvia *et al.* (2016). The domain wall can be displaced by current flow through the nanowire, the magnetization under the MTJ decides the resistance state of MTJ. In theory, DWM device can store multi-bit data by containing more than one domain walls. The research of DWM is still on preliminary stage. It physical mechanism and device characteristics are still

need to be explored.

1.4 Static CMOS Logic vs Threshold Logic

Realizing systems with low power consumption and high performance is the main research topic in recent years, especially on mobile applications. Large amount of techniques have been thoroughly studied to balance performance and power consumption, including voltage scaling, power gating, clock gating, etc. Further optimization on digital circuit becomes very difficult since most possible techniques have been already tried. The only exception is how logic function is computed. In static CMOS logic gate, the boolean function is evaluated by establishing a conducting path from VDD (GND) to the output though a stack of PMOS (NMOS) transistors.The commercial digital circuit synthesis and optimization tools are all based on static CMOS logic. An alternate logic family called threshold logic provides another promising way to compute boolean function, which has the potential to realize the same function with a more compact circuit. By integrating threshold logic gates into main stream automated design flow, we demonstrated that further power and area reduction is feasible without scarifying performance Kulkarni *et al.* (2012); Yang *et al.* (2015a); Kulkarni *et al.* (2016).

Threshold logic is represented by a kind of computing units that evaluate threshold function. A Boolean function $f(x_1, x_2, \cdots, x_n)$ is a threshold function if there exist n weights $(w_1, w_2, \cdots, w_n)$ and a threshold T such that

$$f(x_1, x_2, \cdots, x_n) = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} w_i x_i \geq T \\ 0 & \text{otherwise.} \end{cases} \tag{1.3}$$

Without loss of generality, we assume that w_i and T are both integers. The function can also be represented by $[w_1, w_2, \cdots, w_n; T]$ or simply $[w; T]$ where w is

weight vector. Threshold logic is a (small) subset of boolean function. For example, $f(a,b,c) = a \vee bc$ is a threshold function, as $f(a,b,c) = a \vee bc \equiv 2a + b + c \geq 2 \equiv [w_a, w_b, w_c; T] = [2,1,1;2]$. On the other hand, a very similar function $f(a,b,c,d) = ab + cd$ is not a threshold function. Because the general primitives (AND/OR) are threshold functions, a general boolean function can be represented by cascaded threshold functions. The extensive discussion of threshold functions in literature dated back to 1950s Muroga (1959, 1971). More applications on discrete neural computing is discussed in Siu *et al.* (1995)

A non-decomposable primitive circuit that evaluates threshold function is called threshold logic gate(TLG). As shown in Fig. 1.2, TLG contains n inputs and one output. A TLG computes the weighted sum of all input physical quantities(charge, current or voltage) and evaluates output based on Equation 1.3. Comparing with general AND/OR gates, TLG has more compact computing capability. To realize common arithmetic function with minimum depth, the size of TLG network is polynomial bounded, comparing with AND/OR gate network which are exponential in size Siu *et al.* (1995). With efficient TLG circuit design, same amount of power and area saving can be expected when implementing large scale circuit by threshold logic based network.

1.5 Research Contribution and Dissertation Outline

In this dissertation, different circuit architectures and design strategies for threshold logic gates are described. The contributions are listed as follows:

1. Designed and evaluated a robust threshold logic gates known as PNAND-n in 65nm. Multiple techniques were involved in PNAND library design, including delay modeling, design trade-offs between energy and delay, symmetric layout floor plan and etc.

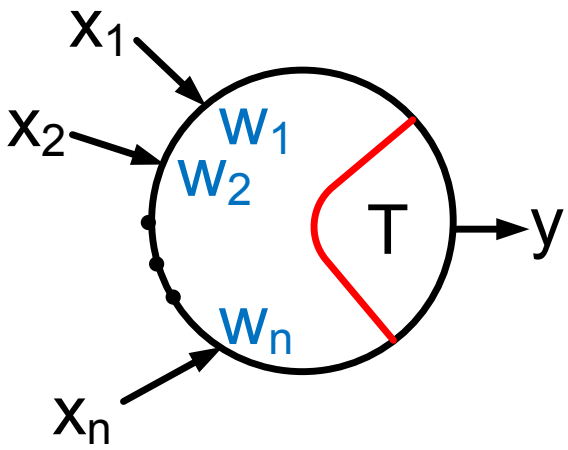


Figure 1.2: Threshold logic gate

2. Designed single input threshold gate KVFF in 65nm. The evaluation of KVFF against D-flipiflop shows its advantage on setup time and total delay.

3. Evaluate the PNAND library and threshold logic synthesis flow on silicon. Two chips were fabricated and tested in 65nm. both test and post-layout results show consistant improvement on area and power consumption in threshold logic based hybrid circuits.

4. A theshold logic gate architecture called TLL was combined with emerging non-volatile device RRAM in order to operate TLL on supply voltage as low as 0.6V. RRAM is a resistive device. On low supply voltage, integrate RRAM with input network increases the rise time margin during evaluation phase, which significantly improves the reliability of sensing difference between left and right input network value.

5. A thoroughly study on non-volatile logic provided a deep insight of how design parameters impact backup energy and yield. When considering process variations, systematic sizing algorithm was proposed to minimize backup energy waste as well as achieving desired yield constraint.

6. Non-volatile flipflop and non-volatile threshold gates were designed according to the proposed strategy. Scan mechanism in flipflop was extended to include non-volatile scan test. The extended scan mechanism could provide optimal backup time on per chip base after fabrication is done. Implementing post fabrication backup tuning can save as high as 78% energy per bit comparing with determine global backup time before fabrication.

The outline of the dissertation is as follows.

1. Chapter 2 describes the earlier work on the design and usage of threshold logic gates in circuit design. It also covers earlier work on threshold logic based synthesis algorithm

2. Chapter 3 describes threshold logic gate architectures and hybrid circuit implementations. Both single input and multi-input sequential threshold gates are designed. The circuit evaluations include delay modeling, energy consumption, robustness check and layout refining. Synthesis algorithm are briefly explained in this chapter. Post P&R simulation and silicon verification results are shown by applying the proposed gates, showing significant area, leakage, and energy reduction comparing with conventional design.

3. Chapter 4 describes integrating RRAM with threshold logic gate for low voltage application.

4. Chapter 5 describes non-volatile flipflops and non-volatile threshold gate design. Non-volatile flipflop is the basic block of NVL system powered by harvested energy. The non-volatility are realized by integrating STT-MTJ unit into flipflop and PNAND. Detailed analysis was done on backup driver circuit with and without considering process variation and an optimized sizing methodology is

proposed to minimize energy without degrade yield requirement. Post fabrication tuning can further reduce the energy by searching backup time by scan chain. Circuit implementations are also included to demonstrate their performance on normal operation.

5. Chapter 6 concludes the dissertation.

Chapter 2

LITERATURE REVIEW

In this chapter, we are going to review some major works been done on threshold logic and non-volatile computing.

2.1 Threshold logic gate design

Exploration of circuit design of threshold logic gates (TLG) have been ongoing since the late 1960s. More than fifty different implementations of TLGs have been reported in Beiu *et al.* (2003). They can be generally classified as being *capacitance* based and *current* based or *conductance* based. Threshold function can also be implemented by using complex CMOS logic Sobelman and Fant (1998) or pass transistor logic Quintana *et al.* (2001). These cells usually compute specific functions such as a majority function or m-of-n function. These gates have low power and large noise margin, and relatively fast with small fan-ins. However, the delay and complexity of the cell would dramatically increase with increase of fan-ins.

Capacitance based or capacitive TLG uses capacitor array to compute weighted sum in Equation 1.3. There are two group of capacitive TLG designs, switched capacitor TLG (CTL) and neuron MOS (νMOS) TLG. CTL used switched capacitor circuit idea implemented in analog circuit to compute threshold function, which has a very regular structure. Instead of a complex analog amplifier, CTL used a saturated inverter to compute the output. Offset cancelling technique was applied in CTL such that fan-ins as large as 255 can be applied Ozdemir *et al.* (1996). νMOS TLG is a capacitive TLG integrated with neuron MOS transistor. This transistor has multiple inputs which are coupled with a buried floating poly-silicon gate. The total voltage

applied on the floating gate can be represented as $V_F = (\Sigma_{i=1}^{n} C_i \cdot V_i)/C_{\text{tot}}$. The output is set to 1 when V_F is higher than threshold. νMOS TLG can be implemented as static or clocked gate Lashevsky *et al.* (1998), or implemented with clocked differential sense amplifier Luck *et al.* (2000). The threshold was determined by inherent inverter switching voltage or could be programmed by applying different voltage references. Both of two group of architectures show large power consumption as well as area and delay, making them less attractive on digital applications.

Current/conductance based TLGs are usually faster than capacitive TLGs. Early implementations include pseudo-nMOS and output-wired inverters. However, both have DC current during operation. Multiple solutions were proposed to reduce DC current. DC power in pseudo-nMOS can be reduced to 14% by dynamic feedback current flow control Kartschoke and Rohrer (1996). A data-dependent self-timed power down mechanism can be applied on output-wired inverter structured to reduce dc current to 25% Beiu (2001). However, high power dissipation and low noise margin are still the main problems in these design.

Differential mode DTGs have become the most popular and convincing architectures recently. Their main advantage is low power consumption. Differential DTGs have two set transistors connected in parallel to compute weighted sum and a CMOS comparator to compare with threshold. The CMOS comparator is usually consist of a pair of cross-coupled inverters. Clock signal is required to start comparing or reset the comparator. A cross-coupled inverters with asymmetrical loads (CIAL) López *et al.* (1995) and a generic latch-type TLG (LCTL) Avedillo *et al.* (1995) were proposed on 1995. Several circuits were proposed later to improve delay, power dissipation and reliability Strandberg and Yuan (2000); Padure *et al.* (2001); Tatapudi and Beiu (2003). These circuits compute threshold functions in two ways: compare the sum of weighted inputs with threshold or compare function f on one bank with its comple-

ment $\overline{f}$ on the other bank. Circuits that implement the former way need to introduce asymmetric weight in input banks to deal with case that sum of weighted inputs are equal to threshold. This is usually done by adding a permanent ON transistor with half weight on one bank Avedillo *et al.* (1995), which would reduce noise margin. The latter way avoids equal case as f and $\overline{f}$ are always turned to opposite side. A design combined split-level precharge differential logic with this method shows significant power reduction ($\tilde{9}0\%$) in Tatapudi and Beiu (2003).

Other technology approaches have also be explored to implement TLG. Single Electron Tunneling (SET) and Resonant Tunneling Devices (RTD) were popular candidates during 1990s and early 2000. SET is a nano device based on quantum mechanical phenomena called Coulomb blockade. N-input linear threshold gate can be implemented by SET, the structure is similar as νMOS. A full adder is reported in Lageweg *et al.* (2002) with combination of SET and coupling capacitors. RTD is another nanoelectronics device that has negative differential resistance. TLG design with RTD were reported as early as 1994 Maezawa *et al.* (1994). Latter a full adder was designed based on the same principle Pacha *et al.* (2000). Quantum cellular automata (QCA) is another popular quantum mechanical device. Each QCA cell consists of five quantum dots. Two electrons are hopping amount these states. Their relative position forms two stable states, named $+1$ and -1. Cell state is influenced by the state of their neighbors. A QCA majority gate was constructed based on this phenomenon Tougaw and Lent (1994). In the gate, output cell state is determined by the majority of the three input cell surrounding it. An AND and an OR gate were constructed by assigning 0 or 1 on one of majority gate input. A 1-bit full adder was also demonstrated in Tougaw and Lent (1994).

However, Major difficulties on fabrication and reliable operation on room temperature make these device hard to commercialize. Recently, new emerging non-volatile

devices have attracted intense interest from both industrial and academic. Some applications with these novel non-volatile devices have already been commercialized by semiconductor company like Intel and Micron. Comparing with previous nanotechnologies, non-volatile device is mature enough to be implemented in circuit design. Threshold logic gate implementations with new emerging device like RRAM, Domain Wall, nanowire are reported in Friedman *et al.* (2016); Fan (2016); James *et al.* (2014).

2.2 Threshold logic based synthesis

Beiu pointed out in Beiu *et al.* (2003) one of the major reason threshold logic is not as popular as CMOS logic on commercial application is lack of high-level synthesis tools. In this section, we would review the efforts on threshold logic synthesis.

Threshold synthesis was popular during 1960s. Network scale was small and it was affordable to do gate implementation manually. Since n-input majority gate is a threshold logic gate and any threshold function can be implemented by majority gate, majority gate synthesis is a particular threshold synthesis methodology. Akers (1962) proposed a reduced unitized table synthesis methodology. Reduced unitized table is constructed from truth table of the function. Two canonical realization and a comprehensive synthesis procedure using majority gates only were descried using the table. Miller and Winder (1962) proposed majority-logic synthesis based on Karnaugh map. It demonstrated that all 3-input function can be implemented by 3-input majority function in two levels with maximum 4 gates. It then extended it to n-input function and n-input majority gate by treating specification of the remaining function as a new problem and repeat the procedure. Muroga (1971) also proposed a shannon decomposition based majority gate synthesis. However, these methods are only suitable on small networks. When the network becomes large, the computing complexity grows exponentially, which makes these methods not applicable for today's

VLSI circuit design.

Threshold synthesis didn't draw much attention during that time. The main reason is that comparing with mature CMOS logic gate, high performance threshold logic gate was lacking during that time. Besides that, there was no efficient algorithm to synthesis multi-level network. After nanoelectronic threshold gates and more efficient algorithm are available, more work on threshold/majority synthesis were published. The first comprehensive multi-level threshold network synthesis methodology and synthesis tool were proposed in Zhang *et al.* (2005). An algebraically-factored combinational boolean network is given as input. For each node that satisfy fan-in constraint, the algorithm would identify if the node is threshold. If the node is not threshold, it would then split the node and map it by multiple threshold gates. Nodes are processed recursively from output until all the nodes are processed. An ILP solver is needed on threshold identification step.

A simplified 3-input majority synthesis was proposed later Zhang *et al.* (2007). It first decomposed the network to only include 3 variables on each node. It then try to find a majority network using the karnaugh map method Miller and Winder (1962). If the trial failed to construct an optimal solution, the methodology would direct map AND/OR gate to majority gate. By restrict to only majority gate, this algorithm doesn't require ILP and and unate identification.

Formulating ILP problem is the most straight forward way to identify if a unate function is threshold. Its solution provides minimum weights and threshold in integer. However, the execution time of solving ILP is usually exponential with respect of input variables. Gowda *et al.* (2007) and Neutzling *et al.* (2013) provides alternative ways to identify threshold function without solving ILP. These non-ILP method can't guarantee neither 100% threshold function identification nor optimal W_i/T assignment, but they run faster.

To further improve synthesis result, Kuo *et al.* (2011); Lin *et al.* (2014) proposed local rewiring techniques that can decompose high fanin or high threshold gate into smaller gate in order to reduce sum of W_i and T. Instead of local optimization, Annampedu and Wagh (2013) demonstrated how to decompose a large fan-in threshold function into polynomial sized threshold network with bounded fan-in. The network has size of $O(n^c/M^2)$ and depth of $O(\log^2 n/\log M)$.

2.3 Non-volatile memory and non-volatile logic

As shown in previous chapter, emerging NVM technologies can replace on-chip and off-chip memory to checkpointing system state. Besides, non-volatile technologies can be implemented to backup and restore intermediate computing results, so called non-volatile logic (NVL). The focus of non-volatile technology is beginning to change, with increasing emphasis on the incorporation of non-volatile logic in both control and datapath circuitry.

Without modifying CMOS logic gate, non-volatile technology can be implemented in two ways. In order to preserve performance of traditional volatile architecture, NVM arrays can be served as a slave memory array for volatile register file and D-flipflops. Before power failure, the data in registers and D-flipflops are serially written to NVM arrays. When the power resumed, the opposite process resume the states from NVM arrays serially. Ref. Khanna *et al.* (2014) reports the design of a microcontroller unit enhanced with non-volatile memory. It has a NVM array (NVMA) that is separated from the local registers (volatile) where the intermediate computation results are stored. The data in the processor's 2,537 flipflops are sequentially saved to, and restored from the NVM arrays. Therefore, the design trade-offs are between the NVMA size, the routing resources between the local registers and the NVMA, and backup time. The number of clock cycles required is the number of bits in the

registers. In contrast to design using NVFFs, the long backup times would preclude its use in severely energy-constrained systems.

Another way is to design non-volatile flipflop (NVFF) by applying non-volatile feature to each D-flipflop Ryu *et al.* (2012); Koga *et al.* (2010); Wang *et al.* (2012); Mahalanabis *et al.* (2015). The NVFF operates as same as normal D-flipflop on normal operation. In back up mode, the output state drives the write control circuitry to backup the state in non-volatile devices locally. During the restore mode, control circuitry restore the saved state to NVFF output. FeRAM based Koga *et al.* (2010); Wang *et al.* (2012), CBRAM based Mahalanabis *et al.* (2015) and STT-RAM based Ryu *et al.* (2012) have been reported. However, the huge area and performance overhead of existing NVFF design is still obstacles for extensive applications.

The earlier efforts Koga *et al.* (2010); Wang *et al.* (2012) using FeRAMs reported substantial penalties in area (10X larger than regular flipflop), performance (delays in $\sim \mu s$) and energy. The emerging spin-based magnetic tunnel junction (MTJ) devices such as STT (Spin Transfer Torque) or SOT (Spin Orbit Torque) with high density, low switching energy, and fast switching times, are promising candidates for NVM and NVL.

Refs. (Ryu *et al.* (2012); Bishnoi *et al.* (2017)) describe the design of a NVFF with STT-MTJ. Ref. Ryu *et al.* (2012) focuses on the design of the write circuit to provide higher driving current thereby reducing the backup time. Ref. Bishnoi *et al.* (2017) explores a fault model for the MTJ (open or a short), and a cell design that can tolerate a single MTJ fault. The design consists of two (for '1' and '0') 2×2MTJ cell arrays associated with each bit. Backup involves two MTJ cells carrying data in opposite directions, which improves the read margin at the cost of doubling backup energy. The focus is on tolerating single failures, without considering the design of the driver circuitry, which would have a significant impact on the energy, performance

and yield.

Techniques that are aimed at the robust design of NVM in the presence of process variations are generally not well-suited for NVL. For instance, the works in Bishnoi *et al.* (2016a); Motaman *et al.* (2015) address the problem of unequal delays between writing a '1' and '0' in an NVM, and the resulting solutions are not applicable to NVFF design. Ref. Yu and Wang (2014) contains a comprehensive treatment of NVM technology, and describes the design of several readout circuits that are tolerant to variations, as well as methods to reduce the read latency. However, these readout circuit architectures are primarily for NVM, and are too complex and require too much energy to be applicable to NVFF. The problem of *read disturb*, which is a common issue, is addressed in Bishnoi *et al.* (2014), using a current mirror and additional control circuitry which renders it unsuitable on digital standard cell design. Ref. Wang *et al.* (2016) explores the issues of STT-based MRAM design in the presence of process variations. Among other things, it proposes a *post-write sensing* strategy that involves a sequence of reads, writes and comparisons to minimize the write error rate. As with other schemes targeting NVM, this is not suitable for NVFFs.

The discovery of Spin Orbit Torque (SOT) switching provides a more efficient way to reverse magnetization Miron *et al.* (2011). SOT switching is induced by applying a current through a heavy metal layer underneath MTJ. An SOT based MTJ cell is a three terminal non-volatile device. Compared with an STT-MTJ, SOT switching is faster Garello *et al.* (2014), and the three terminal structure allows for separate optimization of the write and read paths. This promises to be much more reliable. Refs. (Kwon *et al.* (2014); Bishnoi *et al.* (2016b)) describe SOT-MTJ based NVFF designs, showing that they have the potential for higher speed, lower energy and higher reliability than STT-MTJ devices. The design optimization and circuit architectures presented herein for STT-MTJ can easily be adapted to SOT-MTJ devices.

2.4 Other non-volatile logic gate

Other than memory and NVFF, emerging device can be used to construct non-volatile logic network directly. STT-MTJ, RRAM and DMW devices are implemented in these design because of their area advantage. Nukala *et al.* (2012) describes a STT-MTJ based threshold logic gate array(STLA). It uses the inherent threshold characteristic of STT-MTJ to compute subset of threshold function. The array is operated as a gate-level nano-pipelined circuit, where the computing result of each gate is evaluated and stored in STT-MTJ. Natsui *et al.* (2013) reports a full adder based non-volatile image processor. Partial operands such as reference data are stored in STT-MTJ. The computing result is evaluated by controlling read current flow through STT-MTJ. Huang *et al.* (2014) reports four STT-MTJ based basic CMOS logic gates, INV, AND, OR and XOR. Their delay and power consumption are orders higher than normal CMOS logic gates.

A relative new field is array like non-volatile device network, referred as logic-in-memory. Currivan-Incorvia *et al.* (2016) demonstrates a inverter chain fabricated with three DWM operated as inverter. Currivan *et al.* (2012) reports the simulation results of applying same DWM in full adder. Chen *et al.* (2015) reports mapping 4-bit multiplier on RRAM crossbar array, and Sengupta *et al.* (2016) explores the possibility of implementing artificial neural network on DWM based crossbar network.

Chapter 3

THRESHOLD LOGIC GATE IMPLEMENTATION

In this chapter, we would discuss how threshold logic gate library is designed. The library consists of single and multiple sequential threshold logic gate. The design trade-off like speed, area, reliability and energy consumption would be discussed. After a brief review of how signal assignment and synthesis are done, both the simulation and fabrication results would be shown.

3.1 Multi-input TLG in Digital Circuit

The requirement of TLG in digital IC are restricted. In digital circuit, TLG is treated as a special standard cell which computes threshold function. All standard cell design restrictions should be applied to TLGs. Area and power are very stringent in digital circuit,therefore DC current and large amplifier design should be avoid. Reliable operation is also critical for VLSI implementation since multiple TLGs are included in one design, single error would cause the whole design to fail.

Sec. 2.1 reviewed most popular CMOS TLG circuits over years. Among these architecture, differential mode TLG is known for its robustness and low power. Differential mode TLGs employs a combination of current mode and capacitive mode TLGs. A block diagram of DTG is shown in Figure 3.1. Two banks of transistors are used to represent the inputs, weights and threshold. The clocked comparator compares the conductance difference between two input networks according to the definition of the implemented threshold function. Differential TLGs are usually designed as a sequential element, a latch structure is required to keep the output during the rest of clock period. Comparing result would set the output of latch.

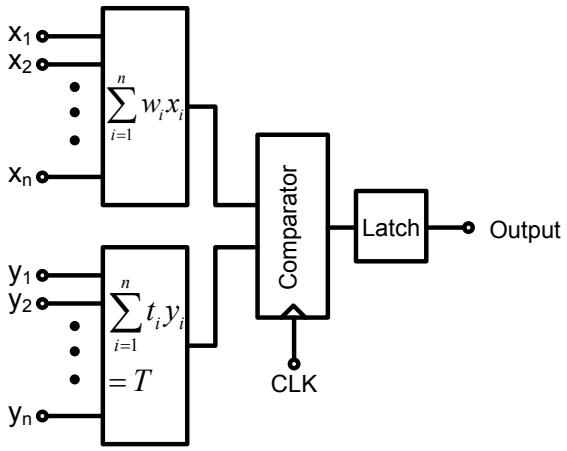


Figure 3.1: Threshold gate circuit block

3.1.1 Circuit Architectures

Figure 3.2 shows a TLG implementation, referred as TLL. It was first published in Ref. Samuel *et al.* (2010). TLL consists of 4 components: (1) a differential sense amplifier, which consists of two cross coupled NAND gates, (2) a SR latch, (3) two discharge devices, and (4) left (**LIN**) and right (**RIN**) input networks. TLL-n refers to a TLL with n inputs in the LIN and the RIN. Clock input signal directly drive source terminals of input network.

- *Reset State*: When clock signal CLK is '0', the sense amplifier is in 'reset' state, node N5 and N6 are discharged to '0' through transistor M_{11} and M_{12}. Therefore nFET M_7 and M_8 are turned off and pFET M_1 and M_4 are turned on, pulling node N1 and N2 to '1'. The SR-latch followed by TLL will keep its previous value.

- *Evaluation State*: When CLK rises from '0' to '1', TLL changes into evaluation state. The rising CLK turns off M_{11}/M_{12} and begins to charge LIN and RIN. Assuming the input corresponds to be in the on-set of threshold function f, the conductance of LIN will be higher than conductance of RIN. When clock

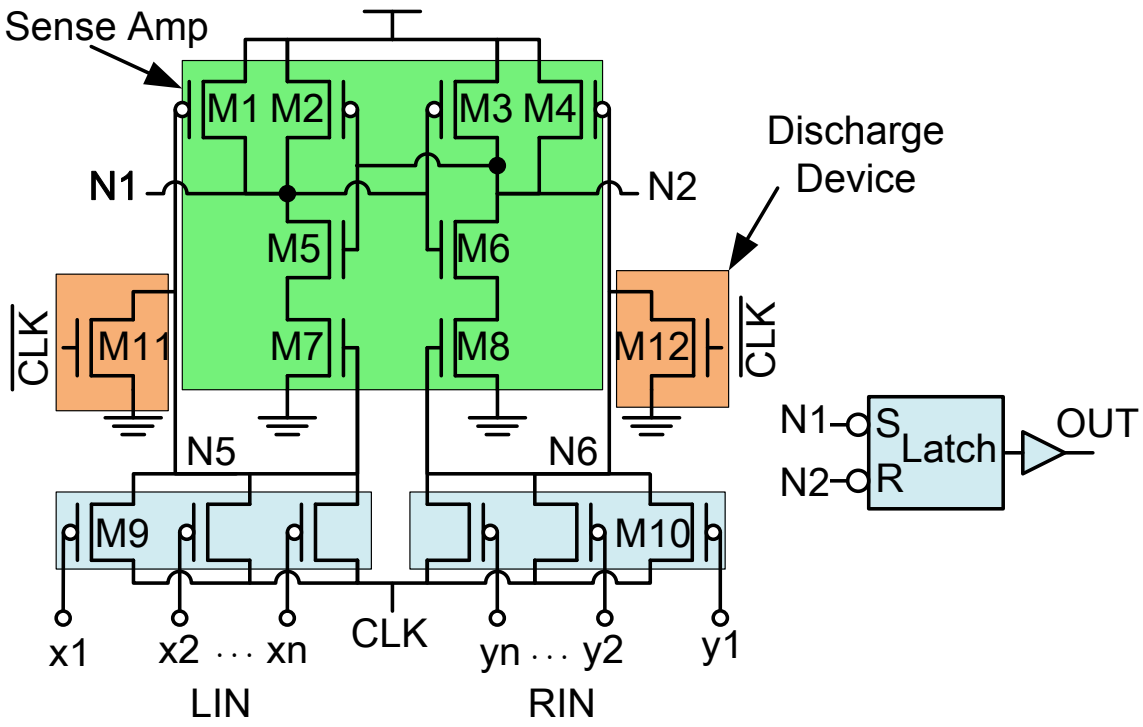


Figure 3.2: Schematic of a TLL circuit.

rise from '0' to '1', the charging speed of N5 will be faster than N6, which will turn M_7 on earlier than M_8. Consequently, node N1 will be discharged before node N2. The falling N1 will then turn off M_6 and turn on M_3, which will stop N2 discharge process and pull it back to '1'. $N1 = 0$ and $N2 = 1$ will set the output SR-latch output to '1'. The similar process happens when the inputs are in off-set, while eventually $N1 = 1$ and $N2 = 0$. In summary, if the number of on-transistors in the LIN exceeds the number in the RIN, then the threshold inequality evaluates to *true* and output is a 1, otherwise the inequality is *false* and the output is 0.

Functionally, TLL can be viewed as a *complex, multi-input* edge-triggered flipflop. In general, a TLL has a lower setup time than a DFF while its *clock-to-Q* delay is comparable. A TLL also presents a lower input capacitance but higher clock capacitance than a D-FF.

In TLL, CLK is directly connected to the source nodes of LIN and RIN. Its capacitance is the sum of source capacitance C_s for all input transistors. When input numbers are large, CLK pin capacitance of TLL can be significantly higher than DFF. Larger clock tree is required to drive TLLs comparing with driving the same number of DFFs. Large clock tree usually consumes higher area and energy as it toggles all the time.

CLK capacitance is also determined by input configurations. C_{gs} varies according to transistor operation region, which is high when transistor is ON. The capacitance variation may introduce extra clock uncertainty to TLL circuit. This is because the clock slew rate delivered to each TLL gate is determined by clock tree driver size and CLK pin load. In normal design with DFFs, Both driver size and load are fixed after place and route (P& R). Timing information such as setup time, hold time requirement and C2Q delay is also determined. However, CLK load in design with TLLs varies depends on input signals, which also changes slew rate of the delivered clock signal. Therefore, the static timing analysis (STA) is not valid anymore. To compensate this variation, threshold function mapping has to guarantee same number of ON transistors for all possible input combinations. The mapping techniques known as CSA would be discussed in the following section. CSA results in a constant load on the clock input regardless of the input vector which is very important for construct clock tree to deliver clock signal.

An improved multi-input differential threshold gate referred as PNAND is designed and its schematic is shown in Fig. 3.3. Two $\overline{(CLK)}$ driven pFETs are stacked on the top of RIN and LIN. An inverter is included in each PNAND cell to generate $\overline{CLK}$ signal, so the clock to output delay (C2Q) is usually higher in PNAND. The advantage of PNAND over TLL is that CLK capacitance is signal independent and its value is equal or less than CLK capacitance of DFF. Therefore, the timing

uncertainty from load variation is eliminated in PNAND.

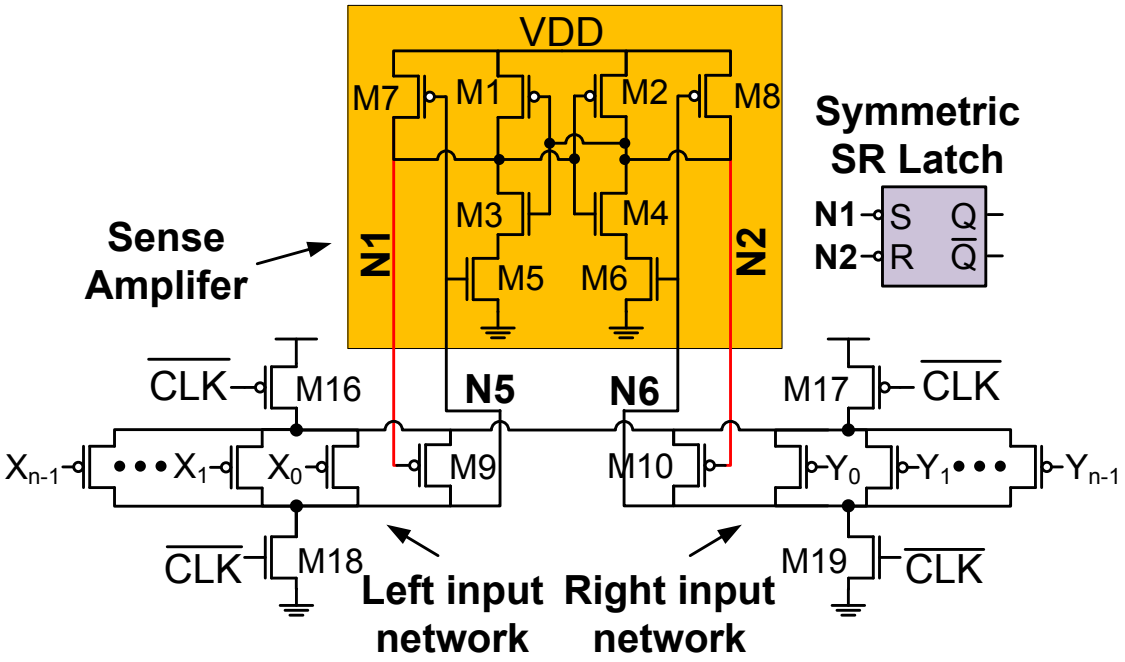


Figure 3.3: Basic PNAND architecture

Transistors M_9/M_{10} in Fig. 3.3 prevent potential floating of N5/N6 on certain input combinations. After the sense amplifier evaluates the state, let's assume that the number of ON transistors on LIN changes from m to 0 while CLK is still high. Without M9, N5 can float at logic 1. Leakage or noise can potentially discharge the node N5 and flip the states of N1 and N2. Transistors M9/M10 are used to strengthen the resolved state, which ensures that state is not disrupted even if inputs change while CLK is high. Fig. 3.4 shows the described case and how M_9/M_{10} prevent output flipping in the same clock period.

SR latch is required for differential mode threshold gate like TLL and PNAND. Latches in Fig. 3.2 and 3.3 hold the evaluation results when CLK is low, which prevent glitches at output. The SR latch is also an important contributor to the clock to output delay. Therefore it is necessary to optimize latch design for minimum delay. There is a plethora of literature available on SR latch optimization. Nikolic

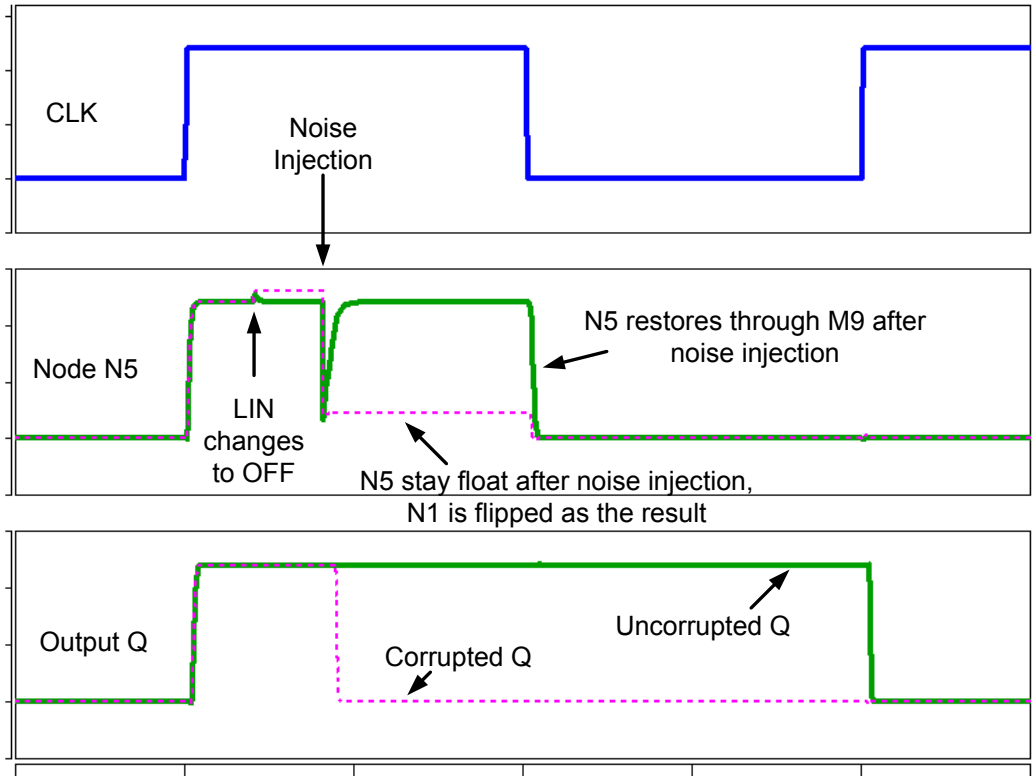


Figure 3.4: Transistor M_9 and M_10 would prevent unexpected state flips in the same clock period

et al. proposed a symmetric latch to balance rise and fall delay Nikolic *et al.* (2000). Kim *et al.* described a NC^2MOS latch Kim *et al.* (2000), which is faster than an SR latch but can lead to glitches at the output. Strollo *et al.* introduced another improved NAND latch design which is glitch freeStrollo *et al.* (2005). For both the improved NAND latch and the NC^2MOS latch design, the input and clock signals were added to the latch, therefore the total input capacitance are higher. Typically, high input pin capacitance, especially for the clock pin, would significantly increase the clock tree size and power consumption. Considering all effects, the symmetric latch proposed by Ref. Nikolic *et al.* (2000) is founded to be a good balance between performance and power.

3.1.2 Implementing Threshold Function

PNAND is equivalent to multi-input flipflop as it computes output at clock rising edge. The multi-input flipflops from standard cell library usually computes a standard two input NAND or NOR function while a single PNAND cell can be configured to compute multiple threshold functions. The configuration simply involves connecting the input x_i and/or their complements to the gates of transistors in the LIN and RIN. It is called *signal assignment* or SA.

PNAND in Fig. 3.3 will be denoted as PNAND-n where both LIN and RIN have n inputs. The function PNAND-n evaluates can be represented as Eqn. 3.1. When the numbers of ON transistors in both LIN and RIN are equal, sense amplifier will be in metastable state during evaluation and the output is not predictable. This equal case should be avoid for all input combinations.

$$f(x_0, x_1, \cdots, x_n, y_0, y_1, \cdots, y_n) = \begin{cases} 1 & \sum_{i=0}^{n} x_i > \sum_{i=0}^{n} y_i \\ 0 & \sum_{i=0}^{n} x_i < \sum_{i=0}^{n} y_i \\ \text{Unknown} & \sum_{i=0}^{n} x_i = \sum_{i=0}^{n} y_i \end{cases} \tag{3.1}$$

Threshold functions are a proper subset of unate functions. Without loss of generality, we can assume that all the weights are positive integers. A threshold function with negative weights can always be transformed to a positive unate form by replacing the input variables with their complements. $f = [w; T]$ is an optimal representation when both weight sum $W = \sum w_i$ and threshold T are minimum. In optimal $f = [w; T]$, the minimum difference between weighted sum of inputs and threshold is 1 ($\min |w'X - T| = 1$). With this observation, equality in 1.3 can be eliminated:

$$\sum_{i=1}^{m} w_i x_i \geq T \equiv \sum_{i=1}^{m} w_i x_i > T - 0.5 \equiv \sum_{i=1}^{m} 2w_i x_i > 2T - 1 \tag{3.2}$$

m is number of input variables in threshold function. The variables of a threshold function can be connected to the inputs of a PNAND-n cell such that output of PNAND is 1 if and only if threshold function evaluates to 1. The actual function implemented by a PNAND-n depends on the input *signal assignment* (SA). The SA procedure always ensures that LIN and RIN never have the same number of ON transistors. For a PNAND-n, a j/k input refers to j and k active transistors in the LIN and RIN, respectively, or vice versa. For example, a PNAND-5 with LIN assigned signals $\bar{a}, \bar{a}, \bar{b}, \bar{c}, \bar{c}$ and RIN assigned signals $a, a, b, logic'1', logic'1'$ (SA $= (\bar{a}, \bar{a}, \bar{b}, \bar{c}, \bar{c} \mid a, a, b, 1, 1)$) implements $f = a \vee bc$, and a PNAND-5 with the SA $= (\bar{a}, \bar{b}, \bar{c}, \bar{d}, \bar{e} \mid a, b, c, d, e)$ implements $abc{+}abd{+}abe{+}cad{+}ace{+}ade{+}bcd{+}bce{+}bde{+}cde$. More than one SA can be implemented, here we discuss two SA techniques for same function.

Optimal Signal Assignment (OSA): All elements before $>$ in Eqn. 3.2 would be assigned to LIN and all elements after would be assigned to RIN. In PNAND, LIN and RIN are consist of PMOS transistors which conduct when gate voltage is '0'. Therefore, all positive variables x_i in equation would be assigned as $\overline{x_i}$ to gate inputs, $1-x_i$ is assigned as x_i and constant is assigned as '0' in PNAND. Assignment Eqn. 3.2 to PNAND-n such that n is minimum is called optimal signal assignment or OSA. A threshold function is not trivial if $W \geq T$. Therefore, we can move $2T-1$ unit literals from left to right in Eqn. 3.2. Then the total weight on left is $2W - 2T + 1$ and total weight on right is $2T-1$. The threshold $2T-1$ on right can be combined with moved $2T-1$ literals as $(2T-1)(1-x_i) \rightarrow (2T-1)\overline{x_i}$. The minimum n that can implement this function on PNAND is $n = \max 2W - 2T + 1, 2T - 1 \leq \max 2W, 2T - 1$. $2W$ is even and $2T-1$ is odd, which makes n an odd number. There are several choices of $2T-1$ literals to be moved from left to right. The choice that leaves each literal to both sides are preferred. It helps to minimize the least robust case in LIN and RIN.

For example, consider threshold function $f = [3, 1, 1, 1 : 3]$. It can be converted as

$$3a + b + c + d \geq 3 \equiv 6a + 2b + 2c + 2d > 5$$

. 5 unit literals would be moved from left to right side. $2a + b + c + d$ is moved to right to ensure all variables on each side. The inequality is then become

$$4a + b + c + d > 2(1 - a) + (1 - b) + (1 - c) + (1 - d)$$

, and the input signals that assigned to PNAND is

$$\{LIN|RIN\} = \{\overline{a}, \overline{a}, \overline{a}, \overline{a}, \overline{b}, \overline{c}, \overline{d}|a, a, b, c, d, 1, 1\}$$

(SA(2)). Another way of literal movement is also applied for comparison. $5a$ is moved to right and the SA becomes $\{\overline{a}, \overline{b}, \overline{b}, \overline{c}, \overline{c}, \overline{d}, \overline{d}|a, a, a, a, a, 1, 1\}$ (SA (1)). Table 3.1 shows the effect on LIN and RIN for both SAs. It is clear that the highest number of ON transistors in SA (1) are 6 on LIN and 5 on RIN as well as 4 and 3 for SA (2).

An important and distinctive aspect of PNAND is that its performance, power and *robustness* are affected not only by process variations, but also by the signal assignment. Its delay is the sum of the input network delay (IND) and sense amplifier/latch delay (SLD). The IND is the RC delay of the network, and the greater the conductivity of the LIN or RIN (i.e. more active transistors in LIN or RIN), the smaller the IND. Thus for a PNAND-n (n odd), a 1/0 input results in maximum IND, and a $k/k-1$ input, for $k = (n+1)/2$, results in a minimum IND. On the other hand, because N5 and N6 both start at 0, and rise to 1, the smaller the difference between N5 and N6, the greater the SLD. Thus, the maximum SLD occurs for a $k/k-1$ input, and for maximum k. This is also the worst-case condition that dictates the robustness of the cell.

Table 3.1: Truth table of threshold function $f = [3, 1, 1, 1 : 3]$. LIN and RIN ON transistor comparison for two signal assignments: (1)$\{\overline{a}, \overline{b}, \overline{b}, \overline{c}, \overline{c}, \overline{d}, \overline{d}|a, a, a, a, a, 1, 1\}$ and (2)$\{\overline{a}, \overline{a}, \overline{a}, \overline{a}, \overline{b}, \overline{c}, \overline{d}|a, a, b, c, d, 1, 1\}$

abcd	SA (1)		SA (2)	
	L	R	L	R
0000	0	5	0	5
0001	2	5	1	4
0010	2	5	1	4
0011	4	5	2	3
0100	2	5	1	4
0101	4	5	2	3
0110	4	5	2	2
0111	**6**	**5**	3	2
1000	1	0	**4**	**3**
1001	3	0	5	2
1010	3	0	5	2
1011	5	0	6	1
1100	3	0	5	2
1101	5	0	6	1
1110	5	0	6	1
1111	7	0	7	0

Complementary Signal Assignment (CSA): All inputs in the RIN will be driven by $x_1, \cdots, x_n$, and all the gates in LIN will be driven by their complements $\overline{x_1}, \cdots, \overline{x_n}$. To ensure that the number of ON transistors in the LIN and RIN are never equal, n must be odd. This is because if n were even, and if r were active in the LIN then $n-r$ will be active in the RIN. Hence if $r = n/2$, an equal number of transistors will be active in the LIN and RIN. Since the LIN and RIN are complementary, for the output to be 1, just over 1/2 (or more) of the transistors in the LIN must be active. That is, the function is 1 if and only if $(n+1)/2$ or more of the inputs are 1. Hence with n being odd, a PNAND-n with this signal assignment (all input gates driven by distinct x_i), implements the threshold function defined by

$$\text{PNAND-n:} \equiv x_1 + x_2 + \cdots + x_n \geq (n+1)/2. \tag{3.3}$$

Consider an arbitrary threshold function $f(z_1, z_2, \cdots, z_m)$ defined by $w_1 z_1 + w_2 z_2 + \cdots + w_m z_m \geq T$, that is to be realized by PNAND-n. Clearly if $T > (n+1)/2$, then f cannot be implemented by PNAND-n. Let $D = (n+1)/2 - T$, $D \geq 0$.

$$z_{1,1} + \cdots + z_{1,w_1} + \cdots + z_{m,1} + \cdots + z_{m,w_m} + D \geq \frac{n+1}{2}. \tag{3.4}$$

Therefore from (3.3) and (3.4), the second condition on PNAND-n to be able to realize $f(z_1, z_2, \cdots, z_m)$ is $W + D \leq n$, or $W - T \leq (n-1)/2$. Given a PNAND-n, if $f(z_1, z_2, \cdots, z_m)$ can be realized, then from (3.4) the assignment of signals can be done as follows: (1) assign D of the inputs of PNAND-n to '0'; (2) for each i, $1 \leq i \leq m$, assign w_i inputs of PNAND-n to the signal $\overline{z_i}$; (3) connect any remaining inputs of PNAND-n to '1'.

Let L denote the number of active transistors in LIN and R denote the same for RIN. It can be shown (and experimentally verified) that the worst case robustness condition will be for an input vector (i.e. $x's$) that results in a unit difference in

conductance i.e. $(L-R)=1$ with the largest number of active transistors $(L+R)$ is maximum. Incidentally, among all the cases for which $(L-R)=1$, the one that maximizes $(L+R)$ results in the least delay. Note that it is impossible to avoid $(L-R)=1$ for any signal assignment.

Justification of CSA: The CSA has three important characteristics that justify its use. First, it maximizes $(L+R)$ resulting in the fastest possible PNAND. Secondly, there is only case for which $(L-R)=1$ and the circuit can be optimized only for this case. Third, irrespective of an input vector $(x's)$, the total number of OFF transistors in the LIN and RIN is always n.

An example is shown here to demonstrate how OSA and CSA is implemented. Consider the threshold function $f(a,b,c)=a+bc \equiv 2a+b+c \geq 2$. In OSA, it can be easily converted into strict inequality function, as shown below:

$$2a+b+c \geq 2 \equiv 4a+2b+2c > 3 \equiv 3a+b+c > (1-a)+(1-b)+(1-c) \quad (3.5)$$

A PNAND-5 is required to implement this function using OSA. LIN is assigned as $\{\overline{a},\overline{a},\overline{a},\overline{b},\overline{c}\}$, and RIN is assigned as $\{a,b,c,1,1\}$.

For CSA implementation, it is easily verified that a PNAND-7 is required since $T=2$, $W=4$, $W-T=2 \leq (7-1)/2$ and $D=(7+1)/2-2=2$. Internally, the transistors in the LIN will be driven by $\{\overline{a},\overline{a},\overline{b},\overline{c},0,0,1\}$, and RIN will be driven by $\{a,a,b,c,1,1,0\}$. The truth table is shown in Table 3.2.

PNAND-7 is usually larger and consumes more energy than PNAND-5 because of extra input pins. Therefore, in most low power digital circuit, OSA is preferred. As shown in Table 3.1 and 3.2, the race condition in OSA is less severe. However, delay of PNAND with CSA is faster and has less variation compare to CSA for all possible input combinations. CSA would be the only solution on low voltage operation where delay variation is a major cause of timing violation.

Table 3.2: Truth table of threshold function $f = [2, 1, 1 : 2]$. LIN and RIN ON transistor comparison for OSA and CSA. OSA: $\{\overline{a}, \overline{a}, \overline{a}, \overline{b}, \overline{c}|a, b, c, 1, 1\}$. CSA: $\{\overline{a}, \overline{a}, \overline{b}, \overline{c}, 0, 0, 1|a, a, b, c, 1, 1, 0\}$

abc	OSA		CSA	
	L	R	L	R
000	0	3	2	5
001	1	2	**3**	**4**
010	1	2	**3**	**4**
011	2	1	**4**	**3**
100	**3**	**2**	**4**	**3**
101	4	1	5	2
110	4	1	5	2
111	5	0	6	1

Both OSA and CSA require odd input n. 4 PNAND-n cells are included in standard cell library where n equals 3, 5, 7 and 9. The higher input is , more threshold function PNAND can implement. However, when n is large, the number of ON transistors in worst case would be high and the race condition on N5 and N6 would be hard to distinguish by sense amplifier. The robust operation would be discussed later.

3.1.3 Delay Modeling

As a multi-input flipflop, characterization of PNAND includes setup time, hold time and CLK to Q delay for all possible input combinations. The evaluation of PNAND includes three steps. When clock edge comes, the first step is node N5 and N6 charging from 0 to V_t of M_5 and M_6. High conductance in input network would speed up the charging delay. Assume N5 reach V_t first, M_5 and M_3 begins to discharge

first. M_5 then reaches its saturation regime and maximizes the draining current. N6 charges to V_t later than N5, which would also cause M_6 to drain current from N2. N2 drop would in turn cause current flaw through M_3 drop, which would slow down N1 discharge. If N6 charging much slower than N5, then N1 would complete its discharge before M_6 starts, preventing N2 discharge by turning off M_4. In this case, the delay from N5 rising to N1 falling depends on N5 rising speed and M_5 draining speed. If N5 and N6 rising are very close, both M_5 and M_6 starts to drain current. The voltage drop on N1 and N2 would suppress the draining current of M_3 and M_4, which slows down discharging on both sides. If this race condition occurs, it would take much longer time for sense amplifier to finally resolve it. On the last step, when N1 reach V_t of SR latch, the latch begin to set its output to 1. The delay between N1 and output Q is determined by N1 falling rate as well as output load. Therefore, the clock to Q (C2Q) delay can be split into three parts, as shown in Eqn. 3.6.

$$\begin{aligned} \text{C2Q Delay} = &\text{input network delay} + \text{SenseAmplifier resolving delay} \\ &+ \text{Latch set delay} \end{aligned} \tag{3.6}$$

Input network delay (IND) is from clock rising to N5/N6 rise. Its value depends on charging current and parasitic load on N5/N6. More ON transistors increase charging current. Less input number and small M_5 and M_6 size reduce parasitic capacitance, both reduce IND. Sense amplifier resolving delay (SAD) depends on input rising time, race between LIN and RIN and output load. Fast and wide separated N5 and N6 rising causes small SAD, slow and congested N5/N6 rising causes large SAD delay. Latch set delay (LSD) is caused by N1/N2 rising time and output load.

Fig. 3.5 shows a clear delay trend for same input case (2:1) on multiple PNAND-ns. The figure shows partial delays such as IND, SAD, LSD and total C2Q delay. More input transistors and large sense amplifier increase all partial delays in large

PNAND. During evaluation, when number of ON transistors is fixed, higher N5/N6 parasitic in PNAND-7 and PNAND-9 increase charging delay and slew rate of N5/N6. Large charging delay causes IND to increase and large slew rate causes SAD to grow.

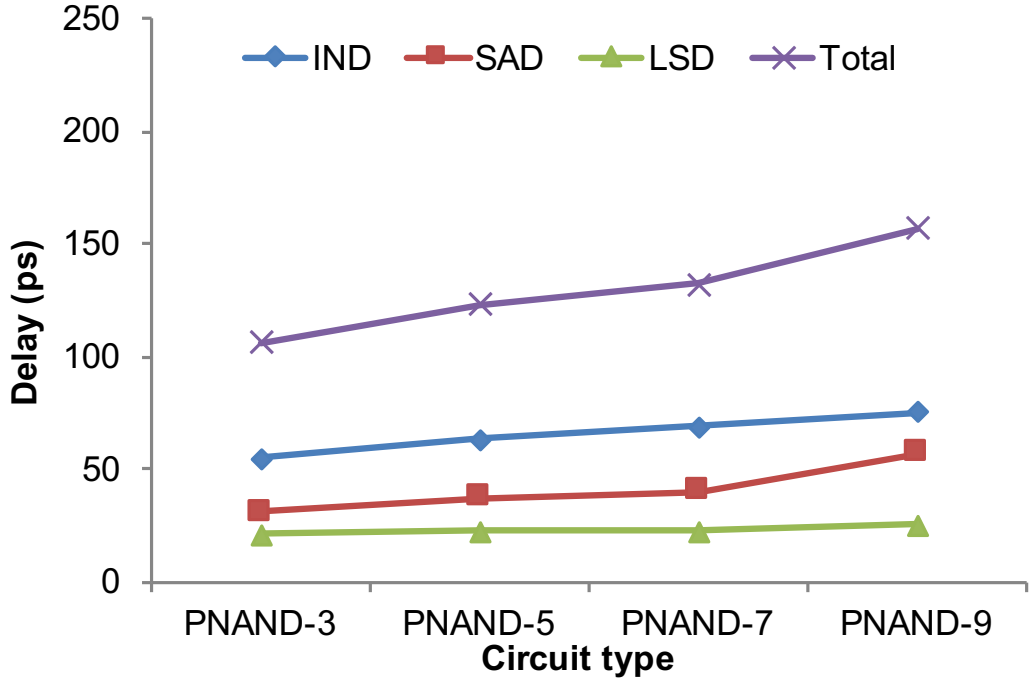


Figure 3.5: PNAND-n C2Q and partial delays for 2:1 input case

Fig. 3.6 and Fig. 3.7 show partial delays of PNAND-7 and PNAND-9 for several typical input cases. All possible input cases when $|\text{LIN} - \text{RIN}| = 1$ is picked. 1:0 to 4:3 cases are picked for PNAND-7 and 1:0 to 5:4 cases are picked for PNAND-9. Note that 5:4 or above in PNAND-7 can always been converted into 1:0 to 4:3 cases by flipping all input signals to its complementary value. Same mechanism can also be applied to PNAND-9. In both figures, IND, SAD and total (C2Q) delay are reduced when number of ON transistors in input network increases. Similar as previous analysis, high charging currents are expected with more ON transistors, cause fast N5/N6 charging speed and small N5/N6 rising slew rate. Therefore, both IND and SAD reduces with more ON input transistors.

Table 3.3, 3.4, 3.5 and 3.6 summarize all delays for PNAND-3,5,7 and 9 for all

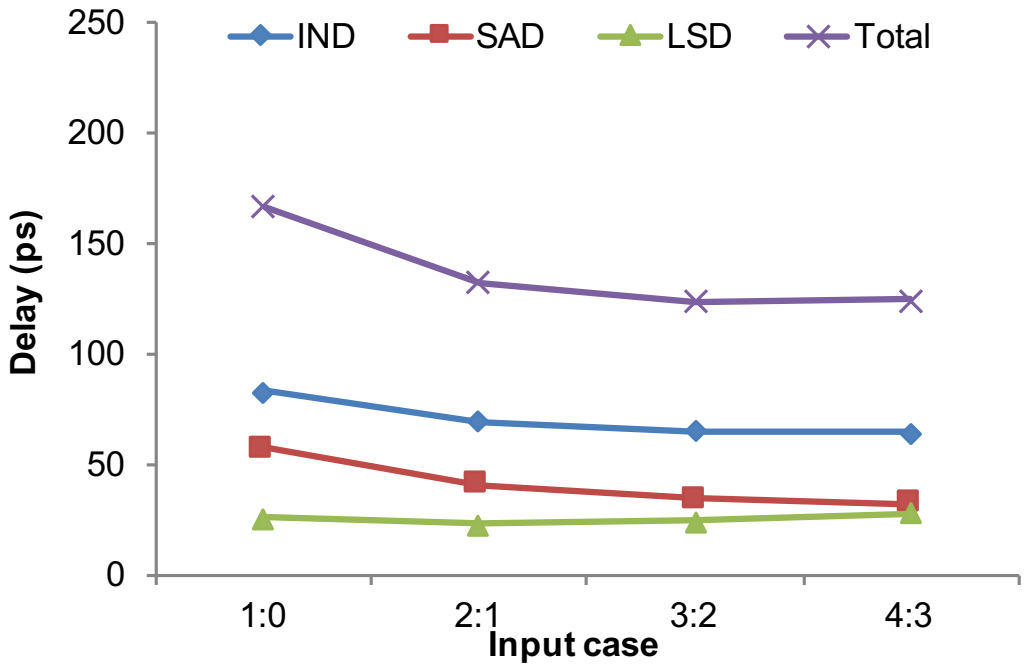


Figure 3.6: Partial and total delay of PNAND-7 for multiple input cases

possible input cases. All delay are simulated by HSPICE under Typical/Typical corner, 25°C degree and 1.2V power supply. No external output load is included in simulation. For all PNAND-n cells, input case 1:0 has the slowest C2Q delay and n:0 has the fastest C2Q. Overall, C2Q is IND dominated where 1:0 and n:0 have smallest and largest IND respectively. And Large n in input network has high IND. C2Q delays in PNAND-7 and PNAND-9 are generally higher than in PNAND-3 and PNAND-5. Same as Fig. 3.5, when number of ON transistors on LIN is the same, Small $|\text{LIN} - \text{RIN}|$ has higher congestion and generally causes higher SAD delay. The worst case congestion for sense amplifier is when LIN and RIN difference is 1 and LIN+RIN is maximum. All PNAND-n cells have same output drive strength, which means the SR latch designs are identical and output load on node N1/N2 are the same. In this case, worst case congestion would lead to slowest N1/N2 falling rate, which would significantly increase LSD delay. As seen in figure, the highest LSD delays for PNAND-5,7,9 happen on the same case as highest IND, which is the worst

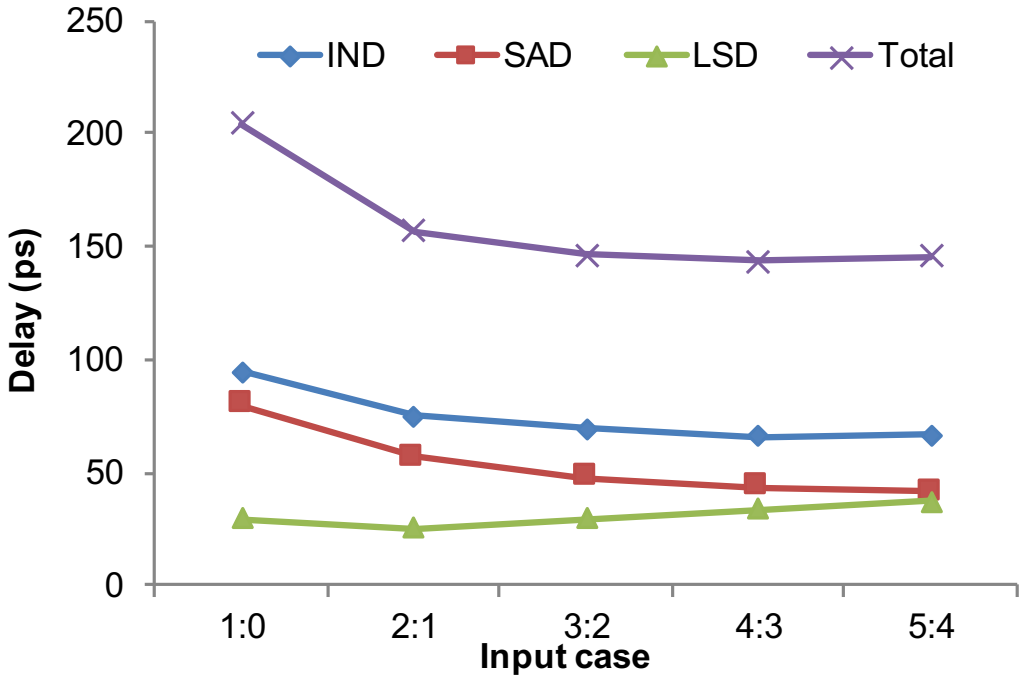


Figure 3.7: Partial and total delay of PNAND-9 for multiple input cases

congestion case $\frac{n+1}{2} : \frac{n-1}{2}$.

Table 3.3: PNAND-3 clock to Q delay (C2Q)for all possible input cases. C2Q delay is split into three parts: a) input network delay (IND); b) sense amplifier delay (SAD); c) latch set delay (LSD).

Input Case (LIN:RIN)	C2Q (ps)	IND (ps)	SAD (ps)	LSD (ps)	Energy (fJ/cyc)
1:0	122.18	60.6	39.16	22.42	30.65
2:1	106.4	54.85	30.6	20.95	34.53
3:0	98.82	51.05	27.4	20.37	31.09

3.1.4 Energy Consumption

Energy consumption is input dependent in both DFF and PNAND cells. The analysis is more complicated in PNAND cells as they include multiple inputs and all input combinations should be considered.

Table 3.4: PNAND-5 clock to Q delay split for all possible input cases.

Input Case (LIN:RIN)	C2Q (ps)	IND (ps)	SAD (ps)	LSD (ps)	Energy (fJ/cyc)
1:0	152	74.32	52.3	25.38	32.82
2:1	123.15	63.35	37.43	22.37	37.2
3:0	111.41	57.85	31.99	21.57	33.56
3:2	116.19	59.1	32.43	24.66	38.16
4:1	107.25	56.2	29.9	21.15	37.61
5:0	103.62	54.73	27.87	21.02	34.09

Energy consumption of PNAND cells are evaluated as total energy consumed in one clock cycle. In Table 3.3, 3.4, 3.5 and 3.6, average energy consumption are assumed to have 30% switching activity. 30% switching activity means that input signals are configured such that output Q flips between 0 and 1 within 30% of total simulation clock cycles. Energy consumption is proportional to parasitic values on signal path. It is also true for cell design. When comparing with same input case, PNAND-9 consumes highest energy while PNAND-3 consumes lowest.

Energy consumption of PNAND cells also depends on input cases. This can be clearly seen in Fig. 3.8. Fig. 3.8 shows range of energy consumption for all PNAND cells in library. Comparing with delays, 1:0 case always has highest delay but lowest energy for all PNANDs. Worst reliability cases always lead to highest energy consumption. Worst reliability case is when $|LIN - RIN| = 1$ and $LIN + RIN = n$. In this case, the falling transition time of N1/N2 is usually higher because it takes more effort for feedback loop in sense amplifier becoming stable. Long transition time costs higher dynamic power in both sense amplifier and latch.

Table 3.5: PNAND-7 clock to Q delay split for all possible input cases.

Input Case (LIN:RIN)	C2Q (ps)	IND (ps)	SAD (ps)	LSD (ps)	Energy (fJ/cyc)
1:0	166.34	83.14	57.23	25.97	35.8
2:1	132.16	69.12	40.32	22.72	40.94
3:0	118.56	62.81	33.97	21.78	36.76
3:2	123.7	64.77	34.39	24.54	42.41
4:1	113.36	60.96	30.99	21.41	41.97
4:3	124.13	64.54	31.77	27.82	43.43
5:0	108.48	58.31	29.16	21.01	37.32
5:2	110.5	60.13	29.34	21.03	42.07
6:1	107.72	58.68	28.07	20.97	42.67
7:0	104.61	56.41	27.46	20.74	38.01

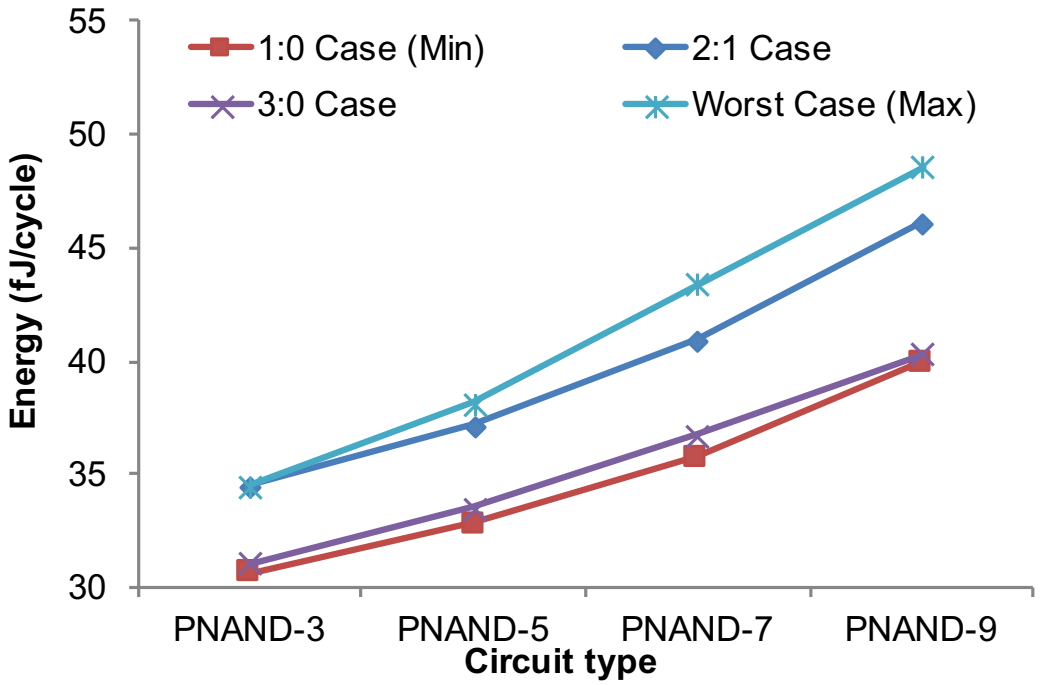


Figure 3.8: Range of energy consumption for PNAND-3, 5, 7 and 9.

Table 3.6: PNAND-9 clock to Q delay split for all possible input cases.

Input Case (LIN:RIN)	C2Q (ps)	IND (ps)	SAD (ps)	LSD (ps)	Energy (fJ/cyc)
1:0	204.34	94.94	79.77	29.63	39.96
2:1	156.85	74.89	56.8	25.16	46.14
3:0	131.58	67.54	48.22	25.82	40.34
3:2	146.69	69.31	47.71	29.67	47.3
4:1	133.91	64.97	43.78	25.16	46.86
4:3	143.39	66.21	43.61	33.57	48.2
5:0	129.07	63	41.17	24.9	41.18
5:2	131.23	65.92	40.61	24.7	47.15
5:4	145.54	66.41	41.61	37.52	48.59
6:1	125.99	61.76	39.36	24.87	47.8
6:3	127.77	63	38.51	26.26	47.71
7:0	123.87	60.77	38.29	24.81	41.84
7:2	123.94	61.63	38.12	24.19	47.49
8:1	123.45	61.68	37.17	24.6	48.19
9:0	120.76	59.02	37.23	24.51	42.12

3.1.5 Robust Operation

Robust operation under process variation is essential for standard cells. Process variations are coming from sources like variations in critical dimensions, random dopant fluctuation and variation of gate oxide thickness. Physical parameter variations lead to variations in electrical parameter such as threshold voltage or gate capacitance. In CMOS logic standard cells and circuits, process variations would cause variations in delay, power and leakage.

As demonstrated in Fig. 3.3, PNAND cell is a sense amplifier based standard cell design. It is known that transistor mismatch would lead to evaluation error in sense amplifier. For example, process variations of transistors M_5 and M_6 cause V_{t5} larger than V_{t6}. Assume inputs are configured as LIN is more conductive than RIN (LIN ¿ RIN). Under normal case, M_5 start conducting first and output of sense amplifier would finally evaluated to $N1 = 0$ and $N2 = 1$. However, higher V_{t5} delays the discharging current through M_3 and M_5. The left branch may lose its discharging priority, leads the sense amplifier to meta-stable state or even flip to the wrong direction.

PNAND cells also have smaller noise margin than standard CMOS cell. Sense amplifier evaluates result by resolving race condition in N5 and N6. For worst input case $\frac{n+1}{2} : \frac{n-1}{2}$, noise injected from inputs or coupling capacitor may disturb the race condition, causing wrong evaluation result. In digital circuit, noise mainly comes from signal coupling and substrate coupling. Substrate coupling affects delay variation while signal coupling affects functional failure. These failures can be improved by careful layout techniques.

Monte Carlo simulation is applied to evaluate circuit sensitivity to process variation. Parameters such as mobility and channel dimension vary in each Monte Carlo

simulation case. Both global and local variations are included in foundry set statistic parameters. Clock to Q delays in both rise and fall conditions are used in performance evaluation. 100,000 Monte Carlo cases are included in statistic analysis. In 65nm, a functional failure is defined as delay larger than 6ns. The delays and yields of PNAND-n (n=3,5,7,9) are shown in Table 3.7. No functional failures are detected in worst input cases, showing robustness of the designed PNAND cell library.

Table 3.7: PNAND cells yield with process variation. The yield is evaluated by 100,000 Monte Carlo simulation. Statistical corner provided by foundry is used in Monte Carlo simulation. Supply voltage is 1.2V and operation temperature is $25°C$. Output load is set to $20fF$.

Input Case (n_LIN:RIN)	Yield	Rise Delay		Fall Delay	
		Mean (ps)	STDEV (%)	Mean (ps)	STDEV (%)
9_5:4	100	229	4.37	224	4.46
9_4:3	100	215	3.72	213	3.29
7_4:3	100	186	3.76	187	3.74
5_3:2	100	176	2.84	189	3.17
3_2:1	100	162	2.47	174	2.87

3.1.6 Layout Technique

PNAND-n cell layouts are critical to cell performance. Applying some analog techniques, sense amplifier mismatch can be significantly reduced by layout matching. To use PNAND-n as standard cells in digital circuits, their layout have to follow the basic standard cell layout guide in order to be used in digital flow. In the mean time, PNAND-n layout should be also as compact as possible to reduce both parasitic and area.

Layout of PNAND-9 is shown in Fig. 3.9. Double height standard cell with GND-VDD-GND supply pattern is applied for PNAND-9. In order to minimize parasitic

mismatch, PNAND cell in Fig.3.3 is split into two sides. LIN, left side of sense amplifier and half of symmetric latch are placed on the top, the other side transistors are placed on the bottom side, make the layout almost horizontal symmetrical. Cross coupled wires are routed vertically through VDD using M2 layer. M2 wires are spaced to reduce coupling capacitance.

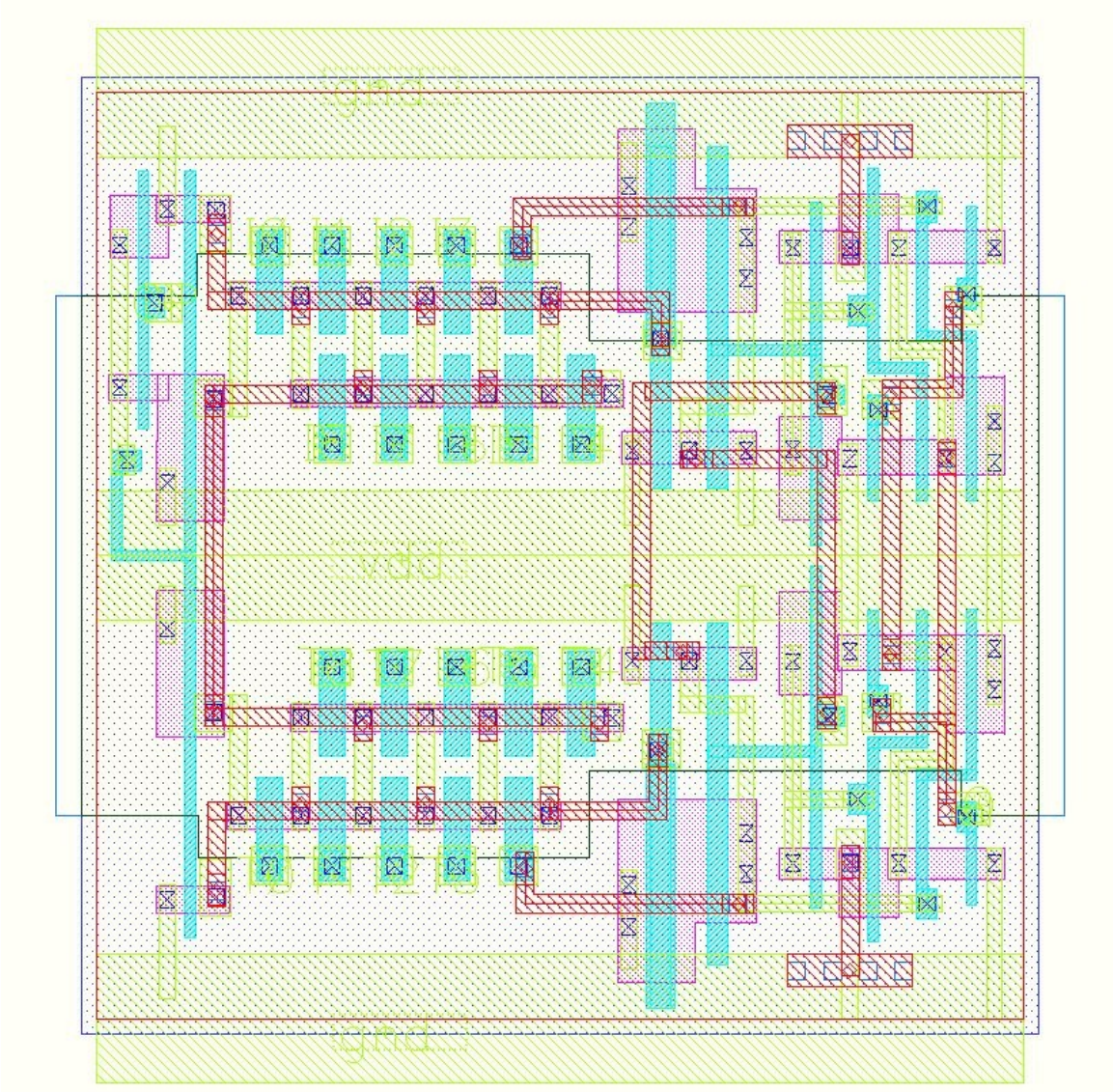


Figure 3.9: Layout of PNAND-9 in double height standard cell format.

Both M1 and M2 layers are used in PNAND cells, and no global routing using M2

is allowed on the top of PNAND-cells. The placement of 18 input pins guarantees that there there are enough space to drop pins from M3 or above metal layers. As shown in Fig. 3.9, nodes N5, N6, N1 and N2 are routed using M2 to reduce internal routing congestion. As discussed before, N5/N6 and N1/N2 are sensitive to noise coupling during sense amplifier evaluation. If there are active signals routed above or close these four nodes, the coupling noise would degrade the cell performance. In worst case, the coupled noise may trigger an erroneous switching, cause circuit malfunction. Fig. 3.10 shows an improved PNAND-9 layout. Instead of using M2 layer for N5 and N6, these two nodes are made to M1 only. To protect N5/N6 from random signal coupling, two M2 plates connected to GND are place on the top of N5 and N6. Any coupling noise would be shielded from N5/N6 and coupled to GND only.

3.1.7 Extension: Preset, Clear and Scan

Similar as D-FF, PNAND cell can be extended to include scan mechanism, asynchronous preset and clear function, as shown in Fig. 3.11. PREZ is active-low preset signal, CLR is active-high clear signal. TE is scan clock and TI is scan data input. It is illegal to active PREZ and CLR on the same time. When PREZ = 0 and CLR = 0, the NOR gate would turn off M_{11} and turn on M_{13}, pulling node N1 to ground. The feedback loop in sense amplifier would then flip N2 to VDD accordingly. $(N1, N2) = (0, 1)$ instantaneously sets SR latch to 0, without CLK involved. The asynchronous clear function operates similar as preset with PREZ = 1 and CLK = 1. In this case, M_{14} is turned on and M_{12} is turned off. The feedback loop sets $(N1, N2) = (1, 0)$ and SR latch sets output Q to 0. The output value would be kept in latch until next clock cycle starting a new input evaluation.

Then scan mechanism is slightly altered for PNAND cell. In Fig. 3.11, TE is scan clock and TI is scan input. TE is kept low during normal mode $TE = 0$. In scan

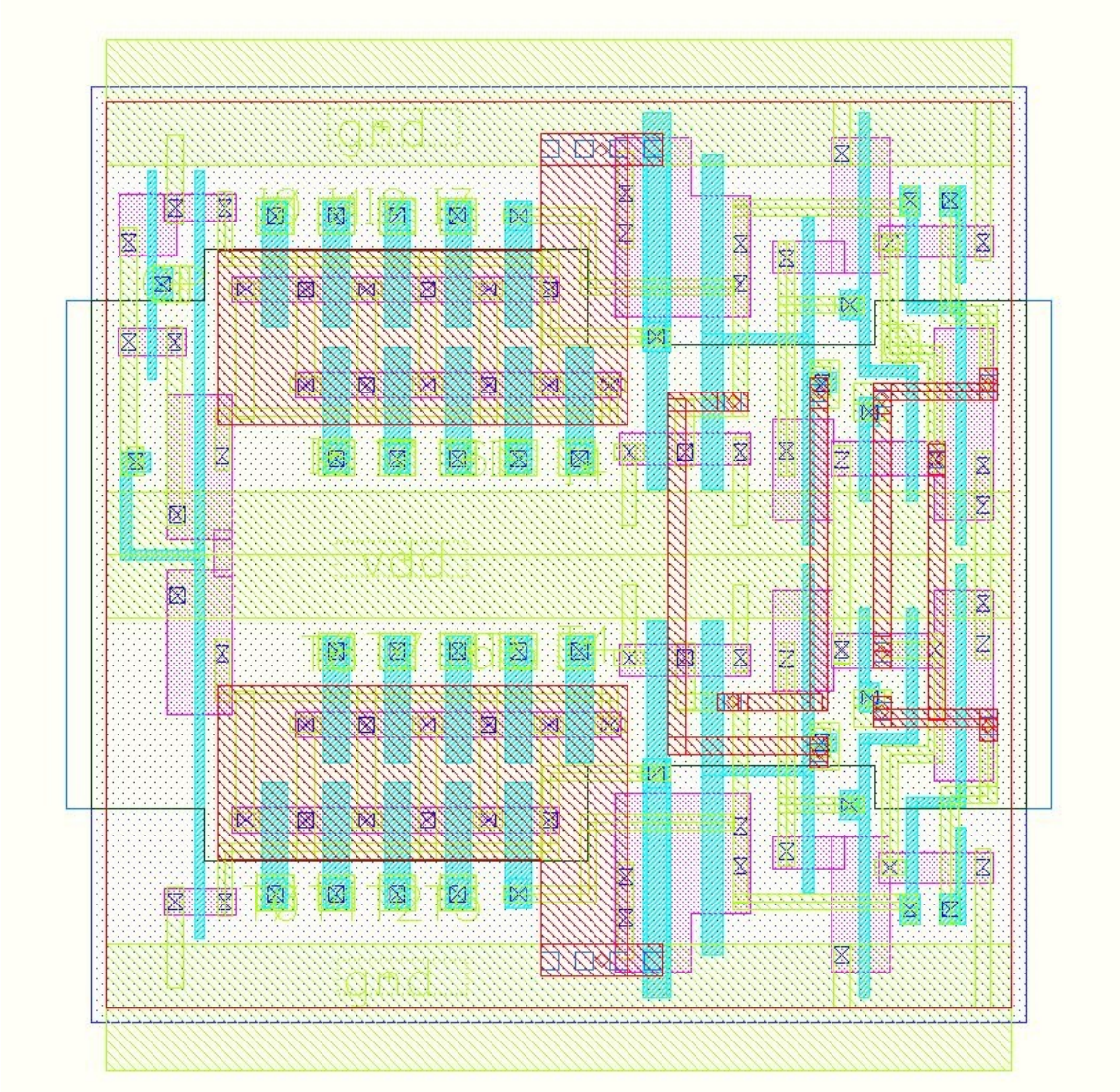


Figure 3.10: PNAND-9 layout with M2 shield on the top of N5 and N6.

mode, input clock CLK is turned off ($CLK = 0$). Instead of toggle CLK, TE acts as clock in scan mode. When $TE = 0$, sense amplifier is on reset mode where node N1 and node N2 are reset to '1'. Since CLK is kept low, node N5 and N6 are kept to ground during the entire scan procedure. When TE switches $0 \rightarrow 1$, M_{20} and M_{21} turns on. Assume $TI = 1$, M_{22} is ON and M_{23} is OFF. When $TE = 0 \rightarrow 1$, node N1 discharges to ground through M_3, M_{20} and M_{22} while N2 is kept to VDD as M_6 and M_{23} are all OFF. Therefore, $(N1, N2) = (0, 1)$, and output Q is set to 1 by SR latch.

If $TI = 0$, M_{23} is ON. When TE rises, Node N2 would be discharged through M_4, M_{21} and M_{23}, therefore $(N1, N2) = (1, 0)$. Output is set to 1. Generally, when scan clock TE rises, the output would follow the TI value, implement a scan mechanism.

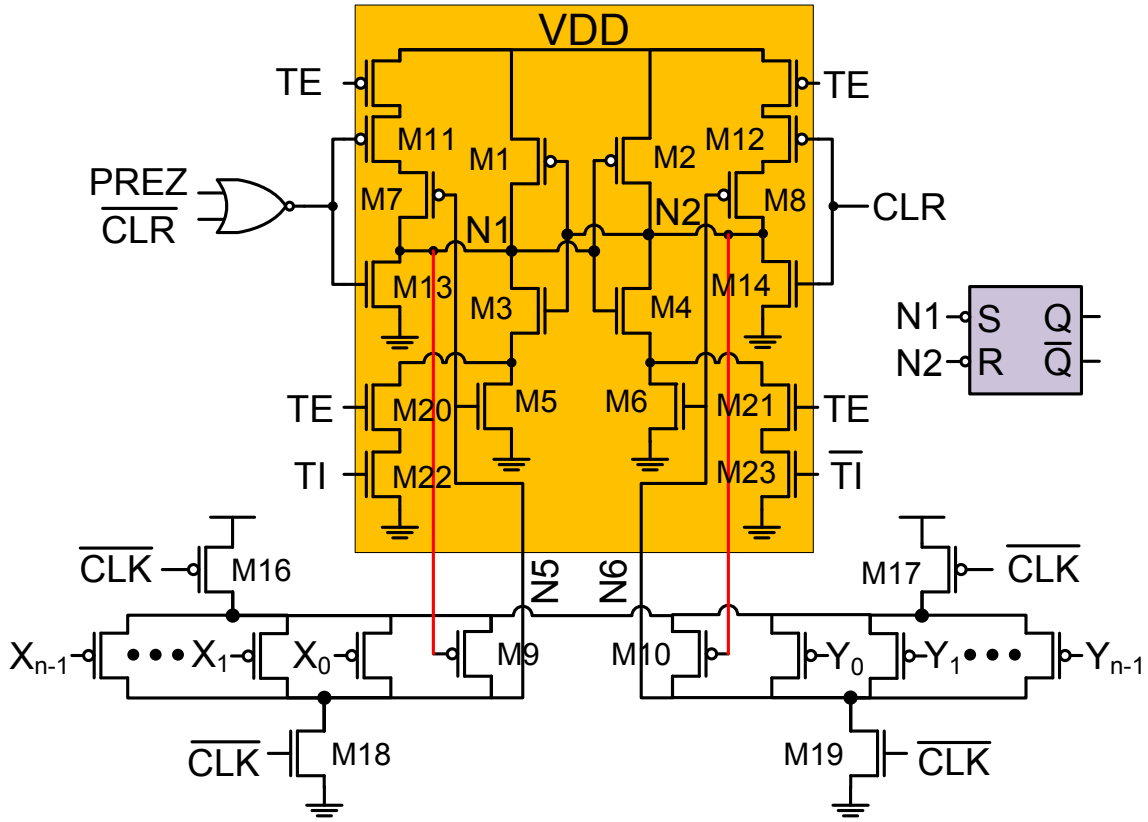


Figure 3.11: PNAND cell with scan, asynchronous preset and clear function.

3.2 Single Input Threshold Gate – Differential D-flipflop

The simplest threshold function is single input identity function $f(x) = x$. It can also be denoted as $[w_x = 1; T = 1]$. Therefore a differential D-flipflop is a special case of a differential mode threshold logic gate. A differential D-flipflop works well with low-swing (not rail to rail voltages) inputs, such as in a register file. The circuit has low clock capacitance and doesn't need an inverted clock signal. However, because it is differential, it needs both the input (D) and its complement ($\overline{D}$) and returns an

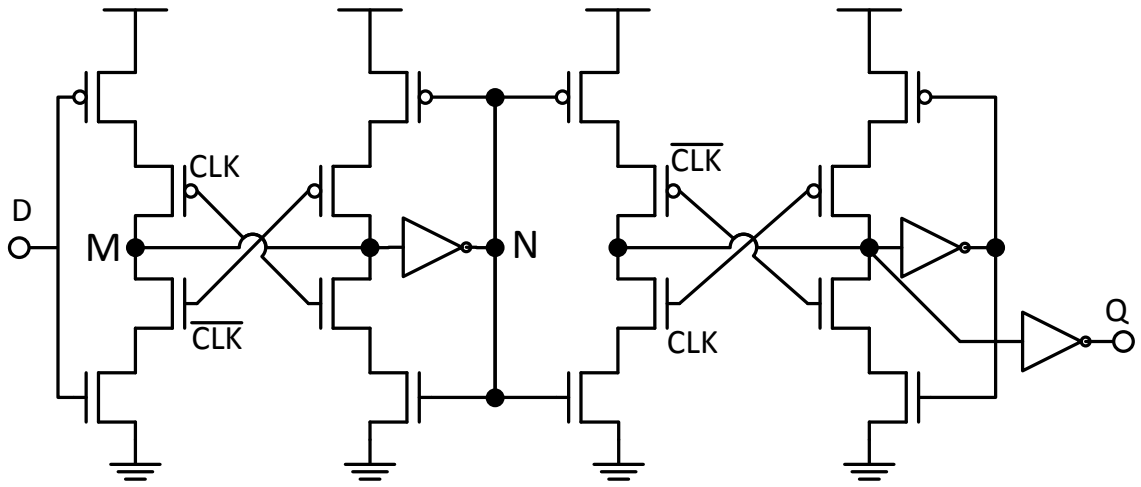


Figure 3.12: Edge triggered flipflop design: master-slave D-flipflop (DFF)

output (Q) and its complement ($\overline{Q}$).

Setup time (t_{su}) is defined as the minimum time required for the input to settle before the rising edge of the clock signal. Clock to output delay or clock to Q propagation delay (t_{c2q}) is defined as the time it takes for the output to settle after the rising edge of the clock.

In this section, two single input threshold gate designs are compared with two D-flipflop designs for delay, energy cross two technology nodes:65nm and 28nm.

3.2.1 Circuit Designs

Master-Slave D-flipflop (DFF) Weste and Harris (2010) is shown in Fig. 3.12 for comparison. When the clock signal (CLK) is low, the input signal (D) propagates through the master latch and reaches the node N. The previous Q value is stored by the slave latch. When CLK rises from low to high, the N value is stored by master latch and output propagates from N to Q and settles as the output Q. Setup time t_{su} is the propagation delay of the master latch, and Clock to output delay t_{c2q} is the propagation delay of the slave latch.

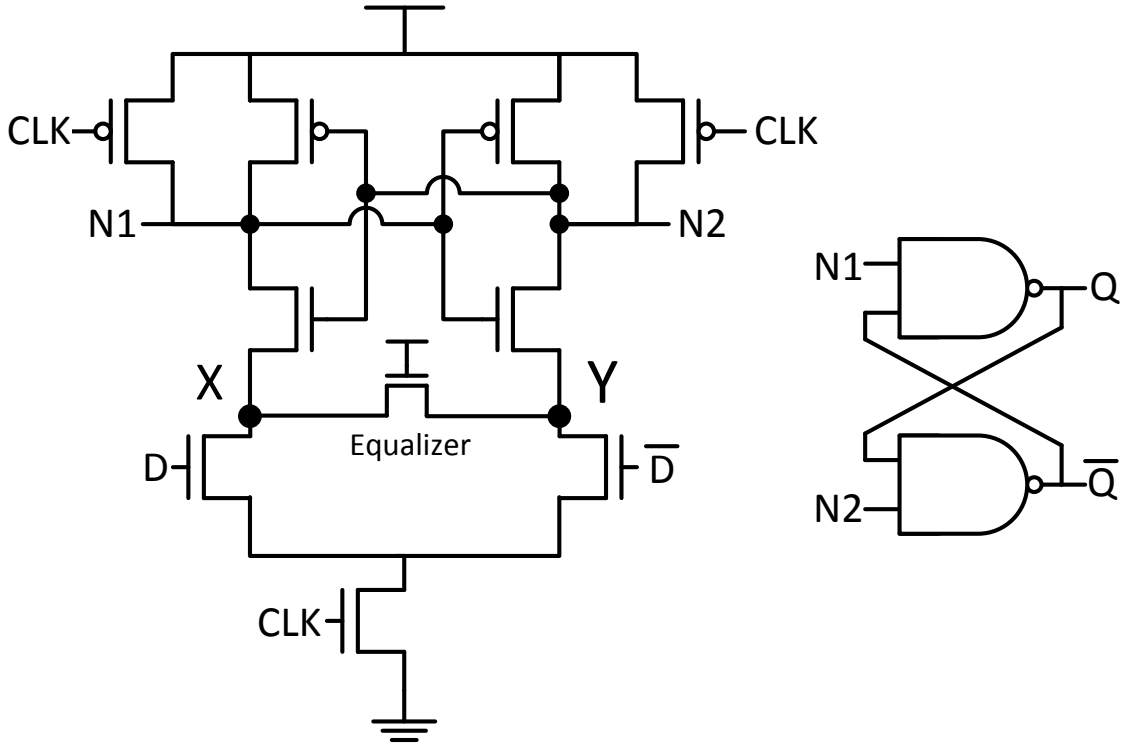


Figure 3.13: Differential Sense-Amplifier flipflop(SAFF) with SR latch

Sense amplifier based flipflop (SAFF) shown in Fig. 3.13 is a differential mode D-flipflop. When the clock signal (CLK) rises from low to high, either node N1 or N2 falls to zero depending on the input D. The weak ON NMOS transistor (Equalizer) between node X and Y is necessary for floating node issues discussed later in this paper. A basic NAND type SR latch is attached to the output nodes N1 and N2, to avoid glitches. For the SAFF, t_{su} is almost zero or even negative, and t_{c2q} is the propagation delay of sense amplifier and latch. Note that both N1 and N2 eventually fall to zero, however the sense-amplifier settles in a state dictated by which node falls faster.

Single-input TLG (TLG-1) shown in Fig. 3.14 has a similar operation mechanism as SAFF. When CLK is low, nodes N1 and N2 will be pulled high. When CLK rises, either the node N1 or N2 will discharge according to input D value. The power

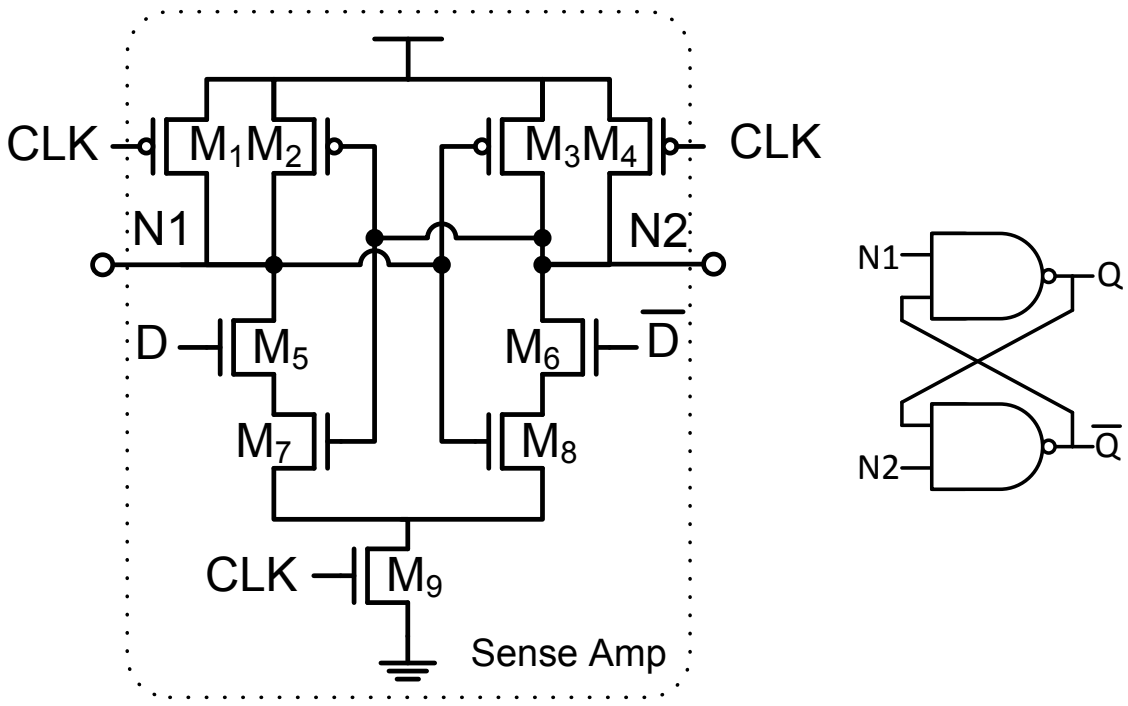


Figure 3.14: Single-input TLG (TLG-1) with SR latch

consumption is be less than the SAFF because there is less parasitic capacitance that is discharged during the evaluation phase. Note that only one of N1 or N2 falls to zero and there is no contention (race) in the evaluation of the outputs.

Reliability enhanced single-input TLG (KVFF) shown in Fig. 3.15 is an improved version of TLG-1, referred as KVFF. The circuit is the same as TLG-1, with two keeper circuits M_{fb1} and M_{fb2}, consisting of minimum sized transistors, are added on both nodes N1 and N2. This avoids N1 and N2 nodes floating at any point when CLK is high. The operation is identical as TLG-1 except for the keeper circuits which are activated after one of nodes N1 or N2 falls to zero. After sense amplifier evaluation is done, the winning side output is discharged to '0', which turns on the feedback transistors on its side. The feedback transistor M_{fb1}/M_{fb2} provides an alternate path from output to ground. The KVFF diminishes the floating node problem of TLG-1. Once the evaluation is complete and N1 or N2 settles at zero,

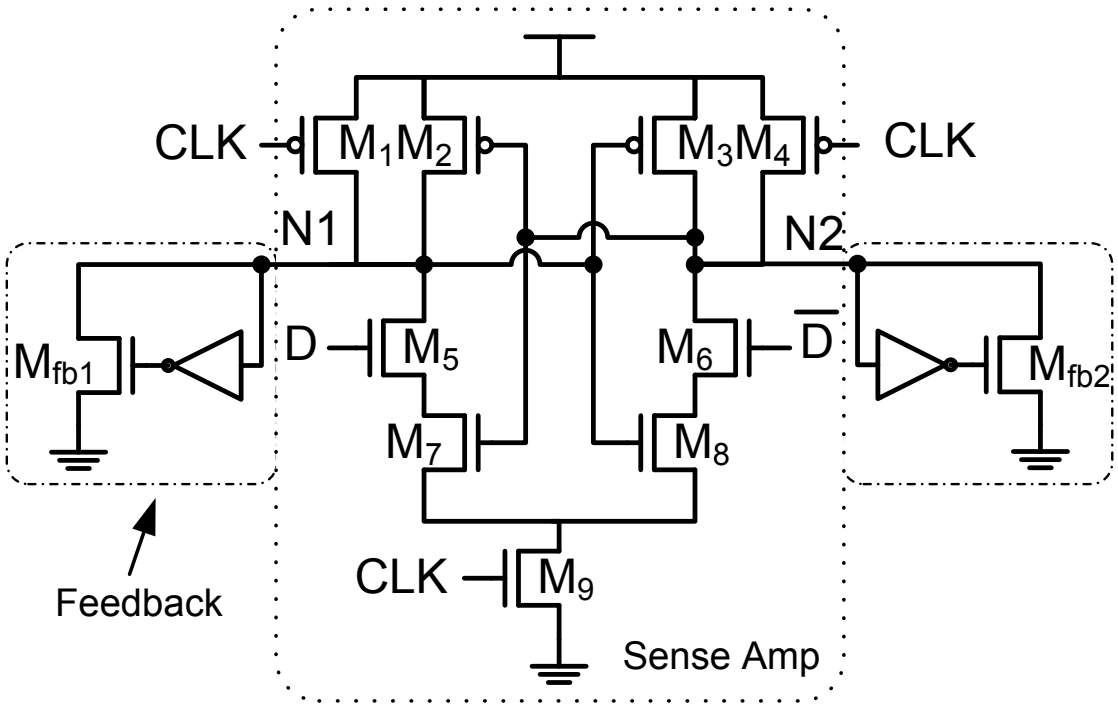


Figure 3.15: The schematic of improved single-input threshold gate (KVFF) without latch

changing input in the same clock cycle won't cause floating nodes because of the keeper circuits. This feedback mechanism also enhances the circuit reliability against noise injection after evaluation.

3.2.2 Experiment Results

We compare four basic designs, master-slave D-flipflop (DFF) from a commercial standard cell library, sense amplifier based flipflop (SAFF), single-input TLG (TLG-1) and enhanced single-input TLG (KVFF).

Performance Comparison

Both the setup time (t_{su}) and the clock to output delay (t_{c2q}) are considered by all synthesis tools when performing timing optimization. For a logic path starting

from a D-flipflop to a D-flipflop, both clock to output as well as setup time must be considered. As such, their sum is important to minimize clock period or enhance timing slack. To do a fair comparison between different sequential elements, the total delay (t_{total}) instead of clock to output delay is used as criteria in our experimental results. $t_{total} = t_{su} + t_{c2q}$

The designs in both 65nm and 28nm process were appropriately sized. We chose the worst case corner for all of the simulations. The worst case corners were both PMOS and NMOS slow (SS), 1.1V supply voltage, and $105°C$ for the design in 65nm process, and SS, 0.9V supply voltage, and $125°C$ for the design in 28nm process.

To make it more realistic in VLSI design, we choose the synthesize corner for our comparison which is the slowest corner in characterized standard CMOS cell library. This corner is used in synthesize procedure to find the maximum operating frequency. The load capacitance and signal transition time are also picked from the characterized library data to make sure the most realistic results.

Table 3.8 and Table 3.9 show the delay, energy and energy delay product (EDP) comparison for the design in 65nm process. As the clock to output delay is similar among all the designs, the total delays are mainly affected by the setup time. In schematic based results, the effective delay of the TLG-1 and KVFF is 28% and 24% faster than the DFF, respectively. Schematic based simulation result shows that EDP of TLG-1 and KVFF is 38% and 21.3% smaller than DFF. In layout based results, these numbers are 33% and 25%. The energy is calculated by integrating the average power consumption over the whole clock period. The input switching activity was 30%.Layout based result shows that EDP TLG-1 and KVFF is 47% and 19% smaller than DFF. The total number of transistors and layout areas are also shown in the respective tables, showing that the KVFF is 20% larger than the Master-Slave D-flipflop and has 7% higher energy consumption.

	t_{su} (ps)	t_{c2q} (ps)	t_{eff} (ps)	Energy (fJ)	EDP $(fJ \times ps)$	MOSFET #
DFF	70	228	298	12	3511	26
SAFF	-7	235	228	11	2489	20
TLG-1	-12	225	213	10	2172	19
KVFF	-16	242	226	12	2763	25

Table 3.8: 65nm technology design comparison (schematic). The simulation is done on slow/slow corner, 1.1V VDD and $105°C$. The load cap is $20fF$. Signal transition time is $70ps$.

	t_{su} (ps)	t_{c2q} (ps)	t_{tot} (ps)	Energy (fJ)	EDP $(fJ \times ps)$	Area (μm^2)
DFF	90	264	354	15	5245	7.8
SAFF	-4	258	254	13	3271	8.32
TLG-1	-6	242	236	12	2805	7.28
KVFF	-11	277	266	16	4233	9.36

Table 3.9: 65nm technology design comparison (layout). The simulation corner is slow/slow, 1.1V VDD and $105°C$. The load cap is $20fF$. Signal transition time is $70ps$.

Table 3.10Table 3.11 show the delay, energy and energy delay product (EDP) comparison for designs in 28nmRVT process. The setup time of three differential mode circuits are again much smaller than the Master-Slave D-flipflop. The difference between rise and fall setup times of input D is significant for designs in 28nm than the same compared to 65nm designs. The reason for this is that the PMOS transistor in the inverter that produces signal $\overline{D}$ is slower than the same in 65nm designs. We can observe that the relative difference between schematic and layout numbers for the four circuits was significantly higher in 28nm designs due to larger contribution of parasitics. The total delay of the proposed TLGs are 25%, 22% faster in layout.

Comparing to SAFF in 65nm, the ETLG-1 is 6ps faster. The energy consumption of TLG-1 and ETLG-1 is 28% and 14% smaller than the Master-Slave D-flipflop, leading to a 46% and 25% improvement in EDP.

	$t_{su}(ps)$	$t_{c2q}(ps)$	$t_{eff}(ps)$	Energy (fJ)	EDP ($fJ \times ps$)	MOSFET #
DFF	28	78	106	4.14	443	26
SAFF	7	82	89	3.67	327	20
TLG-1	4	78	82	3.48	287	19
KVFF	3	82	85	3.93	334	25

Table 3.10: 28nmRVT technology design comparison (schematic). The simulation corner is slow/slow, 0.9V VDD and $125°C$. The load cap is $7.5fF$. Transition time of all signals is $65ps$.

	t_{su} (ps)	t_{c2q} (ps)	t_{tot} (ps)	Energy (fJ)	EDP ($fJ \times ps$)	Area (μm^2)
DFF	30	100	130	5.93	768	2.6112
SAFF	10	97	107	4.60	490	2.9725
TLG-1	10	87	97	4.27	415	2.7982
KVFF	8	93	101	5.08	513	3.6194

Table 3.11: 28nmRVT technology design comparison (layout). The simulation corner slow/slow, 0.9V VDD and $125°C$. The load cap is $7.5fF$. Transition time of all signals is $65ps$.

Table 3.12, 3.13 and 3.14 show the delay, energy and EDP comparison for DFF as well as two KVFF cells in 40nmGP technology with three input transition times. Both KVFF cells(KVFF1 and KVFF2) have same transistor size. The layout of KVFF2 follows CMOS standard cell design rule in 40nm while KVFF1 is designed for symmetric parasitic matching. CLK transition time and load cap are the same. In Table 3.12, the input transition is $4.64ps$. The setup time of all three designs

are negative in this case. 40nmGP technology has fast nFET and pFET. The total delay of KVFF1 and KVFF2 are 28% and 14.2% faster than DFF. Standard cell layout constrain such as limited well height and power rail increase local routing inside KVFF. The parasitic in KVFF2 is more than KVFF1, which increase energy consumption during switching. As a result, EDP of KVFF1 is 19% lower than DFF while KVFF2 is 2% higher than DFF.

Input transition time has significant impact on setup time of both DFF and KVFF. Long input transition time would lead to long setup time. Comparing with Table 3.12 which has fast input transition ($4.64ps$), input rising/falling time in Table 3.13 and 3.14 are much higher ($216.6ps$ and $806.7ps$). The setup time of DFF changes from $-12ps$ to $198ps$ and setup time of KVFF increases from $-14ps$ to more than $200ps$. In Table 3.13 where input slew is moderate, the total delay of KVFF1 is 13.2% lower than DFF and EDP is almost the same as DFF. While the EDP of KVFF2 is 12.7% worse than DFF. In Table 3.14 where input is slow, EDP of both KVFF1 and KVFF2 are both higher than DFF.

	t_{su} (ps)	t_{c2q} (ps)	t_{tot} (ps)	Energy (fJ)	EDP $(fJ \times ps)$
DFF	-12.3	145.8	133.5	6.37	850
KVFF1 (symmetric layout)	-11.7	107.8	96.1	7.16	688
KVFF2 (standard layout)	-14.6	129.3	114.6	7.57	867

Table 3.12: 40nmGP technology design comparison (layout). The simulation corner typical/typical, 0.9V VDD and $25°C$. The load cap is $7.3fF$. CLK transition time is 110ps and input transition time is $4.64ps$.

	t_{su} (ps)	t_{c2q} (ps)	t_{tot} (ps)	Energy (fJ)	EDP $(fJ \times ps)$
DFF	40.7	146.2	186.7	6.40	1195
KVFF1 (symmetric layout)	53.8	108.2	162.0	7.17	1161
KVFF2 (standard layout)	48.6	129.3	177.9	7.57	1347

Table 3.13: 40nmGP technology design comparison (layout). The simulation corner typical/typical, 0.9V VDD and $25°C$. The load cap is $7.3fF$. CLK transition time is 110ps and input transition time is 216.6ps.

	t_{su} (ps)	t_{c2q} (ps)	t_{tot} (ps)	Energy (fJ)	EDP $(fJ \times ps)$
DFF	197.9	144.9	342.8	6.70	2297
KVFF1 (symmetric layout)	234.4	108.1	342.5	7.39	2531
KVFF2 (standard layout)	224.1	129.2	353.3	7.85	2773

Table 3.14: 40nm technology design comparison (layout). The simulation corner typical/typical, 0.9V VDD and $25°C$. The load cap is $7.3fF$. CLK transition time is 110ps and input transition time is 806.7ps.

Setup and hold time distribution

MonteCarlo simulation is applied to evaluate setup time and hold time distributions. Fig. 3.16 shows statistic features of setup and hold time for different designs. Master-slave D-flipflop has a large positive setup time and negative hold time. All threshold gate and differential flipflop have positive hold time and small setup time. All multi-input threshold gates have negative setup time. The standard deviation cross 100 MonteCarlo samples are also shown in figure. PNAND-3 has minimum MonteCarlo variation for both setup time and hold time. In large circuit synthesis, it is the total delay $(t_{su} + t_{c2q})$ that mostly impact circuit's clock frequency and area. With small setup time and relative similar C2Q delay, the differential mode flipflops and threshold gates would improve area and power comparing to masterslave D-flipflop.

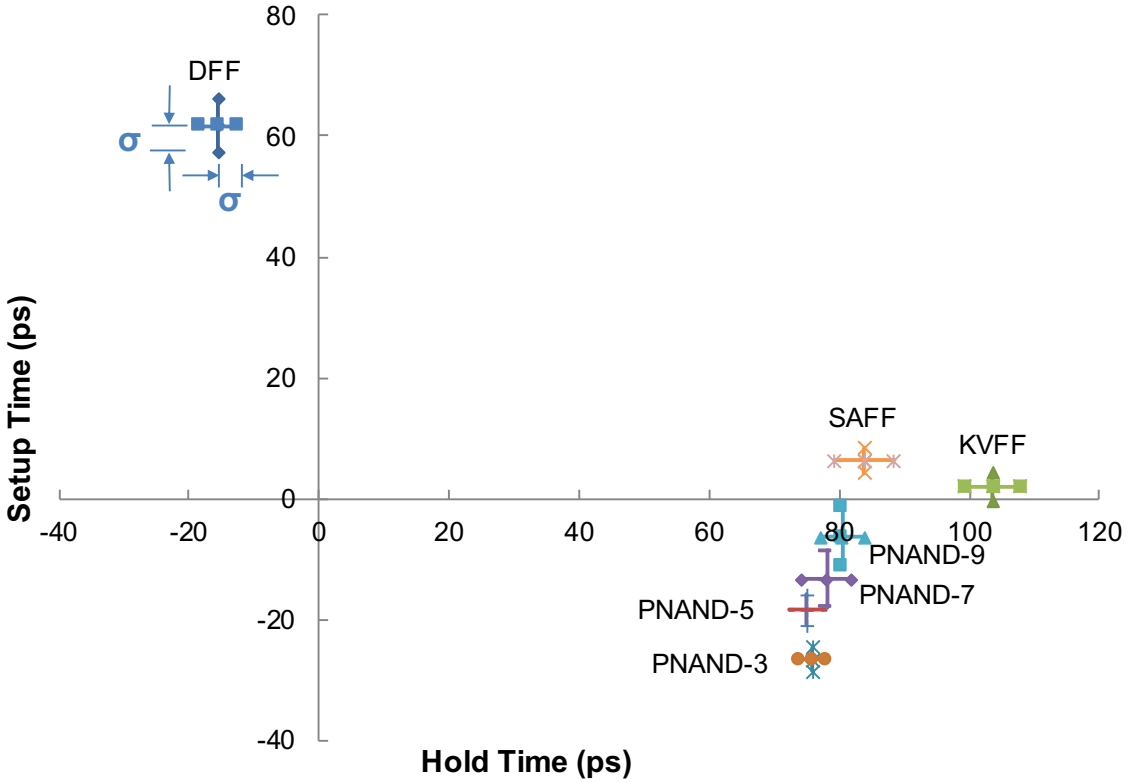


Figure 3.16: Setup time and hold time distribution comparison on 65nm designs. 100 MonteCarlo simulations are applied on foundry set statistic corner, 1.2V VDD and $-40°C$. The data point center is the mean value (hold time, setup time) and the vertical/horizontal bar is the standard deviation.

Reliability Comparison

We compare the four circuits with respect to two reliability criteria. The first criteria is sensitivity to process variation. The sense amplifier in (SAFF) Fig. 3.13 works well without process variation. However, if there is transistor mismatch due to process variation, the evaluation may prefer one side compared to the other. Therefore the size of transistors may affect the reliability of the SAFF under process variation.

The second criteria is sensitivity to noise. Noise sensitive circuits may suffer an evaluation error or a so called soft error. A soft error is caused by a sudden injection of charge from alpha particles, cosmic radiations or even coupling signal noise. A floating node at any point is vulnerable to charge injection. The equalizer in Fig. 3.13 is used

to avoid floating nodes N1 or N2 when CLK is high (evaluation phase) and the input D switches. There is a trade off concerning the size of the equalizer transistor. If the equalizer transistor is too large, the evaluation by sense-amplifier may be affected. If the equalizer transistor is too small, it may not be able to counteract the charge injection on nodes N1 and N2.

In order to assess sensitivity to process variations, 100,000 MonteCarlo simulations were run on all four circuits. All the parameter settings were set by statistical corner which uses the foundry's fabrication data. DFF, TLG-1 and ETLG-1 had **zero** out of 100,000 functional failures for MonteCarlo simulations indicating highly reliable operation. While SAFF had **five** out of 100,000 functional failures in 28nm processes, indicating the influence on reliability by equalizer transistor. Please note that the length of the equalizer transistor in SAFF was increased beyond minimum length in order to obtain zero functional failures in MonteCarlo simulations.

A breaking capacitor experiment was carried out to demonstrate the sensitivity to noise. In this experiment, a pre-charged capacitor is connected to a sensitive node through an ideal switch (Fig. 3.17 (a)). After CLK is settled to high, input D is switched to an opposite value and current pulse is injected on the sensitive node under test. The sensitivity of the circuit depends on how large a current pulse it can tolerate without switching the state. The total charge injected on the node can be increased by increasing the capacitance value. The sensitivity is directly proportional to the minimum charge (Q_{crit}) that can switch the state of the circuit.

To test differential mode circuits, the capacitor was connected to N1 (or N2 since the circuit is symmetric) through ideal switches. The input signals are shown in Fig. 3.17 (b). The minimum Q_{crit} between two nodes is the critical charge of the circuit. A similar experiment was applied to all nodes in the Master-Slave D-flipflop to find the weakest node, which was node M in Fig. 3.12. The Q_{crit} for all four designs

in both 65nm and 28nm process is shown in Table 3.15. The Q_{crit} of KVFF is three times bigger than TLG-1, which is on the same order as Master-Slave D-flipflop.

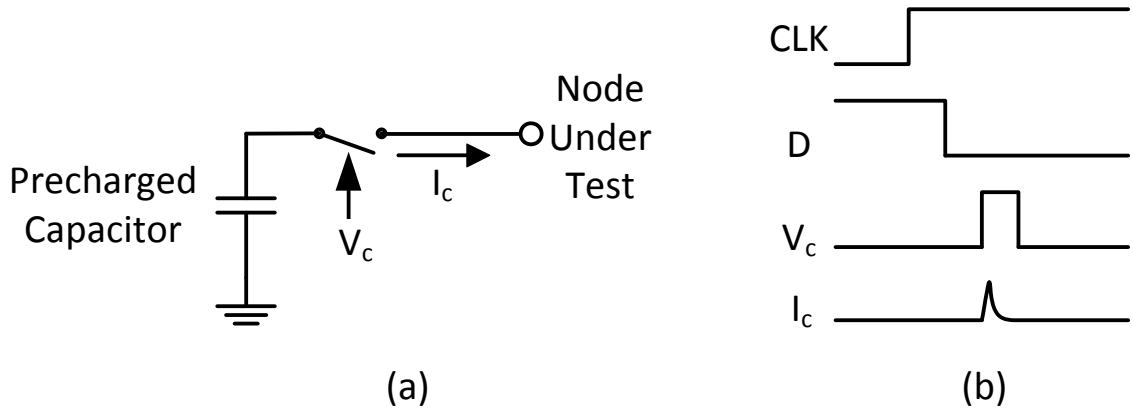


Figure 3.17: The breaking capacitor experiment to determine the node reliability against radiation and coupling noise: (a) The test circuit; (b) Signals applied in the test.

Q_{crit} (fC)	65nm Technology		28nm Technology	
	Schematic	Layout	Schematic	Layout
DFF	9.8	11.4	4.2	5.0
SAFF	8.2	11.8	3.9	5.5
TLG-1	2.7	3.7	1.2	1.5
KVFF	8.7	12.1	4.2	5.6

Table 3.15: Breaking capacitor experiment(schematic and layout). The simulation corner is the same as other tables. All four circuits have zero out of 100,000 Monte-Carlo functional failures on foundry provided statistical corner.

3.3 Circuit Implementation and Silicon Verification

Our OSA algorithm combined with transistor sizing takes into account all non-ideal cases of circuit operation. The result in a robust library of PNAND-n cells for $n = 3, 5, 7, 9$ and KVFF cell, collectively realizing a total of 72 threshold functions and

all of their NPN equivalent functions [1] . Using process variation statistics provided by the foundry, no failures were found in 100K Monte Carlo simulations accounting for global variations and local mismatch, ensuring the robustness of PNAND cells.

3.3.1 Automated Design Flow with Threshold Gates

As stated in the introduction, a PNAND cell can be viewed as an edge-triggered, multi-input flip-flop that computes a threshold function. With this view, the synthesis methodology consists of first searching through cones of logic and performing functional decomposition to extract a threshold function that can be implemented by a PNAND Kulkarni *et al.* (2012). The resulting network, referred to as a *hybrid* circuit, consists of conventional and PNAND cells. The method is referred to as *hybridization*, and an example is shown in Fig. 3.18. Table 3.16 shows the results of the replacing the logic in Fig. 3.18 by a threshold gate. The delay is measured by simulation at slow/slow, 105 °C, 1.1V power supply corner, including clock to Q delay and setup time of sequential elements. The leakage is measured at the typical/typical, 25 °C, 1.2V corner. The standard synthesis and physical design tools then further optimize the resulting netlist without modifying the PNAND cells. The hybridization step improves power, area and leakage by absorbing the logic into the PNAND, as well as reducing the output load on the feeder circuit C, which gets reduced in size during the synthesis step. The design flow is shown on Fig. 3.19.

3.3.2 Post P&R Simulation Results

A standard cell library of PNAND-n cells, for $n = 1, 3, 5, 7, 9$ was designed in a 65nm LP technology. An accurate method for setup, hold time and power characterization of the PNAND cells was also developed. Table 3.17 shows performance

[1] input **N**egation, input **P**ermutation, output **N**egation

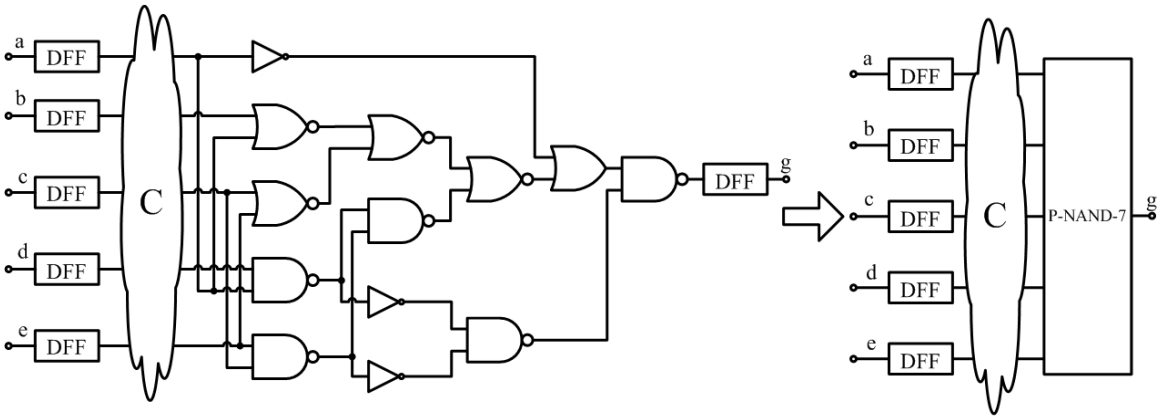


Figure 3.18: Hybridization example: A threshold function replaced by PNAND

Table 3.16: Results of transformation in Fig. 3.18($C = \Phi$)

Parameter	**Conv.**	**Hybrid**	**Improvement**
Delay(ps)	515	276	49%
Area(μm^2)	54	33	38%
Energy(fJ)	63	45.6	27%
Leakage(nW)	5.8	1.8	70%
Total input cap (fF)	77.9	56.1	28%

comparison (post layout simulation) for a 28-bit 4-tap digital FIR filter, an AES crypto-chip, a 32-bit MIPS CPU, and a 64-bit floating-point multiplier using the same PNAND libraries and automated design flow. By introducing our proposed threshold logic based synthesis method, we can further reduce power, power-variation, area, leakage and wire-length of general digital circuits without sacrificing speed.

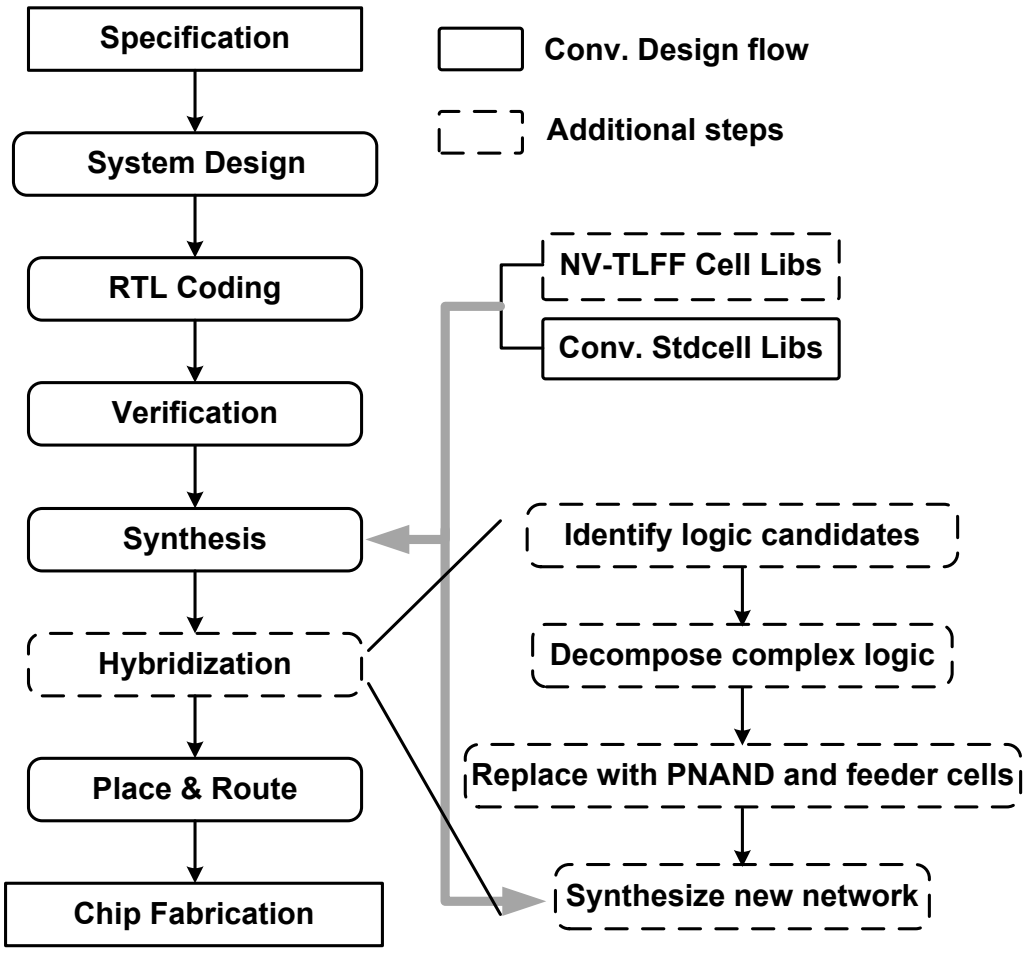


Figure 3.19: Synthesis and hybridization steps

3.3.3 Silicon Verification

The original and improved PNAND cell library and design flow are validated and evaluated on silicon by two separate tapeouts. The first chip included a two stage Wallace Tree multiplier and cell array. Based on the measurement and post layout simulation, both PNAND cell and chip architecture are improved. The improved architecture were applied on second silicon verification where a Booth multiplier and improved cell array were included. The second measurement was on the design's full speed. The results show that the hybrid circuit would be able to run on higher clock frequency than its CMOS counterpart.

Table 3.17: 65nm LP technique mapping improvements of hybrid over conventional for various circuits

Circuit	Power	P Stdev	Leakage	Area	Wirelength
FIR Filter	36%	44%	51%	27%	30%
32-bit MIPS	29%	35%	31%	11%	9%
FP Multiplier	23%	30%	27%	13%	19%
AES crypto-chip	16%	33%	18%	15%	33%

Chip Architecture

Fig. 3.20 shows the chip structure of the first design. The designs under test (DUT) include two functional identical multiplier, one follows the standard CMOS design flow (CMOS multiplier), the other is hybrid generated by threshold hybridization (Hybrid multiplier). The two multipliers are place-and-routed within their own supply domain. Therefore, the power consumption is calculated by measuring current flow through the two separate power supply source.

The PNAND cells in this chip are same as Fig. 3.9. PNAND cells array include two array banks. Each bank includes 32 copies of DFF, TLG-1, PNAND-3,5,9,11, 13, shown in Fig. 3.21. All inputs are directly connected to register files. The outputs are connected to 32 8-bit multiplexer. 3-bit row control signals are applied to select the column under test, and send outputs back to register file.

Register files consist of input register file (IRF) and output register file (ORF). IRF is 16 64-bit register which contains preloaded input test data. ORF has same size and IRF and stores output observed data. value in both IRF and ORF are reached by scan chain before and after test run. During run time, address generator generates Read Addr for IRF and Write Addr for ORF, ensuring data are read and stored in

proper sequence.

The clock signal are generated by two sources, on-chip clock generator and external source. On-chip clock generator includes 8 ring oscillators with various levels and two clock dividers. 24 frequencies can be generated by clock generator, which is sufficient to cover various process corners and test frequencies. The on-chip clock generator is calibrated for each die to provide accurate results. The external clock source is for function verification and obtain data from/to register file. The external clock frequency is lower than 200 MHz because of digital IO limitation. The register file is reached through scan mechanism. since the external clock is synchronized with data sending and receiving, the chip is clocked by external clock during scan process. Fig. 3.22 shows a picture of the fabricated chip. The chip consists of 37 IOs, input/output register files, two multiplier, and a PNAND cell array, the total area is $1mm \times 1mm$.

Fig. 3.23 shows the test signal sequence. Since IRF length is 16, 16 vectors can be tested each round. The single test sequence consists of 2118 external clock cycles. A global reset signal resets all sequential elements at the beginning of each test. Then the scan mechanism is enabled for 1024 cycles, allowing input vectors to be scanned into IRF. Then the address generator is reset again and clock source is switched from external clock source to on-chip clock. After Addr_reset is released, test vectors are sent to DUT in sequence and ORF receives outputs. The read and store address are controlled by address generator. The address generator stops on the last address to prevent address overflow. After the test is done, clock is switched to external clock source. Then the data in ORF is scanned out by Scan_mode signal.

The first design doesn't consider clock glitches during clock source transition. Since the external and internal clock sources are not synchronized, a simple switch from one to the other may generate glitches on CLK signal. Fig. 3.24 shows the

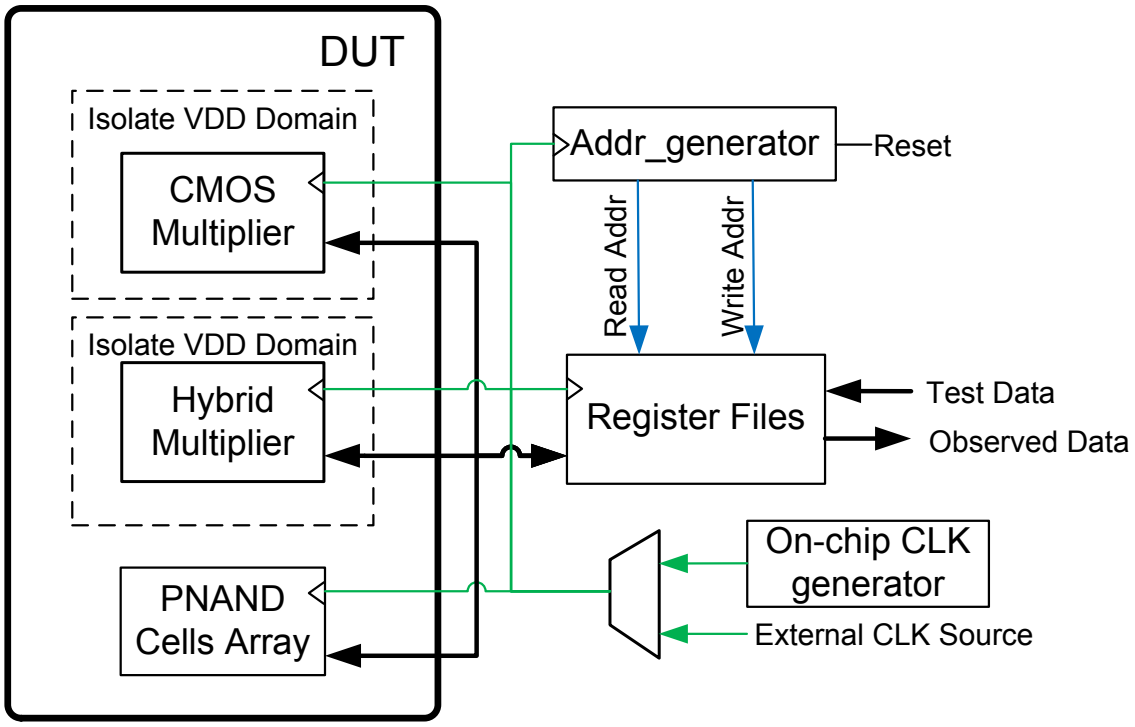


Figure 3.20: Test chip architecture with Wallace tree multiplier and original PNAND cell array

improved architecture in second chip. Booth multiplier and improved PNAND cell library 3.10 are included. Comparing with the first chip, signal synchronizers and an enable signal (EnClk) are added between clock sources and the multiplexer. During the operation, EnClk is set to 1. The internal clock is disabled when CLK Enable is 0. The synchronizer consists of two serial connected flipflops. The input D of first flipflop is driven by EnClk and clock pins are driven by clock signal that need to be synchronized with.

Fig. 3.25 shows the new test sequence. EnClk is turned into 0 when clock source switching is required. As being synchronized with current clock, it would turn off the chip clock without any signal glitch. After clock transition is complete, EnClk is turned back to 1, passing new clock to the entire chip.

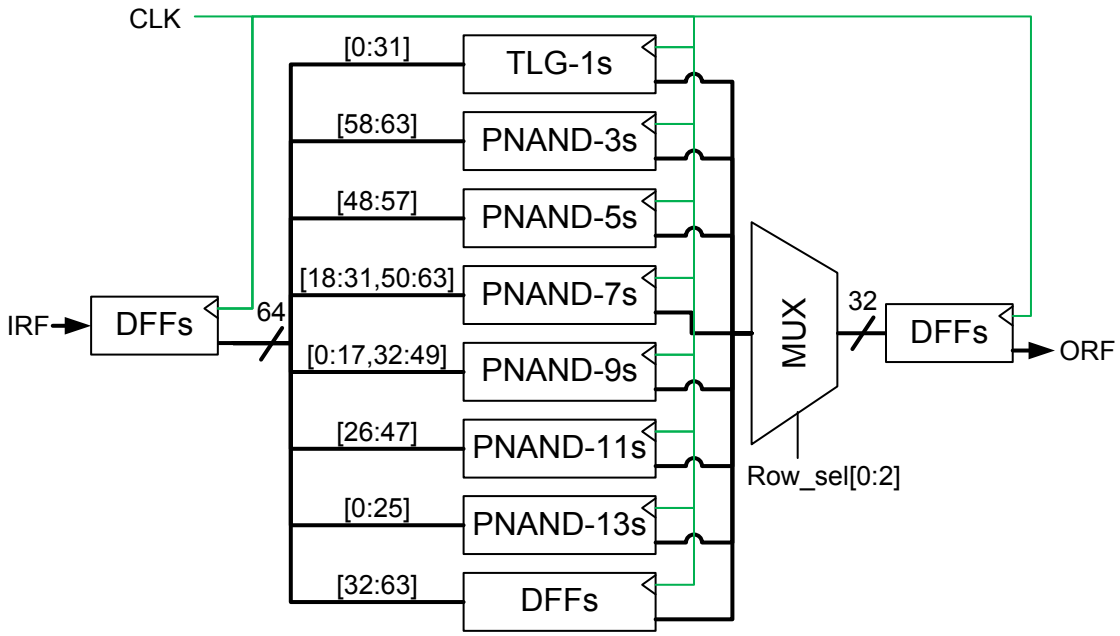


Figure 3.21: PNAND cell array structure

Two-stage Wallace Tree Multiplier

A two-stage 32-bit signed Wallace Tree multiplier was designed by our automated design flow. The conceptual idea of applying hybridization to the two-stage multiplier is shown on Fig. 3.26. The hybrid, as well as a conventional multiplier design, were fabricated in the same technology for design flow verification.

Fig. 3.27a, Fig. 3.27b and Fig. 3.27c show measured and simulated values of dynamic power as a function of frequency for 3 different switching activities. The average improvement of hybrid over conventional multiplier is 34.7%. Fig. 3.27d, Fig. 3.27e and Fig. 3.27f show power as a function of switching activity for 3 different switching frequencies. The average power improvement is 34.4%.

Fig. 3.28 shows the average (over 19 dies) energy-delay product (EDP) versus frequency, and the improvement of hybrid multiplier over conventional multiplier at 30% input switching activity. The vertical bar in the top part of Fig. 3.28 show the 3σ range of the EDP around the mean μ over the 19 dies. Note the $\mu - 3\sigma$ EDP

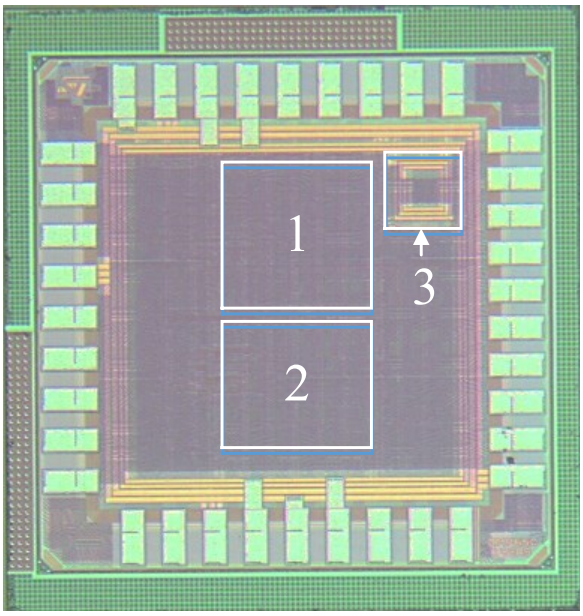


Figure 3.22: Die photo of prototype chip:(1) Conventional multiplier; (2) Hybrid multiplier; (3) Clock generator

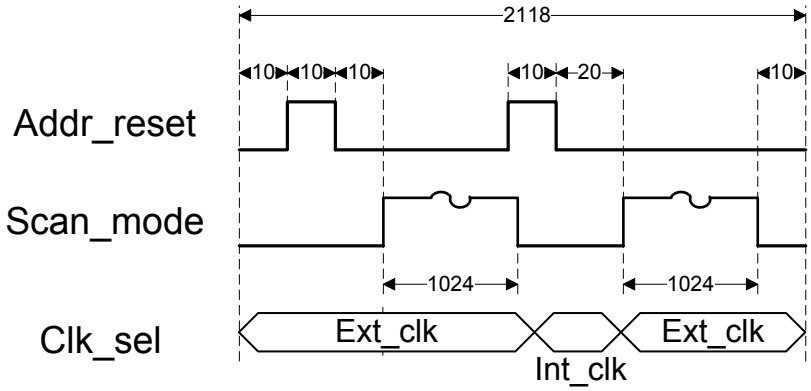


Figure 3.23: Input sequences for single test round

point of the conventional multiplier is significantly higher than the $\mu + 3\sigma$ of the hybrid multiplier, at all frequencies. The lower part of Fig. 3.28 shows the percentage improvement in the EDP of the hybrid multiplier, and the standard deviation over the 19 dies, at each frequency. The energy-delay product for the hybrid is about 34% lower than the conventional design. Table 3.18 summarizes the two multiplier measurement results.

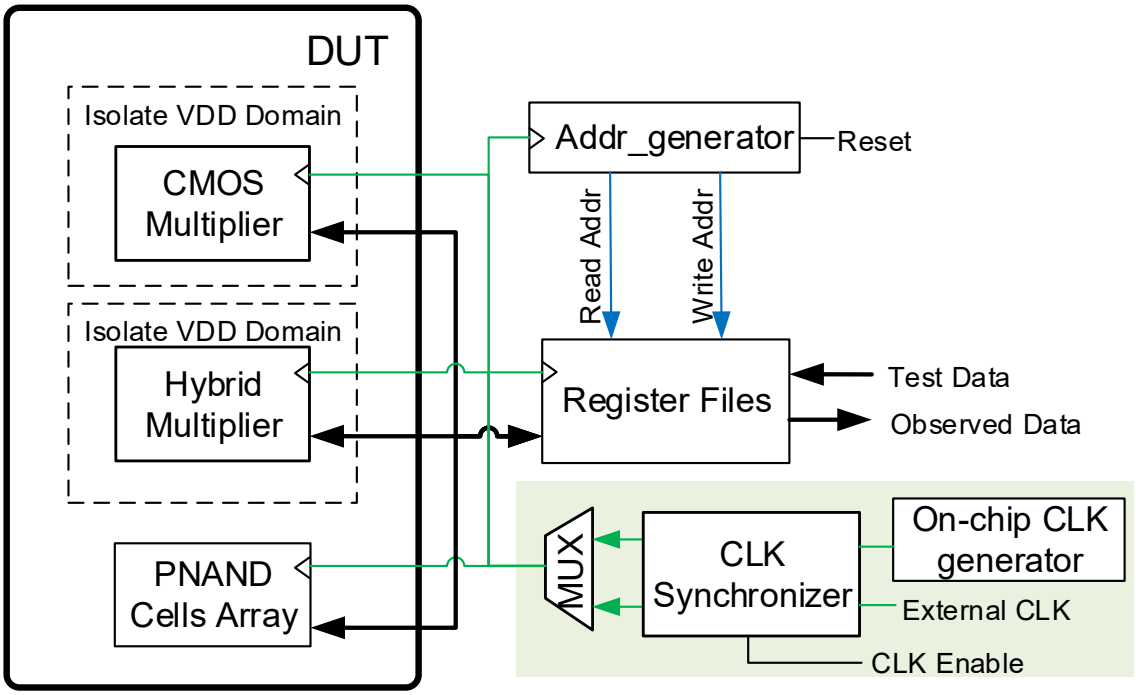


Figure 3.24: Test chip with Booth multiplier and improved PNAND cell array

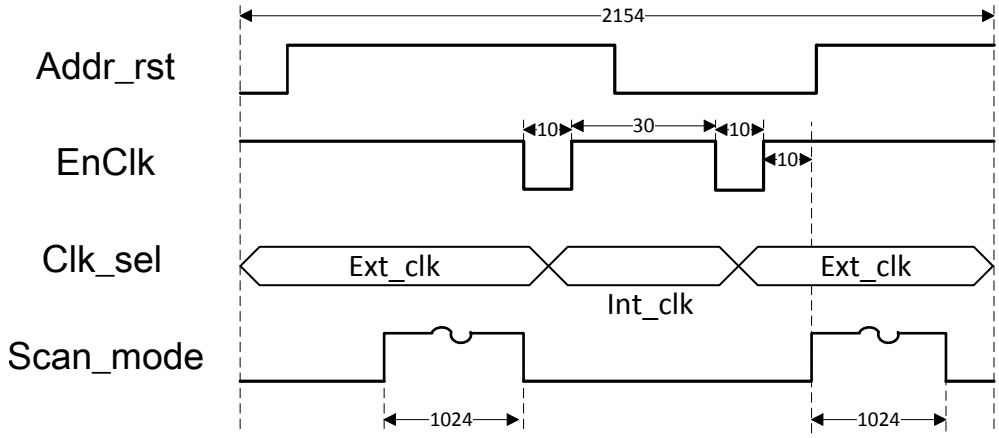


Figure 3.25: EnClk signal is included in test sequence to remove glitches during clock transition

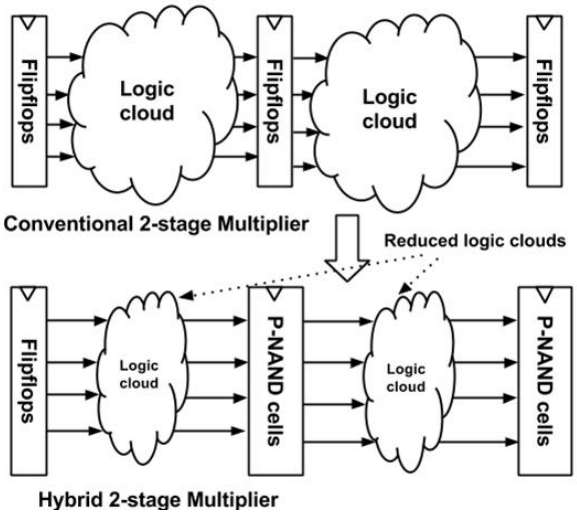
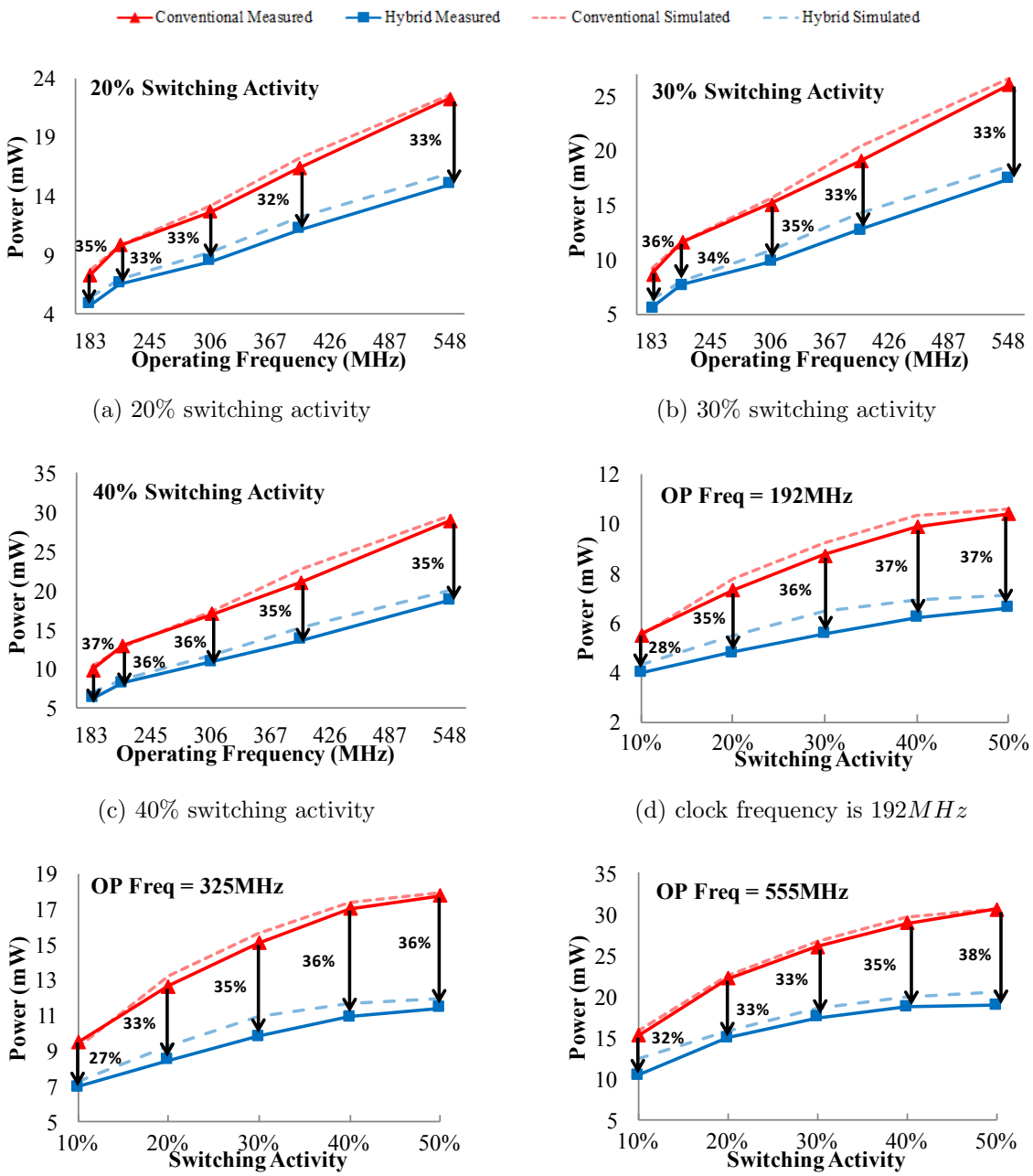


Figure 3.26: Hybridization of multiplier

Table 3.18: Test results of conventional and hybrid multipliers

Specification	**Conv.**	**Hybrid**	**Imp.(%)**
Supply V_{DD}	1.2V	1.2V	–
Area(μm^2)	41814	31680	24%
Leakage(μW)	8.1	4.1	49%
Wire-length(μm)	160160	87243	45%
# Std. Cells	5546	4003 (212 PNANDs)	28%
Clock frequency	642MHz, 30% switching activity		
Power (mW)	31.1	20.7	33.6%
Average EDP ($pJ \times ns$)	75.6	50.2	34%



(a) 20% switching activity

(b) 30% switching activity

(c) 40% switching activity

(d) clock frequency is $192MHz$

(e) clock frequency is $325MHz$

(f) clock frequency is $555MHz$

Figure 3.27: Multiplier dynamic power

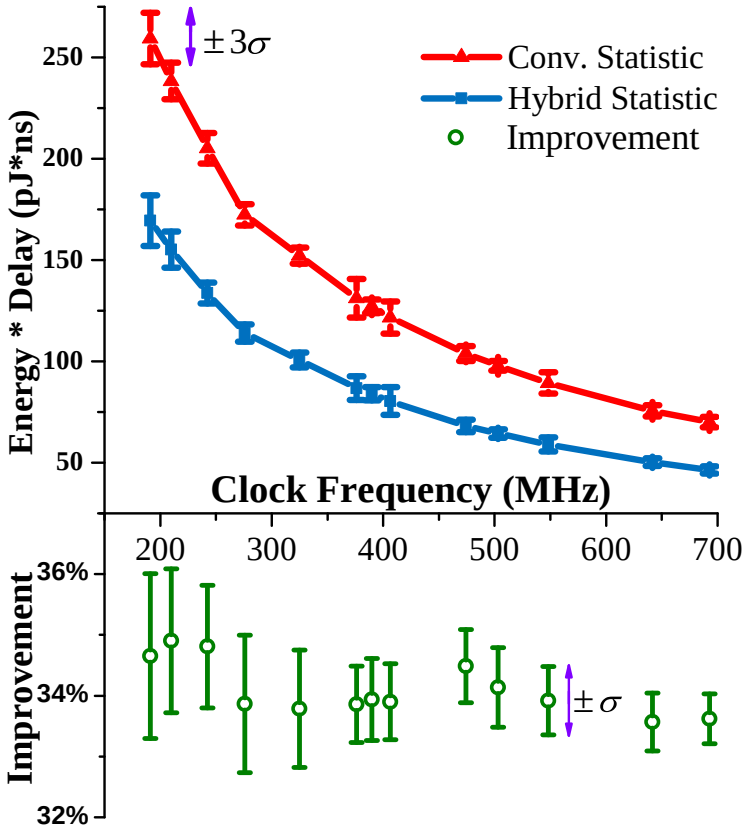


Figure 3.28: Measured energy delay product (EDP) vs frequency over 19 dies. Mean and $\pm 3\sigma$ boundary are shown for EDP, mean and $\pm\sigma$ boundary are shown on improvements below.

Two-stage Booth Multiplier

A 32-bit two stage booth multiplier with similar structure as Fig. 3.26 was designed and fabricated on another batch. This chip includes 36 IOs. The total chip area is $1.024mm^2$.

Fig. 3.29a, Fig. 3.29b and Fig. 3.29c show measured and simulated values of average dynamic power as a function of frequency for 3 different input switching activities. The average improvement of hybrid over conventional multiplier is 30.1%. Fig. 3.29d, Fig. 3.29e and Fig. 3.29f show average power as a function of input switching activity for 3 different operating frequencies. The average power improvement is 29.2%. The vertical bars in Fig. 3.29 show the 3σ range of the power around the mean μ over the 24 dies. Fig. 3.30 shows the average EDP versus frequency with 30% input switching activity. The average EDP of hybrid multiplier is 30.5% lower than CMOS multiplier. Table 3.19 summarizes the two multiplier measurement results. The maximum operation frequency the the maximum clock frequency each multiplier can run without erroneous output bit.

Cell yield

PNAND cell arrays are fabricated to test the functionality. An exhaustive test vectors are applied according to input number n. Meaning that the test vectors include all possible combinations can occur on PNAND-n. The cell functional fail is determined by if any of its exhaustive vectors generates wrong bit.

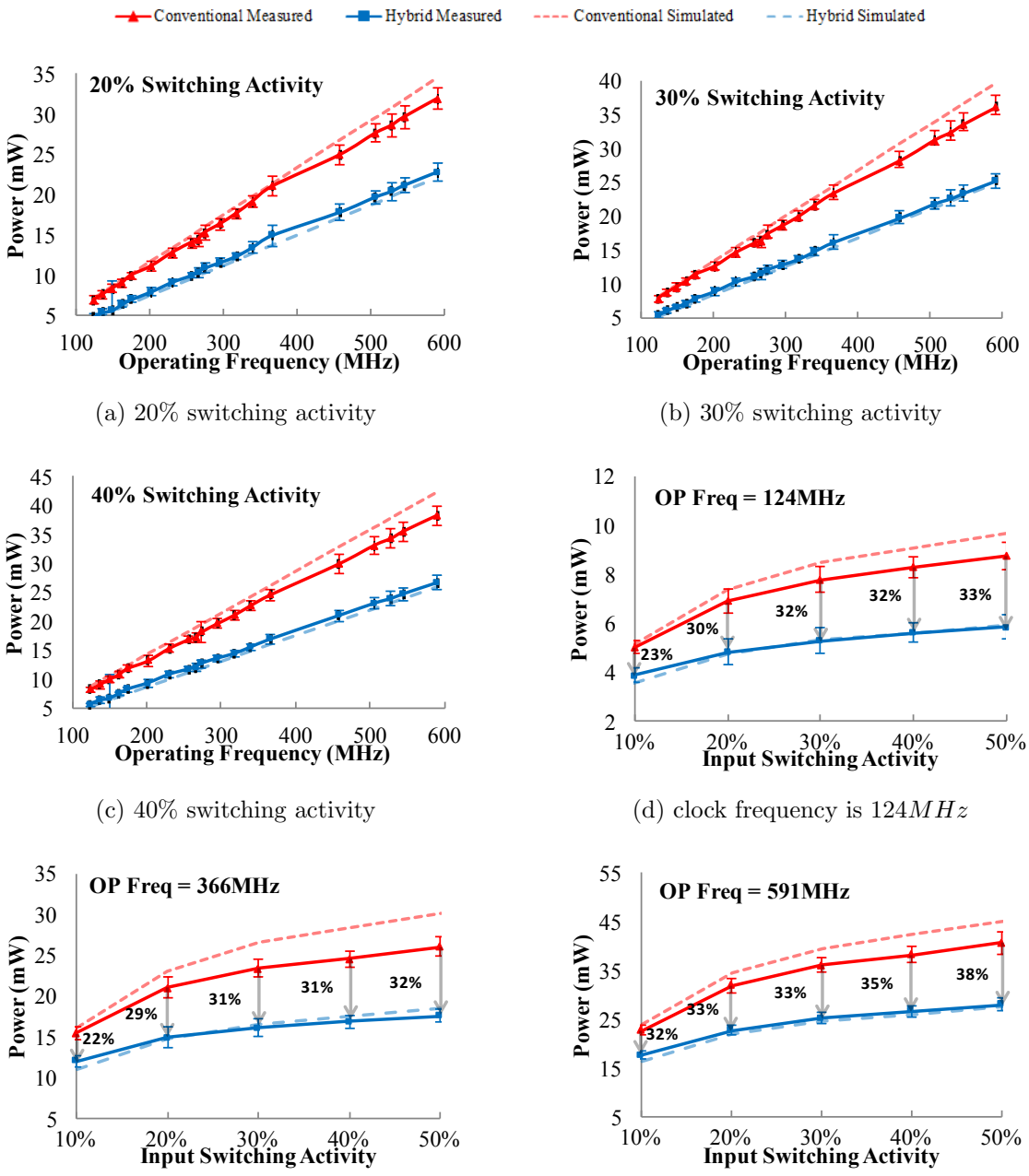
In the first chip, there are total 608 copies of each PNAND over 19 dies. The cell function test are done on 20MHz clock frequency. The yield of PNAND-7 is 98% and the yield of PNAND-9 is 96%.

The second chip was tested on its full speed. The yield of PNAND-7 is 93.1% and

Table 3.19: Test results of booth multipliers

Specification	**Conv.**	**Hybrid**	**Imp.(%)**
Supply V_{DD}	1.2V	1.2V	–
Area(μm^2)	45982	30240	34.2%
Leakage(μW)	11.4	5.7	50%
Wire-length(μm)	194133	130314	32.9%
# Std. Cells	5522	4385 (167 PNANDs)	20.6%
Clock frequency	591MHz, 30% input switching activity		
Power (mW)	36.2	25.3	30.2%
Average EDP ($pJ \times ns$)	103.7	72.4	30.2%
Average Maximum Frequency (MHz)	646	789	22.1%

the yield of PNAND-9 is 96.13%. Yield of the rest cells is 100%.



(a) 20% switching activity

(b) 30% switching activity

(c) 40% switching activity

(d) clock frequency is $124MHz$

(e) clock frequency is $336MHz$

(f) clock frequency is $591MHz$

Figure 3.29: Multiplier dynamic power for booth multiplier

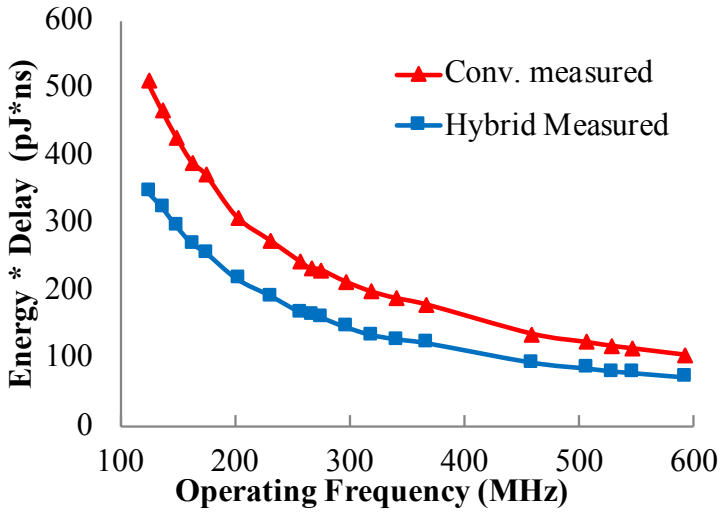


Figure 3.30: Measured energy delay product (EDP) vs frequency over 24 dies.

Chapter 4

TECHNOLOGY SCALING OF THRESHOLD LOGIC GATE

In previous chapter, PNAND cells are designed on 65nmLP technology. The cell and circuit implementation shows significant power and area improvement on normal operation voltage. Exploring threshold logic performance on low geometry and low supply voltage is also very important. In this chapter, we demonstrate how PNAND perform on two process, 40nmGP and 28nmFDSOI technology. We also demonstrate how threshold gate can be modified to increase robustness on low voltage operation.

4.1 PNAND Performance on 40nm GP Technology

PNAND-3, 5, 7, 9 are designed on 40nm GP technology. The normal power supply is 0.9V. Comparing with 65nm designs, the sequential cells in 40nm are characterized in a much wider input transition range. In 40nm, the transition time for input signal is defined as the time difference between signal reach 30% and 70% of supply voltage. In 40nm standard cell library, the cell is characterized with input transition time range from $1.86ps$ to $384.3ps$. In DFF, input signal is passed into master-latch by CLK enabled inverter. When input transition is slow, the inverter recovers the slow input into a sharp signal. Therefore, the setup time degradation is controlled by the signal recovery of inverter chain in master latch. In PNAND cell, the inputs are directly connecting input network without any recovery mechanism. When inputs switch slowly and when the transition is close to CLK rise edge, the slow input transition would slow down the charging of node N5 and N6. Voltages on N5 and N6 control transistors M_5 and M_6 (Fig. 3.3) and discharging speed of node N1 and N2. Therefore, slow input transition causes sense amplifier evaluation time to be

extended. As the result, the setup time of PNAND degrades when input transition is longer. CLK impact on setup time is not as significant as input signals. This is because the sense amplifier in Fig. 3.3 is driven by inverted CLK signal $\overline{CLK}$. The CLK inverter regenerates a much sharper edge from CLK transition to drive the input network. On the other hand, slow clock transition slow down rise time of both N5 and N6 on the same amount, which help sense amplifier to separate them

Table 4.1 shows setup time changes with both input and CLK transition time. It clearly shows that setup increase dramatically with input transition. When Input transition is as sharp as $1.86ps$, setup time of all PNAND cells are negative. When Input transition is median like $86.7ps$, the setup times become positive, from single digit in PNAND-3 to double digits. When input transition is as slow as $384.3ps$, the setup times increase to as high as $500ps$, which is even higher than C2Q delay shown in Table 4.1. The setup time of PNAND cell with small input numbers are generally smaller than PNAND-9. With same input transition, relative slow CLK transition would reduce setup time.

Table 4.1 shows C2Q delay with respect to CLK transition and output load. It is easy to understand that the C2Q delay increases with large output load as it takes more time to charge large load. When CLK transition is slow, the transition time of CLK inverter is also slow. Even though CLK inverter can recover CLK edge on some extent, the transition time of $\overline{CLK}$ which drives input network and sense amplifier is still proportional to CLK transition. A slow $\overline{CLK}$ slows down sense amplifier evaluation. Therefore, C2Q delay is larger when CLK input is slower. The delay difference among different input configurations are not significant. In 40nm GP technology, it is the slew rate that determines PNAND performance.

Fig. 4.1 shows the overall delay of DFF and PNANDs for input transition less than 100ps. The total delays of PNANDs are smaller than DFF when input transition is

Table 4.1: Setup time vs clock transition time of PNAND-3, 5, 7 and 9 in 40nmGP technology. Output load is set to $7.3fF$. The simulation corner typical/typical, 0.9V VDD and $25°C$.

Setup time (ps)			Input transition (ps)		
			1.86	86.65	384.3
CLK transition (ps)	PNAND-3	1.86	-22.4	79.7	461.9
		44.05	-47.5	48.3	422.9
		192.15	-66.9	9.1	377.4
	PNAND-5	1.86	-15.9	94.7	517.1
		44.05	-41.4	63.7	483.9
		192.15	-59.5	26.5	433.1
	PNAND-7	1.86	-17.2	94.7	518.6
		44.05	-42.1	64.0	486.1
		192.15	-60.5	26.9	429.1
	PNAND-9	1.86	-4.9	97.8	527.6
		44.05	-33.2	66.8	495.4
		192.15	-52.6	33.6	440.6

less than 40ps. However, it is not a fair comparison because DFF only computes identity function and PNANDs usually compute a more complex multi-input threshold function. When including delay of equivalent CMOS logic in DFF, the total delay is much higher for CMOS logic equivalent circuit. In this figure, PNAND-3 computes the smallest function, 3-input majority function. The total delay of PNAND-3 is less than single DFF when input transition is less than 65ps.

Table 4.2: C2Q delay vs clock transition time and output load of PNAND-3, 5, 7 and 9 in 40nmGP technology. The simulation corner typical/typical, 0.9V VDD and $25°C$.

C2Q delay (ps)			CLK transition (ps)		
			1.86	44.05	192.15
Output load (fF)	PNAND-3	0.1	91.0	109.7	133
		7.3	113.2	132.6	154.4
		35.8	209.5	228.7	251.2
	PNAND-5	0.1	90.8	110.4	132.0
		7.3	112.9	132.9	156.3
		35.8	209.1	229.1	251.8
	PNAND-7	0.1	99.3	119.4	141.6
		7.3	120.2	140.4	165.4
		35.8	216.4	236.4	257.9
	PNAND-9	0.1	99.8	119.9	145.5
		7.3	122.9	142.9	168.5
		35.8	218.7	239.4	265.1

4.2 PNAND Performance on 28nm FD-SOI Technology

4.2.1 28nm FD-SOI Technology

In recent years, one major challenge of developing next generation of technology is controlling leakage current. Develop bulk transistor with high performance and low leakage for next generation become much more complex. Two possible candidates to replace bulk techniques are 3D architecture and FD-SOI. Fully Depleted Silicon On Insulator, or FD-SOI, is a planar process technology that delivers the benefits of reduced silicon geometries while keep the manufacturing process simple STMicroelec-

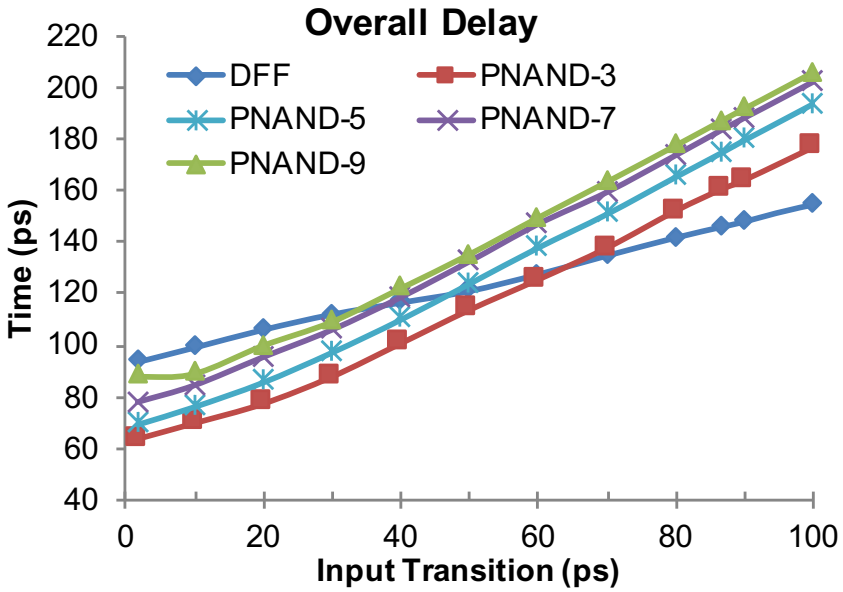


Figure 4.1: The total delay of DFF and PNANDs vs input transition

tronics (2018).

Fig. 4.2 shows sectional views of both bulk and an advanced FDSOI process called Ultra-Thin Body and Buried oxide Fully Depleted SOI or UTBB-FD-SOI. Comparing with conventional bulk process, the UTBB-FD-SOI has better electrostatic characteristics. The buried oxide layer reduces source and drain parasitic and refines current flow, which improves transistor behavior especially at low supply. The fully depleted channel is very thin and undoped. The variations caused by doping fluctuation is then eliminated. In other SOI technology such as Thick SOI or Extremely Thin SOI, the thickness of oxide box is around 150nm. In UTBB-FD-SOI, the box is as thin as 25nm, which enables usage of body bias techniques to dynamically control transistor threshold voltage.

In UTBB-FD-SOI (referred as FDSOI in the remaining of the chapter for short.), threshold voltage control can be delivered by multiple strategies. V_t modulation in FDSOI can be delivered by modifying gate oxide, well type, poly biasing and body biasing. Gate oxide can be chosen from core oxide for core circuit, IO oxide

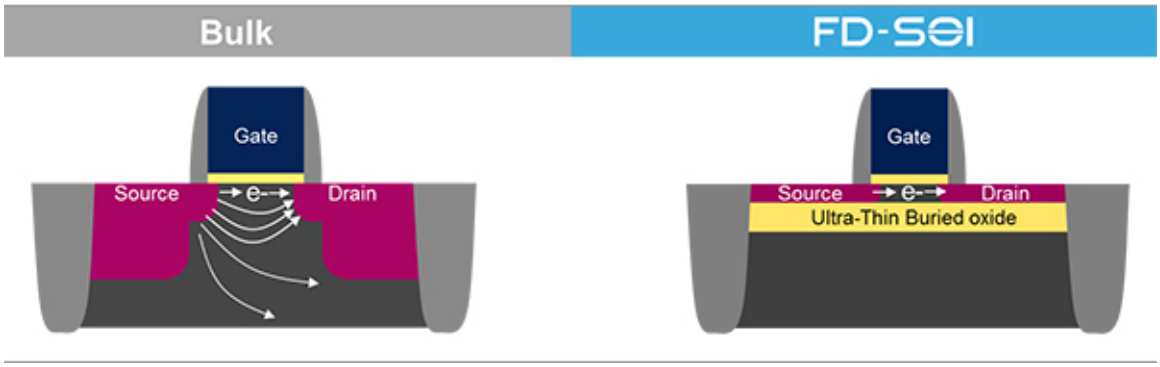


Figure 4.2: Bulk vs FD-SOI technology STMicroelectronics (2018).

for IO transistors and LP core oxide for high density SRAM. IO transistors have high nominal voltage and SRAM has low leakage and high density. The second modification method is poly biasing. Without change the active area, transistor gate length can be increased by adding certain polybiasing layer to layout. This is a convenient way of modifying standard cells without geometry modification. The range of effective length is from 24nm to 40nm. Poly biasing helps on reduce leakage with a cost of slowing down transistors, which provides extra trade-off decision between speed and leakage.

Different than bulk transistors that pFET has to be within n-type ground plane and nFET within p-type ground plane, the implant type of ground plane in FDSOI can be exchanged to realize different V_t flavors. Fig. 4.3 shows how to use ground plane implant for V_t adjustment. For RVT flavor, the ground plane implant is same as bulk process where p-type ground plane is under nFET and n-type ground plane is under pFET. This configuration is also called standard well. The ground planes are exchanged in LVT flavors where n-type ground plane is placed under nFET and p-type ground plane is placed under pFET. It is also called flip well. For a nFET with width of $0.21\mu m$ and length of $30nm$, the V_t is around $480mV$ for RVT and $400mV$ for LVT.

The threshold voltage can also be shifted by body bias. Biasing voltage can be

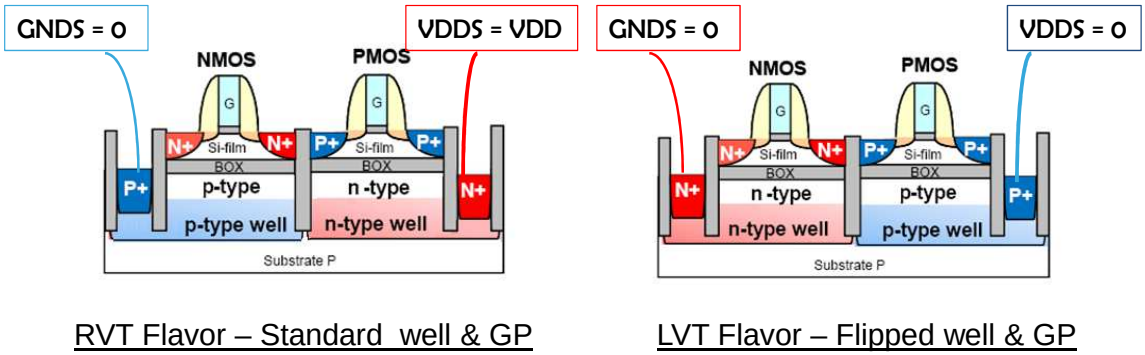


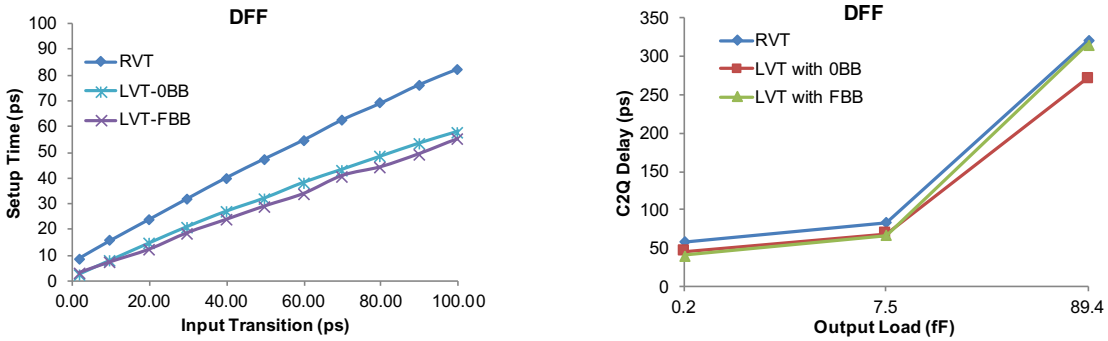
Figure 4.3: Use ground plane implant adjustment for RVT and LVT transistors in FDSOI technology.

applied on VDDS and GNDS terminals in Fig. 4.3. A positive (negative) body-to-source voltage can be applied on nFET (pFET) for forward body bias(FBB). FBB lowers V_t, makes transistors faster and leakier. A negative (positive) body-to-source voltage on nFET (pFET) sets reverse body bias (RBB). RBB increases V_t, make transistors slower and less leaky.

4.2.2 Performance and Design Challenges

Before discuss design trade-offs in 28nm PNAND cell design. The conventional DFF from standard cell library is evaluated for comparison. Fig. 4.4 shows setup time and C2Q delay of DFF in three design scenarios: design with RVT transistors, LVT transistor under zero body bias, and LVT transistors under 1.1V forward body bias. In these three scenarios, relation of threshold voltages are $V_{t\_RVT} > V_{t\_LVT-0BB} > V_{t\_LVT-FBB}$. Similar as in 40nm, the thresholds of computing input/CLK transition are 30% and 70% of supply voltage and the threshold of computing C2Q delay is 50% of supply voltage. In Fig. 4.4, DFF with RVT transistors has highest C2Q delay and setup time cross all data points when comparing with LVT flavor with and without body bias. The setup time of two LVT DFFs is very close. The C2Q delay of LVT with FBB is slightly faster than the one without body bias with lightest

output load. However, the difference diminishes when output load becomes larger. With output load as large as $98.4fF$, the C2Q delay of LVT DFF with FBB is 44.5ps higher than the one without body bias. The simulation results give the conclusion that performance boost on DFF by FBB is very limited. In contrast with the V_t comparison of single transistor, the overall delay under slow input transition and large load for LVT favor without body bias is even better than with FBB.



(a) Setup Time vs Input Transition

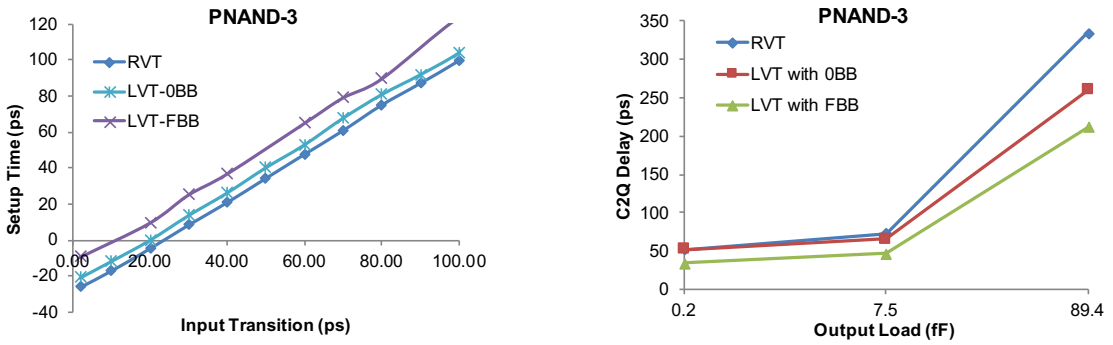
(b) C2Q Delay vs Capacitive Load

Figure 4.4: DFF performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and $25°C$.

Fig. 4.5, 4.6, 4.7 and 4.8 show PNAND-3,5,7,9 setup time and delay. All four circuits have similar trend on both setup time and C2Q delay. The setup time is negative when input transition is as fast as 2ps and increases with higher input transition time. As discussed previously, the setup time of PNAND cells is strongly related to input transition. Unlike DFF, the setup time of RVT flavor is the lowest and the one of LVT with forward body bias is the highest.

V_t value has a large impact on sense amplifier based design. Under same power supply, the benefit of small V_t is directly reflected on larger operation headroom $V_{gs} - V_t$. This effect can be observed on C2Q delay of all PNAND cell designs. The C2Q delay of PNANDs with RVT transistors is the slowest and the LVT PNANDs

with FBB is the fastest. This effect is more dominate with large output load. For PNAND-9, the difference between LVT with FBB and RVT is 60% when output load is $89.4fF$. The difference is 45% for PNAND-7, 44% for PNAND-5 and 37% for PNAND-3. All C2Q delay is simulated with input combination of $(\frac{n+1}{2} : \frac{n-1}{2})$. Comparing with DFF, C2Q delay of PNAND cells are generally faster, especially for LVT. With $7.5fF$ load, LVT PNAND-9 without body bias is 30% faster than LVT DFF. With 1.1V FBB, PNAND-9 is 54% faster.

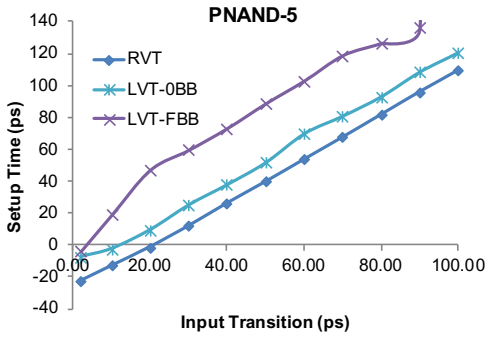


(a) Setup Time vs Input Transition

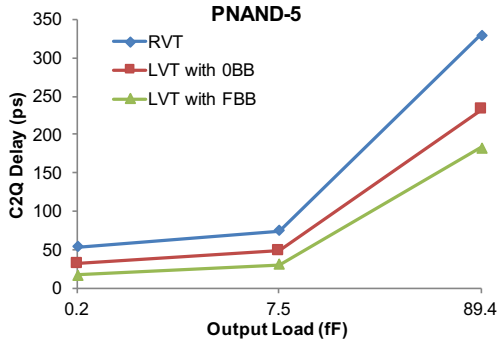
(b) C2Q Delay vs Capacitive Load

Figure 4.5: PNAND-3 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and $25°C$.

Fig. 4.9a shows the overall delay comparison between DFF and PNAND-3 with three V_t flavor. For RVT, the overall delay of PNAND-3 is lower than DFF when input transition is lower than 80ps. DFF on LVT is much faster than RVT. The overall delay of DFF would be faster if input transition time is higher than 30 to 35ps. Fig. 4.9b shows energy delay product (EDP) of hybrid and CMOS equivalent circuit for threshold function y=[22111;4]. The EDP of two circuits are close when input switching activity is low (10%). When input switching activity is as high as 30%, the EDP of hybrid is 25.6% less than its CMOS counterpart. This is due to the fact that the energy consumption of PNAND doesn't change much with respect

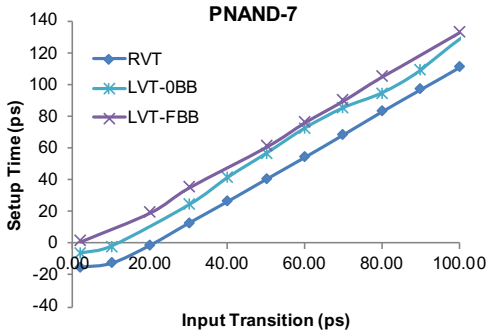


(a) Setup Time vs Input Transition

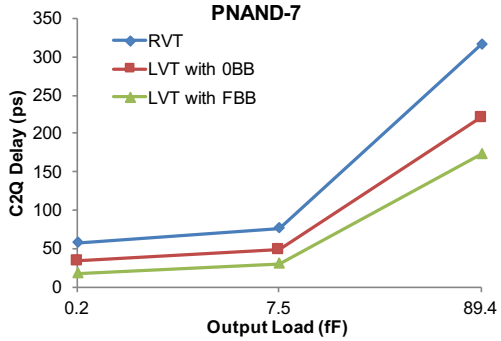


(b) C2Q Delay vs Capacitive Load

Figure 4.6: PNAND-5 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C.



(a) Setup Time vs Input Transition

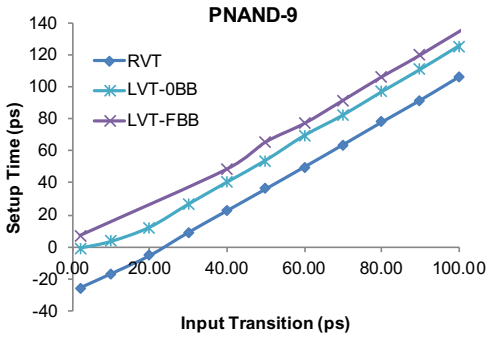


(b) C2Q Delay vs Capacitive Load

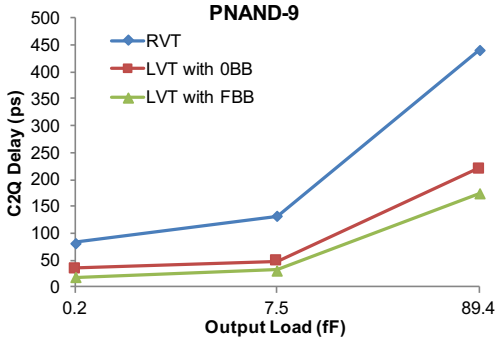
Figure 4.7: PNAND-7 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C.

to input switching activity while energy consumption of DFF increases significantly when input switches more often.

Leakage power of PNAND cells and DFF on three scenarios are shown in Fig. 4.10. Reducing V_t speeds up transistors with the cost of increasing leakage. Leakage of LVT cells with 1.1V FBB is one order higher than LVT cell without body bias and more than two orders higher than RVT cells. All PNAND cells leaks more than DFF. This is



(a) Setup Time vs Input Transition



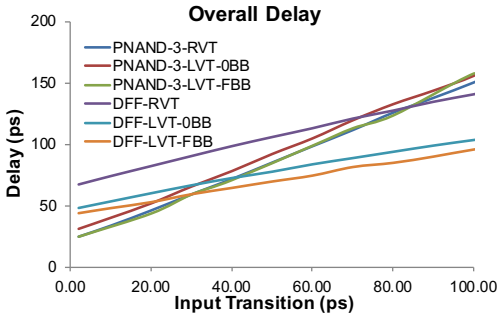
(b) C2Q Delay vs Capacitive Load

Figure 4.8: PNAND-9 performance comparison in three scenarios : design on RVT transistors, design on LVT transistor with zero body bias and design on LVT transistors with 1.1V forward body bias. CLK transition is set to 33ps. The simulation corner typical/typical, 0.9V VDD and 25°C.

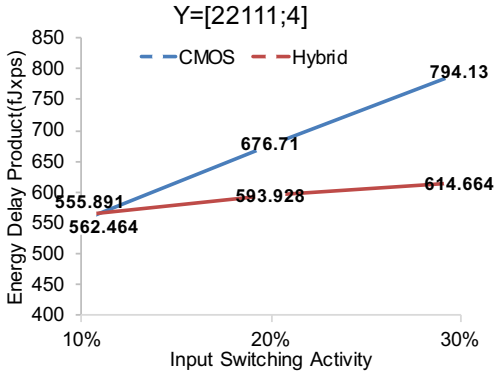
understandable that PNAND cell computes larger function than DFF. The transistor sizes in sense amplifier are also larger. PNAND cells with more inputs leaks more because of the same reason. Sense amplifier in large PNANDs are usually larger than small PNANDs. Combining setup time and C2Q delay, PNAND cells usually have better overall delay when input transition is fast. In hybridization, input transition is set to lower than 40ps in order to show delay benefit.

4.3 65nm Low Voltage Operation

The major drawbacks shared by nearly all threshold logic circuit architectures is their sensitivity to process variations and unsuitability for low voltage operation. The transistor overhead voltage drops when lowing supply voltage. For differential threshold gates like PNAND, their operation robustness would significantly degrade at low voltages due to process variation. In this section, we present a solution to these problems, and show that the potential advantages of TLGs, namely, smaller, faster, lower power and robust circuits are possible at low voltage.



(a) The overall delay comparison between DFF and PNAND-3 in 28nm FDSOI



(b) EDP vs input switching activity

Figure 4.9: Overall performance comparison between CMOS and threshold cell. a) Overall delay comparison between DFF and PNAND-3. b) Energy delay product comparison between a hybrid circuit and its CMOS equivalence for 5-input function y=[22111;4]. The hybrid circuits consists of inverters and a PNAND-7 while the CMOS equivalent circuit consists of standard logic gates and a DFF.

4.3.1 Threshold Logic Gate Architecture

Fig. 4.11 shows the schematic of a TLG. The circuit structure without the resistors is referred to as TLL. Details of this circuit and comparison with other implementations based on the same principle can be found in Samuel *et al.* (2010). Similar as PNAND, TLL also consists of 5 components: (1) a differential sense amplifier, which consists of two cross coupled NAND gates, (2) a SR latch, (3) two discharge devices, (4) left (**LIN**) and right (**RIN**) input networks, and (5) a network of resistors. TLL-n refers to a TLL with n inputs in the LIN and the RIN. Clock input signal directly drive source terminals of input network.

Functionally, TLL can also be viewed as a *complex, multi-input* edge-triggered DFF, identical to PNAND. In general, a TLL has a lower setup time than a DFF while its *clock-to-Q* delay is comparable. A TLL also presents a lower input capacitance but higher clock capacitance than a DFF. In TLL, clock pin capacitance changes

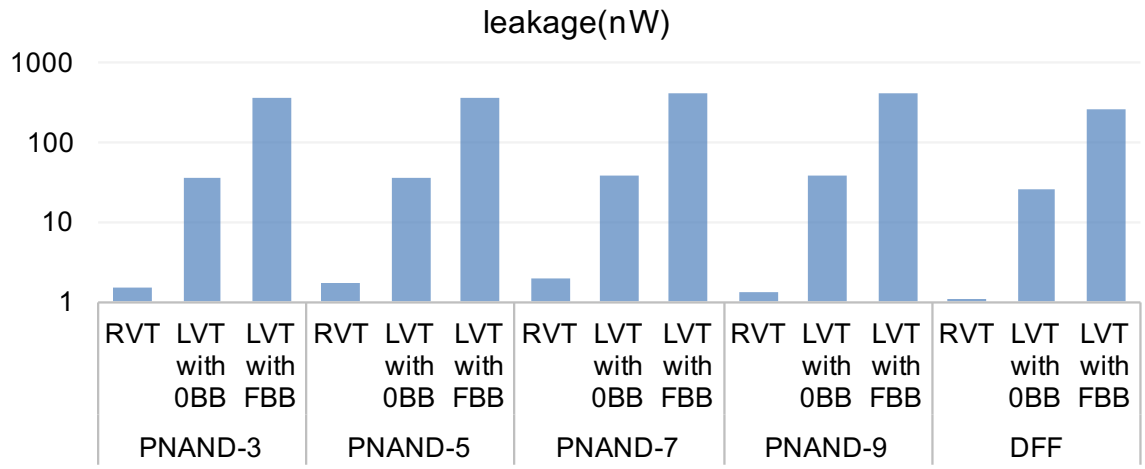


Figure 4.10: Average leakage power for PNAND cells and DFF with same drive strength.

when number of ON transistors in input network is different. Therefore, CSA is required for threshold function mapping. CSA results in a constant load on the clock input regardless of the input vector which is very important for construct clock tree to deliver clock signal. TLL can also be implemented in hybridization, which contribute to reducing the area and power when TLL are judiciously incorporated in logic networks.

4.3.2 Low Voltage Operation

Standard cell layouts of TLL circuits based on Fig. 4.11 (without the resistor network) were carried out for $n = 3, 5, 7, 9$, using a commercial 65nm LP process, with nominal $V_{dd} = 1.2V$. Optimal sizing of the input network, the sense-amp and the output latch were performed to minimize delay and minimize the circuit functional failures in the presence of process variations based on 100,000 Monte Carlo simulations. Unfortunately, failures begin to manifest as soon as the supply voltage is reduced below $1.08V$. This characteristic is shared by all TLG architectures including PNAND.

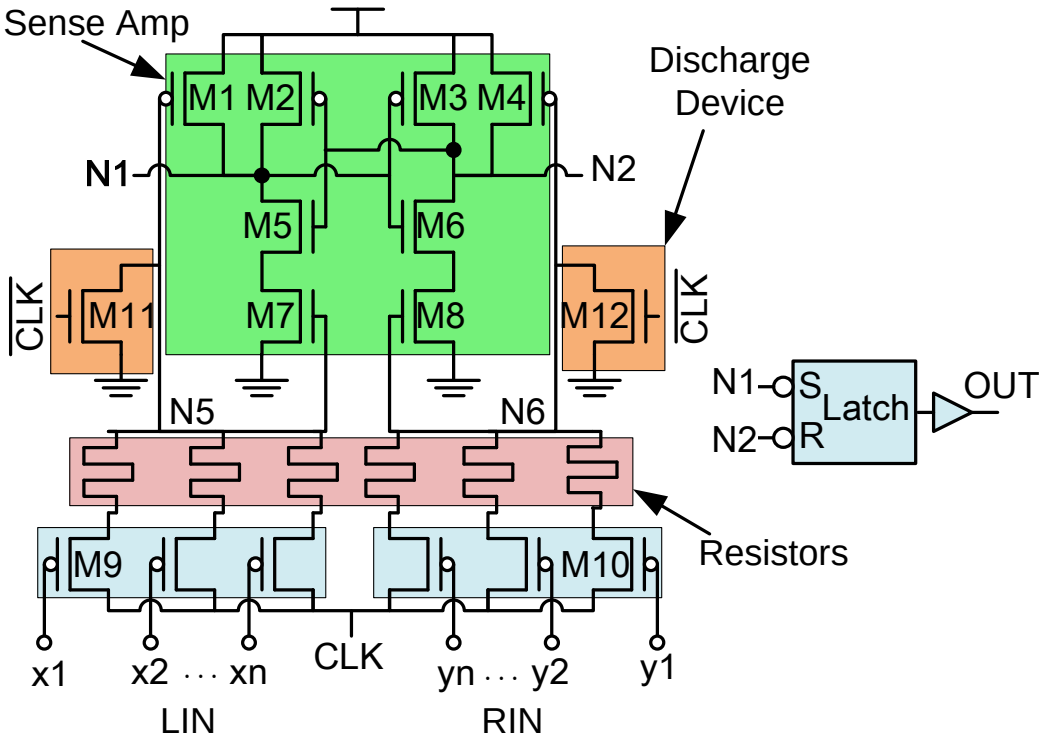


Figure 4.11: Schematic of a TLL circuit with resistor network.

Table 4.3: TLL failures in 100K MC simulations without resistor network

VDD	Cases			
	2:1	3:2	4:3	5:4
0.7	33	713	3232	8478
0.65	253	2185	6697	13056
0.6	1500	5842	12153	19590

In this section, we describe the necessity of the resistor network and how it helps low voltage operation of the TLL. Table 4.3 shows the results of 100,000 Monte Carlo simulations of TLL-7 (schematic only) at low voltages at nominal corner, 25 °C, with minimum size input transistors and the output buffer sized to match the minimum drive strength of standard DFF in CMOS library. Table 4.3 shows significant functional failures at low voltages. The case K:$K-1$ represents the situation where there

are K active transistors in the LIN and $K-1$ active transistors in the RIN, and vice-versa. Note $K=5$ requires a TLL-9. The higher the value of K, the greater the number of input transistors and the complexity of the functions that can be implemented with a TLL. The number of functions with $K \leq 3$ is too small and provide no substantial advantage over conventional logic implementations. It is only with $K \geq 4$, do we see a significant compaction of logic and reduction in power with the use of TLLs.

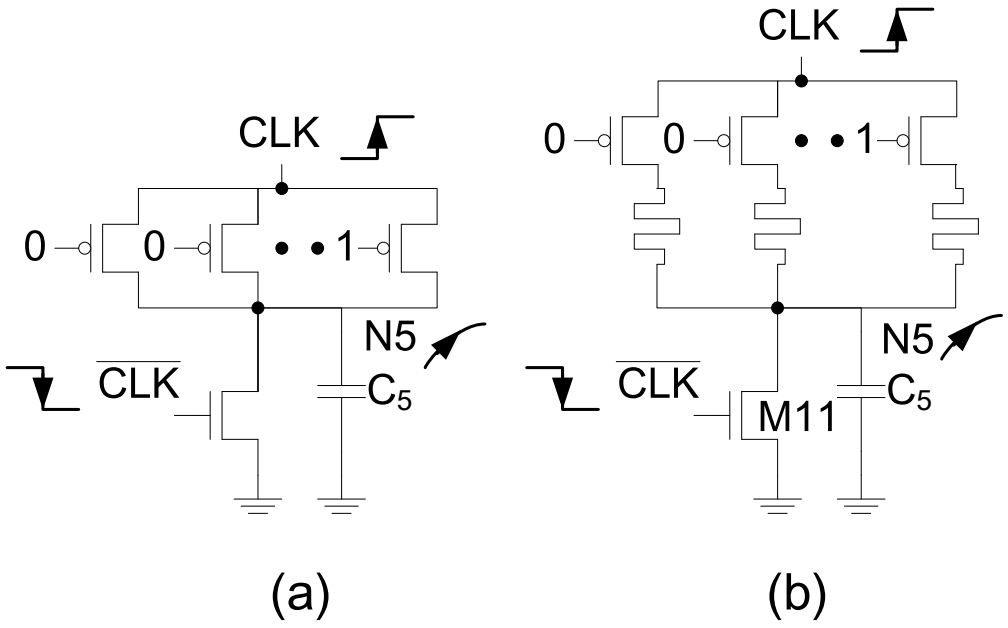


Figure 4.12: Simplified input network of TLL

We now examine how the resistor network shown in Fig. 4.11 might help in improving the robustness of the TLL at low voltages. Fig. 4.12 shows a simplified version of one of the input networks, with and without the resistor network. Consider Fig. 4.12(a), which depicts the evaluation of the TLL. Let t_5 (t_6) denote the time when $N5$ ($N6$) reach the threshold voltage of M_7 (M_8), and t_{sen} denote the minimum time difference between t_5 and t_6 for the sense amplifier to correctly determine the

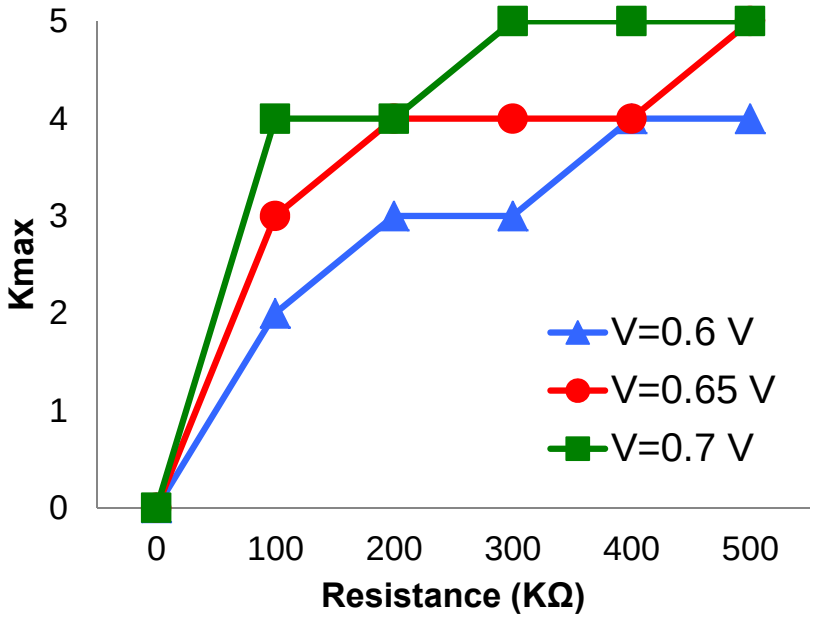


Figure 4.13: TLL functionality vs R

output. Thus the TLL correctly computes the function if

$$\Delta t = \begin{cases} t_6 - t_5 \geq t_{sen} & \text{output} = 1, \\ t_5 - t_6 \geq t_{sen} & \text{output} = 0. \end{cases} \tag{4.1}$$

Let C_5 denote the total capacitance of node $N5$, which includes the gate capacitances of M_1, M_7, the drain capacitances of M_{11} and those of the transistors in the LIN. $N5$ and $N6$ are initially discharged to 0. When the clock rises from $0 \rightarrow 1$, $N5$ and $N6$ rise to V_{dd}. The active pFETs in the LIN and RIN immediately enter the saturation region, where current is $I_s = \mu_p C_{ox}(W/L)(V_{dd} - \mid V_{tp} \mid)^2$. Assuming that there are K and $K-1$ active pFETS in the LIN and RIN, $C_5 = C_6 = C$, and $V_{t7} = V_{t8} = V_{tn}$, then t_5, t_6 and the corresponding Δt are approximately given by

$$\begin{aligned} t_5 \approx \frac{C_5 V_{t7}}{K I_s}, \quad t_6 \approx \frac{C_6 V_{t8}}{(K-1) I_s} \\ \Delta t = t_6 - t_5 \approx \frac{C V_{tn}}{K(K-1) I_s} \end{aligned} \tag{4.2}$$

Note that the above approximate relations explain the trend in Table 4.3 for a fixed V_{dd}, and as K varies. Similarly for a fixed K and as V_{dd} decreases, failures increase because t_{sen} increases as a result of reduced discharge currents through the sense amplifier.

Fig. 4.12(b) shows a resistance R_H in series with each pFET. If R_H is relatively very large, when the clock transitions from $0 \rightarrow 1$, most of the voltage drop will be across the resistor, and the small V_{ds} across the pFET will force it to operate in the linear region. In this case, pFET is very close to a linear resistor, with current $I \propto (V_{dd} - \mid V_{tp} \mid)V_{ds}$. If R_H is sufficiently large, then the resistance of the pFET, which is $R_{lin} \approx 1/(\mu_p C_{ox}(W/L)(V_{dd} - \mid V_{tp} \mid)$, is negligible relative to R_H. In this case t_5, t_6 and Δt are approximated by

$$\begin{aligned} t_5 = -\frac{R_H C_5}{K} \ln\left(1 - \frac{V_{t7}}{V_{dd}}\right), \quad t_6 = -\frac{R_H C_6}{K-1} \ln\left(1 - \frac{V_{t8}}{V_{dd}}\right), \\ \Delta t' = t_6 - t_5 \approx -\frac{R_H C}{K(K-1)} \ln\left(1 - \frac{V_{tn}}{V_{dd}}\right) \end{aligned} \tag{4.3}$$

In a 65nm LP process, $I_s \approx 4.11\mu A$, for a minimum size pFET, when $V_{dd} = 0.6V$. And typically, $V_{tn} = 0.4233V$, $V_{tp} = -0.43V$. For $R_H = 500K\Omega$, $\Delta t' \approx 6.12\Delta t$, showing a substantial improvement in the robustness of a TLL at low voltages. To verify this, 100,000 Monte Carlo simulations were performed on TLL-7, considering the global variations and local mismatch in the CMOS devices, for various values of R_H. For each value of R_H, the maximum value of K in K:K-1 case that satisfied the robustness criterion of 99.99% successes was computed. The results of the simulations are shown in Fig. 4.13.

The plot shows a significant improvement in the robustness of a TLL with the addition of resistors in series with the input network. At $V_{dd} = 0.6V$, with minimum $400K\Omega$ resistor, all threshold functions with a 4:3 combination of active devices in

the LIN and RIN can be implemented. Restricted to TLL-7, this constitutes a total of 29 functions. The set of functions implementable is actually larger because the subset of TLL-9, TLL-11 and TLL-13 functions with worst-case 4 : 3 combinations can also be implemented. To implement all TLL-9 functions (additional 42 functions) the V_{dd} has to be increased to 0.65, and R_H to $500K\Omega$. The important take-away from this discussion is that R_H can be reduced and compensated by increasing the supply voltage.

In general, lowering the supply voltage will significantly increase the delay. This is true for both conventional CMOS logic and TLLs. However, reducing the supply voltage of a TLL requires larger resistances in the input networks. To realize R_H in the range of a few $100K\Omega$, as required by TLLs, in a CMOS process is impractical. Fortunately, emerging memory technologies offer a solution. In the following section, we propose the use of oxide based resistive random access memory (RRAM) devices as resistors for the TLL. As demonstrated in Fang *et al.* (2011), excellent and stable resistance values were achieved for $R_H = 500K\Omega$. At this value, the minimum V_{dd} that met the robustness criteria for TLLs was 0.6V. This is the reason for using $V_{dd} = 0.6V$ as the minimum supply voltage for the TLLs.

4.3.3 Resistor Network

The oxide-based resistive random access memory (RRAM) technology Wong *et al.* (2012) is an emerging candidate for next-generation non-volatile memory (NVM). Here we use RRAM as a CMOS compatible nano-scale resistor. For our application, its resistance need only be set once. Hence, technically speaking we are using an RRAM as a RROM. However to avoid confusion we will continue to refer to it as an RRAM. Used in this way, an RRAM has excellent scalability (<10 nm) and good retention (>10 years). Other NVM candidates such as spin-torque-transfer magnetic

random access memory (STT-MRAM) Zhu (2008) and phase change memory (PCM) Wong *et al.* (2010) can also be used. However, they have some undesirable features: the resistance of STT-MRAM is relatively low (a few kilo Ohm), and PCM has a well-known time-dependent resistance drift (even without voltage stress) problem.

One of the concerns about RRAM is the resistance variation from device to device. This is largely related to the manufacturing process control. Materials engineering such as multi-layer oxide design can restrict the resistance variation to <10% standard deviation around the medium off-state resistance of $500K\Omega$ Fang *et al.* (2011). It can be expected that as the RRAM technology matures, the manufacturing yield and process variation will be further improved. The second concern is the time-dependent resistance drift under voltage stress. There is a well-known exponential voltage-time relationship in the switching dynamics of RRAM Yu *et al.* (2011): the switching time exponentially depends on the applied voltage. To ensure a lifetime of at least 10 years at low voltage stress, we employed an RRAM compact device model Guan *et al.* (2012) to study the dynamics of the resistance drift. Extrapolating from the experimental data shown in Fang *et al.* (2011), and using $1/E$ model, we can see from Fig. 4.14 that the lifetime of RRAM device is 10 years under continuous voltage stress of 0.51V. The voltage drop across RRAM in a TLL is much less than 0.51V, ensuring significantly longer lifetime.

Fig. 4.15 shows the yield of a RRAM based TLL-7 circuit in the presence of process variations. The yield calculation is based on 100,000 Monte Carlo simulations, which includes variations in both transistors and the RRAMs. The mean RRAM R_H value is $500k\Omega$ and the simulations were carried out for σ/μ = 1%, 5% and 10%. The simulations indicate that for high circuit yields with $K = 4$ the required σ/μ should be no more than 5% which is expected in near future.

The RRAM devices need to be initially programmed to their high resistance state

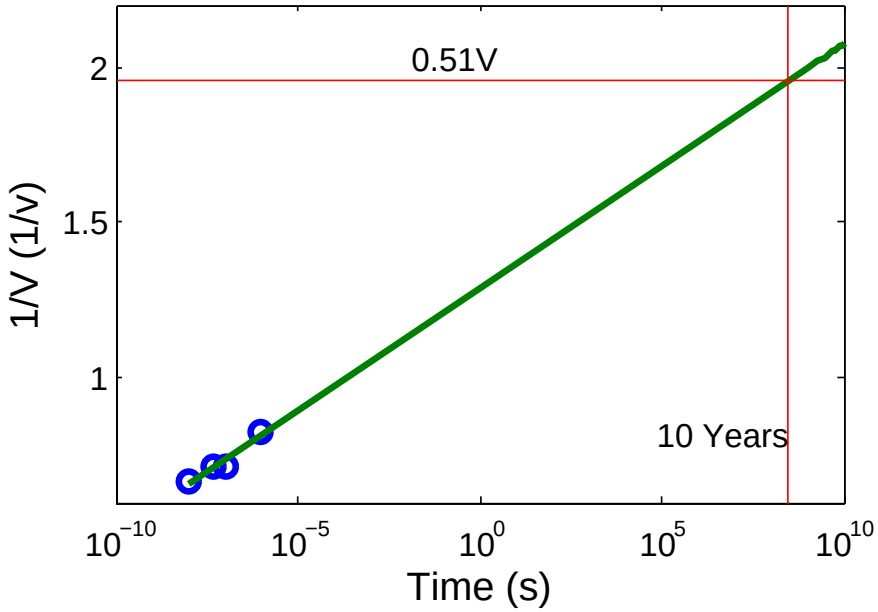


Figure 4.14: RRAM lifetime vs stress voltage

(HRS) only once after fabrication, and the programming circuitry for doing this has to be part of the TLL. Note that RRAM devices are in the top metal layer and do not contribute to the silicon area. The schematic and its 3D structure are shown in Fig. 4.16. Two selection elements Deng *et al.* (2013) are connected on both bottom and top electrodes. In Fig. 4.16 (a), the top electrode is connected to one selector and node N5 of the TLL, and the bottom electrode is connected to another selector and pFETs in the input network. The CLK is first set to 1 and a large positive forming voltage pulse is applied, to set the RRAM device to a low resistance state (LRS). Following this, the CLK is set to 0 and a large negative reset pulse is applied which resets the RRAM device to the high resistance state (HRS).

Fig. 4.16 (b) shows the 3-D arrangement of the different elements. Of the 3 pairs of pillars, the first and the last pair serve as selectors while the middle pair are the RRAM resistors. Although they all have the same structure, the resistance of selectors have high non-linear relation with voltage drop, which ensures that the programming does not affect the normal operation.

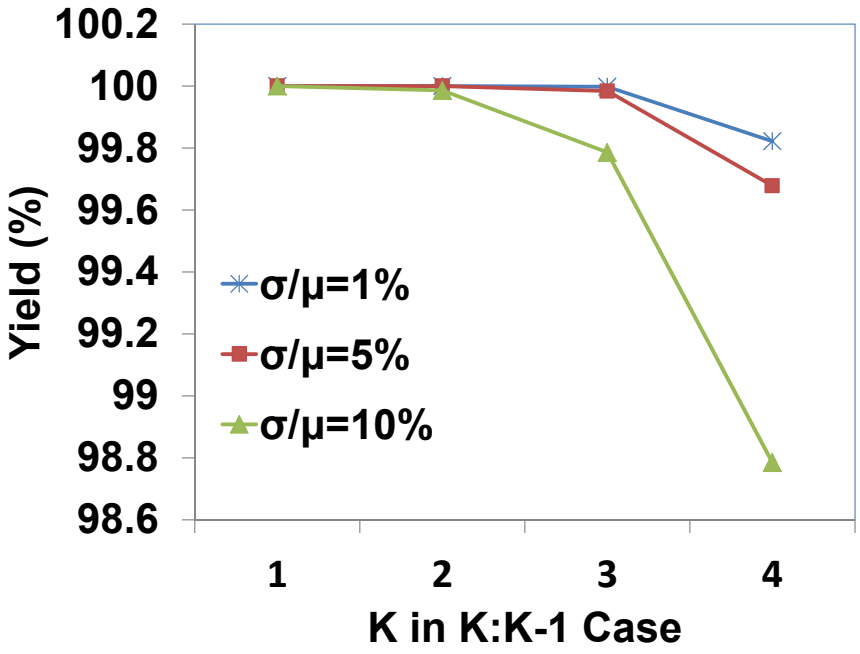


Figure 4.15: TLL-7 yield with RRAM

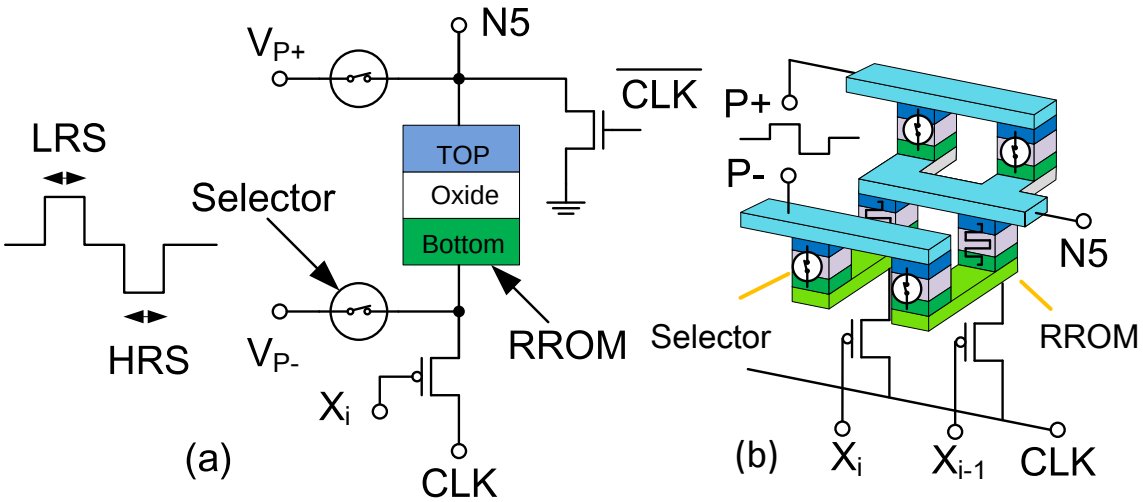


Figure 4.16: (a) Programming RRAM (b) 3-D structure

4.3.4 Cell Comparison of Energy, Delay and Area

Circuits implemented using conventional CMOS logic gates only will be denoted as CCL where those employing TLLs will be referred to as hybrid circuits (they may include inverters). Table 4.4 compares CCL and hybrid implementations of each threshold function.

For CCL circuits, each of the functions was synthesized using a commercial 65nm standard cell library. Since TLL is equivalent to an edge-triggered flipflop, each of CCL implementations contain a DFF at the output of the function. The synthesized implementations were simulated using HSPICE for 100 random vectors having 30% switching activity for each primary input. The simulation corner was nominal, 25 °C, and $V_{dd} = 0.6V$. The average energy-delay product (EDP) of each circuit is shown in the table. The delay of each circuit was determined by applying the critical input vector and reducing the clock period until the function ceases to simulate correctly in SPICE. The TLL configuration contains RRAM with $500K\Omega$ resistances.

The column labeled Ratio denotes the ratio of the energy-delay product (EDP) of CCL to hybrid. Two important things to note are : (1) the energy-delay product of hybrid cells is almost independent of the function, switching activity and input vectors. Therefore the standard deviation of EDP is much less for hybrid circuits than for CCL; (2) except for four functions (shown in bold) where CMOS circuits have a slightly better energy-delay product, hybrid circuits show a consistent and significant improvement in EDP. These four functions are mostly small AND/OR functions. However even for these four functions, as the switching activity increases, the hybrid implementations of these functions start to show improvement over corresponding CCL counterparts. Significant reduction in silicon area was also achieved. Note that the hybrid implementations of all the functions are the worst case implementations

as they employ TLL-7 for all functions. While many of which can actually be implemented with smaller TLL (TLL-5 and TLL-3) which would further reduce area and EDP. The total area each of hybrid function using TLL-7 and 7 inverters was $29.12\mu m^2$. However it should be noted that every function doesn't need all 7 inverters especially if inverter inputs are driven by constants. Similarly multiple inverters driven by the same signal can be merged. Comparing to the average area of CMOS circuit which was $39.77\mu m^2$, it is a 26.7% reduction. When TLLs replace threshold logic cones driving DFFs, the area savings will actually be greater because the input capacitance that TLL cell exhibits compared to the CCL counterpart is significantly reduced (approximately 30% for larger circuits). Synthesis tools can take advantage of this to reduce the size of logic that feeds TLLs in a ASIC design, providing additional reductions in area and power, without any performance degradation.

Table 4.4: Comparison of functions

Function	**Boolean Expression**	**Energy × Delay** ($fJ \cdot ns$)		**Ratio**	**CCL Area** (μm^2)
		CCL	**TLL**		
1111111;4	abcd + abce + ⋯ (35 terms)	102.2	12.45	8.2	138.4
111111;4	abcd + abce + ⋯ (15 terms)	90.4	15.45	5.9	119.2
211111;4	abc + abd + ⋯ (15 terms)	71.1	12.37	5.7	95.2
21111;4	abc + abd + ⋯ (10 terms)	51.8	12.41	4.2	72
111111;3	abc + abd + ⋯ (20 terms)	48.8	13.56	3.6	79.2
11111;3	abc + abd + ⋯ (10 terms)	43.5	13.54	3.2	53
11111;4	abcd + abce + abde + acde + bcde	37.1	14.8	2.5	36.4
22111;4	ab + (a+b) (cd +de + ce)	29.4	12.33	2.4	20.4
11111;2	ab + bc + ⋯ (10 terms)	31	13.78	2.2	48
31111;4	a (b+c+d+e) + bcde	25.9	12.36	2.1	37
2211;4	ab+ acd +bcd	29.5	15.47	1.9	31.8
3111;4	a(b+c+d)	23.4	12.27	1.9	27.8

Continued on next page

Table 4.4 – continued from previous page

Function	Boolean Expression	Energy × Delay ($fJ \cdot ns$)		Ratio	CCL Area (μm^2)
		CCL	TLL		
21111;3	a(b+c+d+e) + bcd + bce + bde + cde	23.7	13.48	1.8	35
1111;2	ab + ac + ad + bc + bd + cd	23.4	13.85	1.7	34.8
2111;4	a(bc+bd+cd)	20.5	12.11	1.7	23.8
2211;3	ab + (a+b) (c+d)	21.4	13.48	1.6	26.8
1111;3	abc + abd + acd + bcd	23.9	16.79	1.4	34.8
2111;2	a+bc+cd+bd	19.6	13.79	1.4	26.8
111;2	ab+bc+ac	24.1	17.48	1.4	23.6
2111;3	a(b+c+d)+bcd	23.5	16.99	1.4	32.8
211;2	a+bc	18	13.97	1.3	19.6
211;3	a(b+c)	16.5	13.36	1.2	13.6
111;1	a+b+c	15.3	13.63	1.1	21.6
1111;1	a+b+c+d	14.5	13.61	1.1	21.8
11;2	ab	14.3	13.64	1	13.4

Continued on next page

Table 4.4 – continued from previous page

Function	Boolean Expression	Energy × Delay ($fJ \cdot ns$)		Ratio	CCL Area (μm^2)
		CCL	TLL		
1111;4	abcd	13.4	14.31	0.9	18.8
11;1	a+b	12.8	13.76	0.9	13.4
3111;3	a+bcd	15.4	16.92	0.9	17.8
111;3	abc	11.9	16.02	0.7	16.6
MEAN		30.9	14.1	2.3	39.8
STDEV		22.6	1.6	1.7	31.7

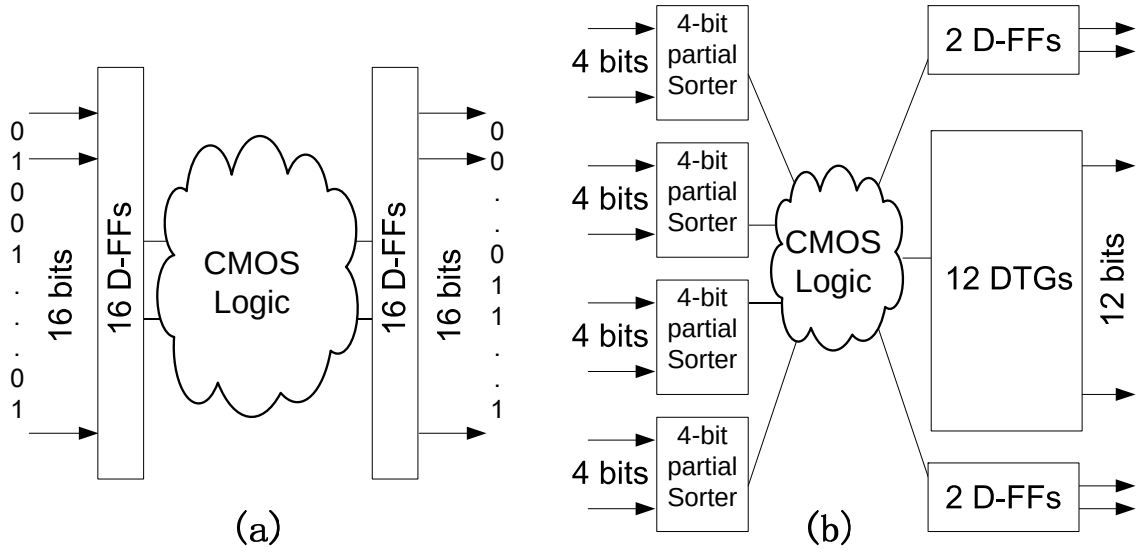


Figure 4.17: A 16 inputs 1-bit sorter

4.3.5 Circuit Implementations and Comparison

In this section, we will show how circuit block implementations can benefit by using RRAM based TLL cell library. The library includes TLL-3, TLL-5 and TLL-7 cells. Two different implementations of each circuit were created using 65nm commercial library : one using CCL and the other using TLLs. The maximum operating frequency of each circuit was determined using SPICE simulation. To estimate the energy consumption of the clock tree, optimally sized clock buffers are included for both CCL and hybrid circuits. CCL implementations were created using Cadence RTL Compiler while the network of TLL cells were interconnected manually.

16-input Single bit sorter

Fig. 4.17 shows the structure of a 16-input single bit sorter. The sorter has 16 1-bit inputs and 16 1-bit outputs. The sorter is especially useful in parity and instruction control circuits and all symmetric functions.

Both circuits are two stage pipelines. For the hybrid circuit, the first stage is

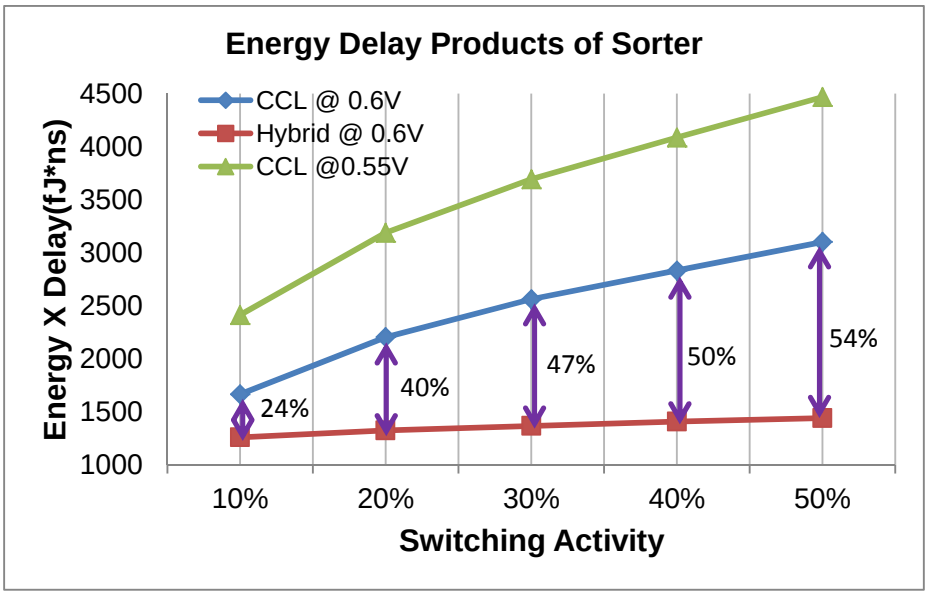


Figure 4.18: Energy × Delay(fJ·ns) of Sorter

implemented by four 4-input sorters, each of which is implemented by four TLL-7 gates. The second stage is implemented by 12 TLLs, 4 DFFs and CMOS logic cells, by suitably replacing the remaining CMOS logic and flipflops with TLLs. The peak frequency of CCL sorter at 0.6V is 125 MHz while the hybrid is 167 MHz. Fig. 4.18 shows the energy-delay product of both circuits over different input switching activities. As the switching activity increases, CCL power increases because more nets toggle and there is greater glitches. On the other hand, TLLs have constant energy irrespective of the input switching activity. Hence the hybrid circuit shows a larger improvement in EDP especially for high switching activity applications. Typical switching activities are between 20 and 30%. For these, the hybrid design shows approximately 40% improvement in the EDP. Finally, the hybrid sorter ($943\mu m^2$) was 18% smaller than the CCL ($1159\mu m^2$).

Fig. 4.18 shows the energy-delay product of CCL is even worse at lower voltages. Therefore even though voltage of TLLs cannot be scaled down as much as CCL circuits, the EDP of TLL is still much lower.

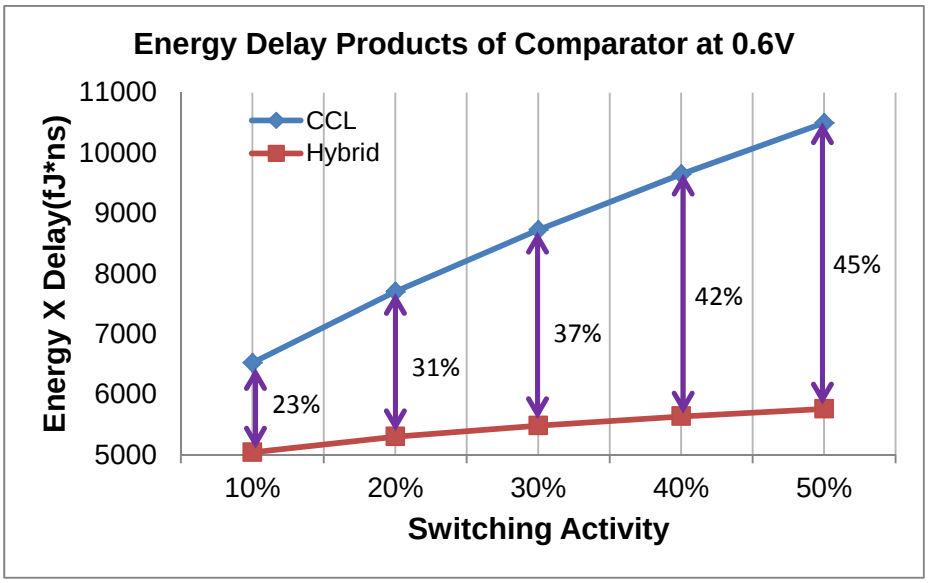


Figure 4.19: Energy × Delay(fJ·ns) of Comparator

Table 4.5: Area Comparison(μm^2)

	Hybrid			Pure CMOS	Imp(%)
	TLL	CMOS	Total	Total	
Sorter	503	440	943	1159	19
Comparator	3501	2114	5615	6822	18

128-bit Comparator

The second circuit implemented was a 128-bit comparator designed as a 4-stage pipeline. The hybrid comparator consists of a hierarchy of several 8-bit comparators. The peak frequency of CCL comparator is 222 MHz while that of the hybrid is 250 MHz. Fig. 4.19 shows the energy-delay product of these two circuits as a function of switching activity. The hybrid comparator required 19% less area than the CCL (5615μm^2 vs 6822μm^2), and a 31-37% improvement in EDP over CCL for switching activities of 20% to 30%. Table 4.3.5 shows the area comparison for both sorter and comparator. Hybrid sorter is 19% smaller than CCL implementation and hybrid comparator is 18% also smaller than CCL.

Chapter 5

ENERGY-EFFICIENT NON-VOLATILE LOGIC

Systems powered by harvested energy must consume very low power and withstand frequent interruptions in power. Non-volatile logic (NVL) addresses the latter by saving the system state in flipflops enhanced with STT-MTJs as the non-volatile storage devices. Manufacturing variations in the STT-MTJs and in CMOS transistors significantly reduce yield, leading to overdesign and high energy consumption. In this chapter, A detailed analysis of the design tradeoffs in the driver circuitry for performing backup and restore, and a novel method to design the energy optimal driver for a given yield is presented.

5.1 NVL System Powered by Harvested Energy

Microelectronic circuits that obtain their energy from ambient energy sources (AES) such as solar, piezoelectric, vibration, airflow, and thermoelectric Priya and Inman (2008) are expected to become essential for the burgeoning field of the Internet of Things (IoT). Although there are substantial differences among them in power density (ranging from tens of μW to tens of mW), as well as variations in the delivered energy over time, it is the intermittent nature of the delivered energy by AES that poses the most difficult challenge for microelectronic systems as they are generally architected for continuous operation. Hence quickly predicting an impending power disruption, and saving the state in some form of non-volatile storage is critical for all but the simplest devices. The emergence of CMOS-compatible non-volatile memory (NVM) technologies (e.g. MRAM, RRAM, PCRAM, CBRAM, FeRAM, STT-RAM, etc.) over the past decade has opened the way for new circuit architectures for near

instantaneous and energy-efficient backup and recovery.

NVM for backup and restoration during a power disruption can be implemented in one of two ways. One option is to have a NVM array (NVMA) that is separate from the local (volatile) registers where the intermediate computation results are stored Khanna *et al.* (2014). Before the power failure, the data in all the registers would be saved serially in the NVMA and later serially restored. The other option (e.g. Koga *et al.* (2010); Wang *et al.* (2012); Ryu *et al.* (2012); Kwon *et al.* (2014); Cai *et al.* (2015); Mahalanabis *et al.* (2015); Kang *et al.* (2016); Bishnoi *et al.* (2016b, 2017)) is to have each register be a non-volatile flipflop (NVFF), which operates like a regular flipflop in normal mode, but has the added capability of storing its state in a local non-volatile device before power failure.

The non-volatile devices that are most often employed in the various NVFF designs have a common characteristic, namely, that they require a *critical current* to be delivered for some minimum duration in order to switch their state. Process variations, including both within die and die-to-die variations pose a major challenge in circuits with NV devices. These, along with variations in the CMOS circuits that drive the NV device, result in statistical variations in the actual current being delivered. Designing with such variations in mind requires quantifying the ensuing trade-offs between reliability (probability of successful backup), area of the driver circuits, backup and restoration time, and power consumption. Optimal driver design of a NVFF considering process variations, and examination of the trade-offs has not been well explored in the existing literature. Ignoring variations in the transistors and MTJ devices will result in poor functional yield. However, the traditional worst-case-corners approach results in significant wastage of energy during backup.

A common required component for storing and restoring data into and from the NV devices is the *backup driver* (Cai *et al.* (2015); Ryu *et al.* (2012)). Fig. (5.1a) shows the key components of such a circuit, without any of the control logic. It consists of two inverters in series with an STT-MTJ device. A brief, high-level description of the behavior of an STT-MTJ, sufficient to explain the design and optimization of the backup driver circuit, follows.

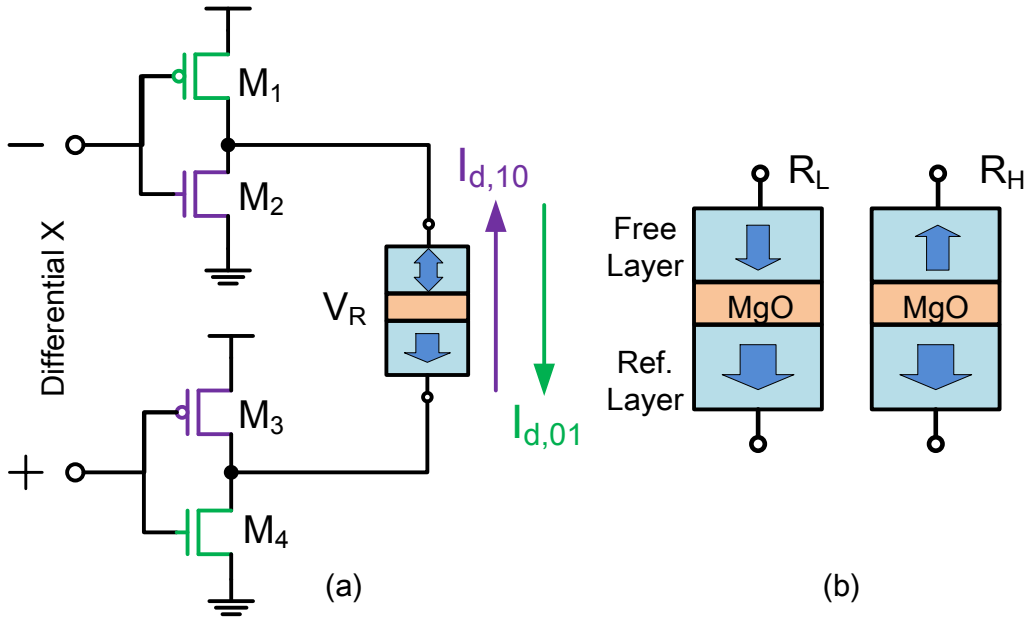


Figure 5.1: (a) Simplified driver circuit providing bidirectional current to switch STT-MTJ cell; (b)The structure of STT-MTJ

5.2.1 The STT-MTJ Cell

An STT-MTJ cell consists of two ferromagnetic layers separated by an oxide insulation layer (usually M_gO) (see Fig. (5.1b)). The magnetization of the reference layer is fixed, whereas that of the free layer can be switched. When the spin orientations in the two layers are parallel (anti-parallel), the STT-MTJ cell has a low

(high) resistance, denoted by R_L (R_H), which represent the logic 0 and 1, respectively. $TMR = (R_H - R_L)/R_L$ represents the relative separation between the two resistance values, with typical values between 50% to 200%, and can be as high as 600% Zhang *et al.* (2012). It is assumed that R_H and R_L are constants, independent of the voltage across the device, and that the change in resistance between R_L and R_H is abrupt. The *switching time* τ is the time at which the abrupt change takes place. Due to thermal fluctuations, the STT-MTJ switching is a stochastic (Munira *et al.* (2012); Bishnoi *et al.* (2016a); Wang *et al.* (2014)). However, deterministic switching is assumed when the device current I_d exceeds a critical value I_c.

Applying $X = 1$ in the backup driver will cause a current $I_{d,01}$ to flow through M_1, the STT-MTJ and M_4. This must exceed a critical current $I_{c,01}$ for a duration of τ_{01} in order for the STT-MTJ to switch from R_L to R_H. Similarly, $X = 0$ will cause a current $I_{d,10}$ to flow in the reverse direction through M_3, the STT-MTJ and M_2. This current must exceed a critical current $I_{c,10}$ for a minimum duration of τ_{10}, in order for the device to switch from R_H to R_L. Thus the four critical parameters associated with an MTJ are R_L, R_H, I_c and τ.

There has been extensive work on the development of compact models of STT-MTJ devices (Sun *et al.* (2011); Xu *et al.* (2015); Wang *et al.* (2014); Zhang *et al.* (2015)). For feature sizes below 40nm, the model described in Wang *et al.* (2014) (also in Zhang *et al.* (2015)) is used here, as it integrates a number of physical models, enabling the analysis of static, dynamic and stochastic behavior, and reports results that show good agreement with experiments. The expressions (Eqn. 5.1, 5.2 and 5.3) for R_L, R_H and switching time τ are taken from Zhang *et al.* (2015). Where $\overline{\varphi}$ is the potential barrier height of crystalline MgO, t_{ox} is the thickness of the oxide barrier, A is the MTJ area. F is a factor calculated from the resistance-area product ($R \cdot A$) value of MTJ. For $R \cdot A = 10\Omega \cdot \mu m^2$, F = 332.2. $C \approx 0.577$ is the Euler's constant,

$\xi = E/k_BT$ the thermal stability factor, V is volume of free layer, M_s is saturation magnetization, P_{ref}, P_{free} the tunneling spin polarizations of the reference and free layers. The parameters α, β and κ include multiple physical parameters. For the purposes herein, they are technology constants.

$$R_L = \frac{t_{ox}}{F \times \overline{\varphi} \times A} \times e^{1.025 \times t_{ox} \times \overline{\varphi}^{-1/2}} \Longrightarrow \alpha t_{ox} e^{\beta t_{ox}}, \tag{5.1}$$

$$R_H = (1 + TMR) \cdot R_L, \tag{5.2}$$

$$\frac{1}{\tau} = [\frac{2}{C + \ln(\frac{\pi^2 \xi}{4})}] \frac{\mu_B P_{ref}}{e M_s V (1 + P_{ref} P_{free})} (I_d - I_c) \Longrightarrow \tau = \kappa \frac{1}{|I_d - I_c|} \tag{5.3}$$

R_L and R_H are comparable to the on-channel resistances of the CMOS transistors in the driver. Therefore, the voltage drop across the MTJ during switching, combined with the fixed power supply V_{dd}, limits the maximum current that a driver can deliver. That driver current depends on the transistor dimensions together with R_L and R_H, which are in turn related to t_{ox} of the MTJ (Equations 5.1 and 5.2). Local and global process variations in transistors and MTJs make the driver current a statistically varying quantity among different devices on the same die and among the same devices on different dice. However, before considering process variations, it will be instructive to examine the factors that affect the transistor sizes in the driver, and how those sizes might be determined.

5.2.2 Driver Sizes Ignoring Process Variations

$I_{d,01}$, and $I_{d,10}$ are functions of R_L, R_H, and the transistor widths W_4, and W_2, where R_L and R_H are determined by t_{ox} (see Equation (5.1)). Writing a 1 (0) in the MTJ will require $I_{d,01}(t_{ox}, W_4) > I_{c,01}$ ($I_{d,10}(t_{ox}, W_2) > I_{c,10}$), and the corresponding switching time τ_{01} (τ_{10}) will be inversely proportional to the excess current (Equation (5.3)).

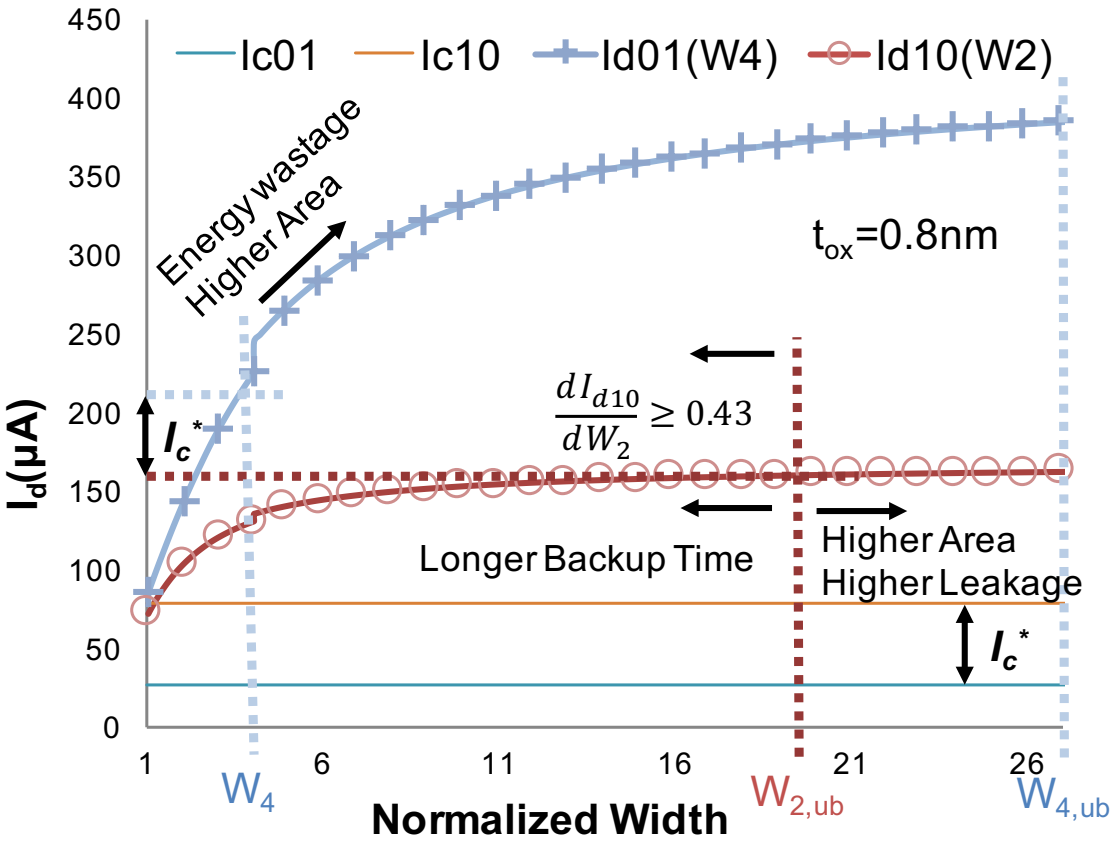


Figure 5.2: $I_{d,01}$ (blue line) and $I_{d,10}$ (red line) vs normalized width. $I_{c,01}$ and $I_{c,10}$ are also included. $\bar{t}_{ox} = 0.8nm$

Let $\gamma = W_1/W_4 = W_3/W_2$ denote the ratio of the width of pFET M_1 (M_3) to the width of nFET M_4 (M_2), and assume that γ is fixed. Fig. (5.2) shows HSPICE generated plots of $I_{d,01}$ and $I_{d,10}$ as a function of the width of the corresponding nFETs W_4, and W_2, respectively, for a specific value of t_{ox}.

From Fig. (5.2), it is seen that any pair of values for W_4, and W_2 are feasible as long as the corresponding $I_{d,01}(W_4) > I_{c,01}$ and $I_{10}(W_2) > I_{c,10}$. The objective is to choose values that minimize the total energy E_{total} required to store a 0 and 1. $E_{\text{total}} = V_{dd}(\tau_{01}I_{d,01}(W_4) + \tau_{10}I_{d,10}(W_2))$. Let $\tau = \max\{\tau_{01}, \tau_{10}\}$ be single time to backup a 0 or a 1. Then

$$E_{\text{total}}(\tau, \tau_{01}, \tau_{01}, I_{d,01}, I_{d,10}) = V_{dd}[\tau_{01} I_{d,01}(W_4) + (\tau - \tau_{01}) I^*_{d,01}(W_4) + \tau_{10} I_{d,10}(W_2) + (\tau - \tau_{10}) I^*_{d,10}(W_2)]. \tag{5.4}$$

$I^*_{d,01}(W_4)$ and $I^*_{d,10}(W_2)$ are the currents after the state transitions have completed. They are different from $I_{d,01}(W_4)$ and $I_{d,10}(W_2)$ because of the change in the device resistances. E_{total} is at least $V_{dd}(\tau_{01} I_{d,01}(W_4) + \tau_{10} I_{d,10}(W_2))$. Hence the minimum of the average or total energy with a single backup time would require that $\tau = \tau_{01} = \tau_{10}$. Then, using Equation (5.3), $I_{d,01}(W_4) - I_{c,01} = I_{d,10}(W_2) - I_{c,10}$, or equivalently, $I_{d,01}(W_4) - I_{d,10}(W_2) = I_{c,01} - I_{c,10} = I^*_c$, where I^*_c is independent of W. Therefore the basic constraint that needs to be satisfied when determining the driver size is

$$I_{d,01}(W_4) = I_{d,10}(W_2) + I^*_c. \tag{5.5}$$

If Equation (5.5) is satisfied, then the total energy is $E_{\text{total}} = V_{dd}\tau(2I_{d,10}(W_2) + I^*_c)$. Now $\tau = \tau_{10} = \kappa/(I_{d,10}(W_2) - I_{c,10})$, and E_{total} can be written as

$$E_{\text{total}} = V_{dd}\kappa \left(\frac{2I_{d,10}(W_2) + I^*_c}{I_{d,10}(W_2) - I_{c,10}} \right). \tag{5.6}$$

Equation (5.6) shows that with equal switching times for storing a 0 and 1, minimizing the total energy is equivalent to maximizing $I_{d,10}(W_2)$. This fact can be used to determine W_2 and $I_{d,10}(W_2)$. W_4 is determined by solving Equation (5.5).

Fig. (5.2) shows plots of $I_{d,10}(W_2)$ (lower curve) and $I_{d,01}(W_4)$ as a function of driver transistor width [1] which are enumerated in discrete increments. W_{min} is the minimum possible width. $W_{2,ub}$ and $W_{4,ub}$ denote widths at which the currents $I_{d,10}$ and $I_{d,01}$ have saturated, i.e., for some small $\epsilon > 0$, $W_{2,ub} = \min\{W \mid dI_{10}/dW \leq \epsilon\}$, and $W_{4,ub} = \min\{W \mid dI_{01}/dW \leq \epsilon\}$. Choosing a value larger than $W_{2,ub}$ or $W_{4,ub}$

[1] Transistor width is normalized to the minimum width allowed in the technology. Therefore, $W_{min} = 1$

will not increase the current appreciably, but increases area. As E_{total} decreases with I_d, and I_d is monotonic with respect to W, the width W_2 that maximizes I_d can be determined by examining the boundary conditions.

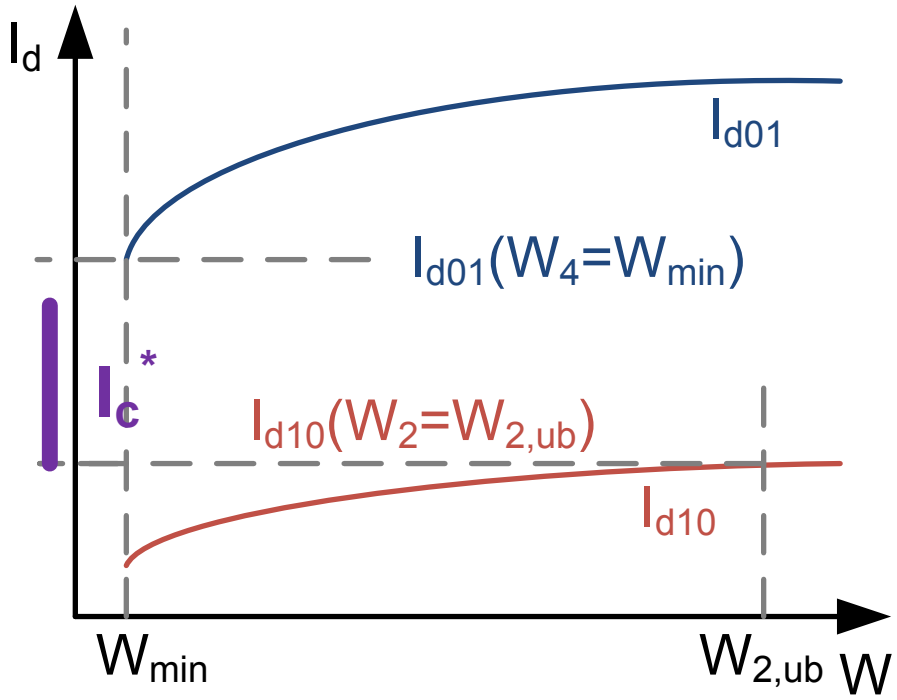


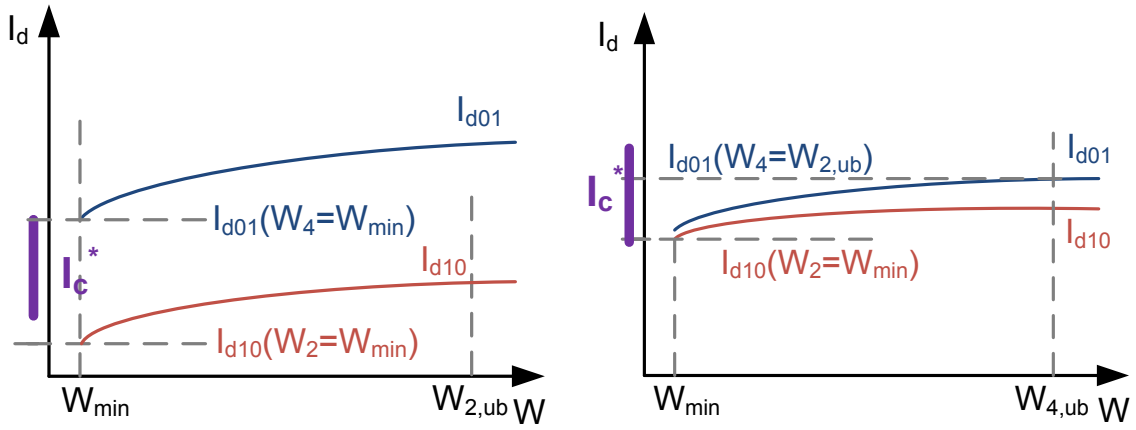
Figure 5.3: Driver current versus transistor width Case I.

Case I: $I_{d,01}(W_4 = W_{min}) > I_{d,10}(W_2 = W_{2,ub}) + I_c^*$.

This is shown in Fig. (5.3), and corresponds to the situation where $R_L << R_H$ (the low and high resistances are widely separated). Even choosing $W_2 = W_{2,ub}$, there is no corresponding value of W_4 for which $I_{d,10}(W_{2,ub}) + I_c^* = I_{d,01}(W_4)$, i.e., equal backup times is not possible, and Equation (5.5) cannot be satisfied. Therefore, the only choice is $W_4 = W_{min}$. Choosing a larger value for W_4 makes writing a logic 1 even faster and wastes energy and area, because the actual backup time is determined by the time required to write a logic 0. Choosing a smaller value for W_2 makes writing a logic 0 even slower.

Note that with $R_L << R_H$, the process of *reading* is more robust, at the expense

of increased energy for writing. This is opposite to the general conclusion on NVM design that wide R_L and R_H separation is always desired. In an AES powered NVL design, devices with widely separated resistance states like an RRAM cell require more energy for writing data than MTJs, while providing greater robustness when reading data.



(a) Case II (b) Case III

Figure 5.4: Driver current versus transistor width Case II and Case III.

Case II: $I_{d,01}(W_4 = W_{min}) > I_{d,10}(W_2 = W_{min}) + I_c^*$.

This is depicted Fig. (5.4a). Since I_d is monotonically increasing, $I_{d,10}(W_2 = W_{2,ub}) > I_{d,10}(W_2 = W_{min})$. Therefore, Case I implies this Case. Hence if Case I fails, and this Case is true, then

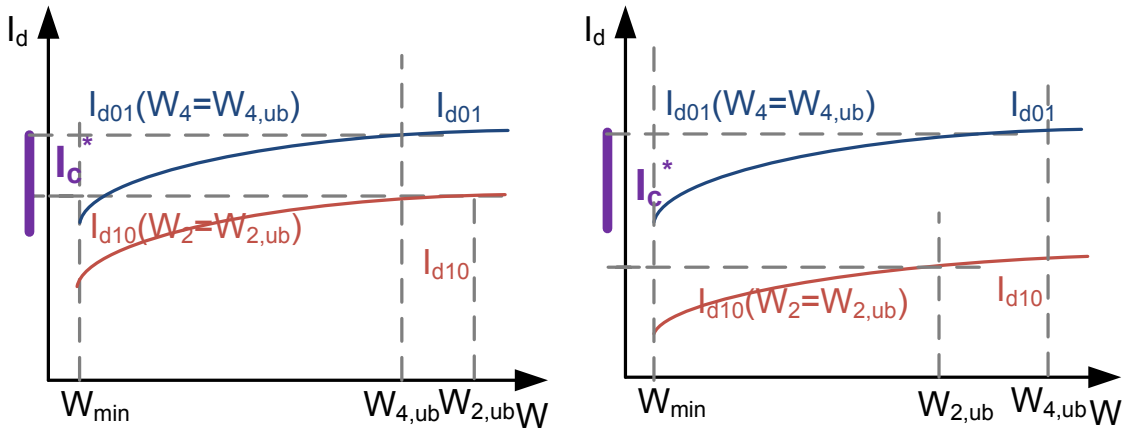
$$\begin{aligned} I_{d,10}(W_2 = W_{2,ub}) &> I_{d,01}(W_4 = W_{min}) - I_c^* \\ &> I_{d,10}(W_2 = W_{min}). \end{aligned}$$

Equation (5.5) has a solution with $W_2 = W_{2,ub}$, and $W_4 = I_{d,01}^{-1}(I_{d,10}(W_2 = W_{2,ub}) + I_c^*)$. Note that choosing $W_4 = W_{4,ub}$ will not satisfy Equation (5.5).

Case III: $I_{d,01}(W_4 = W_{4,ub}) < I_{d,10}(W_2 = W_{min}) + I_c^*$.

This is shown in Fig. (5.4b), and corresponds to the situation when R_L and R_H

are very close and their magnitudes are high, resulting in lower and flatter I_d curves. Higher resistances might be desired so as to reduce the possibility of a *read disturb* and improve thermal stability. In this situation, Equation (5.5) has no solution, and the only option is $W_4 = W_{4,ub}$, and $W_2 = W_{min}$. This speeds up the writing of a logic 1, and slows the writing of a logic 0, when compared to both transistors being of minimum size.



(a) Case IV (b) Case V

Figure 5.5: Driver current versus transistor width Case IV and Case V.

Case IV: $I_{d,01}(W_4 = W_{4,ub}) < I_{d,10}(W_2 = W_{2,ub}) + I_c^*$.

This is shown in Fig. (5.5a). Since $I_{d,10}(W_2 = W_{min}) < I_{d,10}(W_2 = W_{2,ub})$, Case III implies this Case. Hence if Case III fails, and this Case holds, then

$$\begin{aligned} I_{d,10}(W_2 = W_{2,ub}) &> I_{d,01}(W_4 = W_{4,ub}) - I_c^* \\ &> I_{d,10}(W_2 = W_{min}). \end{aligned}$$

Equation (5.5) has a solution, which is $W_4 = W_{4,ub}$ and $W_2 = I_{d,10}^{-1}(I_{d,01}(W_4 = W_{4,ub}) - I_c^*)$.

Case V: $I_{d,01}(W_4 = W_{4,ub}) > I_{d,10}(W_2 = W_{2,ub}) + I_c^*$.

From Fig. (5.5b) it is apparent that there is solution to Equation (5.5), given by

$W_2 = W_{2,ub}$ and $W_4 = I_{d,01}^{-1}(I_{d,10}(W_2 = W_{2,ub}) + I_c^*)$. Once again, note that choosing $W_4 = W_{4,ub}$ first, does not lead to a solution.

These five cases are summarized in Procedure EoptDriverSize shown in Algorithm 1.

5.2.3 Driver Sizes Considering Process Variations

The algorithm for driver sizing described in the previous section is now adapted for the case where the parameters of the transistors in the driver and the MTJ device are subject to manufacturing variations. For an MTJ device, the primary design parameter is its dimension and for the driver circuit, they are the dimensions of the transistors M_1–M_4. There are several secondary non-design parameters associated with the MTJ, such as localized fluctuation of magnetic anisotropy, thermally activated initial procession angle, thermal component of internal energy and etc. Raychowdhury *et al.* (2009), whose variations are not modeled in present work.

For an MTJ device, it has been shown that variations in t_{ox} have the most significant impact on energy consumption Munira *et al.* (2012). This is due to fact that R_H and R_L have an exponential dependence on t_{ox} (see Equations (5.1) and (5.2)). During fabrication, the oxide is grown over the entire die, and consequently, it is assumed that the variation in its thickness is the same for all devices. Hence, following (Munira *et al.* (2012); Wirnshofer (2013)), t_{ox} variation is assumed to global. Consequently, the length L_{MTJ} and width W_{MTJ} of the MTJ can be assumed to be fixed at the minimum feature size of the technology, and that the deviations in t_{ox} among different MTJs on a given die will be the same. On the other hand, the dimensions of the CMOS transistors in the driver are assumed to be subject to both local and global variations. Thus, the widths W_2 and W_4 are modeled as independent random variables centered around their respective nominal values $\overline{W}_2$ and $\overline{W}_4$, which

1 EoptDriverSize($W_{min}, W_{2,ub}, W_{4,ub}$);

output: Energy optimal values of W_2, W_4

2 **if** $I_{d,01}(W_4 = W_{min}) > I_{d,10}(W_2 = W_{2,ub}) + I_c^*$ **then**

3 $\quad W_2 = W_{2,ub}$;

4 $\quad W_4 = W_{min}$; /* case I */

5 **endif**

6 **else if** $I_{d,10}(W_2 = W_{2,ub}) > I_{d,01}(W_4 = W_{min}) - I_c^* > I_{d,10}(W_2 = W_{min})$ **then**

7 $\quad W_2 = W_{2,ub}$;

8 $\quad W_4 = I_{d,01}^{-1}(I_{d,10}(W_2 = W_{2,ub}) + I_c^*)$; /* case II */

9 **endif**

10 **else if** $I_{d,01}(W_4 = W_{4,ub}) < I_{d,10}(W_2 = W_{min}) + I_c^*$ **then**

11 $\quad W_2 = W_{min}$;

12 $\quad W_4 = W_{4,ub}$; /* case III */

13 **endif**

14 **else if** $I_{d,10}(W_2 = W_{min}) < I_{d,01}(W_4 = W_{4,ub}) - I_c^* < I_{d,10}(W_2 = W_{2,ub})$ **then**

15 $\quad W_4 = W_{4,ub}$;

16 $\quad W_2 = W_2 = I_{d,10}^{-1}(I_{d,01}(W_4 = W_{4,ub}) - I_c^*)$; /* case IV */

17 **endif**

18 **else**

19 $\quad W_2 = W_{2,ub}$;

20 $\quad W_4 = I_{d,01}^{-1}(I_{d,10}(W_2 = W_{2,ub}) + I_c^*)$; /* case V */

21 **endif**

Algorithm 1: Computes optimal transistors sizes W_2, W_4

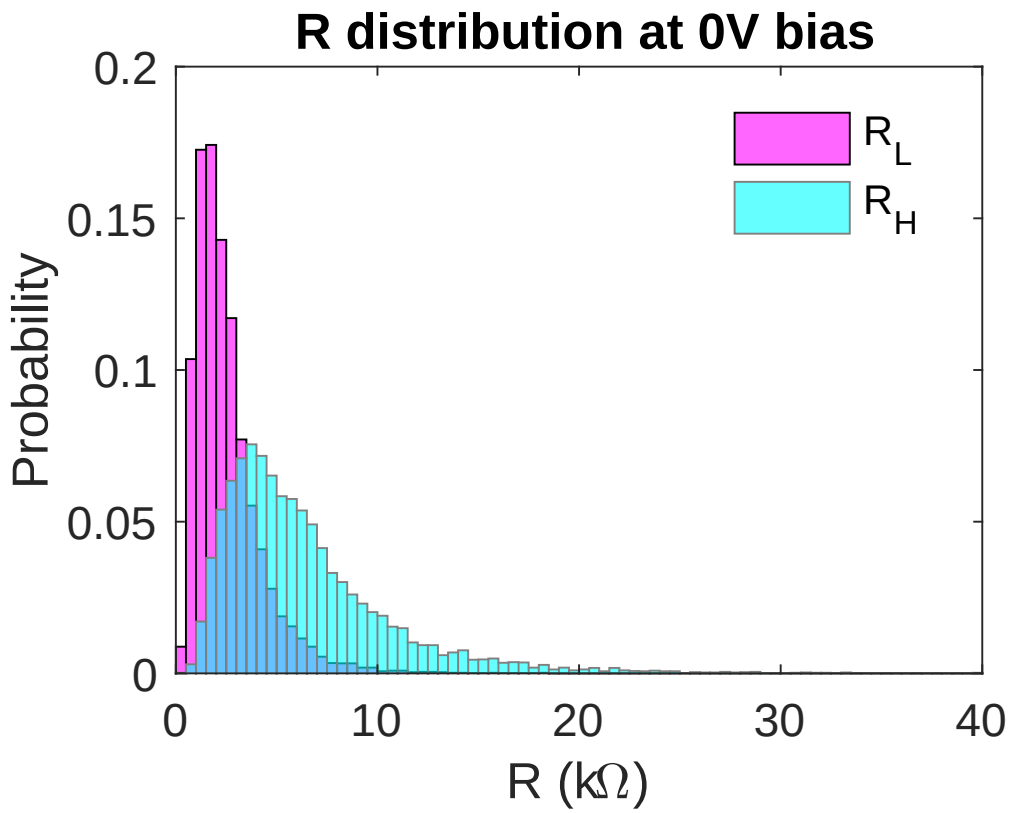


Figure 5.6: Frequency histograms of R_L and R_H using 10K MonteCarlo samples. Data for CMOS is from a 40nm commercial library with foundry supplied parameters and HSPICE models. Data for MTJ variations was generated assuming $\bar{t}_{ox} = 0.8nm$ and $\sigma_{tox} = 0.1\bar{t}_{ox}$, and using models in (Wang *et al.* (2014); Zhang *et al.* (2015)).

are to be specified as part of the design.

Variations in t_{ox} result in variations in R_L and R_H (see Fig. (5.6)), and variations in t_{ox}, W_2, and W_4 will result in corresponding variations in the driver currents. Fig. (5.7) shows frequency histograms of I_d in a driver, assuming different sources of variations. The inset plot shows the histogram of I_d considering local and global variations only in the driver transistors, and the outer plot includes variations in the transistor dimensions and t_{ox} of the MTJ. The plots indicate that variations in t_{ox} overwhelm the effect of variations in the transistors' dimensions. However, in the interest of generality and applicability to scaled geometries, the currents $I_{d,01}$ and $I_{d,10}$ are modeled as a function of a collection of random variables over the parameter

space (W_2, W_4, t_{ox}).

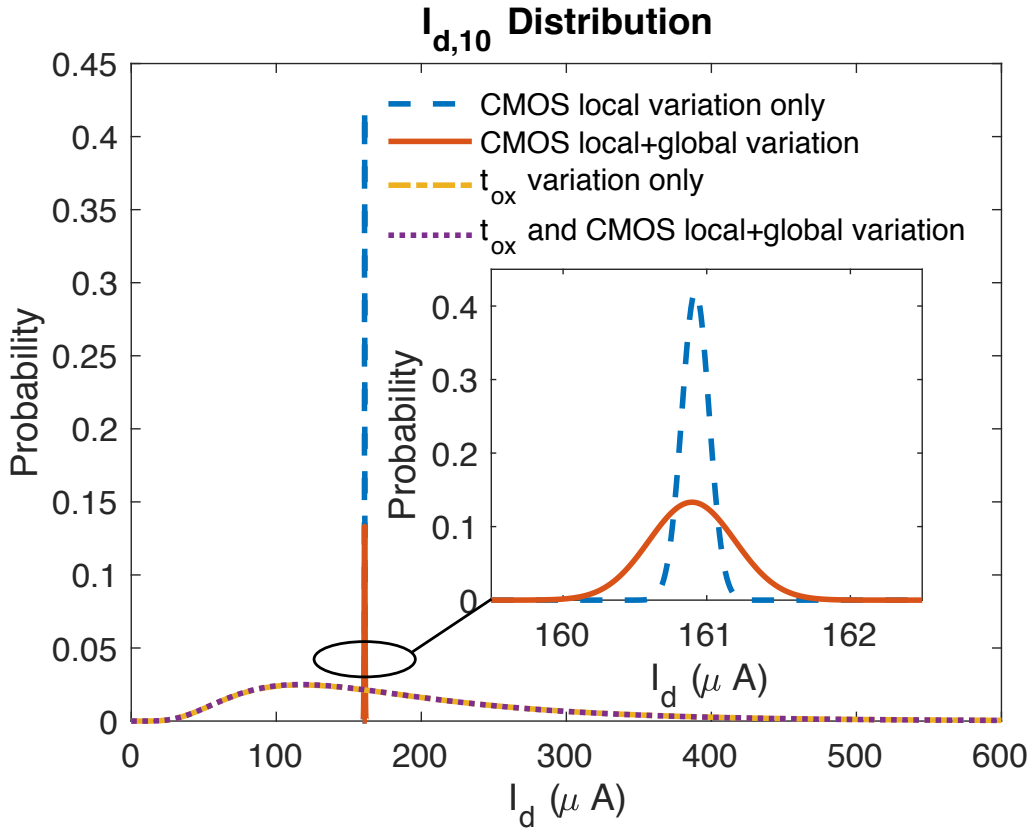


Figure 5.7: Frequency histogram of I_d using 10K MonteCarlo samples. MonteCarlo configuration is the same as 5.6.

Fig. (5.8) shows plots of I_d as a function of the (normalized) widths of the driver's transistors. The red ($I_{d,10}$) and blue ($I_{d,01}$) solid curves correspond to the case where no variations are considered in the transistor dimensions nor in the t_{ox} of the MTJ. These plots are similar to those shown in Fig. (5.2). The plots also show individual populations (10K) of the $I_{d,10}$ and $I_{d,01}$ values generated by Monte Carlo simulations, by varying (W_2, W_4, t_{ox}) around their nominal values $[\overline{W}_{2,i}, \overline{W}_{4,j}, \bar{t}_{ox}]$, for $(i, j) \in [1, n]$. Let $S(\overline{W}_2, \overline{W}_4, \bar{t}_{ox})$ denote the population of samples centered at $(\overline{W}_2, \overline{W}_4, \bar{t}_{ox})$.

The problem to determine the energy-optimal driver size in the presence of process variations is to identify the population (i.e. the nominal values $\overline{W}_2, \overline{W}_4$) that have at least $y\%$ (y being the yield) of the samples resulting in a successful backup and

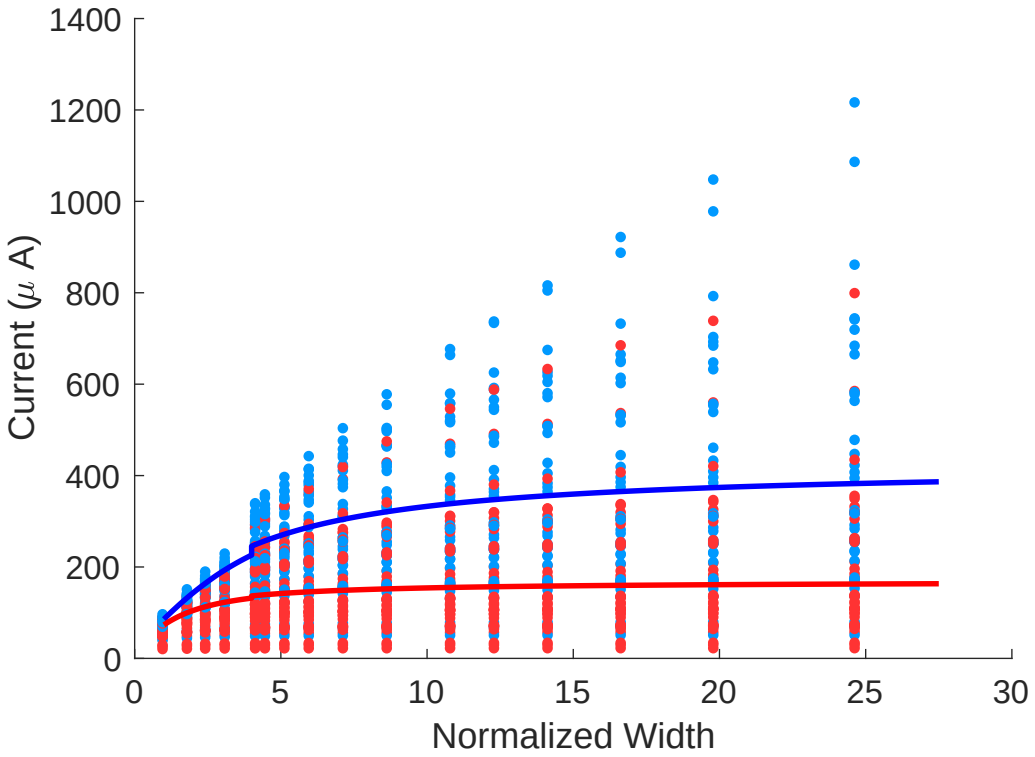


Figure 5.8: I_d current vs driver width. Blue line (dots) is $I_{d,10}$, Red line (dots) is $I_{d,01}$. Lines are with no process variation, dots are currents with t_{ox} and CMOS (local and global) variation. Note: To avoid clutter, only a subset of widths are plotted.

restore, and have minimum average energy. Yield and energy are related. To see how to compute energy as a function of yield, consider samples of I_d shown in Fig. (5.8). Each pair of data points (red and blue dots) within a population has an associated backup time τ_{01} and τ_{10}, that can computed using Equation (5.3). The corresponding total energy would be calculated by Equation (5.4) where $\tau = \tau_y$. This energy is computed for all the samples in a given population whose backup times fall within the y percentile, for a given yield y.

Fig. 5.9 shows plots of the average energy versus the driver width, for several values of the yield y. It is clear that unlike the deterministic case (see Fig. (5.2)), the minimum of the average energy does not necessarily correspond to the largest value of the transistor width (i.e. maximum current) but instead to some intermediate value.

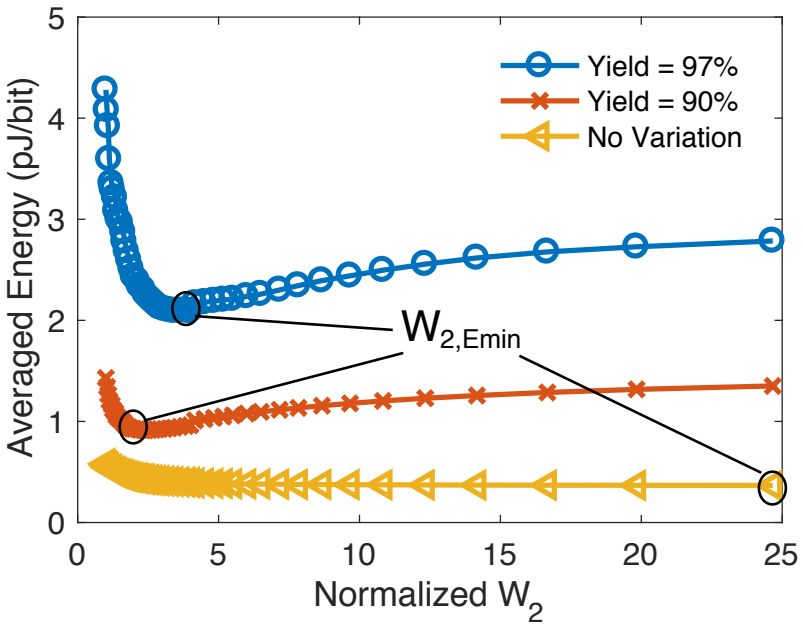


Figure 5.9: Average total energy versus driver width, for different yields, accounting for process variations. Minimum energy is achieved with $W_{2,Emin} = W_{2,ub}$ when no variation is included. In the presence of process variations, yield constrained minimum energy can be achieved with smaller $W_{2,Emin}$, whose value depends on the target yield.

The smaller W_2 implies lower current and longer backup time.

The procedure to determine the nominal widths of the driver transistors in the presence of process variations is shown in Algorithm (2). The objective is to identify the nominal values $(\overline{W}_2, \overline{W}_4)$ that define a population $S(\overline{W}_2, \overline{W}_4, \bar{t}_{ox})$ whose ensemble average energy computed over all those outcomes whose backup times fall below τ_y (the y percentile value of the backup time) is minimum. The procedure is a non-parametric or data-driven approach, using the empirical distribution of currents generated by Monte Carlo simulations to compute averages. As the set of transistor widths form a discrete set, the procedure starts with setting the nominal values to their respective upper bounds (lines 2), and iterates over the discrete set (line 3). Procedure EoptDriverSize is used to determine the next nominal value around which to generate the sample population (line 5,6), and then the backup times and currents are computed for each sample point (lines 7-11). The average of the samples

whose backup times are within the y percentile value is computed (lines 12-18). The minimum average energy value is retained, and the procedure terminates as soon the average starts to increase (lines 19, 20).

5.3 Non-Volatile Flipflops with Scan

5.3.1 Yield versus Energy Consumption

Fig. 5.9 shows that higher yield requires higher energy expenditure. One way to reduce backup energy is to boost the voltage Motaman *et al.* (2015). However this is not practical for the type of low voltage, low power ASICs employing energy harvesting that are the target of this work. Techniques for improving the energy efficiency by balancing the backup times used in NVM as described in (Motaman *et al.* (2015); Bishnoi *et al.* (2016a); Zhang *et al.* (2011)) are not applicable for NVFFs. For this reason the method described in Section 5.2, minimizes the average energy under a yield constraint by sizing the drivers separately. Other techniques that improve the write margin by increasing the driver size (to increase I_d) and the backup time, result in high energy consumption (Bishnoi *et al.* (2016a); Zhang *et al.* (2011)). Device engineering as in Halawani *et al.* (2016) can also be done to trade retention time with write energy. However that is outside the scope of this chapter.

The backup time τ_y determined by procedure EoptDriverSizeWPR ensures that, with a high probability, y% of the dice will succeed in backup of a '1' and '0'. However, the conservative choice of τ_y results in wasted energy for most of the dice. This motivates the adaptive approach of determining the backup time on a per-chip basis. This section presents the architecture of a NVFF equipped with a scan mechanism which allows for dynamically testing and adjusting the backup time to minimize the backup energy. This scan mechanism is compatible with the normal

EoptDriverSizeWPR($[W_{min}, W_{ub}], t_{ox}, y$) ;

output: Energy optimal values of $\overline{W}_2$, $\overline{W}_4$ and τ_y

$i = 1$, $\overline{W}_{2,0} = \overline{W}_{2,ub}$, $\overline{W}_{4,0} = \overline{W}_{4,ub}$, $E_{avg,0} = \infty$;

while $\overline{W}_{min} \leq \overline{W}_i \leq \overline{W}_{ub}$ **do**

  $[\overline{W}_{2,i}, \overline{W}_{4,i}] =$

  EoptDriverSize($\overline{W}_{min}$, $\overline{W}_{2,i-1}$, $\overline{W}_{4,i-1}$);

  $S_j = (W_{2,i,j}, W_{4,i,j}, t_{ox,j})$ =MC($\overline{W}_{2,i}$, $\overline{W}_{4,i}$, $\bar{t}_{ox}$) ; `/* Gen MC samples */`

  for *j=1:N* **do**

    ($I_{d,01,j}$, $I_{d,10,j}$) = HSPICE(S_j) ; `/* Find Id by HSPICE */`

    ($\tau_{01,j}$, $\tau_{10,j}$) = Eqn 5.3 ($I_{d,01,j}$, $I_{d,10,j}$) ;

    $\tau_j = \max(\tau_{01,j}, \tau_{10,j})$;

  end

  $\tau_y : \text{Prob}(\tau \leq \tau_y) = y$; `/* y% of switching times` $\leq \tau_y$ `*/`

  for *j=1:N* **do**

    if $\tau_j \leq \tau_y$ **then**

      E_j= Eqn 5.4 (τ_y,$\tau_{01,j}$, $\tau_{10,j}$,$I_{d,01,j}$, $I_{d,10,j}$) ;

    endif

  end

  $E_{avg,i} = (E_1 + E_2 + \cdots + E_N)/(yN)$; `/* N is number of samples */`

  if $E_{avg,i} > E_{avg,i-1}$ **then**

    return $\overline{W}_{2,i-1} + \Delta W$, $\overline{W}_{4,i-1} + \Delta W$, τ_y;

  endif

  $\overline{W}_{2,i} = \overline{W}_{2,i} - \Delta W$, $\overline{W}_{4,i} = \overline{W}_{4,i} - \Delta W$,$i = i + 1$;

end

Algorithm 2: Procedure to compute optimal sizes of driver transistors W_1, W_4, W_3, W_2 considering process variations.

scan available on traditional flipflops, and hence has minimum hardware cost.

5.3.2 NVSFF Basic Structure

The general structure of a non-volatile scan flipflop (NVSFF) is shown in Fig. 5.10. A non-volatile storage unit (NVSU) is attached to a volatile flipflop. This NVSFF has five modes of operation. In the *normal* mode (regular operation) and *normal scan* mode, it acts like an edge-triggered scan flipflop. In these modes $RES = 0$ and $SAV = 0$, which together disconnect the path between the NVSU and the volatile flipflop. During the *backup* mode, the flipflop state is stored into the NVSU. After the backup mode is completed, the system can be safely powered off without losing the intermediate results. During the *restore* mode, the previous stored state is read out and presented on the flipflop output Q. The *non-volatile test* mode is a combination of the normal scan mode, the backup mode and the restore mode. This operation mode is mainly for performing the non-volatile device test and determining the backup time. Details of the circuit operation and design considerations are presented next.

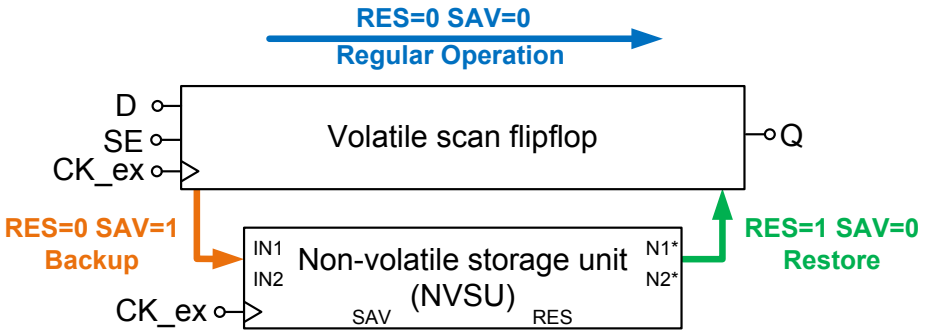


Figure 5.10: The basic structure of non-volatile scan flipflop (NVSFF).

5.3.3 Non-Volatile Storage Unit (NVSU)

The architecture of the NVSU is shown in Fig. 5.11. It takes two differential signals $IN1$ and $IN2$, and produces two differential outputs $N1^*$ and $N2^*$. RES and

SAV control the operation mode. Two STT-MTJ devices are included in the NVSU. The one labeled STT_data stores the state during backup mode. The one labeled STT_ref serves as a reference, used during the restore mode.

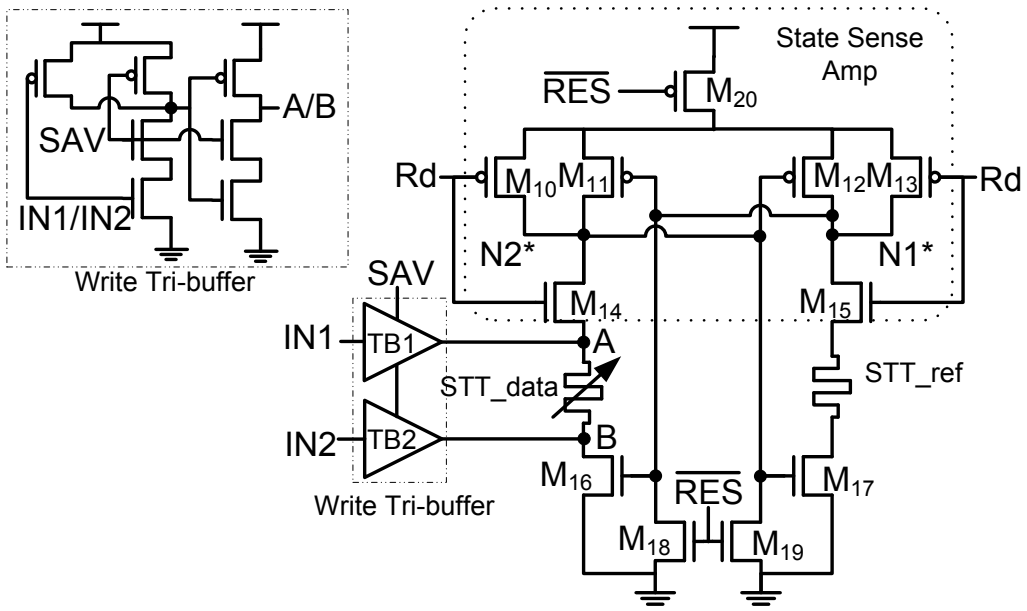


Figure 5.11: The schematic of NVSU. The NVSU includes a write buffer, two STT-MTJ devices and a state sense amplifier.

Normal Mode and Normal Scan Mode: The NVSU is inactive during the normal mode, and is turned off to save power. The input and output transistors are sized small to reduce the parasitics on the normal signal path.

Backup Mode: $RES = 0$ and $SAV = 1$ sets the NVSU to the backup mode. The unit labeled as *state sense amplifier* is inactive in this mode. Current will flow through write tri-state buffers $TB1$ and $TB2$ and set the state of STT_data. The current direction is determined by $IN1$ and $IN2$. Compared to the driver shown in Fig. (5.1), $TB1$ and $TB2$ consist of one pFET and two nFETs in a stack. The extra SAV driven pFET eliminates a false path to MTJ during restore mode.

The SAV signal is independent of the clock, and as long as $SAV = 1$ and inputs are differential, $TB1$ and $TB2$ will provide the necessary current to store the data.

Note that consistent with other works on NVM and NVL (Khanna *et al.* (2014); Ma *et al.* (2015); Balsamo *et al.* (2015)), it is assumed that there exists a mechanism that will predict an impending power system failure and will initiate the backup by setting $SAV = 1$ during the period with $CK = 1$. A method to predict such a failure can be found in Balsamo *et al.* (2015).

Restore Mode: When the power is re-established, the state of the flipflop can be restored by setting $SAV = 0$ and $RES = 1$. The two tri-state buffers $TB1$ and $TB2$ are disabled. When $Rd = 0$, $N1^* = 1$ and $N2^* = 1$. When $Rd : 0 \rightarrow 1$, $M14$ and $M15$ in Fig. 5.11 become active, creating discharge paths to ground for both $N1^*$ and $N2^*$. Assuming $STT\_data = R_L \ll R_{ref}$, the positive feedback in the state sense amplifier will sense the conductance difference between two discharging paths and set $N2^* = 0$ and $N1^* = 1$, which drive a regular flipflop and set its output to $Q = 0$.

A *read disturb* occurs when the stored state in an STT-MTJ is flipped on a read operation. The probability of a read disturb in the NVSU can be reduced by using smaller transistors or lowering the power supply voltage for the state sense amplifier, at the cost of a longer restoration time. Unlike NVM implementations in which the stored data would be read more than once, in the NVFF with backup and restore, the stored data would only be restored to the datapath once. When the next power interrupt occurs, new data would be backed up. Therefore, the read disturb is not the primary concern in NVFF design.

Non-volatile Test Mode: This mode is applied to test the functionality of other two modes as well as determine an optimal backup time. Unlike the other operation modes, this involves a sequence of operations. It starts in the normal scan mode ($SE = 1$, $SAV = 0$ and $RES = 0$) that scans in the test data, resulting in the data appearing at each output Q. After the data has been scanned in, the NVSFF is switched to the backup mode and restore mode. After a backup and restore step, the

previous test data will be present at the output, if both steps completed successfully. Then the output data is scanned out for verification by switching to the normal scan mode. The backup time is the duration when $SAV = 1$. The control signal sequence is shown in Fig. 5.12.

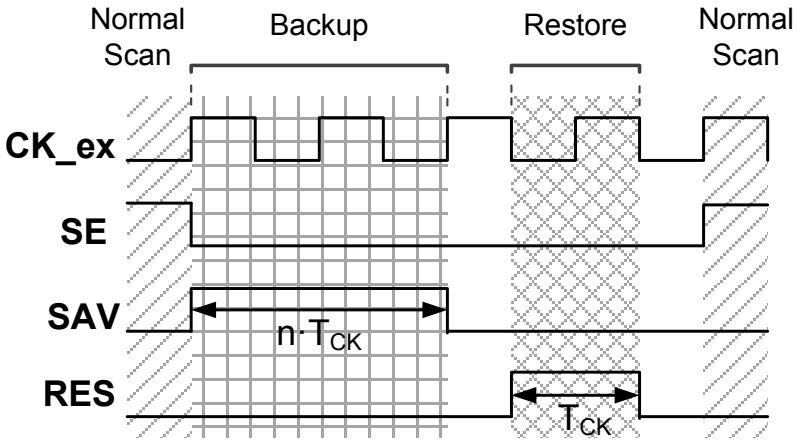


Figure 5.12: The control signal sequence during non-volatile test mode.

Timing of Control Signals: RES is synchronized with the falling edge of the clock, and therefore can be easily generated by a negative edge triggered flipflop. Rd is derived from both RES and CK, which feeds into state sense amplifier. SAV controls $TB1$ and $TB2$. When input signals $IN1$ and $IN2$ are stable, the duration of SAV determines backup time τ. Although SAV can be synchronous or asynchronous, a synchronous signal is preferred as it can easily be generated by a counter followed by a flipflop, and the total backup time would simply be $\lceil \tau/T \rceil \times T$ or roundup$(\tau/T) \times T$, where T is the clock period. An asynchronous SAV can be generated by a separate pulse generation circuit, where τ is controlled by the pulse width. In an energy-area-constrained digital system, a synchronous SAV would be preferred because control circuitry would be smaller and consume less power than an on-chip pulse generator. The one disadvantage of using a synchronous SAV is that granularity with which τ can be adjusted is one clock period. Therefore, if the clock period is large, an

asynchronous SAV may actually result in lower energy expenditure.

Timing of Input/Output Signals: During backup mode, $IN1$ and $IN2$ should be differential and stable. No current would flow through STT_data if $IN1 = IN2$. If both signals flip, the current direction would change. During the restore mode, the outputs $N1^*$ and $N2^*$ will become differential after the state sense amplifier evaluates, when $CK = 1$ and $Rd = 1$. When $CK = 0$ and $Rd = 0$, both $N1^*$ and $N2^*$ are reset to 1. A latch is required to maintain the evaluation results on the NVSFF outputs when CK is low.

5.3.4 Non-Volatile Scan Differential Flipflop (NVSFF-DM)

The NVSU takes a pair of differential inputs during the backup mode, and produces a pair of differential outputs during the restore mode. Therefore, the simplest type of flipflop to interface with the NVSU would be a differential or sense-amp based flipflop. Fig. 5.13 shows such a modified version of KVFF Yang *et al.* (2015b) interfaced with the NVSU. The combined unit is referred as NVSFF-DM. The circuit includes a differential sense amplifier with its output $N1$ and $N2$ connected to both the SR-latch and the NVSU. The inputs to the SR-latch can be switched from either the sense amplifier or the NVSU outputs. The tri-state buffers between SR-latch and the two sources are not shown. In the normal mode, when $CK = 0$, it is easy to verify that $(N1, N2) = (1, 1)$. When $CK : 0 \rightarrow 1$, $(N1, N2) = (0, 1)$ or $(N1, N2) = (1, 0)$, depending on the input D. $(N1, N2)$ set the output of SR-latch accordingly. The two feedback loops in Fig. (5.13) are there to eliminate potential floating nodes as explained in Section 3.2.1. $(N1, N2)$ become differential and stable after evaluation is completed.

The internal CK is gated by SAV and Rd is gated by RES. SAV ensures that CK remains at 1 during the backup mode, and RES ensures that Rd follows

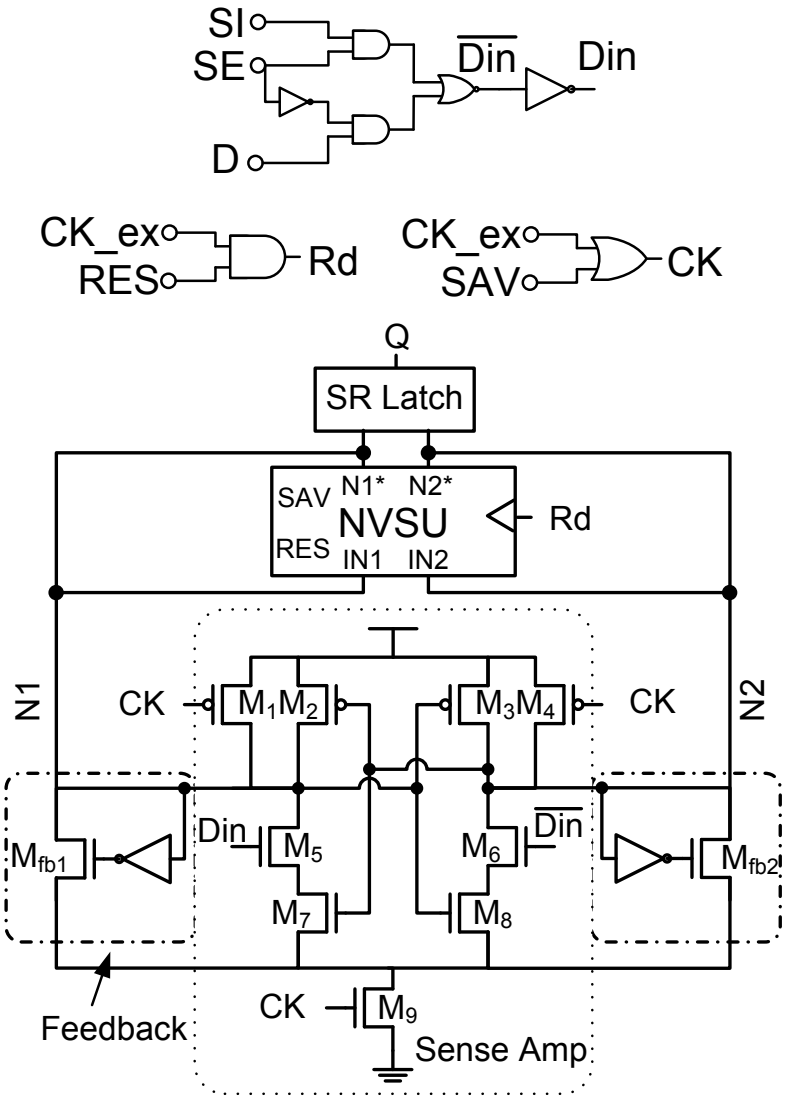


Figure 5.13: Schematic of the NVSFF-DM. The tri-state buffers between NVSU and SR latch are not shown.

the external clock CK_ex only during restore mode. CK-gating makes sure that $(N1, N2)$ change from $(1, 1)$ to $(0, 1)$ or $(1, 0)$ only once. Rd gating ensures that the state sense amplifier will operate and consume power only during the restore mode. The SR-latch latches the output either from the sense amplifier or the NVSU. The requirements imposed by the NVSU on its inputs and outputs are satisfied with these settings of the NVSFF-DM.

5.3.5 Non-Volatile Scan Master-Slave Flipflop (NVSFF-MS)

With some modification, the NVSU can also be combined with a conventional master-slave flipflop to form a non-volatile scan master-slave flipflop (NVSFF-MS). This is shown in Fig. (5.14). The scan mechanism is the same as in a conventional D-flipflop. However, the NVSU needs to be properly interfaced with the master and slave latches. The NVSU receives inputs (X and Y) from the slave latch during backup mode and sends its output back to the same node (X) during restore mode. To prevent the NVSU from interfering with the slave latch during normal mode and backup mode, a tri-state buffer is used to buffer the output of NVSU. This buffer should be turned on only when NVSU is in the restore mode and its outputs are ready. Since the outputs of NVSU would become differential only when they are ready, a completion detection signal CD is derived from $N1^*$ and $N2^*$ to drive the tri-state buffer. Unlike the NVSFF-DM, the slave latch and transmission gate between master and slave latch in NVSFF-MS are driven by different derived clocks derived from the master clock CK_ex. During the restore mode, the transmission gate should be turned off to block the signal from master latch. After the state is restored into the slave latch, the slave latch should be able to latch the data when external clock goes to 0.

The schematic of NVSFF-MS is shown in Fig. (5.14). In normal mode and normal scan mode, both SAV and RES are 0, NVSFF-MS operates the same as normal scan flipflop. The internal clock signals CK, CK', $\overline{CK}$ and $\overline{CK'}$ follow the external CK_ex under different conditions. CK follows CK_ex when both $SAV = 0$ and $RES = 0$, and CK' follows CK_ex when $SAV = 0$.

Nodes X and Y are fed into NVSU as differential inputs $IN1$ and $IN2$. In the backup mode, $SAV = 1$, $RES = 0$. Then $CK = CK' = 1$ and $\overline{CK} = \overline{CK'} = 0$. This

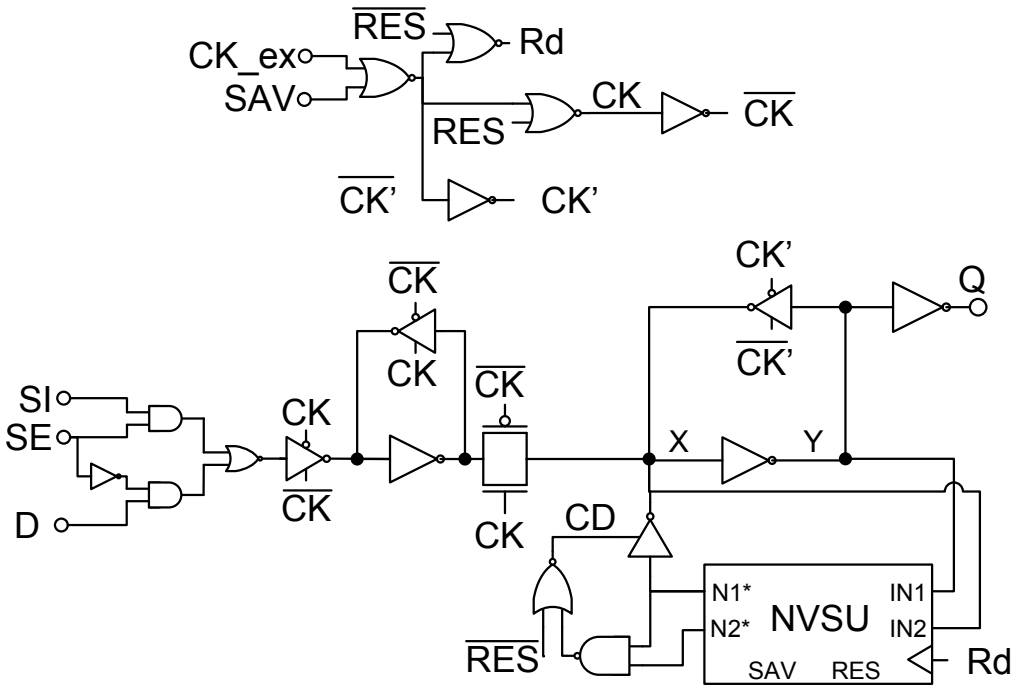


Figure 5.14: Schematic of NVSFF-MS.

disconnects the master from its inputs and the slave, so that the value of the master can be saved in the NVSU. $RES = 0$, $CD = 0$ blocks $N1^*$ to node X. It ensures that X and Y are kept differential and stable during entire backup mode.

During restore mode, $RES = 1$, $CK = 0$ and $\overline{CK} = 1$. The transmission gate between master and slave latches is blocked. In the meantime, Rd, CK' and $\overline{CK'}$ follow CK_ex. When $CK\_ex = 0$, $N1^* = N2^* = 1$, and $CD = 0$. The slave latches its previous state. When $CK\_ex : 0 \rightarrow 1$, the state sense amplifier in NVSU sets $N1^*$ and $N2^*$ into opposite values. These two differential signals set $CD = 1$, which enables the tri-state buffer between the NVSU and the slave. The value of $\overline{N1}^*$ is therefore sent to the slave latch to set its output Q.

5.3.6 Extension of Scan for Non-volatile Test

The conventional scan mechanism can be extended to include the test of non-volatile devices in each NVFF. The test procedure shown in Fig. (5.15) allows determining the *actual* or *chip-specific* backup time, after fabrication. Procedure EoptDriverSizeWPR described in Section 5.2 returns the nominal driver size that minimizes the average energy, and τ_y, which is the backup time for y percentile of the corresponding population. By definition, setting the backup time for all chips to τ_y would, with high probability, result in $y\%$ of the chips being successfully backed up. However, each specific chip might be successfully backed up with a smaller backup time. This smaller backup time, denoted by τ^*, is computed by using the scan mechanism on each chip. Once it is computed, it can be saved and used for backup whenever required. The energy savings using τ^* versus using τ_y can be substantial.

Fig. (5.15) shows an outline of the scan procedure to determine τ^*. If τ is the backup time computed by Procedure EoptDriverSizeWPR, then the least number of clock cycles whose total duration exceeds τ is $m(\tau) = roundup(\tau/T_{CK})$. In Fig. (5.15) this is initialized to $m = m_0 = m(\tau_y)$. Then data is scanned into all the NVSFFs, and the backup mode is made active (i.e. $SAV = 1$) for m cycles. Next, a restore is performed, and the data is scanned out. If there are no differences between the data scanned in and scanned out, then m cycles was sufficient. Otherwise m is decremented, the procedure is repeated. If on some iteration, the scanned out values differ from the scanned in values, then the number of cycles was not sufficient. If this happens on the first iteration, where $m = m_0$, then this chip is considered to have not met the yield criterion and deemed to have failed. On the other hand if the error appears on some value of m other than the first, then the previous iteration succeeded, and the minimum backup time is $\tau^* = (m + 1)T_{CK}$.

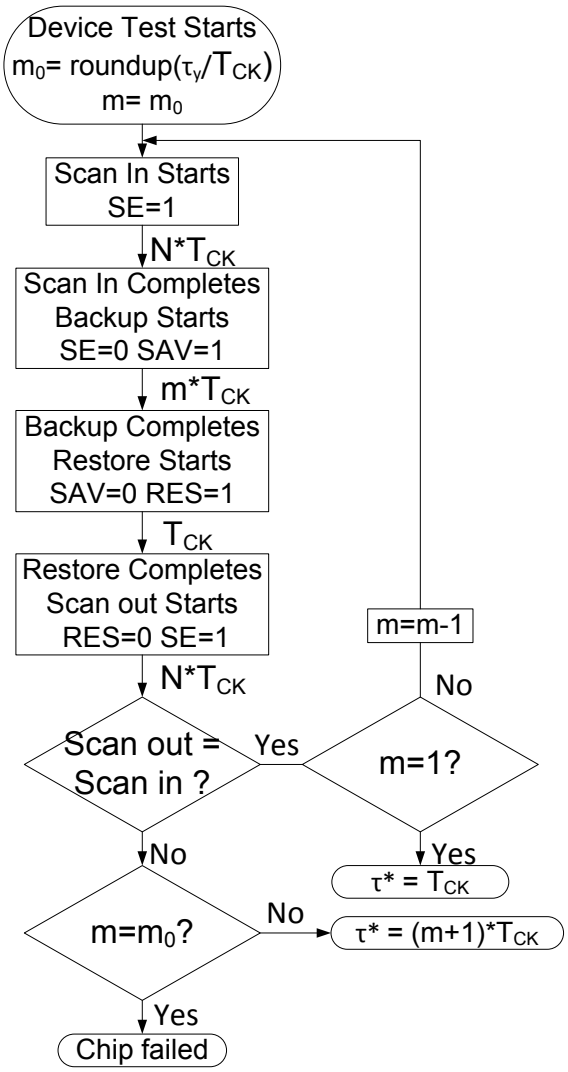


Figure 5.15: Non-volatile scan test procedure. N is number of flipflops in design.

Fig. (5.16) shows the energy expenditure using the two different backup times – one using τ_y which is termed as GBT for *global backup time*, and the other using τ^* which is termed as PFT for *post-fabrication tuning*. The savings in energy using PFT for a yield of 98% is nearly 80% compared to using GBT. Note that τ^* was computed using Procedure EoptDriverSizeWPR with τ_y in line 15 being replaced by τ_j, and updating E_j only if $\tau_j \leq \tau_y$.

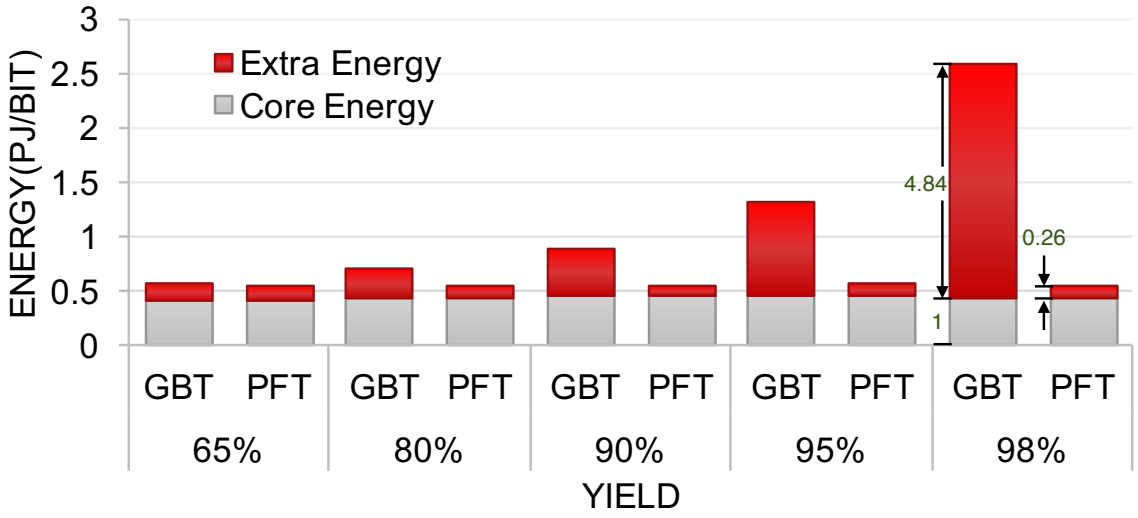


Figure 5.16: GBT: Single, global backup time, PFT: post-fabrication tuning. Core energy is the same for GBT and PFT. For achieving high yield, the energy wastage with PFT is much less than GBT.

5.3.7 Robustness of the Restore Operation

The focus of this chapter has been on the energy efficiency and robustness of the write or backup operation, because it results in greater power consumption than the read or restore operation. To read the state of a NVFF the data has to be sensed and compared with a reference. Hence process variations can result in a failure of this operation as well. In a NVFF the read circuitry is independent of the driver circuit used for the write operation. Consequently, techniques used to improve the robustness of the NVFF read operation by device parameter optimization Zhang *et al.* (2011) or by introducing redundancy Bishnoi *et al.* (2017) can be directly applied to the proposed NVFF design. Note that the proposed post-fabrication tuning method shown in Fig. 5.15 verifies that both the backup and restore operations are successful as it searches from the smallest backup time.

All the components excluding backup and restore circuits are standard CMOS logic blocks, and hence are subject to process variations. Their yields are generally orders of magnitude higher than the STT-MTJ and other emerging devices. Con-

sequently, reduction of yield in the CMOS blocks due to process variations was not considered.

5.4 Experimental Results

This section contains simulation based evaluations of the proposed NVFF circuits as well as the results on a larger design incorporating the NVSFFs. The circuits were designed using a commercial PDK for 40nm GP process. Other standard cells in 40nm were used in circuit automated synthesis. The power and delay values were obtained using HSPICE.

5.4.1 STT-MTJ Cell

The device simulations are based on the models in (Zhang *et al.* (2015); Munira *et al.* (2012)). The STT-MTJ has a square shape top view with both width and length equal to 40 nm. Other parameters are shown in Table 5.1. As t_{ox} is the most significant factor on energy consumption, to simplify the analysis, perturbations in t_{ox} are assumed to be Gaussian. To study the impact of the variations in t_{ox} on the resistances of the MTJ, 10,000 Monte Carlo simulations were performed with the mean μ_{tox} and sigma σ_{tox} of t_{ox} set to $.8nm$ and 10% of mean according to the study in Ref. Raychowdhury *et al.* (2009). Other physical parameters remained constant. Fig. 5.6 shows the distribution of $R_L(0)$ and $R_H(0)$. The mean and sigma of the resistances are summarized in Table 5.2. $I_{c,01}$ is $78.71\mu A$ and $I_{c,10}$ is $27.77\mu A$. If a single power supply is used in NVSFF design, the maximum voltage drop cross MTJ could not exceed than its V_{dd}, which is $0.9V$ in used 40nm technology. Therefore, the maximum resistance can be calculated as

$$R_{H,max} = V_{dd}/I_{c,10} = 32.4k\Omega,$$
$$R_{L,max} = V_{dd}/I_{c,01} = 11.43k\Omega.$$

Table 5.2 shows the mean and standard deviation of resistances for two different mean values of t_{ox}. A smaller t_{ox} is preferred to ensure that the 3σ of R_L and R_H are below the maximum resistances dictated by the power supply. Based on Table 5.2, $\mu_{tox} = 0.8nm$ and $\sigma_{tox}/\mu_{tox} = 0.1$ is assumed.

Table 5.1: STT-MTJ parameters.

Parameter	Value
MgO thickness(μ)	0.8 nm, 0.85 nm
Free layer thickness	1.3 nm
Area	40 $nm \times$ 40 nm
Resistance area product)	5 $\Omega \cdot \mu m^2$
TMR at zero bias	150 %
STD of variation(σ)	3%, 5%, 10 %
MonteCarlo cases	10000

Table 5.2: Mean and standard deviations of STT-MTJ resistances versus t_{ox}. The mean of random variable t_{ox} is set to two values, $.85nm$ and $.8nm$, with sigma equal to 3%, 5% and 10% of $\bar{t}_{ox}$.

μ_{tox} (nm)	σ_{tox} (%)	μ_{RH} ($k\Omega$)	σ_{RH} ($k\Omega$)	μ_{RL} ($k\Omega$)	σ_{RL} ($k\Omega$)
0.85	10	9.59	6.91	3.84	2.76
	5	8.23	2.74	3.29	1.09
	3	7.96	1.57	3.18	0.62
0.8	10	6.39	4.33	2.56	1.73
	5	5.57	1.75	2.23	0.70
	3	5.41	1.01	2.16	0.40

5.4.2 Performance Evaluation of Proposed NVSFFs

Table 5.3 shows the delay and the energy delay product of the two NVSFF as well as a volatile master-slave scan flipflop (SFF-MS) designs. The setup time (T_{setup}) of the NVSFF-DM is negative, in contrast to the positive setup time of the NVSFF-MS. Hence the total delay of NVSFF-DM is less than that of a NVSFF-MS. Compared to the NVSFF-MS, the average energy consumption (measured with 30% input switching activity) is higher in NVSFF-DM, but the EDP is similar due to the lower total delay of the NVSFF-DM. The total delay of SFF-MS is between the two NVSFFs, but its energy and EDP are much less than both NVSFFs. The area overhead of the NVSU in the NVSFF-DM and NVSFF-MS, makes their size about twice that of the SFF-MS. However, this does not translate to a similar increase in area of a whole circuit with either of the NVSFF cells (see Table 5.7).

Table 5.3: Performance of NVSFF-MS, NVSFF-DM and SFF-MS. The average energy is based on 30% input switching activity. Simulation conditions are: $25°C$, 0.9V, TT corner, and output load of $3fF$.

	T_{C2Q} (ps)	T_{setup} (ps)	T_{total} (ps)	Energy (fJ/cyc)	EDP (fJ·ps)
NVSFF-MS	60.28	6.90	67.18	4.10	275.56
NVSFF-DM	46.99	-2.99	44.00	5.99	263.51
SFF-MS	38.08	16.74	54.82	2.218	121.59

A reference MTJ (STT_ref) is required in the state sense amplifier (see Fig. (5.11)). The resistance of STT_ref is between R_H and R_L. Since the state recovery is implemented by the sensing current flow, R_{ref} is set to be harmonic mean of R_H and R_L. $1/R_{ref} = 2(1/R_H + 1/R_L)$. The resistance of STT_ref is achieved by changing the dimension of the MTJ to $55nm \times 50nm$, and R_{ref} is $3.09k\Omega$. The recovery time of two designs are shown in Table 5.4. In this work, global perturbations in t_{ox} are the most significant source of variations in the device resistances. Therefore, relative

differences between R_{ref} and R_H/R_L would remain constant on a die.

Table 5.4: Delay and energy consumption of restore from NVSU to output Q.

	Recover '0'		Recover '1'	
	Delay (ps)	Energy (fJ/bit)	Delay (ps)	Energy (fJ/bit)
NVSFF-MS	107.7	15.92	142.3	13.5
NVSFF-DM	83.87	17.75	82.84	19.41

Table 5.5 shows a comparison of NVSFF-MS and NVSFF-DM with published data on two other designs. The setup time and delay of the sense amplifier based NVFF (SA-MFF) in Ref. Cai *et al.* (2015) are similar to the NVSFF-DM. Although the forward body bias feature of FDSOI can improve the energy delay product, the SA-MFF uses a fixed write pulse for backup, which has a significantly high failure rate (24.6%) due to MTJ variations. The NVFF in Ryu *et al.* (2012) has a large positive setup time, and exhibits a DC current during a read operation.

Table 5.5: Comparison of non-volatile flipflop with prior reported data. a) Ref. Cai *et al.* (2015), b) Ref. Ryu *et al.* (2012).

	NVSFF-MS	NVSFF-DM	(a)	(b)
Technology	40nm	40 nm	28 nm FDSOI	45 nm
T_{setup}	6.9 ps	-3.0 ps	-4.9ps	75.2 ps
T_{C2Q}	60.3 ps	47.0 ps	50.1	203.3 ps
Backup time	Tunable		Fixed	Fixed
Backup energy	504fJ/bit		N/A	N/A
Restore time	142.3ps	83.9ps	N/A	2.01 ns
Restore energy	15.92 fJ/bit	19.41 fJ/bit	N/A	170.9 fJ/bit

Table 5.6 shows the energy consumption of NVSU during the backup mode. Three driver sizes were examined to evaluate their effects on the energy consumption. The driver sizes were determined based on method described in Section 5.2. Ignoring variations, the minimum energy is achieved with the largest driver size (107.5). When both CMOS and MTJ variations are included, the single global backup time $\tau_{97} = 14.6ns$, whereas the chip-specific backup times ranged from $1.96ns$ to $12.84ns$ (over 10K samples). However the energy expenditure of the former was more than $3.5X$ than the latter. Moreover the sizing and PFT approach results in an energy expenditure that is close to the ideal case with no variations.

Table 5.6: Comparison of backup schemes. (a) and (b) use single backup time for all dice, and (c) refers to chip-specific backup time. (b) and (c) include variations in both CMOS and MTJ.

	Yield	Driver Size	τ (ns)	Energy (pJ/bit)
(A) No Variation	100%	107.5	2.17	0.367
(B) Global Backup Time	97%	20.9	14.6	1.811
(C) Post Fab. Tuning	97%	32.8	1.96-12.84	0.504

5.4.3 Performance Evaluation of Circuits

Both NVSFFs are characterized using a standard characterization tool. To demonstrate the performance impact of NVSFFs on larger circuits, two circuits, an 8-bit multiply-and-accumulate (MAC) unit, and a 32-bit adder were synthesized using the two different NVSFFs and a SFF-MS.

MAC unit: The circuit structure is shown in Fig. 5.17. The MAC unit was synthesized using Genus from Cadence, with two different combinations of standard cells: (1) standard logic with NVSFF-MSs, (2) standard logic with NVSFF-DMs. Note that the total number of flipflops (16 input and output) in both designs is the same,

and both were synthesized for the same target clock period of $1.835ns$.

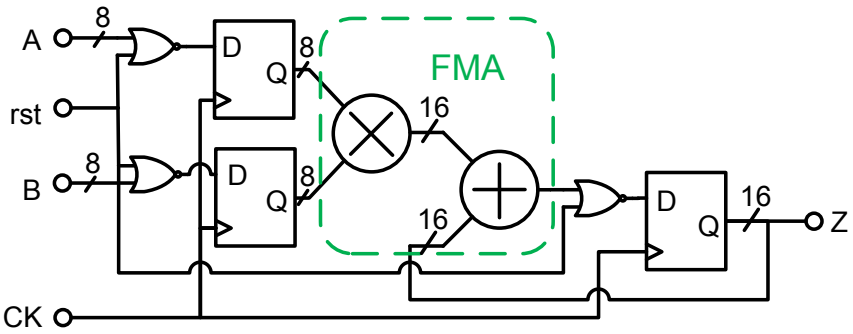


Figure 5.17: 8-bit MAC unit. It includes input and output flipflops, a synchronous reset and FMA (fused multiply-add) unit.

Table 5.7 shows of the results of the synthesis. The column *Cell Count* indicates the total number of standard cells. The designs with NVSFF-DMs have 11.6% fewer cell counts and 16% less area compared with the one with NVSFF-MSs. Even though NVSFF-DM consumes greater power, its smaller (negative) setup time allows the synthesis tool to reduce the logic cone driving the flipflop to a greater degree than in the case of the NVSFF-MS.

Power estimation was done by PTPX from Synopsys, using the library characterized data. Input sequences with 10%, 20% and 30% switching activities were supplied to the circuit. The average energy was measured by averaging the energy consumption across more than 100 cycles. Fig. (5.18) shows the total energy consumption of the two circuits versus input switching activity. The NV-MAC unit with NVSFF-DM consumed about 18.7%, 18.9% and 19% less energy than the NV-MAC unit with NVSFF-MS. As with delay (see Table 5.3), both area and energy consumption of the MAC with SFF-MS are between those with NVSFF-DM and NVSFF-MS.

32-bit adder: Two 32-bit adders are designed and synthesized in the same way as the MAC unit. There are 97 flipflops in the design. The synthesized results are shown in Table 5.7. The design with NVSFF-DMs has only 3.5% fewer cells and 7% smaller area than the one with NVSFF-MSs. The energy consumption with three switching

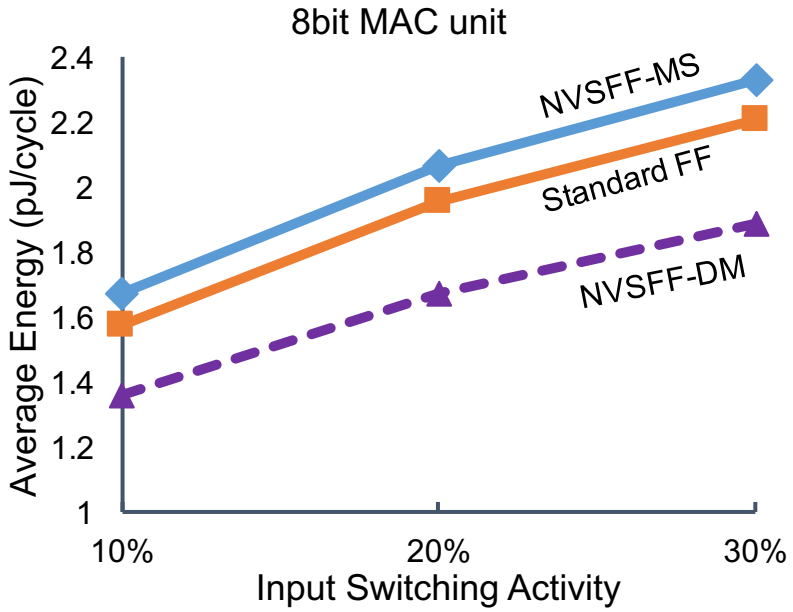


Figure 5.18: MAC total energy vs input switching activity under normal operation. The simulation is done by PTPX under 25°C typical corner.

activities are also very close, about 0.9%, 5.8% and 7.2% fewer on NVSFF-DMs, shown in Fig. 5.19. Compared with the MAC unit, the 32-bit adder has fewer logic cells and more flipflops. The NVSFF-DM has lower total delay (setup plus clock-to-Q) but slightly higher power consumption than the NVSFF-MS. The reduced delay allows synthesis tools to absorb the extra slack by reducing the size of the logic cone driving the flipflop. Note that for the 32-bit adder, the reduction in the size of its logic cones when using NVSFF-DM was not sufficient to compensate for it larger power consumption due to its greater number of flipflops. Since SFF-MS is smaller than NVSFFs, the total area of the adder with SFF-MS is 10.4% and 16.6% smaller than the one with NVSFF-DMs and NVSFF-MSs, respectively.

5.5 Non-Volatile Majority Flipflop (NV-MJFF)

Single input NVFF-DM can be extended to multi-input flip-flops that embed a more complex function than the simple identity function. One such class of functions

Table 5.7: Comparison of logic cell count and area using different flipflops in MAC and adder.

	MAC unit		32-bit Adder	
Flipflop Type	Cell Count	Area (μm^2)	Cell Count	Area (μm^2)
NVSFF-MS	603	3040	482	2517
NVSFF-DM	533	2555	465	2342
SFF-MS	580	2795	477	2098

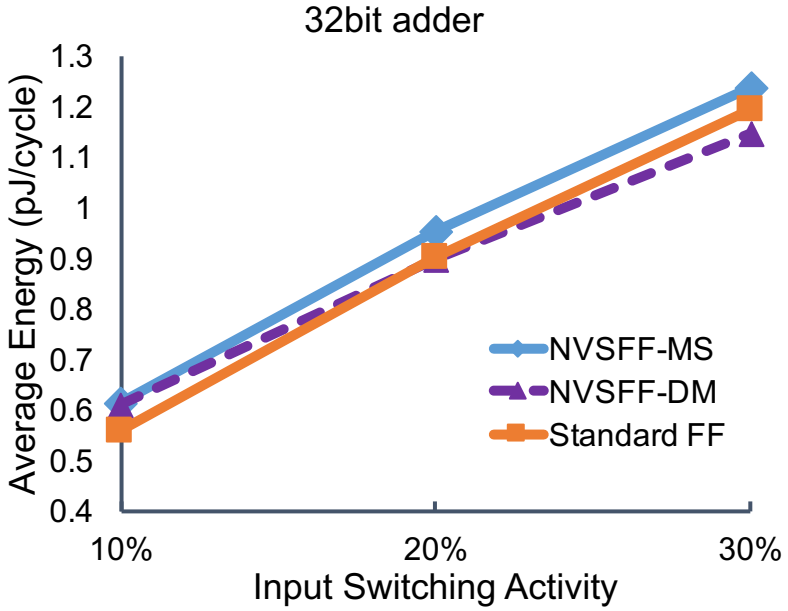


Figure 5.19: 32-bit adder total energy vs input switching activity under normal operation. The simulation is done by PTPX under 25°C typical corner.

is the majority function. Majority/minority functions are ubiquitous in arithmetic circuits as well as general combinational logic blocks Vemuru *et al.* (2014), and there is a substantial body of work on synthesis of majority/minority networks Vemuru *et al.* (2014); Amar *et al.* (2014).

5.5.1 NV-MJFF Structure

The circuit shown in Fig. 5.20, referred to as an NV-MJFF, is a non-volatile edge-triggered flipflop which computes a three input majority function $f(a, b, c) = ab + ac + bc$.

NV-MJFF is a non-volatile extension of PNAND-3. Its non-volatile operation (backup and restore) are similar as NVSFF-DM in previous section. Assume that two or more of the inputs a, b, c, are high. Then the LIN (*left input network*) will have at least two active devices and the RIN (*right input network*) will have at most one active device. As a result, the conductance of the LIN is higher than that of the RIN. On the rising edge of the clock, the sense amplifier sets $N2$ to 1 and $N1$ to 0. This corresponds to $f(a, b, c) = 1$. As the circuit and its operation are symmetric, if only one input is high, then the evaluation will result in $(N1, N2) = (1, 0)$, which corresponds to $f(a, b, c) = 0$. Thus the circuit computes a 2-out-3 majority function.

5.5.2 Performance Evaluation

NV-MJFF, NVFF-DM(no scan) and NVFF-MS(no scan) were designed using a commercial PDK for 65nm LP process. Other standard cells in 65nm were used in circuit automated synthesis. The power and delay values were obtained using HSPICE.

Table 5.8 shows the delay and the energy delay product of the three NVFF designs. The CLK-to-Q delay (T_{C2Q}) of NV-MJFF is larger than that of the NVFF-MS and NVFF-DM due to the fact that it also computes a majority function. The setup times (T_{setup}) of the NVFF-DM and NV-MJFF were negative, in contrast to the positive setup time of the NVFF-MS.Similar as designs in 40nm, the total delay of NVFF-DM is smaller. Compared to the NVFF-MS, the average energy consumption (measured

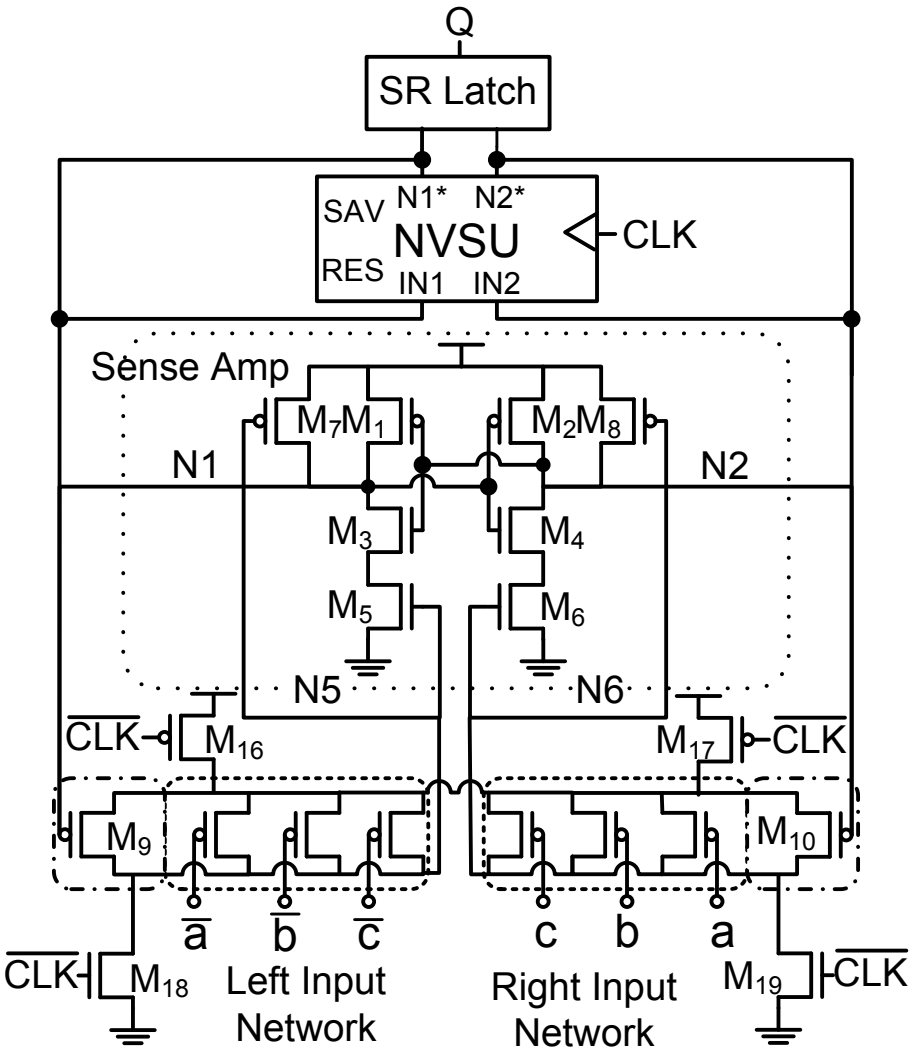


Figure 5.20: Structure of NV-MJFF

with 30% input switching activity) was similar in NVFF-DM, but the EDP was lower due to the much lower total delay of the NVFF-DM. NV-MJFF (including 3 inverters of the inputs) is functionally equivalent to a 3-input majority circuit driving a NVFF-MS. The EDP of the NV-MJFF, including the three inverters is 8231.8 fJ·ps, whereas the EDP of the equivalent majority circuit driving a NVFF-MS is 1.015 pJ·ps - which is a 18.9% reduction.

Similar as previous experiments in 40nm, the MAC unit shown in Fig. 5.17 was

Table 5.8: The performance comparison between NVFF-MS and NVFF-DMs. The average energy is based on 30% input switching activity. The simulations is done under $105°C$, 1.1V, SS corner. Output load is set to $20fF$.

	T_{C2Q} (ps)	T_{setup} (ps)	T_{total} (ps)	Energy (fJ/cyc)	EDP (fJ·ps)
NV-MSFF	282.4	58.11	340.51	18.17	6187.1
NV-DMFF	285.2	-20.57	264.69	16.49	4364.2
NV-MJFF	315.4	-53.56	261.84	30.88	8084.8

synthesized using three different combinations of standard cells: (1) standard logic with NVFF-MSs, (2) standard logic with NVFF-DMs, (3) standard logic with NVFF-DMs and NV-MJFFs. Note that the total number of flipflops in all three designs is the same.

The third design requires simplified hybridization. Specifically, if there was a flipflop driven by a three input majority function, then both the flipflop and the majority function logic can be replaced by a NV-MJFF. This logic absorption was performed automatically, but the details of the absorption algorithm are beyond the scope of this paper. Those DFFs that were not replaced with a NV-MJFF, were replaced by NVFF-DMs.

The three circuits were synthesized with the same target clock period $1.76ns$. Table 5.9 shows the statistics. The row *Comb. Cell Count* indicates the total number of standard cells excluding the flipflops since all designs have the 32 flipflops. The design with NVFF-DMs have 12.3% fewer cell counts and 14% less area compared with the one with NVFF-MSs. Keep in mind that NVFF-MSs also consume higher energy during backup. What is more interesting is the design that had both NVFF-DM and NV-MJFF. The logic absorption resulted in a significant reduction in cell count and total area. There were 8 NV-MJFF cells introduced in the circuit. The

remaining 24 flipflops were NVFF-DMs. Because of logic absorption, the 8 NV-MJFF cells created positive timing slacks on the critical paths. A re-synthesis step exploited this and further reduced the size of logic cells on these paths. The non-volatile MAC unit synthesized with NVFF-DMs and NV-MJFFs resulted in a 22.2% reduction in cell count and a 22.3% reduction in area compared with the one synthesized with NVFF-MSs.

Table 5.9: MAC combinational cell count and area comparison. 32 flipflops are not included in cell count. 8 NV-MJFF are included in the third circuit.

Flipflop Type	NVFF-MS	NVFF-DM	NVFF-DM/MJFF
Comb. Cell Count	528	463	411
Total Area (μm^2)	3494.4	3005.1	2716.0

HSPICE simulation was used to evaluate the energy consumption. Input sequences with 10%, 20% and 30% switching activities were supplied to the circuit. The average energy was measured by averaging the energy consumption across more than 100 cycles. Fig. 5.21 shows the energy breakdown between combinational logic and flipflops. The energy consumption of the logic feeders reduced substantially because of the negative setup time and logic absorption. Fig. 5.22 shows the total energy consumption of three circuits versus input switching activity. The NV-MAC unit with NVFF-DM consumed about 15.1%, 14.8% and 14.7% less energy than the NV-MAC unit with NVFF-MS. With the logic absorption, the NV-MAC with both NVFF-DM and NV-MJFF consumed 21.8%, 22.5% and 23.1% less energy than the same design synthesized with NVFF-MS.

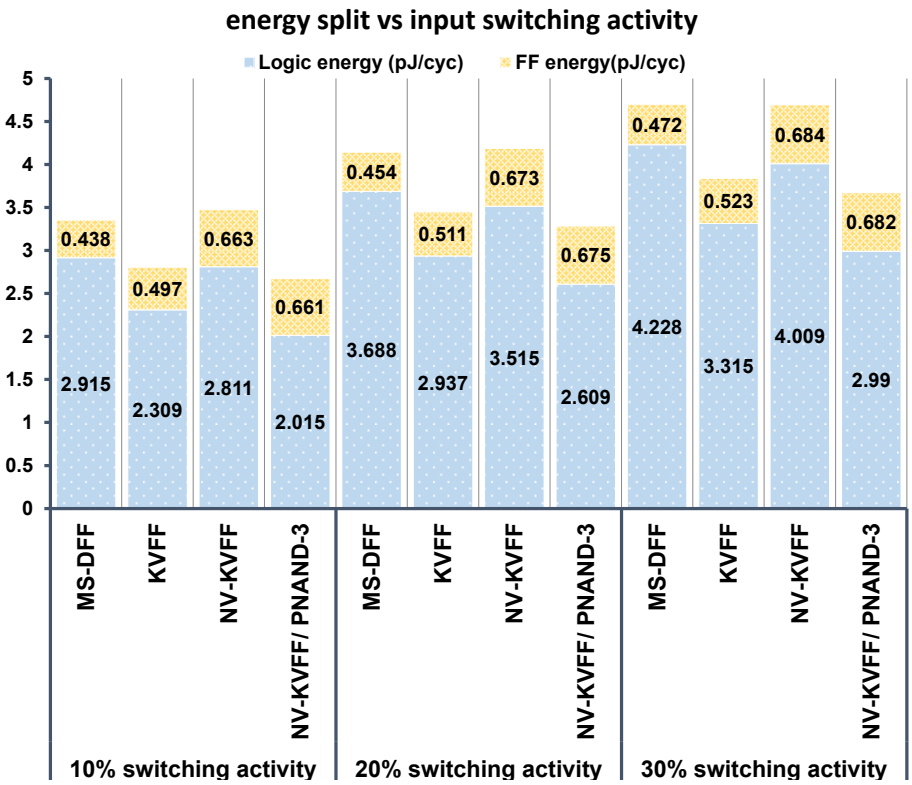


Figure 5.21: MAC energy breakdown vs input switching activity, the simulations is done under 25°C, 1.2V, TT corner. Numbers in *italics* denote totals.

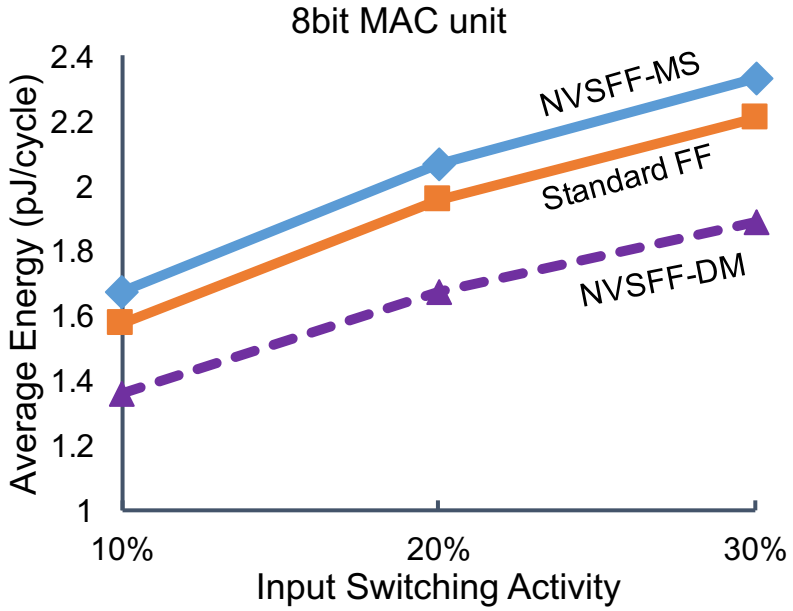


Figure 5.22: MAC total energy vs input switching activity under normal operation. The simulation is done by HSPICE under 25°C typical corner

Chapter 6

CONCLUSIONS AND FUTURE WORK

Threshold logic based synthesis is proved in theory Muroga (1971) to be more efficient on gate counts and logic depth comparing with CMOS logic synthesis. Threshold logic gates are assumed to have unlimited fan-ins and similar delay and power comparing with simple CMOS logic gate. However, the performance of threshold logic gate in real life are constrained by multiple factors, including limit fan-ins and weights, high power consumption, high area, poor robustness with process variation and poor in technology scaling. It also lack a mature digital design flow considering these gate limitations. This work is trying to address some of the critical issues in threshold logic gate design. The PNANDs proposed in this work is believed to have a very good balance among area, power, delay and robustness. To verify the gate performance and PNAND based threshold logic synthesis flow, two multipliers and cell arrays are fabricated in two chips. The silicon results show consistent area (24%, 34%) and power (33%, 30%) improvement on hybrid multiplier comparing with the functional identical multiplier designed by conventional digital flow. To the best of our knowledge, the two chips are the first silicon implementations for threshold logic based design.

Conventional differential mode circuits are not robust at low voltage. Direct applying low supply voltage to PNAND would lead to high failures. Chapter 4 have shown how we can circumvent this problem by integrating threshold gates known as TLLs with emerging RRAM memory technology and improved the robustness of the TLLs at low voltages. The performance of TLLs are not affected by this robust enhance technique. Hybrid circuit using the proposed TLLs can result in significant

improvement in area and energy-delay product.

NVL with almost instant backup and restore operations has gained great attention due to its applicability for systems that are powered by harvested energy. The key components in NVL are the flipflops that represent the state of the system at any given time. For near instantaneous backup and restoration of the state, it is best to enhance the flipflops with NV storage. The optimal design of the driver circuit to save the state in a NV device is critically important for energy efficiency, and robustness due to process variations. Chapter 5 presented a systematic approach to the energy optimal design of the backup driver and the determination of the corresponding backup subject to satisfying a yield constraint. To further reduce energy wastage, a novel method is presented that adjusts the backup time on a per-chip basis, after fabrication. This substantially reduces the energy wasted when compared to using a single backup time for all chips. Also included is the design of NVFFs that enables the post- fabrication tuning of the backup time through the use of a scan mechanism. Significant energy reduction with post fabrication tuning is demonstrated both in theory and in two circuit implementations, a 32-bit adder and a 8-bit MAC unit. The proposed methodology allows conversion of any ASIC design to one that is completely non-volatile using commercial synthesis flows. NVFF can also be extended to non-volatile threshold logic gate. A majority gate NV-MJFF was designed to demonstrate the advantages from threshold logic. The simulation results of non-volatile MAC unit using both NV-MJFF and NVFF-DM show that it can achieve same performance as conventional volatile design but with smaller area and lower energy consumption.

Several future works are listed to further explore the potential of combining non-volatile memory and threshold logic. Since extensive amount of works have been done on NVM, it will be valuable to explore non-volatile threshold gates using different NV-devices. For example, the resistance of DWM device can be tuned continuously

Sengupta *et al.* (2016). This device can be used to apply process compensation on low voltage operation. Multi-bit NVM devices can also be used to store more than one state. Therefore multiple check points can be made for system recovery.

Non-volatile TLGs can be extended to a FPGA structure. The primary work was published in Kulkarni *et al.* (2014). A customized layout can be used to maximize the performance. Implementing non-volatile threshold gate in FPGA can reduce leakage on the tiles that are not been used.

NVL can also be implemented in neural networks ASIC implementations like CNN or RNN. These applications have high volume data transfer and storage. In an interrupting power supply scenario, store data fast, secure and with low energy consumption is important for these application. Threshold logic can also be implemented as it would reduce area and power consumption without alternate the circuit functionality.

REFERENCES

Abbas, Z. and M. Olivieri, "Impact of technology scaling on leakage power in nano-scale bulk cmos digital standard cells", Microelectronics Journal **45**, 2, 179 – 195, URL http://www.sciencedirect.com/science/article/pii/S002626921300253X (2014). (document), 1.1, 1.1

Akers, S. B., "Synthesis of combinational logic using three-input majority gates", in "3rd Annual Symposium on Switching Circuit Theory and Logical Design (SWCT 1962)", pp. 149–158 (1962). 2.2

Amar, L., P. E. Gaillardon and G. D. Micheli, "Majority-inverter graph: A novel data-structure and algorithms for efficient logic optimization", in "2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC)", pp. 1–6 (2014). 5.5

Annampedu, V. and M. D. Wagh, "Decomposition of threshold functions into bounded fan-in threshold functions", Information and Computation **227**, Supplement C, 84 – 101, URL http://www.sciencedirect.com/science/article/pii/S0890540113000503 (2013). 2.2

Avedillo, M. J., J. M. Quintana, A. Rueda and E. Jimenez, "Low-power cmos threshold-logic gate", Electronics Letters **31**, 25, 2157–2159 (1995). 2.1

Balsamo, D., A. S. Weddell, G. V. Merrett, B. M. Al-Hashimi, D. Brunelli and L. Benini, "Hibernus: Sustaining computation during intermittent supply for energy-harvesting systems", IEEE Embedded Systems Letters **7**, 1, 15–18 (2015). 3, 5.3.3

Beiu, V., "Logic gate having reduced power dissipation and method of operation thereof", URL http://www.google.tl/patents/US6259275, uS Patent 6,259,275 (2001). 2.1

Beiu, V., J. M. Quintana and M. J. Avedillo, "VLSI Implementations of Threshold Logic - A Comprehensive Survey", IEEE Trans. Neural Networks **14**, 1217–1243 (2003). 2.1, 2.2

Bishnoi, R., M. Ebrahimi, F. Oboril and M. B. Tahoori, "Read disturb fault detection in STT-MRAM", in "2014 International Test Conference", pp. 1–7 (2014). 2.3

Bishnoi, R., M. Ebrahimi, F. Oboril and M. B. Tahoori, "Improving write performance for STT-MRAM", IEEE Transactions on Magnetics **52**, 8, 1–11 (2016a). 2.3, 5.2.1, 5.3.1

Bishnoi, R., F. Oboril and M. B. Tahoori, "Non-volatile non-shadow flip-flop using spin orbit torque for efficient normally-off computing", in "2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)", pp. 769–774 (2016b). 2.3, 5.1

Bishnoi, R., F. Oboril and M. B. Tahoori, "Design of defect and fault-tolerant non-volatile spintronic flip-flops", IEEE Transactions on Very Large Scale Integration (VLSI) Systems **25**, 4, 1421–1432 (2017). 2.3, 5.1, 5.3.7

Cai, H., Y. Wang, W. Zhao and L. A. de Barros Naviner, "Multiplexing sense-amplifier-based magnetic flip-flop in a 28-nm FDSOI technology", IEEE Transactions on Nanotechnology **14**, 4, 761–767 (2015). (document), 5.1, 5.2, 5.4.2, 5.5

Chen, B., F. Cai, J. Zhou, W. Ma, P. Sheridan and W. D. Lu, "Efficient in-memory computing architecture based on crossbar arrays", in "2015 IEEE International Electron Devices Meeting (IEDM)", pp. 17.5.1–17.5.4 (2015). 2.4

Currivan, J. A., Y. Jang, M. D. Mascaro, M. A. Baldo and C. A. Ross, "Low energy magnetic domain wall logic in short, narrow, ferromagnetic wires", IEEE Magnetics Letters **3**, 3000104–3000104 (2012). 2.4

Currivan-Incorvia, J. A., S. Siddiqui, S. Dutta, E. R. Evarts, J. Zhang, D. Bono, C. A. Ross and M. A. Baldo, "Logic circuit prototypes for three-terminal magnetic tunnel junctions with mobile domain walls", Nat Commun **7**, article (2016). 7, 2.4

Deng, Y., P. Huang, B. Chen, X. Yang, B. Gao, J. Wang, L. Zeng, G. Du, J. Kang and X. Liu, "Rram crossbar array with cell selection device: A device and circuit interaction study", Electron Devices, IEEE Transactions on **60**, 2, 719–726 (2013). 4.3.3

Endoh, T., H. Koike, S. Ikeda, T. Hanyu and H. Ohno, "An overview of nonvolatile emerging memories spintronics for working memories", IEEE Journal on Emerging and Selected Topics in Circuits and Systems **6**, 2, 109–119 (2016). 5

Fan, D., "Ultra-low energy reconfigurable spintronic threshold logic gate", in "2016 International Great Lakes Symposium on VLSI (GLSVLSI)", pp. 385–388 (2016). 2.1

Fang, Z., H. Yu, X. Li, N. Singh, G. Q. Lo and D. L. Kwong, "$HfO_x/TiO_x/HfO_x/TiO_x$ multilayer-based forming-free rram devices with excellent uniformity", Electron Device Letters, IEEE **32**, 4, 566–568 (2011). 4.3.2, 4.3.3

Fong, X., S. H. Choday and K. Roy, "Bit-cell level optimization for non-volatile memories using magnetic tunnel junctions and spin-transfer torque switching", IEEE Transactions on Nanotechnology **11**, 1, 172–181 (2012). 6

Friedman, J. S., A. Godkin, A. Henning, Y. Vaknin, Y. Rosenwaks and A. V. Sahakian, "Threshold logic with electrostatically formed nanowires", IEEE Transactions on Electron Devices **63**, 3, 1388–1391 (2016). 2.1

Garello, K., C. O. Avci, I. M. Miron, M. Baumgartner, A. Ghosh, S. Auffret, O. Boulle, G. Gaudin and P. Gambardella, "Ultrafast magnetization switching by spin-orbit torques", Applied physics letters **105**, 21, 212402– (2014). 2.3

Gowda, T., S. Vrudhula and G. Konjevod, "Combinational equivalence checking for threshold logic circuits", in "Proceedings of the 17th ACM Great Lakes Symposium on VLSI", GLSVLSI '07, pp. 102–107 (ACM, New York, NY, USA, 2007), URL http://doi.acm.org/10.1145/1228784.1228813. 2.2

Grupp, L. M., J. D. Davis and S. Swanson, "The bleak future of nand flash memory", in "Proceedings of the 10th USENIX Conference on File and Storage Technologies", FAST'12, pp. 2–2 (USENIX Association, Berkeley, CA, USA, 2012), URL http://dl.acm.org/citation.cfm?id=2208461.2208463. 1

Guan, X., S. Yu and H.-S. Wong, "A spice compact model of metal oxide resistive switching memory with variations", Electron Device Letters, IEEE **33**, 10, 1405–1407 (2012). 4.3.3

Halawani, Y., B. Mohammad, D. Homouz, M. Al-Qutayri and H. Saleh, "Modeling and optimization of memristor and STT-RAM-based memory for low-power applications", IEEE Transactions on Very Large Scale Integration (VLSI) Systems **24**, 3, 1003–1014 (2016). 5.3.1

Huang, K., R. Zhao and Y. Lian, "STT-MRAM based low power synchronous non-volatile logic with timing demultiplexing", in "2014 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)", pp. 31–36 (2014). 2.4

James, A. P., L. R. V. J. Francis and D. S. Kumar, "Resistive threshold logic", IEEE Transactions on Very Large Scale Integration (VLSI) Systems **22**, 1, 190–195 (2014). 2.1

Jayakumar, H., A. Raha and V. Raghunathan, "Quickrecall: A low overhead hw/sw approach for enabling computations across power cycles in transiently powered computers", in "2014 27th International Conference on VLSI Design and 2014 13th International Conference on Embedded Systems", pp. 330–335 (2014). 2

Kang, W., Y. Ran, W. Lv, Y. Zhang and W. Zhao, "High-speed, low-power, magnetic non-volatile flip-flop with voltage-controlled, magnetic anisotropy assistance", IEEE Magnetics Letters **7**, 1–5 (2016). 5.1

Kartschoke, P. and N. Rohrer, "Techniques for reduced power and increased speed in dynamic and ratio logic circuits", in "Proceedings of the 39th Midwest Symposium on Circuits and Systems", vol. 1, pp. 175–178 vol.1 (1996). 2.1

Khanna, S., S. C. Bartling, M. Clinton, S. Summerfelt, J. A. Rodriguez and H. P. McAdams, "An fram-based nonvolatile logic mcu soc exhibiting 100% digital state retention at $VDD = 0$v achieving zero leakage with $>$ 400-ns wakeup time for ulp applications", IEEE Journal of Solid-State Circuits **49**, 1, 95–106 (2014). 2.3, 5.1, 5.3.3

Kim, J.-C., Y.-C. Jang and H.-J. Park, "Cmos sense amplifier-based flip-flop with two n-c2mos output latches", Electronics Letters **36**, 6, 498–500 (2000). 3.1.1

Koga, M., M. Iida, M. Amagasaki, Y. Ichida, M. Saji, J. Iida and T. Sueyoshi, "First prototype of a genuine power-gatable reconfigurable logic chip with FeRAM cells", in "2010 International Conference on Field Programmable Logic and Applications", pp. 298–303 (2010). 2.3, 5.1

Kulkarni, N., N. Nukala and S. Vrudhula, "Minimizing area and power of sequential cmos circuits using threshold decomposition", in "2012 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)", pp. 605–612 (2012). 1.4, 3.3.1

Kulkarni, N., J. Yang, J. S. Seo and S. Vrudhula, "Reducing power, leakage, and area of standard-cell asics using threshold logic flip-flops", IEEE Transactions on Very Large Scale Integration (VLSI) Systems **24**, 9, 2873–2886 (2016). 1.4

Kulkarni, N., J. Yang and S. Vrudhula, "A fast, energy efficient, field programmable threshold-logic array", in "2014 International Conference on Field-Programmable Technology (FPT)", pp. 300–305 (2014). 6

Kuo, P. Y., C. Y. Wang and C. Y. Huang, "On rewiring and simplification for canonicity in threshold logic circuits", in "2011 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)", pp. 396–403 (2011). 2.2

Kwon, K. W., S. H. Choday, Y. Kim, X. Fong, S. P. Park and K. Roy, "SHE-NVFF: Spin hall effect-based nonvolatile flip-flop for power gating architecture", IEEE Electron Device Letters **35**, 4, 488–490 (2014). 2.3, 5.1

Lageweg, C., S. Cotofana and S. Vassiliadis, "A full adder implementation using set based linear threshold gates", in "9th International Conference on Electronics, Circuits and Systems", vol. 2, pp. 665–668 vol.2 (2002). 2.1

Lashevsky, R., K. Takaara and M. Souma, "Neuron mosfet as a way to design a threshold gates with the threshold and input weights alterable in real time", in "IEEE. APCCAS 1998. 1998 IEEE Asia-Pacific Conference on Circuits and Systems. Microelectronics and Integrating Systems. Proceedings (Cat. No.98EX242)", pp. 263–266 (1998). 2.1

Lin, C. C., C. Y. Wang, Y. C. Chen and C. Y. Huang, "Rewiring for threshold logic circuit minimization", in "2014 Design, Automation Test in Europe Conference Exhibition (DATE)", pp. 1–6 (2014). 2.2

López, J. A. H., J. G. Tejero, J. F. Ramos and A. G. Bohórquez, "New types of digital comparators", in "International Symposium on Circuits and Systems", vol. 1, pp. 29–32 (1995). 2.1

Luck, A., S. Jung, R. Brederlow, R. Thewes, K. Goser and W. Weber, "On the design robustness of threshold logic gates using multi-input floating gate mos transistors", IEEE Transactions on Electron Devices **47**, 6, 1231–1240 (2000). 2.1

Ma, K., Y. Zheng, S. Li, K. Swaminathan, X. Li, Y. Liu, J. Sampson, Y. Xie and V. Narayanan, "Architecture exploration for ambient energy harvesting nonvolatile processors", in "2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA)", pp. 526–537 (2015). 1.2, 5.3.3

Maezawa, K., T. Akeyoshi and T. Mizutani, "Functions and applications of monostable-bistable transition logic elements (mobile's) having multiple-input terminals", IEEE Transactions on Electron Devices **41**, 2, 148–154 (1994). 2.1

Mahalanabis, D., V. Bharadwaj, H. J. Barnaby, S. Vrudhula and M. N. Kozicki, "A nonvolatile sense amplifier flip-flop using programmable metallization cells", IEEE Journal on Emerging and Selected Topics in Circuits and Systems **5**, 2, 205–213 (2015). 2.3, 5.1

Mahalanabis, D., Y. Gonzalez-Velo, H. J. Barnaby, M. N. Kozicki, P. Dandamudi and S. Vrudhula, "Impedance measurement and characterization of ag-ge30se70-based programmable metallization cells", IEEE Transactions on Electron Devices **61**, 11, 3723–3730 (2014). 3

Miller, H. S. and R. O. Winder, "Majority-logic synthesis by geometric methods", IRE Transactions on Electronic Computers **EC-11**, 1, 89–90 (1962). 2.2

Miron, I. M., K. Garello, G. Gaudin, P.-J. Zermatten, M. V. Costache, S. Auffret, S. Bandiera, B. Rodmacq, A. Schuhl and P. Gambardella, "Perpendicular switching of a single ferromagnetic layer induced by in-plane current injection", Nature **476**, 7359, 189–193 (2011). 2.3

Mittal, S., J. S. Vetter and D. Li, "A survey of architectural approaches for managing embedded dram and non-volatile on-chip caches", IEEE Transactions on Parallel and Distributed Systems **26**, 6, 1524–1537 (2015). 2, 6

Motaman, S., S. Ghosh and N. Rathi, "Impact of process-variations in sttram and adaptive boosting for robustness", in "2015 Design, Automation Test in Europe Conference Exhibition (DATE)", pp. 1431–1436 (2015). 2.3, 5.3.1

Munira, K., W. H. Butler and A. W. Ghosh, "A quasi-analytical model for energy-delay-reliability tradeoff studies during write operations in a perpendicular STT-RAM cell", IEEE Transactions on Electron Devices **59**, 8, 2221–2226 (2012). 5.2.1, 5.2.3, 5.4.1

Muroga, S., "The principle of majority decision logical elements and the complexity of their circuits", in "IFIP Congress", pp. 400–406 (1959). 1.4

Muroga, S., *Threshold Logic and its Applications* (John Wiley, New York, NY, 1971). 1.4, 2.2, 6

Natsui, M., D. Suzuki, N. Sakimura, R. Nebashi, Y. Tsuji, A. Morioka, T. Sugibayashi, S. Miura, H. Honjo, K. Kinoshita, S. Ikeda, T. Endoh, H. Ohno and T. Hanyu, "Nonvolatile logic-in-memory array processor in 90nm MTJ/MOS achieving 75% leakage reduction using cycle-based power gating", in "2013 IEEE International Solid-State Circuits Conference Digest of Technical Papers", pp. 194–195 (2013). 2.4

Neutzling, A., M. G. A. Martins, R. P. Ribas and A. I. Reis, "Synthesis of threshold logic gates to nanoelectronics", in "2013 26th Symposium on Integrated Circuits and Systems Design (SBCCI)", pp. 1–6 (2013). 2.2

Nikolic, B., V. G. Oklobdzija, V. Stojanovic, W. Jia, J. K.-S. Chiu and M. Ming-Tak Leung, "Improved sense-amplifier-based flip-flop: design and measurements", Solid-State Circuits, IEEE Journal of **35**, 6, 876–884 (2000). 3.1.1

Nukala, N. S., N. Kulkarni and S. Vrudhula, "Spintronic threshold logic array (stla) - a compact, low leakage, non-volatile gate array architecture", in "2012 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)", pp. 188–195 (2012). 2.4

Ozdemir, H., A. Kepkep, B. Pamir, Y. Leblebici and U. Cilingiroglu, "A capacitive threshold-logic gate", IEEE Journal of Solid-State Circuits **31**, 8, 1141–1150 (1996). 2.1

Pacha, C., U. Auer, C. Burwick, P. Glosekotter, A. Brennemann, W. Prost, F. J. Tegude and K. F. Goser, "Threshold logic circuit design of parallel adders using resonant tunneling devices", IEEE Transactions on Very Large Scale Integration (VLSI) Systems **8**, 5, 558–572 (2000). 2.1

Padure, M., S. Cotofana, C. Dan, M. Bodea and S. Vassiliadis, "A new Latch-based Threshold Logic Family", in "CAS 2001 Proceedings. International Semiconductor Conference, 2001.", vol. 2, pp. 531 –534 vol.2 (2001). 2.1

Priya, S. and D. J. Inman, *Energy Harvesting Technologies* (Springer Publishing Company, Incorporated, 2008), 1st edn. 1.2, 5.1

Quintana, J. M., M. J. Avedillo, R. Jiménez and E. Rodríguez-Villegas, "Practical low-cost cpl implementations threshold logic functions", in "Proceedings of the 11th Great Lakes Symposium on VLSI", GLSVLSI '01, pp. 139–144 (ACM, New York, NY, USA, 2001), URL `http://doi.acm.org/10.1145/368122.368903`. 2.1

Ransford, B., J. Sorber and K. Fu, "Mementos: System support for long-running computation on rfid-scale devices", SIGARCH Comput. Archit. News **39**, 1, 159–170, URL `http://doi.acm.org/10.1145/1961295.1950386` (2011). 1

Raychowdhury, A., D. Somasekhar, T. Karnik and V. De, "Design space and scalability exploration of 1T-1STT MTJ memory arrays in the presence of variability and disturbances", in "2009 IEEE International Electron Devices Meeting (IEDM)", pp. 1–4 (2009). 5.2.3, 5.4.1

Rodriguez Arreola, A., D. Balsamo, A. K. Das, A. S. Weddell, D. Brunelli, B. M. Al-Hashimi and G. V. Merrett, "Approaches to transient computing for energy harvesting systems: A quantitative evaluation", in "Proceedings of the 3rd International Workshop on Energy Harvesting & Energy Neutral Sensing Systems", ENSsys '15, pp. 3–8 (ACM, New York, NY, USA, 2015), URL `http://doi.acm.org/10.1145/2820645.2820652`. 1.2

Ryu, K., J. Kim, J. Jung, J. P. Kim, S. H. Kang and S. O. Jung, "A magnetic tunnel junction based zero standby leakage current retention flip-flop", IEEE Transactions on Very Large Scale Integration (VLSI) Systems **20**, 11, 2044–2053 (2012). (document), 2.3, 5.1, 5.2, 5.4.2, 5.5

Samuel, L., B. Krzysztof and S. Vrudhula, "Design of a robust, high performance standard cell threshold logic family for deep sub-micron technology", in "Proceedings of the IEEE International Conference on Microelectronics", (Cairo, Egypt, 2010). 3.1.1, 4.3.1

Sengupta, A., Y. Shim and K. Roy, "Proposal for an all-spin artificial neural network: Emulating neural and synaptic functionalities through domain wall motion in ferromagnets", IEEE Transactions on Biomedical Circuits and Systems **PP**, 99, 1–9 (2016). 2.4, 6

Siu, K.-Y., V. Roychowdhury and T. Kailath, *Discrete Neural Computation: A Theoretical Foundation*, Information and Systems Sciences Series (Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1995). 1.4

Sobelman, G. E. and K. Fant, "Cmos circuit design of threshold gates with hysteresis", in "Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on", vol. 2, pp. 61–64 vol.2 (1998). 2.1

STMicroelectronics, "Fdsoi", `http://www.st.com/content/st_com/en/about/innovation---technology/FD-SOI.html` (2018). (document), 4.2.1, 4.2

Strandberg, R. and J. Yuan, ""Single input current-sensing differential logic (SCSDL)"", in "International Symposium on Circuits and Systems", vol. 1, pp. 764–767 (2000). 2.1

Strollo, A., D. De Caro, E. Napoli and N. Petra, "A novel high-speed sense-amplifier-based flip-flop", Very Large Scale Integration (VLSI) Systems, IEEE Transactions on **13**, 11, 1266–1274 (2005). 3.1.1

Sun, J. Z., R. P. Robertazzi, J. Nowak, P. L. Trouilloud, G. Hu, D. W. Abraham, M. C. Gaidis, S. L. Brown, E. J. O'Sullivan, W. J. Gallagher and D. C. Worledge, "Effect of subvolume excitation and spin-torque efficiency on magnetic switching", Phys. Rev. B **84**, 064413, URL `https://link.aps.org/doi/10.1103/PhysRevB.84.064413` (2011). 5.2.1

Tatapudi, S. and V. Beiu, "Split-precharge differential noise-immune threshold logic gate (spd-ntl)", in "Artificial Neural Nets Problem Solving Methods", edited by J. Mira and J. R. Álvarez, pp. 49–56 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2003). 2.1

Tougaw, P. D. and C. S. Lent, "Logical devices implemented using quantum cellular automata", Journal of Applied Physics **75**, 3, 1818–1825, URL `https://doi.org/10.1063/1.356375` (1994). 2.1

Valov, I., R. Waser, J. R. Jameson and M. N. Kozicki, "Electrochemical metallization memories – fundamentals, applications, prospects", Nanotechnology **22**, 25, 254003, URL `http://stacks.iop.org/0957-4484/22/i=25/a=254003` (2011). 3

Vemuru, S., P. Wang and M. Niamat, "Majority logic gate synthesis approaches for post-cmos logic circuits: A review", in "IEEE International Conference on Electro/Information Technology", pp. 284–289 (2014). 5.5

Wang, S., H. Lee, F. Ebrahimi, P. K. Amiri, K. L. Wang and P. Gupta, "Comparative evaluation of spin-transfer-torque and magnetoelectric random access memory", IEEE Journal on Emerging and Selected Topics in Circuits and Systems **6**, 2, 134–145 (2016). 2.3

Wang, Y., Y. Liu, S. Li, D. Zhang, B. Zhao, M. F. Chiang, Y. Yan, B. Sai and H. Yang, "A 3us wake-up time nonvolatile processor based on ferroelectric flip-flops", in "ESSCIRC (ESSCIRC), 2012 Proceedings of the", pp. 149–152 (2012). 2.3, 5.1

Wang, Y., Y. Zhang, E. Deng, J.-O. Klein, L. A. B. Naviner and W. Zhao, "Compact model of magnetic tunnel junction with stochastic spin transfer torque switching for reliability analyses", Microelectronics Reliability **54**, 1774–1778 (2014). (document), 5.2.1, 5.6

Weste, N. and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective* (Addison-Wesley Publishing Company, USA, 2010), 4th edn. 3.2.1

Wirnshofer, M., *Variation-Aware Adaptive Voltage Scaling for Digital CMOS Circuits*, chap. 2, Springer Series in Advanced Microelectronics (Springer Netherlands, 2013). 5.2.3

Wong, H. S. P., H.-Y. Lee, S. Yu, Y.-S. Chen, Y. Wu, P.-S. Chen, B. Lee, F. Chen and M.-J. Tsai, "Metal oxide rram", Proceedings of the IEEE **100**, 6, 1951–1970 (2012). 4.3.3

Wong, H. S. P., S. Raoux, S. Kim, J. Liang, J. P. Reifenberg, B. Rajendran, M. Asheghi and K. E. Goodson, "Phase change memory", Proceedings of the IEEE **98**, 12, 2201–2227 (2010). 4.3.3

Xu, N., J. Wang, Y. Lu, H. H. Park, B. Fu, R. Chen, W. Choi, D. Apalkov, S. Lee, S. Ahn, Y. Kim, Y. Nishizawa, K. H. Lee, Y. Park and E. S. Jung, "Physics-based compact modeling framework for state-of-the-art and emerging STT-MRAM technology", in "2015 IEEE International Electron Devices Meeting (IEDM)", pp. 28.5.1–28.5.4 (2015). 5.2.1

Yang, J., J. Davis, N. Kulkarni, J. s. Seo and S. Vrudhula, "Dynamic and leakage power reduction of asics using configurable threshold logic gates", in "Custom Integrated Circuits Conference (CICC), 2015 IEEE", pp. 1–4 (2015a). 1.4

Yang, J., N. Kulkarni, J. Davis and S. Vrudhula, "Fast and robust differential flipflops and their extension to multi-input threshold gates", in "2015 IEEE International Symposium on Circuits and Systems (ISCAS)", pp. 822–825 (2015b). 5.3.4

Yu, H. and Y. Wang, *Design Exploration of Emerging Nano-scale Non-volatile Memory* (Springer-Verlag New York, 2014), 1st edn. 2.3

Yu, S., Y. Wu and H.-S. Wong, "Investigating the switching dynamics and multilevel capability of bipolar metal oxide resistive switching memory", Applied Physics Letters **98**, 10, 103514–103514–3 (2011). 4.3.3

Zhang, R., P. Gupta and N. K. Jha, "Majority and minority network synthesis with application to qca-, set-, and tpl-based nanotechnologies", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **26**, 7, 1233–1245 (2007). 2.2

Zhang, R., P. Gupta, L. Zhong and N. K. Jha, "Threshold network synthesis and optimization and its application to nanotechnologies", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **24**, 1, 107–118 (2005). 2.2

Zhang, Y., X. Wang, H. Li and Y. Chen, "STT-RAM cell optimization considering MTJ and CMOS variations", IEEE Transactions on Magnetics **47**, 10, 2962–2965 (2011). 5.3.1, 5.3.7

Zhang, Y., B. Yan, W. Kang, Y. Cheng, J. O. Klein, Y. Zhang, Y. Chen and W. Zhao, "Compact model of subvolume MTJ and its design application at nanoscale technology nodes", IEEE Transactions on Electron Devices **62**, 6, 2048–2055 (2015). (document), 5.2.1, 5.6, 5.4.1

Zhang, Y., W. Zhao, Y. Lakys, J. O. Klein, J. V. Kim, D. Ravelosona and C. Chappert, "Compact modeling of perpendicular-anisotropy CoFeB/MgO magnetic tunnel junctions", IEEE Transactions on Electron Devices **59**, 3, 819–826 (2012). 5.2.1

Zhu, J.-G., "Magnetoresistive random access memory: The path to competitiveness and scalability", Proceedings of the IEEE **96**, 11, 1786–1798 (2008). 4.3.3