Method to Address Performance Decline due to Process, Voltage, and Temperature Variations in Integrated Circuits

Submitted by

Mohsen Radfar

M.Sc. Computer Architecture

B.Sc. Applied Mathematics

A thesis submitted in total fulfilment
of the requirements for the degree of

Doctor of Philosophy

School of Engineering and Mathematical Sciences

Faculty of Science, Technology and Engineering

La Trobe University

Bundoora, Victoria 3086

Australia

November 2013

Dedicated to my warm-hearted beautiful wife,

Samane

Table of Contents

List of Figures

List of Tables

List of Acronyms

ABB	adaptive body-biasing
ADSL	asymmetric digital subscriber line
ALU	arithmetic logic unit
AMS	analogue mixed signal
AR-mode	accessed retention mode
BB	body biasing
BIPS	billion instructions per second
BPTM	Berkeley predictive technology model
CHLFF	complementary hybrid latch flip-flop
CMOS	complementary MOSFET
CSAFF	complementary-SAFF
DCVSL	differential cascade voltage switch logic
DCVSL	differential cascade voltage switch logic
DIBL	drain-induced barrier lowering
D-MOSFETs	double metal-oxide-semiconductor field-effect transistors
DRC	design rule check
DVS	dynamic voltage scaling
EDA	electronic design automation
EDP	energy-delay product
EX	execution stage
FBB	forward body bias
FF	flip-flop
FF corner	Fast NMOS-Fast PMOS corner
FFT	Fast Fourier Transform
FIFO	first-in first-out
FIR	finite impulse response
FM	frame memory
FO4	fan out of 4
FPU	floating-point unit
FS corner	Fast NMOS- Slow PMOS corner
GDSII	graphic database format−version II
GIPS	Giga instructions per second
GP	general-purpose
HMA	hybrid memory architecture
IA32	Intel Architecture, 32-bit
IC	integrated circuit
iPDK	interoperable process design kit
IROM	instruction read only memory
ISA	instruction set architecture
ITRS	international technology roadmap for semiconductors
LEF	library exchange format
LP	low-power
LUT	look-up table

LVS	layout versus schematic
MC	Monte Carlo
MOSFET	Metal–oxide–semiconductor field-effect transistor
NBB	NMOS network body bias
NM	number of multipliers
NMOS	n-channel MOSFET
PBB	PMOS network body bias
PDF	probability distribution function
PMOS	p-channel MOSFET
PT	process and temperature variations
PTM	predictive technology model
PV	process and voltage
PVT	process, voltage, and temperature
RBB	reverse body bias
RISC	reduced instruction set computing
ROM	read only memory
RTL	register transfer level
SA	sense amplifiers
SAFF	sense-amplifier based flip-flop
SEFF	soft edge flip-flop
SF corner	Slow NMOS-Fast PMOS corner
SIMD	single instruction multiple data
SM	scratchpad memory
SNM	static noise margin
SOC	system on chip
SPICE	simulation program with integrated circuit emphasis
SRAM	static random access memories
SS corner	Slow NMOS- Slow PMOS corner
ST	Schmitt trigger
SULP	sensitive and ultra-low power
TFF	transparent FF
TSMC	Taiwan Semiconductor Manufacturing Company
TT corner	Typical NMOS- Typical PMOS corner
UPF	unified power format
VGA	video graphics array
VLSI	very large scale integrated
VT	voltage and temperature variations
WLAN	wireless local area network
ZBB	zero body biased

List of Variable Definitions

μ	mobility of the majority carriers in channel
μ	mean value
μ_X	mean value of random variable X
C_D	capacitance of the depletion layer
C_{ox}	capacitance of the oxide layer
C_S	a given inverter's switching load capacitance
D_0	$\frac{\frac{1}{2}\eta C_s V_{DD}}{I_{0p} e^{\frac{V_{DD}}{m_p v_T}} \left(1 - e^{-\frac{V_{DD}}{v_T}}\right)}$
$E(f(X,Y))$	expected (mean) value of function of random variables X and Y
$E(X)$	expected (mean) value of random variable X
E_{act}	active energy
$E_{act}DP$	active energy-delay product
$E_{ACT}DP$	Random variable representing active energy-delay product
EDP_{SULP}	Energy-delay product when SULP FBB technique is applied
$EDPSULP$	random variable representing EDP_{SULP}
EDP_{ZBB}	Energy-delay product when ZBB technique is applied
$EDPZBB$	random variable representing EDP_{ZBB}
E_{leak}	leakage energy
$E_{leak}DP$	leakage energy-delay product
$E_{LEAK}DP$	Random variable representing leakage energy-delay product
$Erf(x)$	error function of variable x
$Erfc(x)$	complementary error function of variable x
$f_{Dx\alpha}(V_{thx0})$	$\frac{1}{2}\left(1 - Erf\left(\frac{\alpha\sigma_{VTHX}}{\sqrt{2}m_x v_T} + \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma^2_{VTHX}}}\right)\right) + \frac{e^{\frac{-\alpha\gamma V_{DD}}{m_x v_T}}}{2}\left(1 + Erf\left(\frac{\alpha\sigma_{VTHX}}{\sqrt{2}m_x v_T} + \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma^2_{VTHX}}}\right)\right)$
$f_{Ex}(V_{thx0})$	$\frac{1}{2}\left(1 + Erf\left(\frac{\sigma_{VTHX}}{\sqrt{2}m_x v_T} - \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma^2_{VTHX}}}\right)\right) + \frac{e^{\frac{\gamma V_{DD}}{m_x v_T}}}{2}\left(1 - Erf\left(\frac{\sigma_{VTHX}}{\sqrt{2}m_x v_T} - \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma^2_{VTHX}}}\right)\right)$
$f_X(x)$	probability distribution function for random variable X which represents variable x
I_0	drain-source current at threshold voltage
I_{0n}	drain-source current of a given PMOS transistor at threshold voltage
I_{0p}	drain-source current of a given NMOS transistor at threshold voltage
I_{0X}	drain-source current of transistor M_X at threshold voltage
I_{leak_n}	leakage current passing through a given NMOS transistor when $V_{DS}=V_{DD}$ and $V_{GS}=0$

I_{leak_p}	leakage current passing through a given PMOS transistor when $V_{DS}=V_{DD}$ and $V_{GS}=0$
I_{on_n}	current passing through a given NMOS transistor when it is switched on (at the subthreshold region)
I_{on_p}	current passing through a given PMOS transistor when it is switched on (at the subthreshold region)
k_B	Boltzmann's constant (=1.38x10^{-3}J/K)
L	channel length
m	slope factor of transistor $\left(1+\frac{C_D}{C_{ox}}\right)$
m_n	slope factor of a given NMOS transistor
m_p	slope factor of a given PMOS transistor
mul_n	number of multipliers in a given NMOS transistor
mul_p	number of multipliers in a given PMOS transistor
mul_X	number of multipliers in transistor M_X
m_X	slope factor of transistor M_X
NDEP	doping concentration in the channel
n_i	intrinsic carrier concentration in an undoped silicon substrate
p	probability of $V_{BSP}=V_{DD}$ or $V_{N2}\leq V_{DD}/2$
P (X)	probability of the event *X*
q	charge of electron (=1.602x10^{-19}C)
T	temperature in Kelvin
t_0	a given timing constraint applied on the variable t_d
t_d	delay of a given gate or data-path (s)
T_d	random variable representing variable t_d
TdZBB	random variable representing the delay of a given inverter when zero body bias (ZBB) is applied
t_{ox}	gate silicon oxide thickness
t_{total}	total time of the examination used for the calculation of total energy consumption
V_{BS}	body-source potential difference (body bias)
V_{BSN}	body-source voltage for transistors in NMOS network
V_{BSP}	body-source voltage for transistors in PMOS network
VBSN	random variable representing V_{BSN}
VBSP	random variable representing V_{BSP}
V_{DD}	supply voltage
V_{DDH}	higher (or nominal) supply voltage in a multiple V_{DD} system
V_{DDL}	lower supply voltage in a multiple V_{DD} system

V_{DS}	drain-source potential difference
V_{GS}	gate-source potential difference
V_M	switching threshold in a given gate
V_{N1}	reference voltage in the SULP FBB generator of Figure 4.1
V_{N2}	SULP FBB generated voltage in Figure 4.1
VN1	random variable representing V_{N1}
VN2	random variable representing V_{N2}
V_{SS}	zero/ground voltage
v_T	thermal voltage
V_{th}	threshold voltage
V_{th0}	zero-bias threshold voltage
V_{thn}	threshold voltage of a given NMOS transistor
VTHN	random variable representing V_{thn}
V_{thn0}	specific threshold voltage of transistor M_{A2} in Figure 4.1.b in which V_{N2} is equal to $V_{DD}/2$
V_{thp}	threshold voltage of a given PMOS transistor
VTHP	random variable representing V_{thp}
V_{thp0}	specific threshold voltage of transistor M_{A2} in Figure 4.1.a in which V_{N2} is equal to $V_{DD}/2$
V_{thX}	threshold voltage of transistor M_X
W	channel width
W_X	channel width of transistor M_X
α	input activity factor for a given gate or data-path ($0 \leq \alpha \leq 1$)
γ	body bias coefficient for a given technology
ε_{si}	permittivity of gate oxide (=345fF/cm)
η	delay factor of a given inverter's non-step input
λ	Drain-Induced Barrier Lowering (DIBL) effect
σ	standard deviation
σ^2	variance
σ_X	standard deviation of random variable X
Φ_s	surface potential for a given technology
σ_X^2	variance of random variable X

Statement of Authorship

Except where reference is made in the text of the thesis, this thesis contains no material published elsewhere or extracted in whole or in part from a thesis submitted for the award of any other degree or diploma.

No other person's work has been used without due acknowledgment in the main text of the thesis.

This thesis has not been submitted for the award of any degree or diploma in any other tertiary institution.

_____ 14/11/2013

Mohsen Radfar

Acknowledgment

The opportunity to be admitted to La Trobe University as a PhD candidate was given to me by Professor Jugdutt (Jack) Singh, and resulted from a simple reply to one of my 1600 worldwide email enquiries. Professor Jack Singh became my principal supervisor and I would like to thank him for his support and for his constant faith in me from start to end.

I would also like to express my gratitude to my co-supervisor, Dr. Kriyang Shah, who was always willing to help me out of my stalemates and kept believing in me no matter how far I pushed the boundaries.

I will also never forget our meetings with Dr David Tay discussing my progress in research as well as other exciting conversations.

Moments that I will always remember are my difficult to answer questions from Vincent, the lunch time debates with Golnar and Soheil, long chats with Harsh, absolute silences with Prajakta, and many other office sharing experiences to evade noise, cold, heat, and sometimes people.

I am also glad that we had May Gu and Tracey Carpenter in the Centre who really cared about us and occasionally held afternoon teas compelling us to socialise.

In addition, I will remain ceaselessly grateful to this beautiful country, Australia, and this magnificent university, La Trobe University, for the wonderful residence and the life-changing scholarship they granted me, based only on merit. I struggle, though, to appreciate the real extent of melancholy my family, especially my mother, has been through and will be feeling as a result of my immigration. I wish this world was not so unjust.

My paramount appreciation, however, goes to the one without whom all this would have been a dream, who overlooked my countless flaws and appreciated my few aptitudes, who endured my never ending hardships from our first days, who encouraged and accompanied me throughout this journey, without whom my life was and would be an utter meaninglessness; I dedicate to her invaluable essence my valued thesis, to my dearest **Samane**.

Mohsen Radfar

Abstract

Since the onset of the new millennium, power consumption related complications, such as heat dissipation, battery lifetime and reliability, have resulted in drastic shifts in silicon industry priorities so that performance is no longer the only pivotal motivation in the design of integrated circuits. Instead, energy consumption restrictions are transforming the design methods, and smart sensor applications are simultaneously exacerbating this trend by pressing for extremely long, if not indefinite, battery lifetime.

However, this shift and proliferation in the semiconductor industry has come at a substantial price which is a growing inaccuracy in the fabrication process of integrated circuits with each technology generation. When operating in ultra-low power situations, the impact of this inaccuracy affects the circuits to a greater extent, makes them vulnerable to voltage and temperature variations, and can simply render them inoperable.

This study proposes a novel technique that is highly sensitive to fabrication process variations while monitoring and appropriately responding to temperature and voltage variations. The body bias generated by this technique was applied on an 8-bit Kogge Stone adder as well as a 1024 point, radix 4, 32x32bit complex input iterative FFT processor. Results showed that the proposed technique reduces the delay-energy product by 20% in the adder and by 400% in the FFT processor, when voltage is scaled from superthreshold to subthreshold levels and while temperature reaches extremes of -15°C to 75°C.

Process variation is also addressed so that circuits' error probability is decreased from 50% to 1% at subthreshold regions and performance variations are dropped seven times with respect to zero body biased circuits.

A pipelined version of the aforementioned FFT also resulted in 40 times less energy consumption per FFT, compared to the latest low energy FFT implementation in the literature, while being only two times slower.

List of Publications

[1] M. Radfar, K. Shah, and J. Singh, “A Yield Improvement Technique in Severe PVT Variations and Extreme Voltage Scaling,” submitted to the journal of Microelectronics Reliability.

[2] Radfar, M.; Shah, K.; Singh, J., “A Reliability Improvement Technique in Severe Process Variations and Ultra Low Voltages,” in Biomedical Circuits and Systems Conference (BioCAS), 2013 IEEE, 2013, pp. 210-213.

[3] M. Radfar, K. Shah, and J. Singh, "A highly sensitive and ultra low-power forward body biasing circuit to overcome severe process, voltage and temperature variations and extreme voltage scaling," Int. J. Circ. Theor. Appl.. doi: 10.1002/cta.1935, 2013.

[4] M. Radfar, K. Shah, and J. Singh, "Recent Subthreshold Design Techniques," Active and Passive Electronic Components, vol. 2012, p. 11, 2012.

[5] M. Radfar, S. P. Mozafari, K. Shah, and J. Singh, "Analysis of geometric and non-linear programming as optimization algorithms for low power VLSI circuits," in Electrical and Computer Engineering (CCECE), 2011 24th Canadian Conference on, 2011, pp. 617-620.

[6] M. Radfar and S. Pourmozaffari, "Faster solution of nonlinear equations using logical effort method and curve fitting in low power design," IEICE Electronics Express, vol. 6, pp. 889-896, 2009.

1 Introduction

1.1 Introduction

Design of high-performance processors has undergone a drastic change since the start of the new millennium due to power consumption related difficulties, such as heat dissipation, battery lifetime and reliability (details in section 1.2.3). Moreover, performance is no longer the key driving factor in the design of recent integrated circuits as energy consumption limitations are revolutionising the design techniques (details in section 1.2.2). On the other hand, low power systems such as mobile phones, tablets and cameras demand longer battery lifetime while tolerating a performance compromise as opposed to laptops. There also exist applications such as implantable medical processors and wireless sensor networks pushing for very limited energy consumptions and very long, if not indefinite, lifetime while accepting low performance processors. This wide spectrum of energy-performance requirement needs various design techniques.

At the same time, sensor/stimulator technology, energy harvesting and ultra low power circuits have experienced massive advances recently, opening new opportunities for many applications. The main driving force behind this recent rapid growth in wireless sensor networks is the advances in low-cost sensor manufacturing, thanks to successes in Micro-Electro-Mechanical Systems (MEMS) technology. Miniature dimensions together with ultra low power consumption are amongst critical characteristics of such sensory systems as, besides sensors and stimulators/actuators, they accommodate other versatile components such as microprocessors, memories, transceivers, and energy sources. Besides, advances in semiconductor industry have met the miniaturisation required for the high level of integration in sensor systems. Tens to thousands of these sensory systems can be connected together by their radio transceivers to form wireless sensor networks with various applications.

For instance, sensing applications now collect their ambient information, such as temperature, pressure, etc, and either partially/fully process it or transmit it for further processes. For example, glaucoma, a disease characterized by excessive fluid pressure within the eye, can be diagnosed early through a sensing system implanted inside the human eye [1]. Stimulators can also trigger events via electrodes in implanted circuits.

One such system is the bionic ear cochlear prosthesis, which gives speech understanding via electrical stimulations [2]. Also a similar project, focusing on the bionic eye, is in progress at Bionic Vision Australia [3]. Figure 1.1 shows numerous applications of wireless sensor networks [4].

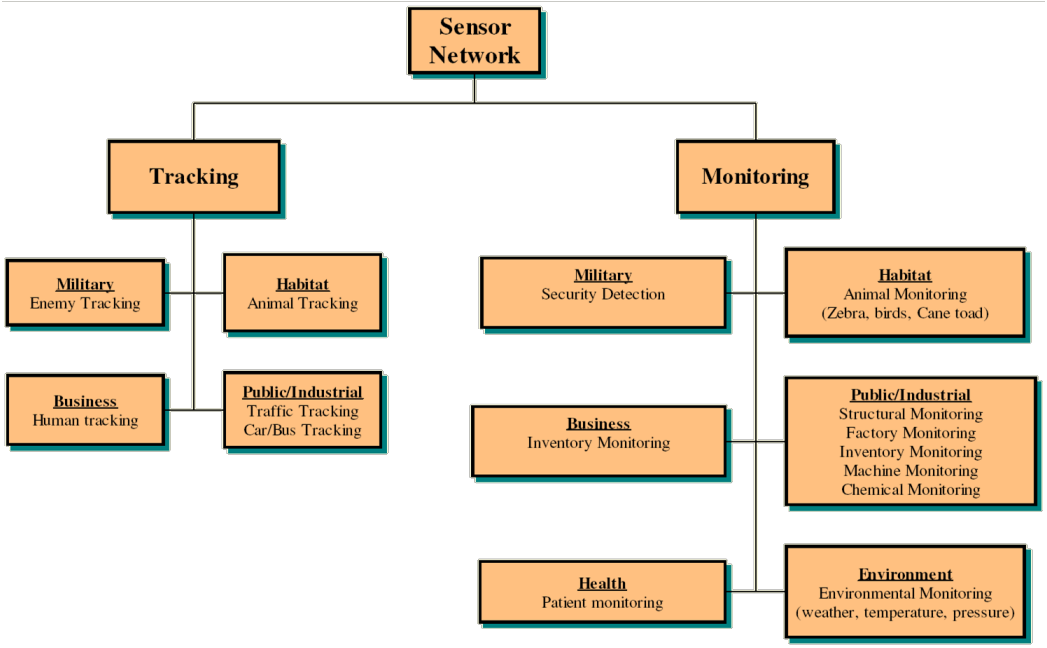


Figure 1.1. Sensor network applications

Beside the aforementioned medical applications, wireless network sensors can be used for surveillance and tracking purposes by detecting and identifying intrusions, or they can be utilised for detecting short-term and long-term environmental changes (like temperature in the forest) to predict disasters (such as fires). They can also be employed to sense seismic variations and help discern upcoming volcanic eruptions or earthquakes [4].

In many of these applications, replacing or recharging the battery is very costly, unsafe or sometimes impossible. This puts an extreme constraint on power consumption as well as reliability as functional failures due to low battery voltages cannot be tolerated. On the other hand, the reliability of fabricated integrated circuits is deteriorating with each technology generation (details in section 1.3.1). As a result, not only these microprocessors and their memories need to be ultra low power, but they also have to be capable of dealing with any conditions which result in functionality failures, one of them being process variations (details in section 1.3). Two other sources of these failures are voltage variations (for example, due to low capacity in the battery to deliver the required power) and temperature variation (which can be either due to ambient temperature variations or circuits' heat dissipations).

1.2 Low Power Design Motivations

1.2.1 Semiconductor Industry Growth Trend

Gordon Moore predicted via an illustration (Figure 1.2) in 1965 [5], that “number of components per integrated function” would be doubled each year "for at least ten years". This period was later referred to as each 18 months rather than each year.

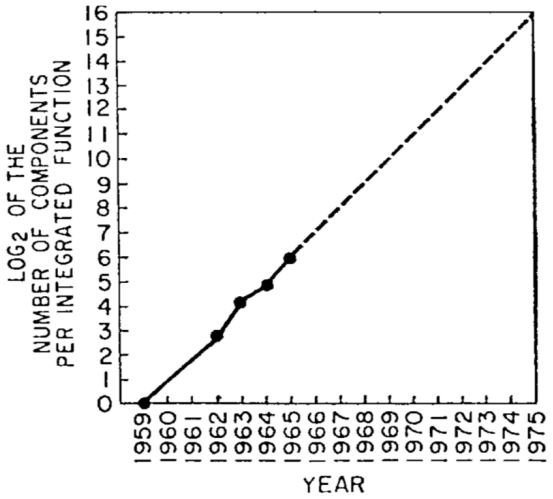


Figure 1.2. Moore’s original figure (1965)

This law has been extended to other aspects of the digital electronic industry such as transistors’ density, channel length, power dissipation, hard drive capacity, etc. For example Figure 1.3 shows that, since 1965, his prediction has been quite reliable so far [6].

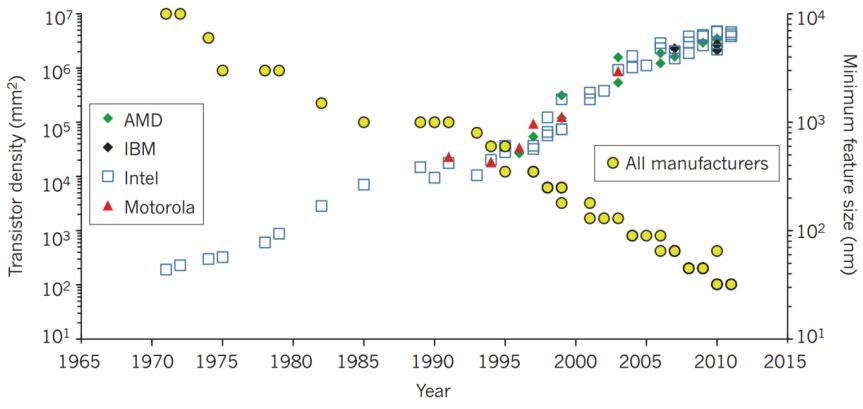


Figure 1.3. Reliable predictions of Moore’s law

1.2.2 Power Implications of Semiconductor Growth

The above mentioned exponential growth in transistors' density, resulting in vast improvements in memory capacities and processing performance, soon brought up some hidden issues. The first high frequency RISC microprocessor in 1992 took many by surprise when it consumed 30W at 200MHz [7]. The new power and thermal problems could no longer be disregarded (these problems are discussed in section 1.2.3). In fact, in 2002, Intel showed that power consumption in its CPUs doubled each 36 months from 1971 to 2000 [8].

However, low power techniques and technologies have allowed further improvements and advances without extreme power consumptions (these techniques are reviewed in section 2.2). Figure 1.4 shows power consumption in all Intel processors since 1995, grouped by fabrication technology [9]. The blue curve, signifying the power dissipation doubling every 36 months, was later capped by the straight line indicating the maximum power of 150W in Intel microprocessors over the last decade. This is mainly due to the enormous cost and consequences incurred by the heat generated in over 150W high performance processors resulting in an unjustifiable price or infeasible implementation. Section 1.2.3 addresses the main consequences of high power consumptions.

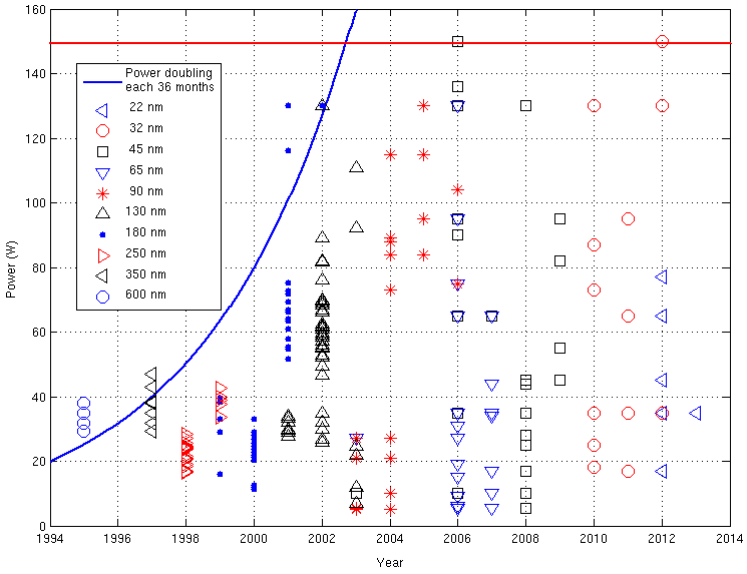


Figure 1.4. All Intel processors and their power consumption since 1995

Despite this cap, newer CPU generations have kept providing better performances while meeting the power restrictions. It can also be seen in Figure 1.4 that, with each generation of technology, power consumption experiences a rise over the course of a couple of years until the next narrower technology is introduced and subsequently a drop in power dissipation occurs. Smaller technologies also have resulted in faster transistors and chips, because the closer transistors are together and the smaller they are, the faster processors become (due to lower propagation delays between transistors and lower capacitance loads). Figure 1.5 demonstrates the trend in the clock speed of Intel processors since 1995 and shows that clock speed has followed the transistor scaling trend and experienced a doubling rate in speed each two years until it was capped by power consumption limitations (as shown in Figure 1.4) around 2004. Since then, clock speed has become more or less the same but other low power techniques (reviewed in section 2.2), like multi-core designs, were introduced to continue taking advantage of the smaller technologies. This highlights power/performance as two major driving forces toward smaller transistors. Technology advances and low power techniques have both played a significant role in satisfying power constraints. Chapter 2 will briefly discuss some of these techniques that are also related to the thesis's main topic. Before that, however, a more detailed look is taken at consequences of higher power consumption (details in section 1.2.3) and narrower transistor technologies (details in section 1.3.1) to find out what areas Chapter 2 has to focus on while reviewing low power techniques (details in section 2.2).

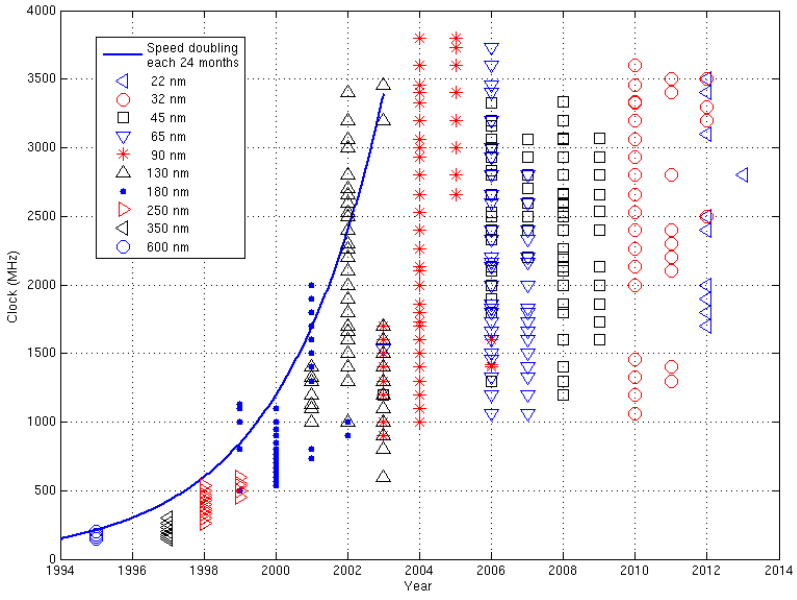


Figure 1.5. Speed trend in Intel processors

1.2.3 Consequences of High Power Consumption

1.2.3.1 *Power Density and Heat Dissipation*

The power density of high performance integrated circuits is usually so high that the ambient air cannot passively dissipate the produced heat. This excessive heat can result in circuit malfunctioning and hence system breakdown or even can destroy the circuit. Therefore, solutions such as heat sinks, fans, or even fluid cooling are utilised to dissipate the extra heat. However, this incurs extra cost as well, and it is highly dependant on power usage. For example, Figure 1.6 exhibits the cost of different cooling systems for Intel microprocessors [10] showing a large commercial restriction in manufacturing of high power consumption processors.

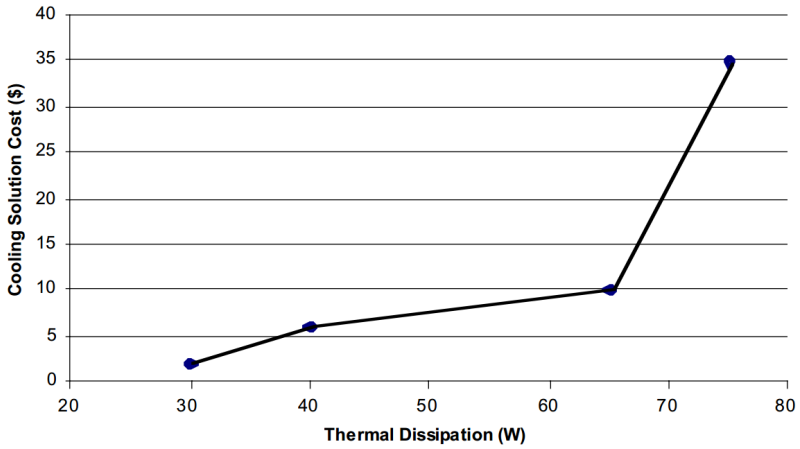


Figure 1.6. Cost of cooling solutions from heat sink to heat pipe

The cost of cooling is expensive in organizations with significant numbers of servers that consume vast amounts of electricity. This cost for data centres in the US alone is in the order of $7 billion annually [11] and, if this trend continues, it is predicted many of them will soon run out of the power and cooling capability to deal with such a growing demand.

1.2.3.2 *Battery Lifetime in Portable Devices*

Higher power consumption results in shorter battery lifetime in portable devices too. Devices such as laptops, mobile phones, tablets, remote sensors, etc, are now playing a vital role in every day life. Therefore, there is a considerable consumer demand as well as

commercial competition to expand the life span of batteries without too much compromise in the devices' performance. However, improving the capacity of batteries has a very slow pace of 5% to 10% each year [12] which is incomparable to the aforementioned silicon industry growth and is lagging significantly behind Moore's law of doubling in computational complexity each 1.5 years. In fact, this limitation is another major motivation behind the advancements in low power techniques to minimise the cost and weight of batteries and maximise their operating lifetime.

1.2.3.3 *Reduced reliability at higher temperatures*

High temperature can in many ways affect the reliability of silicon devices. Increased heat exponentially accelerates electromigration [13], which is the mass transport of metal atoms in ICs causing open circuits in the conductors and functional failure. It also increases delay in conductors and slows transistors down at nominal threshold voltage domains. As Figure 1.7 shows, leakage current also increases as temperature rises. This creates a positive feedback because current increase leads to extra energy consumption and hence extra temperature rise. Unless excess heat is dissipated, reliability of the circuit is significantly compromised and eventually can render the whole system unresponsive or malfunctioning.

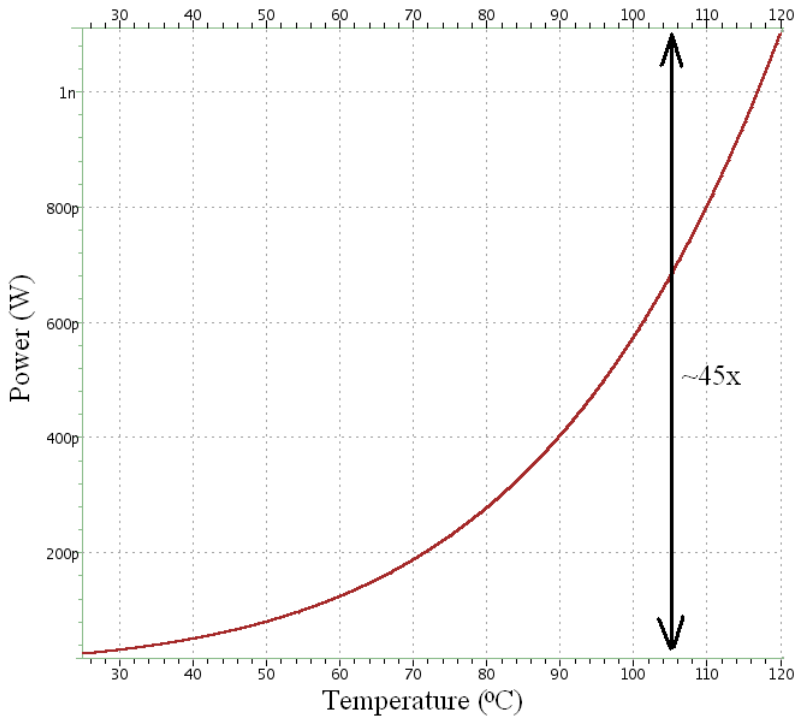


Figure 1.7. Leakage power rises exponentially with temperature rise in a 60nm/120nm NMOS

1.3 Variation Aware Design Motivations

As discussed in sections 1.2.1 and 1.2.2, the new narrower technologies offer many benefits in terms of speed and power consumption improvements. However, this improvement comes at a price and that is the reliability degradation. This section studies the consequences of the new technologies and how to address them.

1.3.1 Semiconductor Scaling Trend and Consequences

Figure 1.8 illustrates the exponential scaling trend in the channel length of metal–oxide–semiconductor field-effect transistors (MOSFET) over the past four decades [14] and also its expected future according to International Technology Roadmap for Semiconductors (ITRS) targets. As discussed before, such scaling trends have enabled exponential rate for the number of transistors compacted inside processors, too.

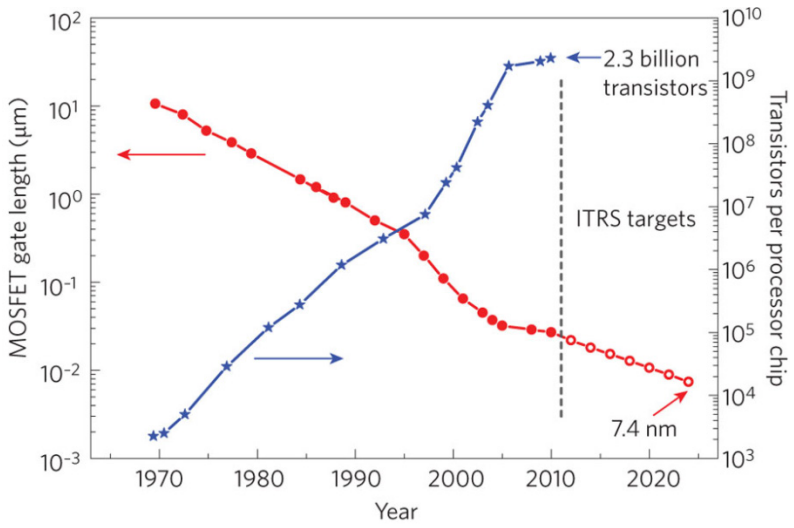


Figure 1.8. Trends of MOSFET channel length scaling and number of transistors per processor

Although large numbers of transistors are favoured, fabricating 32nm or smaller channels is simply becoming harder as photolithography techniques are not catching up with such tiny dimensions [15]. The procedure of utilising light to transfer the pattern of the circuit onto the surface of silicon, which is called lithography, has reached a hurdle because the wavelength of the light is no longer smaller than minimum transistor size [15]. The low resolution of light wavelength results in lithographic variations which play one

contributing part in process variations[1]. The other one is the doping process in which impurities are added to the silicon substrate. Hence, there are manufacturing issues whose nature is random and therefore result in some unpredictable characteristics. This compels engineers to consider statistical analysis when designing circuits to make sure they are capable of tolerating these process variations. Section 2.1.2 explains how statistical analysis helps characterise transistors' behaviour.

Process variations are generally divided to inter-die and intra-die variations [16]. Inter-die variations change all transistors' parameters within a chip in one direction (for instance, the gate length of all devices is increased leading to a slow and low power chip). These changes follow a normal probability distribution (presented in section 2.1.2). In sub-90nm technologies, inter-die variations in threshold voltage have a crucial impact even in high frequency ICs, such as microprocessors, as Figure 1.9 suggests [17]. In this case, for example, threshold voltage variation leads to around 30% variation in frequency and 20 times variation in leakage current, and therefore, high leakage or low frequency chips must be rejected which affects the production yield severely.

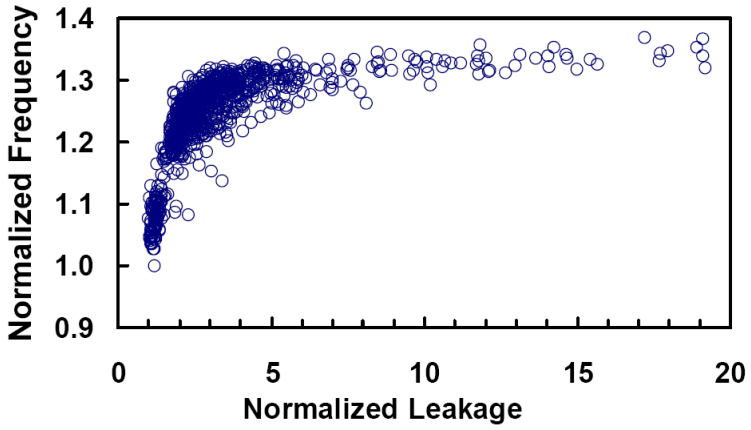


Figure 1.9. Variation of standby leakage current and frequency of microprocessors in a wafer

Intra-die variations, on the other hand, randomly affect parameters within a die (individual chips) and, therefore, two adjacent transistors can have different sizes even though designed to be the same. However, intra-die variations have a low influence on circuit delay, when compared to inter-die variations [18], and, as section 2.2.3 discusses in detail, dramatic delay increase is a major challenge of ultra low power design. Therefore, the proposed techniques in this thesis are mainly focusing on dealing with inter-die variations.

[1] The variations occurring in the fabrication process of dies resulting in random differences in characteristics of devices in chips

In addition to process variations, environmental variability caused by temperature and supply voltage variations are also two other main contributors to the unpredictability of fabricated circuits [19]. Temperature of an integrated circuit (IC) can change due to changes in surrounding temperature. Very large scale integrated (VLSI) circuits consist of many different parts inside the IC with various energy requirements. Depending on the work load, these parts lead to burst-in-nature hot spots, which is another reason for temperature variations. These different and fluctuating energy requirements also cause sudden voltage drops as power supplies cannot deliver the required current surge [20]. Handling these variations is also a major duty of the proposed technique in this thesis.

When technology scales in each generation, not only channel length/width and gate-oxide thickness all scale down, but supply and threshold voltages need to be scaled too [21]. However, scaling of supply voltages cannot be realised without threshold voltage scaling to support high performance requirements [22]. As it will be seen in section 2.1.1, threshold voltage has an exponential correlation with subthreshold current (otherwise known as leakage current), that is, threshold voltage reduction results in exponential leakage increase.

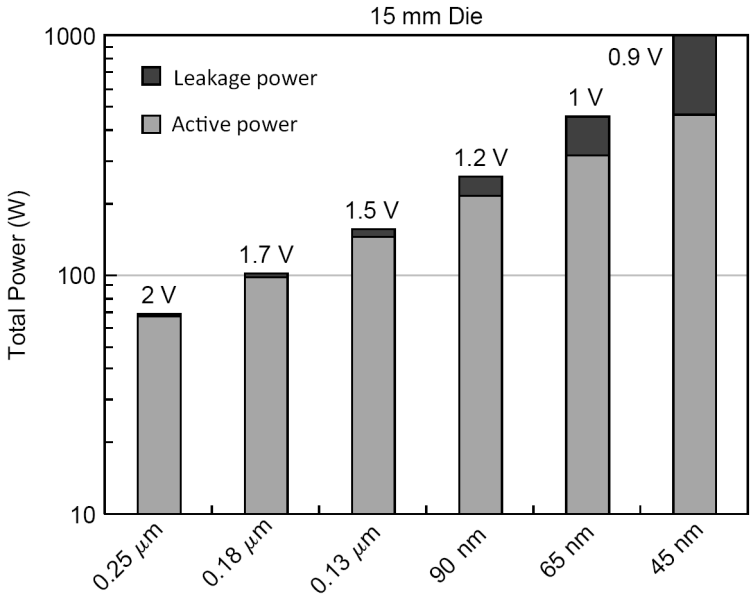


Figure 1.10. Supply voltage, leakage and active power trends over different technologies

As Figure 1.10 shows, this leakage current rise is to an extent that no longer can be ignored as it has a considerable fraction of the total consumed power and, therefore, is of a major concern in ultra low power designs [23]. Hence, reducing power supply has hit a

barrier and circuits can no longer enjoy the energy reduction due to voltage reductions. This presses for new techniques to lower the supply voltage and, at the same time, manage the leakage concerns.

1.4 Research aims

It can be seen from the above discussions that low power and variation-aware designing is motivating many studies in the semiconductor field. This research proposes a technique to address process, voltage, and temperature (PVT) variations specifically arising in ultra low power architectures. To be able to do so, however, the following aims have to be accomplished:

- Locating the existing gaps in this field through a comprehensive literature review
- Acquiring the necessary resources and tools for this research, training on how to employ these special design library and tools, creating the low power library required for designing test circuits, and many other steps before having all the final simulation results analysed and verified
- Developing the mathematical and statistical formulations in order to analyse and comprehend the proposed circuit's statistical behaviour when facing PVT variations and low voltage circumstances
- Designing and implementing a test circuit in order to extensively simulate all possible circumstances occurring during PVT variations at very low voltages and verify the mathematical analyses
- Design and implementation of a test circuit similar to processors often found in wireless sensor applications, analysing the simulation outcomes and comparing with recent works to show the extent of the improvements

1.5 Original Contribution of the Thesis

This research develops a mathematical platform for studying the statistical behaviour of the proposed ultra low power circuit when facing the PVT variations. This platform is capable of precisely and accurately predicting the functionality, performance, and energy consumption of such low power techniques in order to analyse and verify the final operation. The analysis results showed the highest sensitivity in this technique to process variations among currently available techniques, and an improved energy-delay product (EDP) in the analysed theoretical inverter.

In addition to sensing the PVT variations, the proposed technique is at the same time capable of reacting to the voltage level under which the circuit is working. This means that if a circuit has a low voltage level or any other performance restricting conditions (due to PVT variations), the proposed technique detects it and helps the system cope with the PVT variations which are more pronounced at low voltages. The originality of this technique is in its capability to handle harsh temperature and process variations while addressing aggressive voltage scaling that is unprecedented in the literature.

On the other hand, this extra help is withdrawn when the system is working in high voltage levels or any other high performance circumstances. This results in considerable reduction of functionality failures at slow conditions and no energy overhead at fast situations which adds to the technique's originality as well. This technique also leads to a significant improvement in production yield as a result of its adaptive approach towards PVT variations as well as voltage reductions.

In fact, a seven times improvement in delay variation was observed in the simulated 8-bit Kogge-Stone adder as well as 23% improvement in its EDP. Additionally, error probability was decreased from 50% to 1% at 0.4V as a result of this technique.

The outcome of mixed signal simulations on a 1024 point, radix 4, 32x32bit complex input iterative FFT processor showed not only the same seven times improvement in delay variations, as before, but EDP was also reduced to around four times, as a result of this technique. Finally, a pipelined version of the FFT showed about 40 times less energy consumption per FFT, compared to the latest low power FFT implementation in the literature, while being only two times slower.

1.6 Significance of the Research

The above mentioned achievements lead to many theoretical and practical benefits in ultra low power design techniques.

The mathematical platform helps designers understand and adjust the parameters and factors which have the highest and lowest sensitivity in the output of a particular sub/nearthreshold circuit. They can also analyse and predict, with a high accuracy, what effects a technique can have on a design under test, and optimise the approach mathematically.

Improvements in EDP of simulated Kogge-Stone adder and FFT processors as well as negligible energy and area overhead of the proposed technique demonstrate that low power circuits, such as the aforementioned smart sensors, can enjoy the benefits of this technique with little or no concern on justifying the shift. Moreover, the negligible energy and area overhead leads to negligible impact on design turnaround and minimum design precautions.

Not only the application of this technique can be easily justified, but there are also other incentives encouraging taking advantage of this body bias generator. For example, seven times delay variation reduction as well as subthreshold error probability reduction from 50% to 1% leads to a massive production yield boost for vulnerable and PVT variation prone ultra low power circuits.

Lastly, the comprehensive literature review delivered in this study can be used as an independent review of recent sub/nearthreshold techniques to help understand all advantages and disadvantages of the available techniques in various aspects of low power design field from circuit to architecture level, and in memory as well as logic components.

1.7 Research Methodologies and Techniques

In order to realise the above mentioned aims in this research, a precise method has been followed which is explained here.

Firstly, the previous literature was investigated to be able to come up with a list of potential gaps. To identify these gaps, however, it was necessary to extend the area of study to a large range of applications. Low power techniques are application-specific and, therefore, are as extensive as integrated circuit applications. They can be applied on different high-performance or low-energy systems in different structures such as logical components or memory arrays. As a result, finding the challenging issues required studying all the techniques in these various systems and structures, comparing the outcomes of their research with previous works, locating the remaining problems and finally choosing the most critical problem and possible solutions.

These possible solutions were, then, taken into account and a technique was proposed. Mathematical analysis of circuits' behaviour together with studying numerous design techniques in previous literature helped in proposing the technique.

With initial simulations and more fine-tunings suggested by mathematical and simulation optimisations, the proposed circuit is verified theoretically, through probability theory, as well as empirically, through Monte Carlo simulations.

After achieving satisfying results by means of mathematically analysing the technique being applied on an inverter, the next step was preparing the requirements for verifying the outcome of the technique on bigger scales and through simulations. This required the costly interoperable Process Design Kit (iPDK) support by Taiwan Semiconductor Manufacturing Company (TSMC) as well as Electronic Design Automation (EDA) tools and simulation environments. Moreover, a very time consuming process was needed to create the standard cell library as a requirement for implementation of different test circuits.

Ultimately, test circuits were designed, implemented, and simulated to verify the final functionality of the technique in various working conditions and system structures. The simulation process depends on the structure of the system under test. A Kogge-Stone adder was first chosen, as the basic block of every processor, to be able to manage the time and processing intensive Monte Carlo simulations.

After meeting the initial targets of the proposed technique, two large FFT architectures were chosen, based on the literature review, to be able to compare the final outcomes with the previous works. These large FFT processors were also implemented and simulated and the results proved the correctness of the initial results.

1.8 Thesis Organisation

The Introduction chapter began with explaining the motivations behind the rapid growth in semiconductor industry and then consequences of this growth on power consumption and process variations were explored. Based on these challenges, the thesis goals and originality were detailed and, later, the methodology followed to reach these goals was reported.

The literature review, Chapter 2, starts with a brief characterisation of subthreshold behaviour of transistors. Then, the major parameters affecting subthreshold current are investigated. Finally, a comprehensive literature review is delivered in two sections addressing the power, performance and reliability issues mentioned in the introduction chapter with the relative merits of each study.

Chapter 3, Design of a Standard Cell Library, firstly illustrates the design flow required to determine the initial design descriptions and embody the final simulation codes. The procedure for realising a specific standard cell library is, then, described in two sections on cell characterisation and design flow. Finally, two recent approaches in creating standard cell libraries are also mentioned.

Chapter 4 presents the theoretical aspects of the proposed body biasing technique and mathematically analyses and predicts the effect of this technique on the delay and error rate of a system undergoing severe process, voltage and temperature variations as well as aggressive voltage scaling.

By employing the standard cell library created in Chapter 3 and applying the technique proposed in Chapter 4 on an 8-bit Kogge-Stone adder, post-layout Monte Carlo simulations are carried out in Chapter 5 to determine the effectiveness of this technique to variations and voltage scaling.

Chapter 6 tests the proposed technique at a much larger scale by means of iterative and pipelined FFT processors to investigate the real energy and area overhead of the technique, when it comes to a realistic implementation of a microprocessor, as well as its credibility and performance.

Chapter 7 summarises the challenges, methodologies and achievements and also offers detailed future works and opportunities which are suggested as a result of this research.

2 Literature Review

2.1 Introduction

To alleviate the higher leakage problems in recent technologies, it is necessary to first study the effect of the main parameters which play a pivotal role in leakage and performance characteristics of the transistors. Therefore, section 2.1.1 briefly demonstrates required formulations necessary for characterisation of threshold voltage and subthreshold current employed throughout this thesis. Subsequently, three sections of 2.1.2, 2.1.3 and 2.1.4 examine the three essential parameters of threshold voltage, body bias and temperature, respectively.

Section 2.2 presents the available techniques which address subthreshold current and PVT variations at the different design levels of abstraction with their respective advantages and disadvantages that led to the proposed technique utilised in this thesis.

2.1.1 Subthreshold Current Characterisation

Threshold voltage can be approximated using following equation [24].

$$V_{th} = V_{th0} + \gamma\left(\sqrt{\phi_s - V_{BS}} - \sqrt{\phi_s}\right)$$
$$\text{where } \phi_s = 2v_T \ln\left(\frac{NDEP}{n_i}\right), \gamma = \frac{t_{ox}}{\epsilon_{ox}}\sqrt{2q\epsilon_{si}NDEP} \text{ and } v_T = k_B\frac{T}{q} \quad (2.1)$$

where γ is the body bias coefficient, Φ_s the surface potential, V_{BS} the body-source potential difference (body bias), v_T the thermal voltage (≈26mV at 25°C), *NDEP* the doping concentration in the channel, n_i the intrinsic carrier concentration in an undoped silicon substrate (=1.45x10^{10}1/cm^3 at 300K), t_{ox} the gate silicon oxide thickness, q is the charge of electron (=1.602x10^{-19}C), ε_{si} the permittivity of gate oxide (=345fF/cm), k_B the Boltzmann's constant (=1.38x10^{-3}J/K) and T the temperature in Kelvin.

On the other hand, if a device is operating in the subthreshold region[1], its drain source current (or the leakage current) can be modelled by the equation (2.2) [25].

[1] where V_{DS} working range is smaller than threshold voltage

$$I_{DS} = I_0 \mathrm{e}^{\frac{V_{GS}-V_{th}+\gamma V_{BS}+\lambda \mathrm{V_{DS}}}{m\, v_T}} \left(1 - \mathrm{e}^{-\frac{V_{DS}}{v_T}}\right) \tag{2.2}$$

where $I_0 = \mu \frac{W}{L} \sqrt{\frac{q\epsilon_{si} NDEP}{2\Phi_s}} v_T^2$, $\mathrm{V_{th}}$ signifies the threshold voltage, λ the Drain-Induced Barrier Lowering (DIBL) [26] effect (which is mostly negligible at subthreshold domain [27]) and m the slope factor of transistor. For I_0, μ is the mobility of the majority carriers, W the width and L the length (presented in Figure 2.1).

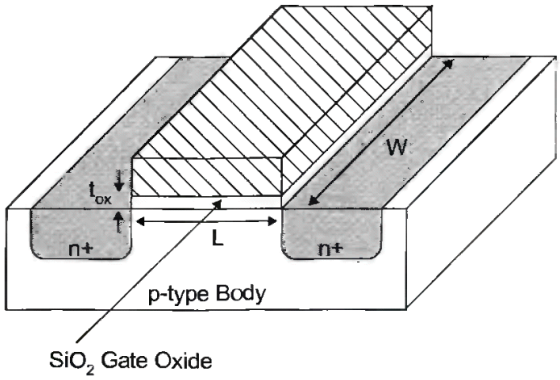


Figure 2.1. NMOS transistor

Figure 2.2 demonstrates the normalised drain-source current in the equation (2.2) focusing on V_{DS} effect in subthreshold domain (the term in parentheses). Referring to Figure 2.2, a similarity can be observed between I-V characteristic of subthreshold domain current to that of superthreshold domain[1] current found in many text books such as [24]. This means that a transistor in the subthreshold domain can be exploited as if it is working in superthreshold domain and all combinational and sequential circuits will act the same in two domains at logic level of abstraction provided the circuit level limitation of subthreshold circuits are fully considered. Instead of drain-source currents in a switched-on gate at a superthreshold voltage, it is the leakage current (drain-source weak inversion current) of a gate, working at a subthreshold voltage, that charges and discharges the capacitance load of subsequent gates. Therefore, at the gate level, a subthreshold circuit can be designed and implemented in exactly the same way that regular VLSI circuits are and it is circuit level design that makes subthreshold circuits different. A subthreshold circuit can also work at a superthreshold voltage and vice versa which means, if multi-voltage design is required, circuit level characterisations and

[1] where V_{DS} working range is bigger than threshold voltage

verifications should be undertaken in both subthreshold and superthreshold voltages (details are discussed in Chapter 3).

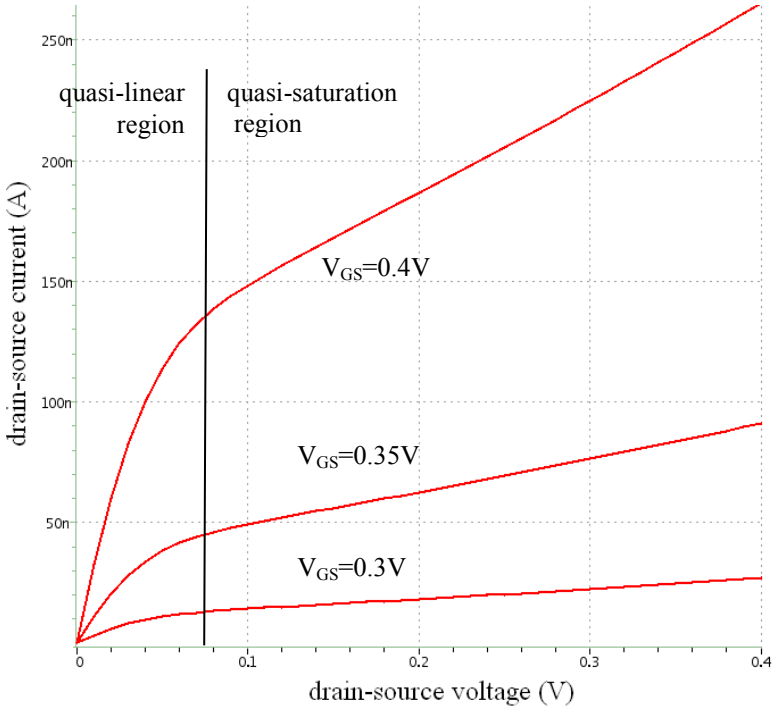


Figure 2.2. Subthreshold I-V characteristic of an NMOS transistor

Figure 2.3, on the other hand, demonstrates the exponential effect of supply voltage on the drain-source on-current [1] (presented in equation (2.2)) [28]. For decades, this exponential effect has been a motivation to ultra low power circuit designers. This is because the further the voltage of circuits moves to the subthreshold domain, the less drain-source current is drawn and, hence, the less possible energy is consumed.

It is clear in Figure 2.3 that this exponential drain-source current drop, causing less energy consumption in subthreshold and nearthreshold [2] voltage domains, results in exponential delay increases as well (which is explained in detail using equation (4.14)). Although subthreshold current reduces with supply voltage (V_{DD}), the above mentioned delay increase leads to exponential leakage current per cycle. On the other hand, switching energy[3], which depends on V_{DD}, decreases quadratically, when V_{DD} scales down. This results in a minimum energy point occurring during the course of voltage scaling.

[1] The drain source current when $V_{GS}=V_{DS}=V_{DD}$
[2] Nearthreshold voltage domain is when V_{DS} working range is very close to, but still under threshold voltage
[3] Defined as $\frac{1}{2}\alpha C_S V_{DD}^2$, where α is switching activity and C_S the load capacitance being switched

In fact, it has been shown for many applications that the minimum energy consumption point is located in the subthreshold domain [29]. This is the main motivation for the rest of this chapter, to follow up recent subthreshold techniques [30] in order to figure out what challenges are facing designers to take advantage of this minimum energy point and if any hurdles are hindering them.

For this purpose, further examination of equation (2.2) shows that it is not only drain-source or gate-source voltages affecting subthreshold current. Sections 2.1.2, 2.1.3 and 2.1.4 explore other important factors which are used to overcome subthreshold critical challenges.

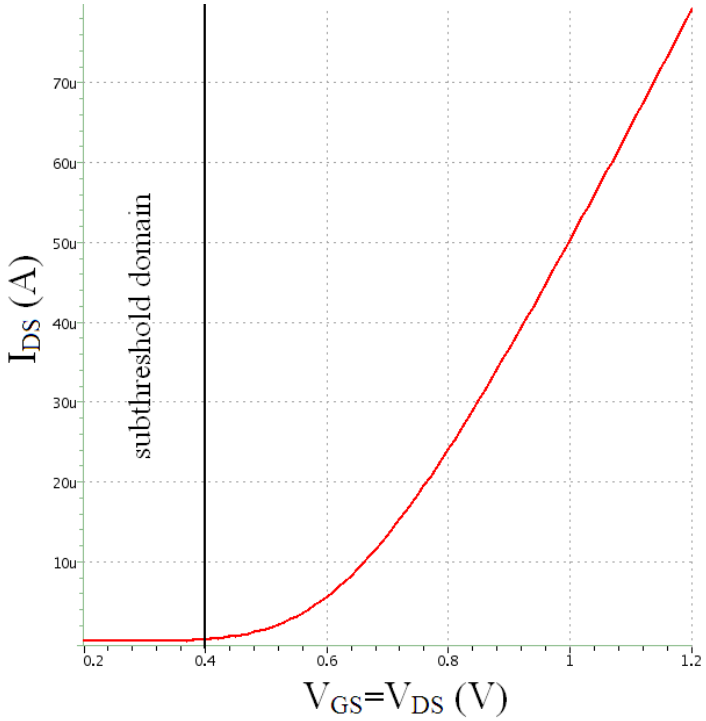


Figure 2.3. Exponential effect of V_{DD} scaling on drain-source current.

2.1.2 Process Variation and Threshold Voltage Effect

As mentioned before and by referring to equation (2.2), an exponential link can be observed between threshold voltage and leakage current. On the other hand, equation (2.1) also shows the effect of process variations on threshold voltage as it depends strongly on both doping concentration in the channel and oxide thickness. Additionally, other process variation dependant parameters are also present in equation (2.2) such as

width and length of transistor's channel, mobility of the majority carriers, etc. As a result, study of process variation effect on subthreshold current can be very complex.

Table 2.1 shows the normalised variability (sigma/mean) in delay and power as a function of different circuit styles, summarising the impact of variation in the mentioned transistor parameters based on data found in [31] from Monte Carlo simulations of common digital blocks, such as adders, implemented in a 90nm process. As can be seen, variation in V_{th} and channel length (L) contribute most heavily to overall process variation (which V_{DD} is not part of). In fact, a study in [32], using HSPICE[1] simulations on a design comprised of row and column access transistors in a 0.18μm process, showed at least 10 times more sensitivity[2] in V_{th} variations than in channel length variations. Despite this, the effect of channel length cannot be ignored and, when process variation resistance is required in a circuit, usually devices with the longer channel lengths are used (as explained in Chapter 4).

Table 2.1. Effect of process and voltage variations on different parameters in devices

Parameter	Delay Variability (%)	Power Variability (%)
V_{th}	2.5-5.6	1.7-4.5
L	2.6-3.2	0.02-0.2
t_{ox}	0.9-2.1	0.7-1.9
W	0.2-0.9	0.4-1.2
V_{DD}	2-3.7	4-5.25

Therefore, as a common practice already in use [33], which results in accurate predictions of design parameters such as delay and power (the simulation results in subsequent chapters also prove this accuracy), it is usually assumed that all process variations can be abstracted in V_{th} variations and this assumption is essential in later discussions.

Figure 2.4 shows the histogram of V_{th} variation occurring in 10,000 Monte Carlo simulations on a 65nm TSMC NMOS transistor. As the bell-shaped curve suggests, threshold voltage can be modelled by a normal Probability Distribution Function (PDF). As mentioned before, the nature of process variations in chip manufacturing is random and this haphazard characteristic is usually modelled by Gaussian (normal) distribution [34]. In this case, foundry provides the model for transistors in which Gaussian models have been widely used for simulating the process variations. Monte Carlo analysis [35]

[1] A SPICE code simulator software by Synopsys, Inc., Mountain View, CA
[2] The ratio of $(\partial I_D / \partial V_{th})$ to $(\partial I_D / \partial L)$

integrated in HSPICE simulations helps exploit these Gaussian models for emulating the stochastic behaviour of process variations (Figure 2.4). If a huge number of random data are generated during a Monte Carlo analysis, the distribution of data converges to a normal distribution.

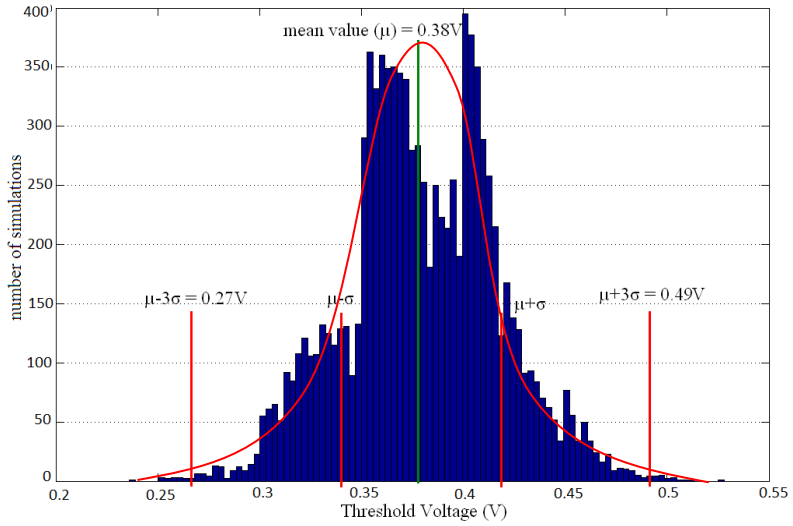


Figure 2.4. 10,000 Monte Carlo simulations signifying threshold voltage variation in an NMOS transistor

Figure 2.4 was (and all Monte Carlo simulations in this thesis were) created by the foundry provided HSPICE model, and therefore, the results resemble the most practical outcome possible. Figure 2.4 also shows two outer boundaries of μ-3σ and μ+3σ that are usually referred to as fast and slow corners, respectively.

Conventionally, and for circuits working in superthreshold domain, worst case scenarios (fast and slow corners) are examined and circuits are designed to deliver the required power and performance in both corners. However, for the process variation simulated and shown in Figure 2.4, the superthreshold and subthreshold on-current can vary substantially. This is shown in Figure 2.5 in which the variability (σ/μ) of on-current in subthreshold 0.3V voltage domain (~1.12 A/A) is 8.5 times higher than superthreshold 1V voltage domain (~0.13 A/A). This is a result of process variation effect, as emphasised before, which is exponential in subthreshold domain, but almost linear in superthreshold domain.

This exponential effect leads to a lognormal probability distribution for subthreshold current (Figure 2.5.b) compared to the normal distribution in superthreshold domain (Figure 2.5.a). Throughout this thesis, the lognormal distribution for subthreshold current helps in the understanding of the power and performance behaviour of subthreshold

domain and explains the proposed techniques. In essence, if random variable X has a normal distribution, then random variable $Y=e^X$ has a lognormal distribution, as a normal distribution has an exponential PDF, the lognormal PDF has the general format of equation (2.3) [36].

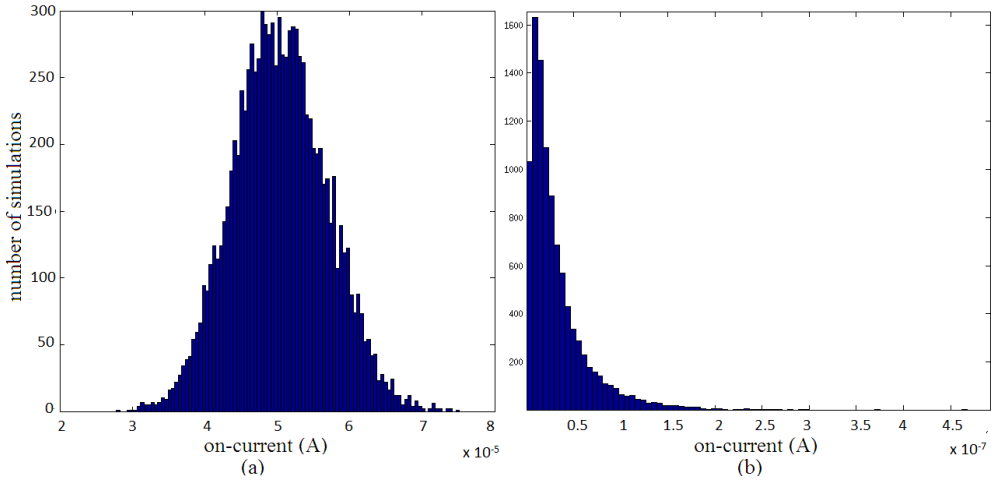


Figure 2.5. 10,000 Monte Carlo simulations signifying on-current variation in an NMOS transistor at a) superthreshold and b) subthreshold domains

$$f_Y(y) = \frac{1}{\sigma\sqrt{2\pi}y} e^{-\frac{(\ln y-\mu)^2}{2\sigma^2}} \tag{2.3}$$

This lognormal effect in subthreshold current is directly sensed in delays of gates working in subthreshold regions (for details refer to equation (4.14)) leading to severe variations in delay (performance) of the chip and, hence, more rejection of dies which do not pass the timing requirements.

The long tail in the on-current lognormal distribution in Figure 2.5.b explains the higher variability in the subthreshold domain whose sources are the worst cases in the manufacturing process and consequences are, therefore, magnified and manifested in this long tail [37]. This, as a result, clearly shows that corner analysis, as opposed to statistical analysis, is not a viable technique for subthreshold designing because it leads to drastic measures and crosses out many otherwise absolutely practical ultra-low power designs and it severely damages the production yield [38].

For example, Figure 2.6 shows that channel length variations, resulting in too narrow lengths, violates the power constraint, despite helping transistors to speed up [39]. Large

channel lengths, however, slow down transistors which reduces yield by failing to meet the required frequency.

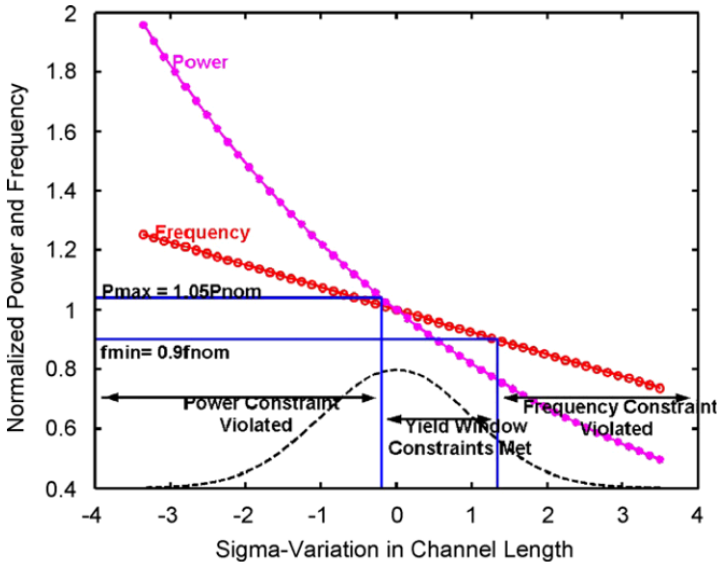


Figure 2.6. With leakage power present, channel length variations can violate power constraint

Therefore, Monte Carlo analysis is at the core of subthreshold designs which requires HSPICE code extracted from the final circuit under test. This HSPICE extraction, as explained in section 3.2, needs all the details of physical design in the standard cell library, which is a library containing the cells used while implementing the design (details explained in Chapter 3). This is one reason why creating a low power standard cell library is needed in this study (presented in section 3.3.4).

2.1.3 Body Bias Effect

As equation (2.1) suggests, the voltage difference between source and substrate (body) of a MOSFET has an effect on the threshold voltage. An NMOS (PMOS) transistor is forward body biased when V_{BS}, the voltage applied between body and source, is positive (negative), and it is reverse body biased when this voltage is negative (positive). When a transistor is forward body biased, its threshold voltage reduces, as equation (2.1) also suggests, and when reverse body biased, its threshold voltage increases. As mentioned before, threshold voltage has a significant effect on subthreshold current. Equation (2.2) helps understand the reason behind this.

Equation (2.2) shows that if threshold voltage is increased, then subthreshold current rises too, which is a direct response to forward body biasing. Figure 2.7 shows the simulation results for a 65nm TSMC modelled feature size transistor which was forward and reverse body biased, as a proof of predictions of equations (2.1) and (2.2).

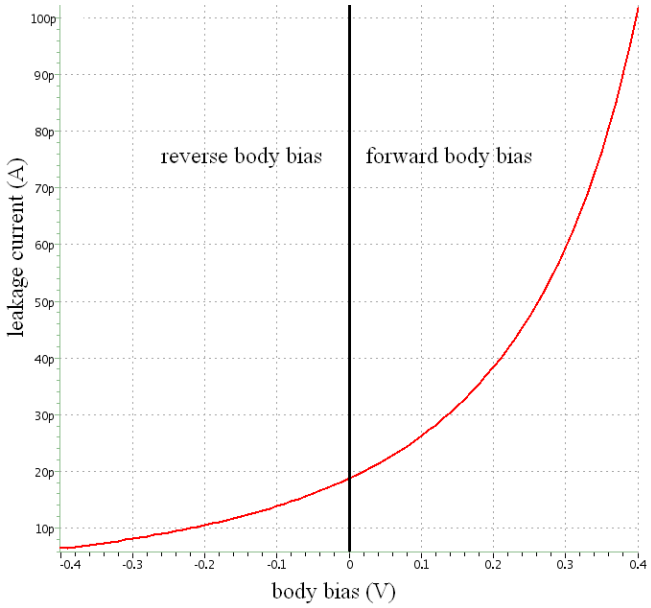


Figure 2.7. Exponential effect of body biasing on leakage current when V_{GS}=0V and V_{DS}=1.2V

2.1.4 Temperature Effect

Temperature has a direct linear influence on thermal voltage and, as a result, a temperature rise degrades threshold voltage linearly (presented in equation (2.1)). However, both threshold voltage decrease and thermal voltage increase lead to an exponential rise in the leakage current (presented in equation (2.2)). Once again, the simulation results in Figure 2.8 show this exponential effect. The exponential effect of body biasing, therefore, can be used to mitigate the exponential effect of temperature which has already been taken advantage of to some extent [40-41].

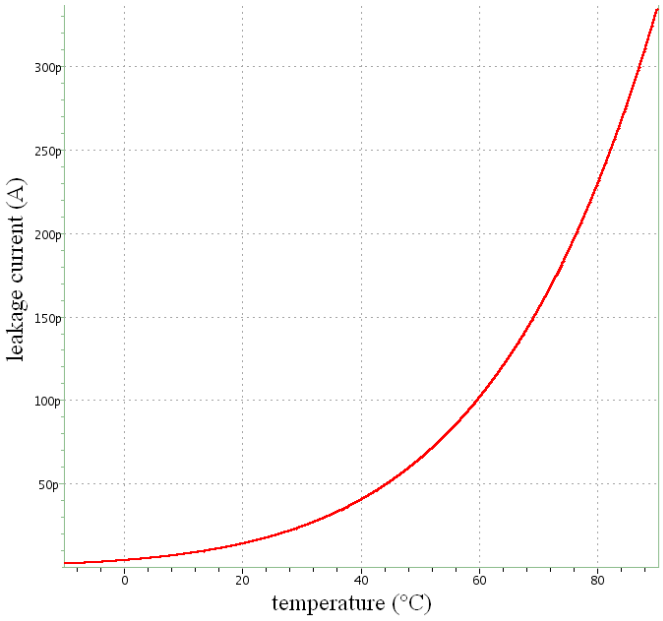


Figure 2.8. Exponential effect of temperature on leakage current when V_{GS}=0V, V_{BS}=0V and V_{DS}=1.2V

2.2 Low power variation-aware design techniques

So far, it has been shown that low power and process variation tolerant IC design is an inevitable requirement for implementation of all designs fabricated in recent silicon technologies. It was also mentioned that the minimum energy point resides in the subthreshold voltage domain and that subthreshold current is exponentially affected by threshold voltage, body bias, and temperature.

There is a strong incentive to have minimum energy consumption, especially for battery powered applications, and this section investigates the barriers in reaching this optimal point. Discusses of how these issues have been addressed in the recent literature and searches for remaining challenges in this area are also detailed.

However, achieving the minimum energy point requires a technique called dynamic voltage scaling (DVS) which is a vital part of any low power system. Therefore, a brief introduction to different DVS strategies is given in the next section before moving onto the literature review sections.

2.2.1 Dynamic Voltage Scaling

Dynamic Voltage Scaling (DVS) has been utilised in the implementation of many low-power processor designs [42] (first proposed in [43]). DVS is an opportunistic technique and has a goal of reduction of power consumption whenever high performance is not needed. That is, supply voltage (and frequency, as a result) is adapted depending on the required performance which leads to delivering no more than required energy, whilst fixed voltage systems are designed based on worst case delays in which performance suffers the most.

Sensory systems are typical users of the DVS technique. It is quite common that a sensory system acquires information from its environment and, after processing, records the final results in its local memory. These results can be wirelessly sent back to servers when some events are triggered, such as an alarming situation or a read request. The raw data collected by sensors usually does not require high speed processing power as long as there is not an overflow risk. That is, processing can be executed as slowly as possible and can take a long time, until the next round of data collection, which means supply voltage can be reduced significantly during this period. While transmitting the recorded results, or at certain periods when data collection is overwhelming, the processor needs higher processing frequency and, as a result, higher voltage levels. This shared behaviour among almost all sensory systems is realised by DVS techniques. However, the challenges of implementing reliable sub/nearthreshold circuits are the major difficulties in implementation of such sensory systems.

Although minimum energy point can be taken advantage of by DVS, it does not always reside in the subthreshold region. For example, minimum energy point can change depending on activity factor, that is, components with less activity such as SRAMs tend to have higher minimum energy V_{DD} near threshold voltage [44]. On the other hand, as will be seen in section 2.2.3.1, a compromise in minimum energy point leads to a large performance gain. Therefore, techniques used for achieving minimum energy can be used in many applications and not only limited to low performance purposes such as sensory systems (as section 2.2.3 discusses).

There are many recent review papers covering the subjects of sub/nearthreshold digital design and their challenges. The goal of this chapter, however, is to cover a broader range

of designs, to show the interrelation of different solutions for low power digital design and review the recent updates and new advances in the ultra low power era.

2.2.2 Previous Literature Reviews

In this section some useful and comprehensive previous reviews are briefly mentioned, as the objective here is to minimise reviewing previous studies if has already been done.

Authors in [29] presented a detailed review which discuss the advantages/disadvantages of subthreshold design, and mathematical equations required for the subthreshold circuits. The paper also covered variation problems briefly and presented subthreshold design techniques for Static Random Access Memories (SRAM) design followed by design techniques for multiprocessing. These discussions constitute [29] one of the most inclusive early papers.

In [45] the authors have focused on variations and leakage reduction, to a greater extent, and DC-DC converters (a fundamental building block in DVS systems [46-47]) have also been discussed.

In one of the recent reviews, the authors in [48] presented a detailed review of subthreshold design by exploring subthreshold modelling methods, a few leakage reduction techniques, a short discussion about pipelining/parallelising and a good review of low power SRAMs. A discussion of other device technologies like Double Metal-Oxide-Semiconductor Field-Effect Transistors (D-MOSFETs) has also been included which will not be taken into account in this chapter.

The remainder of this chapter focuses on challenges that have come to light since the above mentioned reviews and will explain the recent advances. Subsequently, results are compared to find out the effectiveness of these solutions and finally attention is drawn to remaining challenges.

2.2.3 Sub/nearthreshold Design Challenges

As previously mentioned and due to ultra low voltages, subthreshold design usually has to deal with two major challenges, namely performance degradation and subthreshold PVT variations. There are many techniques already published by researchers in these two areas. Pipelining (details in section 2.2.3.1.1) is the main technique used to address the

performance degradation. However, techniques for addressing PVT variation are more diverse and have been divided into two sections of SRAM (details in section 2.2.3.2.1) and logic design (details in section 2.2.3.2.2). The techniques proposed in this thesis have been developed as a result of this extensive literature review which aims to identify the challenges and provide potential ideas to address them.

2.2.3.1 *Addressing Performance Degradation*

Pipelining is popular in superthreshold designs because it usually needs many circuitries for controlling the pipeline, resulting in a leaky system, when operating in subthreshold voltages. As a result, nearthreshold voltage operation is considered for pipelined circuits by many researchers.

In the superthreshold region, energy is extremely sensitive to V_{DD} due to the quadratic dependence of active energy on V_{DD}. Therefore, voltage scaling down to the nearthreshold voltages yields a 10 times energy reduction at the expense of nearly 10 times performance decrease [49]. Interestingly, energy reduces by only ~2 times when V_{DD} is scaled further down from the nearthreshold region to the subthreshold region, but at the same time delay rises dramatically by 50–100 times. As a result, authors in [49] concluded that a huge amount of performance could be recovered by backing off a bit and working in nearthreshold region, when higher performance is needed.

Authors in [50] also found that the rate of delay change with respect to supply voltage change ($\delta t_d/\delta V_{DD}$) is very significant in the nearthreshold regime. A 200mV change in supply voltage from 0.3V to 0.5V leads to approximately a 30 times change in performance. The concept was proved by offering a two V_{DD} design that were only 50mV apart and, as suggested, a small voltage supply rise caused very considerable speed-up. Dual-V_{DD} assignment was applied at the level of entire rows in the layout in order to restrict the surplus cost of dual voltage distribution and no level shifters were utilised because these dual voltages were not more than 100mV apart. It was shown that the maximum speed-up (with V_{DDL} = 0.4V and V_{DDH} = 0.45V) was ~45%, which was equal to what is obtained by powering up all cells to the V_{DDH}.

The above studies demonstrated that nearthreshold voltages are necessary for higher speed demands. Keeping this fact in mind, this section continues with articles that have tried to increase performance more by pipelining within the nearthreshold region.

2.2.3.1.1 *Pipelining*

Pipelining in the subthreshold region leads to leaky circuits, as discussed before, and as a result, if a subthreshold pipeline is still necessary due to the performance demand, it has to be very simple. For example in [51], whilst considering subthreshold operation difficulties like PVT variations, it was concluded that variations are distributed over the length of a path which makes shallow pipelines with high FO4 delay per stage more advantageous. Hence, in [52] a 2 stage pipeline implementation was selected for the processor. Shallow pipelining is encouraged in [53] too, with 2-4 stages for the implemented FIR filter operating in the minimum energy point, suggesting that pipelining can improve performance with little to no negative effect on energy per operation.

Generally, there are three main sources of energy consumption in pipeline: instructions, circuits (including datapath, registers and control), and the synchronisation plan. As will be explained in detail, each instruction has its own specific energy usage and depending on synchronisation strategy running between stages, active energy differs. While section 2.2.3.2.2 covers data path and control circuits, some specific concerns are discussed in this section. Moreover, as some techniques useful for nearthreshold pipelining (such as how to cope with PVT variations in pipeline) are located in low power superthreshold researches, inevitably these studies have also been included (presented in Table 2.2).

Considering instruction isolation first, it has been reported in [54] that as supply voltage reduces, ADD instruction operates correctly until 0.74V, while logical instructions (XOR and AND) tolerate V_{DD} scaling down to 0.68V. Therefore it was proposed that isolating ADD operation lets the ALU operate at 0.68V by providing 2-cycles for ADD operation and 1-cycle for other instructions, resulting in more power savings. For ADD, the ALU saved another 23% power because halving the frequency at the reduced V_{DD} also decreased power consumption at the cost of performance degradation.

Authors in [55-57] have considered both instructions and datapath. Any possible delay failure in specific instructions such as ADD (under process variation and voltage scaling) was prevented by adaptively extending the clock period to two-cycles while all standard operations were single-cycle. Execution datapath was changed so that whenever a failure in operations became probable, those operations could be executed in two cycles. The prediction was done by utilising a small pre-decoding logic. In addition, if the temperature exceeded a threshold value, a lower supply voltage (V_{DDL}) was applied to the execution unit. Once the temperature fell below the threshold, nominal supply (V_{DDH})

would be restored. It is also interesting that during execution, only execution stage received V_{DDL} while all other pipeline stages received V_{DDH}. In this study, however, effect of process and voltage variations has not been discussed.

Table 2.2. Comparison of pipelining strategies

References→ / Criteria↓	[54]	[55]	[58]	[59]	[60]
Technique	Instruction Isolation	DVS and critical path isolation under temperature variations	Variable clock in times of process variations	soft edge flip-flop	Flow-through latch between stages and selection of different voltages
Technology	45nm	BPTM 70nm	90nm	PTM 65nm	PTM 32nm
Circuit	32-bit in-order 5-stage dual-pipeline processor with IA32	in-order superscalar pipeline with the Alpha ISA	32-bit microprocessor	34-bit pipelined adder	6 stages pipelined FPU
Frequency	1.25GHz	1.5-3GHz	~0.1-1GHz	2-2.5GHz	Improves BIPS/W by 47% (actual frequency not reported)
Area and/or frequency impact	28% performance reduction due to instruction isolation	~4.5% area overhead / 3.4-11% frequency overhead	2.6% area overhead/13% -50% performance improvement	5-20% performance improvement	40% performance improvement
Energy/Temperature impact	13% power reduction	Reduces temperature by 6.6-9%	3% energy overhead	19% power saving (4.9mW)	Not reported
Min Voltage	740mV for ADD 680mV for XOR and AND	700mV	Scaling from 1.2V to 1V	Scaling from 1.2V to 1.05V (5-20% V_{DD} reduction)	Scaling from 1.4V to 0.95V
Advantages	DVS enabled	Temperature variations tolerant, DVS enabled	Low energy and area overhead	Rather large power reduction	Rather large performance improvement
Disadvantages	Performance reduction	PV variations not discussed	Not supporting very low voltages	Not supporting very low voltages, PVT variations not discussed	Not supporting very low voltages

In synchronisation, it is often attempted to modify the clock so that the slack time in datapath is used for compensation of variations or voltage scaling. For example, by considering instruction, datapath and clocking, authors in [58, 61] associated a variable

delay with each pipeline stage, and a table of delays was adjusted to meet the delay of each specific instruction. Whenever the delays of all stages were elapsed, a new clock was created and therefore, some clocks were shortened and the overall speed was increased. This variable delay unit was located close to the corresponding datapath so as to be subject to similar PVT conditions. A delay selector reads the inputs of the pipeline to choose an appropriate delay value from the operation selection table. Despite being low energy and area overhead, sub/nearthreshold voltage domain operation is lacking in this research.

In [59], which focuses on synchronisation plan, a new soft edge flip-flop (SEFF) was proposed to postpone the clock of the master latch to produce a window along which both master and slave latches were active. This window, which is called the transparency window, allowed timing slacks to pass between adjacent pipeline stages. The delayed clock was created by employing an inverter chain and sizing them in order to maintain the desired delay. Available slacks at stages were passed to the previous stages, providing previous stages with a surplus of borrowed time. Since positive slacks were available in all stages of the pipeline, as a result of this time borrowing, the clock could be increased or circuit voltage could be decreased to reduce the power consumption. In spite of proposing a very energy saving technique, authors have not however discussed the PVT variations effect and sub/nearthreshold voltages.

In [60, 62], a design was presented with a flow-through latch between two stages so that clocking of that latch added an extra half cycle to the pipeline. This half cycle provided extra time borrowing to absorb delays due to process variation in the previous stages. Gating the latch and switching between the modes with and without the extra latency allows for post-fabrication tuning. A voltage interpolation was also used to deliberately select different effective voltages needed for each stage to run at a single nominal frequency. Therefore, if the pipeline ran slowly due to process variation, there were two ways to obtain the nominal operating frequency. One option was connecting more stages to V_{DDH} so that the effective voltage increased. Another option (as discussed before) was to extend two stages with a latch in between to a single stage to provide additional time for execution while decreasing energy by switching more stages to V_{DDL}. Once again, the effect of sub/nearthreshold voltages have not been considered or reported in this study.

Considering the effect of clock in power consumption, it should be pointed out that asynchronous pipelines are playing a vital role in recent designs too. Although they are

beyond the scope of this review, the idea of eliminating the clock in a synchronised world can be found in [63] with a Moebius pipeline proposed. In this pipeline, a stage sent a signal called “COMPLETE” to the previous stage when the computation was done and at the same time held the result until the “COMPLETE” signal from the next stage came. This way, in addition to saving clock power, the available slack in the path was utilised very efficiently and, besides, variations were dealt with in a better way.

In a survey of pipelining [64], different architectures were also examined. This survey states that the single issue in-order architecture, (similar to the above studies), is appropriate for very low energy design points (and is used in the FFT of Chapter 6), whilst the quad-issue out-of-order is only suitable at very high performance applications [64]. It was also discovered that the dual-issue in-order and out-of-order processors were efficient for many different kinds of design performances.

By referring back to Table 2.2 it can be concluded from reviewing the above studies that for optimising power and speed, using synchronisation strategies are as important as instruction isolation. It is also suggested that minimum achievable voltage when using energy costly pipelining techniques is still beyond the subthreshold regions. As DVS techniques, used in sensory products, seek the lowest energy consumption by working in subthreshold regions, this proves that performance increase in pipelining cannot justify the high energy overhead of their implementation, unless new techniques are put forward.

Having said that, authors in [65-66] claim a new and opposite idea to [51] and [52] (developed in the same research group and reviewed above) by suggesting the concept of super-pipelining, that is, reducing the depth of the pipeline from long delays of 63 FO4 to 7 FO4 delays and increasing the number of stages from un-pipeline to 16 stages which gains ~46% energy saving because of switching energy and leakage current reduction. With the higher performance of about 31%, the higher delay resulting from lowering the voltage is outweighed, and 35% lower minimum voltages can be achieved provided the register cells do not create high delay overhead. Therefore, this idea of super-pipelining (already examined by the above authors in an FFT implementation [67]) is employed in Chapter 6 too and a register cell (along with a low power standard cell library) is implemented for this purpose in Chapter 2 to verify this most recent concept in subthreshold design and to make further improvements using the proposed technique in Chapter 3.

2.2.3.2 *Addressing PVT Variations*

Minimum energy can be typically achieved when V_{DD} scales down to the subthreshold region [42]. Subthreshold systems have been proven to be fully functional below 200 mV [68]. Nevertheless, there are still some important challenges for subthreshold designers and the most significant one is variations. Process, voltage and temperature (PVT) have exponential correlations with the subthreshold current, as mentioned before, and tiny PVT variations, as a result, have a huge impact on performance and energy consumption. In this section, recent techniques to overcome variations are reviewed in the two sections of SRAM and logic design.

2.2.3.2.1 *SRAM Design*

SRAMs maintain a large proportion of power consumption in chips from 30% in runtime to 90% in standby time [69-70] . This is because more area (more than 50%) is usually allocated to on-chip caches with every new processor generation. This is due to the attractive characteristics of SRAMs such as low activity and high transistor density [19] and is also due to power and performance optimisation as a result of placing memory as close as possible to the processor.

The commercially available high density SRAMs are inoperable below 0.7V [71-72] and if an SRAM is compiled and created by a commercial memory compiler, it cannot be used in sub/nearthreshold voltage domains. As a result, when such SRAMs are used, level shifters are required to shift the voltage level of those signals going to an SRAM from a subthreshold domain. Section 3.3.3, therefore, designs a level shifter, as part of the created low power standard cell library, which remains functional from 0.3V to 1.2V. This level shifter is used in the FFT structure with two voltage domains (presented in Chapter 6). A circuit with two or more voltage domains also has to follow standard designing steps (called IEEE 1801 Standard) which are discussed in section 3.4.

As explained before, low power applications like wireless mobiles or sensor processors usually need to have two modes of working i.e. high performance and low power/standby. The latter mode is usually the source of leakage power consumption which mostly happens in SRAMs (especially in standby time). As mentioned before, when the activity and voltage reduce in low power design, leakage and PVT variations will become the most important factors. This rule is applicable to subthreshold SRAMs too and this

section continues with a review of recent techniques in dealing with these issues. It is worth pointing out that leakage reduction techniques always help tackle PVT variations in SRAM because they usually make SRAMs more robust and error free; therefore some leakage reduction techniques have also been reviewed. Table 2.3 compares the most important criteria for subthreshold SRAM designs.

Stacking, which is a technique that stacks redundant transistors to reduce leakage (by increasing the series resistance), can reduce leakage in the bitcell by stacking in the cross-coupled inverters of SRAM as well as other retentive gates [69]. It was found that the leakage sensitivity to the number of stacked devices becomes linear for more than two stacked devices. Therefore, a stack height of two was utilised. Furthermore, it was shown that increasing the length (L) of the devices in the cross-coupled inverters leads to a more area-efficient reduction in leakage; this represents a technique utilised in the proposed circuit in this thesis too. It was also observed that IMEM (Instruction MEMory) and DMEM (Data MEMory) consume 89% of the standby power while the CPU consumes only 7% of the power when it is power gated. A particular architecture was proposed for storing frequently used procedures in Instruction Read Only Memory (IROM) while storing application specific instructions in IMEM. Since ROM can be power gated during standby mode, it is beneficial to put as many instructions in IROM as possible. For further leakage reduction, in [69], a specific entry in DMEM is power gated only if a special free-list indicates that the entry is idle. Despite being low energy, this techniques used 14 transistors for a memory cell which created a large area overhead and also Static Noise Margin (SNM) has not been discussed (a critical factor on susceptibility of SRAM to noise).

Apart from gating, sizing, stacking and usually various write/read assists are used for preventing subthreshold region failures in SRAMs. Write/read assists that are designed for subthreshold region, however, might severely impact high-voltage performance [73]. To address this fact, authors in [73] proposed a reconfigurable SRAM with three different write-assist architectures. By combining different circuits optimized for both subthreshold and superthreshold voltages and employing reconfigurability to switch between them, their SRAM operated from 1.2V down to 250mV.

Table 2.3. Comparison of effects of different techniques in SRAM designing

References→ / Criteria↓	[69]	[73]	[74]	[75]	[76]	[77]	[78]
Technique	Stacking and length sizing, power gating in standby and compression	Buffered read, and reconfigurable DVS support	Buffered read, control of supply, buffered voltages and SA	Single-ended, 2% bit redundancy, body, header and footer bias	Segmented Virtual Grounding	column-wise write, DCVSL read control	Schmitt Trigger based
Main Novelty	Using ROM, new cell design	Reconfigurability	Redundancy in SAs	New cell design	Superthreshold read	Soft-error addressing	Using ST Design
Technology	0.18µm	65nm	65nm	0.13µm	0.13µm	90nm	0.13µm
Size (bits)	64	64K	256K	2K	40K	32K and 49K	4K
Frequency (KHz)	~35@450mV 121@500mV	200,000@1.2V 500@250mV	25@350mV	205@300mV 21.5@210mV	100,000@ 400mV	581.4@300mV 0.5@160mV	620@ 400mV
Area overhead	910% to 6T	Not reported	30% to 6T	42% to 6T [79]	8% to 6T	61% to 8T	~200% to 6T
Total leakage/size (pA)	Not reported	~700@1.2V ~30.5@250mV	~24@350mV ~21@300mV	~122@ 300mV	27@400mV	~24.11@ 300mV	~90@ 400mV
Energy / access/size (fJ)	~0.000058@ 500mV	0.167@400mV	~0.396@350mV	0.488@ 340mV 0.38@300mV	0.17@400mV	0.056@ 300mV (Write) 0.094@ 300mV(Read)	50% and 18% lower dynamic and leakage power to 6T@ 175mV
Min Voltage (mV)	450	250	350	193	360	160	160
transistors	14T	8T	8T	6T	6T	10T	10T
Bit error rate	Not reported	Read SNM eliminated	Read SNM eliminated	2%@120mV	3.5%@ 330mV	60.3mV mean Read and ~91mV mean Hold SNM@ 300mV	~56.5mV mean Read and ~118mV mean Hold SNM@ 400mV [80]
min energy voltage (mV)	450	400	350	340	Not reported	160	160
Pros	Low energy	High performance	Low read error rate	Variability aware design	Very high performance	Low energy, High read SNM	Low voltage, High read SNM
Cons	Large area overhead, SNM not discussed	PVT variations not discussed	Low frequency	Still high leakage current	Not DVS enabled	Leakage increase at typical temp	Large area overhead

Effectiveness of DVS was also examined in [73]. Consider a memory in low power mode (0.4V) and accessed every 2μs with each access causing active energy consumption. It was observed that leakage power decreased 40 times by scaling from 1.2V (without DVS) to 0.4V (with DVS). But DVS circuitry consumed energy as well and energy consumption in both with and without DVS during low power mode became equal after just five accesses (or 10μs). As a result, only if a system lingers on in the low-power mode for longer than 10μs, then is it justifiable to utilise DVS, otherwise it will consume more energy than the without a DVS system. Although a great example of DVS has been presented in this paper, the effect of variations, as a result of working in subthreshold domain, has not been discussed.

The work done in [74] (which has already been reviewed in review papers [45] and [48]) is similar to [73]. A buffered read was employed to guarantee read stability, and, in order to enable subthreshold write and read, a peripheral control on both the bit-cell voltage and the read-buffer's foot voltage was performed without degrading the bit-cell's density. The authors also amended the Sense Amplifiers (SA) and, by means of redundancy, the problem of area-offset tradeoff in SAs was mitigated, which in return decreased read errors by 5 times compared to up-sizing. Despite working in the subthreshold domain, frequency of the implemented SRAM can be improved with the techniques similar to what is proposed in this thesis.

Instead of using the traditional differential structure, authors in [75] used a single-ended cell with a full transmission gate on one side. By the elimination of the second bitline, the cost of having one additional wordline was balanced. One obvious benefit of this design was the ability of the bitline to be driven from rail-to-rail removing the necessity of using a sense amplifier (which usually leads to density and variability problems in differential designs). Furthermore, the noise was isolated, during a read operation, to the single bitline which made this design essentially more robust to read failures than differential design. During the write operation, the supply voltage was gated on the feedback inverter to make up for the degraded write margins. Upsizing was also utilised to handle process variations. As it will be seen in detail, upsizing is among the techniques that have been adapted in the proposed ideas of this thesis too.

Continuing the discussion on read assists, in [76] four operational modes: Retention, Read, Write and a new proposed mode called Accessed Retention mode (AR-mode) for the SRAM cell were defined. This new mode signified SRAM cells located on an

accessed row which were not selected to be read or written, a similar approach to [73] and [74]. These cells did not discharge their bitlines and hence saved energy. This also increased the read noise margin of the accessed cell. In addition, it was shown that using RBB in the subthreshold region led to a low leakage current for all non-selected cells. Furthermore, due to the superthreshold voltage setting for the selected cells (for read operation) the cell access time was reduced dramatically while the stability of the AR-mode cells was maintained. However, to be able to switch between high performance and low energy modes, it is necessary to use DVS, a technique that was not discussed in their study.

Moving to write assists, in [77, 81] the authors offered a differential 10T bitcell that efficiently split read and write operations and as a result achieved high cell stability. The write assist transistors in the cell were boosted to make up for weak writability. Each four columns was connected to a common ground voltage driver with dynamic-threshold MOS to lessen process variations. The driver's pull-down device was forward-biased during read to increase the drive current. Forward body biasing is also the main technique used in this thesis to increase the performance when PVT variations result in a slow operation.

As was seen before, other cells sharing a word line in some SRAMs are subject to a hold stability problem while a specific cell is being written [77]. Some solutions, that implement adjacent bits as the same logic word, make the SRAMs exposed to multiple bit soft-errors (which is more critical in subthreshold SRAMs). The offered column-by-column write control in [77] caused the hold stability of adjacent cells not to be affected during a write. Dynamic Differential Cascade Voltage Switch Logic (DCVSL) scheme was also used for read access. In this scheme, bitline leakage noise is offset by the drive current of a keeper, providing large bitline swing. While holding, bitline leakage subthreshold current was considerably decreased because of the stacked bitline leakage path.

In [78], however, instead of using read/write assists, a Schmitt Trigger (ST) based 10 transistor SRAM cell was proposed with the idea of making the characteristics of the inverter pair of the bitcell near the ideal inverter, which is fundamental for a robust cell operation. The positive feedback from extra transistors adaptively altered the V_M of the inverter depending on the direction of input transition ($0 \rightarrow 1$ input transition or vice versa). The proposed ST bitcell took advantage of differential operation and delivered a better noise immunity. The idea of altering the V_M of gates to reach a minimum noise

situation will be discussed later in detail as it contributes to the understanding of the proposed techniques in this thesis.

Although cell design plays an important role in decreasing delay and energy, the SRAM architecture is also another effective factor. Multi-tier SRAMs in System On Chip (SOC) design is a common technique used to prevent costly out of chip memory accesses as well as to increase performance.

Besides, the low speed of subthreshold SRAMs limits the ability of subthreshold cores whose speed is usually more than subthreshold SRAMs [44]. As a result, a discussion about optimum subthreshold SRAM architecture is necessary.

Authors in [44] observed that optimal L1 (Level 1) cache size increases from 64KB to 128KB for targets below 76MHz, since L2 (Level 2) starts to relatively consume more energy. Even though a larger L1 causes more energy per access, the energy saved from decreasing L2 accesses (in lower frequencies and because of larger L1) outweighs any increase in the L1. Furthermore, it was observed that optimal energy consumption is obtained at nearthreshold region (400–500mV) and at a frequency of ~15MHz-50MHz.

Another example is given in [82-83] in which by means of a multi level SRAM, a high performance design has become possible. The proposed design supported ultra V_{DD} scaling from a nominal to sub/nearthreshold voltages. In order to reduce off-chip traffic and improve performance and energy efficiency, a large on-chip frame memory (FM) of 10Mbit was embedded, which allowed keeping Video Graphics Array (VGA) frames. However, as discussed before, when dealing with V_{DD} scaling, usual SRAMs cannot work reliably below 700mV. Therefore, a Hybrid Memory Architecture (HMA) was proposed to decrease the access rate from processors to the FM by employing the data locality in the scratchpad memory (SM). Within the proposed HMA, there existed three characterized memories to hold the data: (1) accumulator register: short-term data; (2) SM: intermediate-term data; and (3) FM: long-term data.

On the other hand, nearthreshold operation in logic decreases frequency compared to superthreshold operation. This speed degradation, however, suggests several new and interesting design opportunities about memory system selection [84-85]. Firstly, memory technologies (like 130nm or 180nm devices as discussed in Logic Design section 2.2.3.2.2) and designs that are slower and more energy efficient can substitute timing critical memory designs. This will help to decrease the total energy of the chip while

memory is working in superthreshold voltage and logic in nearthreshold. Furthermore, multiple accesses to memory can be carried out in one nearthreshold clock cycle of logic. This means that more parallel data can be fetched and be processed in one cycle. Finally, the slower memory possibility allows caches, register files and other elements that are originally designed to compensate long memory latency, to be turned off or removed. Therefore, a pipeline was implemented so that in a single cycle of the Single Instruction Multiple Data (SIMD) pipeline, multiple memory access was possible. It was also shown that wider SIMD widths do not always provide less energy consumption because of the additional hardware and increase of critical path delay involved.

However, it must be noted that multi-level SRAMs and caches are intended for near/superthreshold applications and are not justifiable for sub/superthreshold systems.

Finally, in one of the FFT processor implementations in section 6.3, a subthreshold SRAM is used which has been chosen from the above reviewed SRAMs based on the structure and energy per access. For example, the SRAM proposed in [75] has a high frequency and supports aggressive DVS which makes it a possible candidate in one of the FFT examinations.

2.2.3.2.2 *Logic Design*

This section discusses techniques that either worsen the variations or eliminate them and helps identify the relative merits of recent subthreshold techniques in logic design. The possible negative effects of these techniques while decreasing variations have also been considered. Also, delay/performance variations due to PVT variations are discussed in this section.

As mentioned before, stacking has been widely used to increase the threshold and hence decrease the subthreshold leakage [69]. However, this technique has not been used in the proposed ideas in this thesis because of some drawbacks which cause variations and are discussed as follows.

Firstly, although stacked devices exhibit lower current variability, they have a higher probability of logic failure due to insufficient output swing especially at lower supply voltages [86].

Nevertheless, it has been reported that insufficient output swing can be compensated by upsizing the stacked devices (which comes with an area overhead price) and authors in

[86] took advantage of this to make up for degraded output levels in stacked devices. The failure rate of 0.13% was targeted and the proposed 32-bit adder with constant yield sizing worked for lower voltages until 300mV [86]. In addition to making up for PVT variations, upsizing also reduces Drain Induced Barrier Lowering (DIBL) effects and as a result leads to lower power consumption [87]. In fact, an upsize by several nanometres at 32nm node in a Fan Out of 4 (FO4) inverter can reduce energy per operation by 65% at 10MHz in 0.3V and by two orders of magnitude at 10kHz in smaller than 0.2V [88].

The second problem in stacking is the reduction in current in the stacked devices which results in a loss of speed in the subthreshold region. However, this can also be offset, should stacking be used, by body biasing [89]. For example, in [89] authors proposed complementary hybrid latch flip-flop (CHLFF) for ultra low power applications and Forward Body Bias (FBB) was used to increase the speed of the PMOS stacked network. It was found that reducing the supply voltage to 0.3V in an NMOS stacked flip-flop (FF) causes some failures in corners. After applying FBB to PMOS network, the supply voltage could be reduced to 0.23V, the speed was increased three times (to 5MHz), and the power consumed was 0.159μW. The same idea was also applied to a sense-amplifier based flip-flop (SAFF) in [89] and the improved Complementary-SAFF (CSAFF) worked properly for supply voltages of even less than 0.3V and consumed 0.144 μW with double the speed (of 5MHz).

FBB is the focus of this study as it enables aggressive DVS, a necessary practice to minimise the consumed energy. In lower frequencies (~100KHz), however, Forward Body Bias (FBB) increases minimum-energy due to threshold voltage reduction [90]. On the other hand, FBB can help make up for performance loss if its energy consumption is controlled with a technique similar to what is proposed in this thesis. Instead, Reverse Body Bias (RBB) was used in [90] for an 8-bit multiplier and at low frequencies with 0.2V power supply. Moreover, it proved to be more efficient (70% less energy overhead) than DVS. It was also found that global process and temperature variations might suggest a wrong frequency estimation of minimum-energy point. This may result in an improper device/V_{th} selection specific for low-power (LP) or general-purpose (GP) design and lead to energy overhead higher than 200% at the worst-case corner for energy [90]. The technique proposed in this thesis is also capable of supporting RBB when accordingly tuned, and will be discussed later.

As stated before, DVS causes a significant energy saving because minimum possible voltage is chosen to meet the required performance. Although it is powerful in controlling dynamic energy, DVS cannot deal with the static energy which is a rapidly growing problem in short-channel devices and is dominant in low activity systems. However, simultaneous static and dynamic energy management are made viable through dynamic voltage and threshold scaling which tunes both supply and body bias voltages at the same time. In fact, minimum total energy for any required performance can be gained by this technique in digital systems fabricated on a 0.1μm technology or lower [91-92]. For low power high speed applications, this technique has been implemented successfully [92] and has also been employed in this thesis.

There are techniques that manage V_{th} and V_{DD} at the same time, to minimise the energy consumed by means of algorithms that identify the appropriate V_{DD} and V_{th} depending on the work load. Sometimes V_{th} adaptive biasing is used to gain the minimum energy as it can help reduce leakage in lower supply voltages (but variations are forgotten) [92] and sometimes it is utilised to compensate PVT variations while scaling is missing [93]. There are cases that claim to handle both at the same time but they use very complicated architectures only suitable for high performance GIPS processors [94].

However, with the increase of dopants in the below 100nm device's channels to cause stronger inversion, technology scaling counteracts the body biasing effect. Despite this fact and as this study proves, the impact of PVT variations still forces designers to employ adaptive body-bias techniques. On the other hand, as devices work using the subthreshold current in the subthreshold region, delay worsens substantially in this region too. These challenges need to be addressed to make the widespread exploitation of subthreshold design possible [95].

As mentioned before, process variation is usually divided to inter-die and intra-die variations [16]. Inter-die and intra-die variations can also occur at the same time. Likewise, intra-die variations are classified into systematic and random. In systematic variations, devices that are spatially correlated (e.g. are close together) experience the same effects while random variations can occur accidentally even for spatially correlated devices. Systematic intra-die (as well as inter-die) variations can be addressed by adaptive body-bias techniques [96].

At superthreshold voltages (nominal voltages), body biasing is a very familiar technique to designers for adjusting delay and leakage to overcome inter-die PVT variations.

Although efficient in the superthreshold region, the impact of body biasing is especially sensed at the subthreshold voltages because of the exponential increase in the sensitivity of devices to the threshold voltage (as mentioned in section 2.1.2). For example in a typical 90nm technology, if threshold voltage is changed by 50mV at a 1V supply voltage, delay varies 13% whereas it results in a 55% delay increase at a 0.45V supply voltage [97].

Moreover, noise margin susceptibility is also a major challenge at subthreshold voltages. As threshold voltage (V_{th}) is managed by an independent doping process, PMOS and NMOS threshold voltage can differ considerably. This, for example, can result in an insufficient high output voltage at the fast NMOS slow PMOS corner (in which NMOS devices are much leakier than PMOS ones) or an insufficient low output voltage at the fast PMOS slow NMOS corner. Consequently, at process corners, not only noise margins can be violated, but also either rising or falling time is substantially lengthened which in return results in increased timing breaches.

In a superthreshold circuit, the ratio between the channel mobility of the majority carriers in PMOS and NMOS transistors, called the β-ratio or P/N ratio, is usually adjusted so that the noise margin of the gate is maximised. This optimal ratio in superthreshold region, however, is not exactly similar to the optimal ratio in subthreshold region. If designed to only maximise the superthreshold noise margin, this P/N ratio can result in a high skew between PMOS and NMOS devices in subthreshold region.

As PMOS and NMOS can be controlled independently using body biasing techniques, this opens up many opportunities for designers to optimally tune the β-ratio and prevent V_{th} mismatch problems. For example, authors in [98-99] used V_{th} balancing schemes which enabled them to implement a supply voltage scaling from superthreshold to subthreshold voltages. Their body biasing technique adapts P/N-ratio dynamically while voltage scaling.

However, in [98] the problem is that there is always an FBB in subthreshold for either NMOS or PMOS network which makes the circuit leaky when at TT, FF, or SS corners. In addition, the body bias circuit in [98] uses large switches to disconnect the circuit from delivering FBB on superthreshold voltages which means both controlling and area, not to mention energy, overhead. In addition to being low power, the FBB technique, proposed in Chapter 4, does not need any controller as PVT variations plus voltage scaling are automatically sensed.

Complexity of body biasing generators can also be another source of energy overhead. Due to its complexity, for instance, the body biasing circuit used in [99] incurs energy overhead on the chip and, therefore, needs to be disabled on superthreshold. However, the proposed body biasing circuit in this thesis works in subthreshold region for any V_{DD} and hence imposes minimal power overhead.

Authors in [29, 95, 100] also studied the capability of adaptive body-bias techniques to address PVT variations and implemented a subthreshold processor to show its effectiveness. They also proved that a body bias which optimises P/N ratio for noise margin also minimises energy per instruction.

However, in high temperatures and subthreshold voltages, transistors become so leaky that it is very damaging, in terms of total energy consumption and not energy per instruction, to provide the optimal noise margin P/N body bias. For example, a simple simulation on an inverter, with PMOS device sized twice the NMOS device, V_{DD}=0.4V and T=75°C, shows that an optimal FBB applied on PMOS improves noise margin by only ~1% while increases static current by 10%. Furthermore, delay is very sensitive to body biasing, at subthreshold voltages, which helps prevent timing violations at extreme voltage scaling. This, therefore, means that a non-optimal FBB is needed to overcome extreme voltage scaling situations. For instance, in the same inverter with V_{DD}=0.3V and T=25°C, when the non-optimal FBB of 0V is applied to PMOS, delay is improved by 2.23 times while noise margin worsens by 1.17 times when compared to the optimal FBB. On the other hand, noise margin for data-path is not of primary concern at superthreshold voltages which means forward body biasing can be cancelled to prevent any extra energy consumption regardless of optimal P/N ratio. Unlike [95], the FBB technique proposed in Chapter 4 is cancelled at high temperatures, FF corners, and superthreshold voltages, and is fully applied when at subthreshold voltages.

Above studies clearly suggest that employing body biasing in controlling PVT variations is essential when it comes to subthreshold designing and DVS techniques.

The following papers have also considered more extensive circuits like processors, and their performance is compared in Table 2.4. For example, the above claims about Body Biasing (BB) can be verified by looking at [78] which uses three BB voltages: forward, zero and reverse BB produced by a BB Generator with a PV monitor which is an inverter (temperature variations were not studied). BB also was utilised to prevent failure caused by NMOS/PMOS mismatches which in practice modulates the β-ratio adaptively in sub-

V_{th} region such that the switching threshold (V_M) will be close to ½V_{DD}. The inverter V_M is compared against two reference voltages. If $V_M<V_{REF1}$, signifying that the NMOS is stronger than the PMOS, forward BB will be then applied to the PMOS network. Conversely, if $V_M > V_{REF2}$, the NMOS network is forward body biased. It was also pointed out that the application of BB can successfully alter the β-ratio and decrease the distribution of V_M, leading to increased robustness in subthreshold circuit [78].

Authors in [101] used the same idea and proposed a configurable V_{th} balancer using BB to reduce the V_{th} mismatch between NMOS and PMOS transistors, so that both the functional and the timing/speed yield were increased. This speed improvement was because both the PMOS and NMOS transistors were forward-biased when the balancer was turned on. The V_{th} balancer, however, applies FBB at Typical-Typical, Fast-Fast and Slow-Slow corners which accounts for a faster design but FBB could have been cancelled in such corners, especially the Fast-Fast one, to save more energy and to be able to scale voltage even more. Voltage and Temperature variation results are also missing. The logic gates with more than four parallel transistors or four stacked transistors were discarded from cell library to decrease leakage current variability, and ratioed logics were substituted with non-ratioed logic [101] (a practice that was adapted in this thesis too). Despite being fast while preventing mismatches, this study [101] is not concerned about other forms of process variations or temperature variation.

Although above mentioned studies decreased voltage for specific circuits, more discussion is needed on use of Dynamic Voltage Scaling (DVS) in the general purpose (GP) designs. DVS is considered inferior to Dynamic Frequency Scaling at facing variability in subthreshold voltages when energy efficiency is the key performance criteria [52]. In fact, it is not limited to the choice of DVS and DFS only. A study in [52] clarifies that many of the area optimal and performance optimal designs, at superthreshold voltages, are not suitable for subthreshold voltages (therefore their library was re-characterised for subthreshold operations and, to maximise the robustness, some cells were eliminated from it). Addressing temperature variations, however, has not been fully explored in this study [52] which is one of the main targets in the proposed technique in this thesis.

Table 2.4. Comparison of effect of different techniques for further power-performance improvements

References→ / Criteria↓	[78]	[101]	[52]	[102]
Technique	ABB	V_{th} balancing by BB	Frequency Scaling	ABB and width/length sizing
Technology	0.13µm	65nm	0.13µm	0.13µm
Circuit	8x8 FIR Filter	JPEG co-processor	general-purpose sensor processor	8-bit processor
Frequency	98KHz@280mV 240Hz@85mV	2.5MHz @400 mV	833KHz@200mV	77-354KHz
Performance Impact	2.6x faster@200mV 1.25x faster@1.2V	Delay improvement from 14 ns to 10 ns	1.09x faster due to library selection	3.6x improvement@300mV
Energy/Power Impact	40nW@85mV	0.75pJ per cycle@400mV 1.0pJ per cycle@450mV	2.6pJ/instruction@ 360 mV	3.5pJ/inst@350mV @354KHz 515fJ/inst@290mV @77KHz
Min Voltage (mV)	85 [99]	350	200	140 (24% reduction to zero BB)
Min Energy Voltage (mV)	85	350	360	350 (total) 290 (core)
Advantages	Mismatch prevention	Mismatch prevention, Fast	Frequency optimisation	Preventing PT variations, L sizing
Disadvantages	Temp variations not addressed	Not energy conservative, VT variations not addressed	Temp variations and mismatches were not fully investigated	A complicated Off-chip BB system

It was also reported in [102] that DVS is more energy efficient for high target frequencies (i.e. GP designs) while BB is more energy efficient for low target frequencies (low power designs) over the frequency range of 30–300 kHz. In fact, [102] verifies [52] for use of DFS and agrees that DVS should be substituted in subthreshold design with more energy efficient techniques. The authors in [102] have again used BB for eliminating performance variations. However, in contrast with the proposed technique in this thesis, the BB solution in [102] is a complicated off-chip BB system incurring energy and implementation overhead.

Using the above studies, it can be concluded (presented in Table 2.4) that body biasing has been used in many low power processors to reduce power consumption by compensating for the variations and mismatch between V_{th} of pullup and pulldown networks. This shows that creating a body biasing technique is the best approach, currently, to tackle subthreshold issues and maintain superthreshold performance; although proposing a comprehensive idea to simultaneously deal with PVT variations while DVS, coping with energy and area overheads of body biasing circuits and

improving severe delay reduction and variation, should be an indispensible part of this technique.

Designed based on the challenges reviewed in logic design, the proposed technique is mathematically analysed in Chapter 4 and simulated in Chapter 5. Chapter 6 takes advantage of the pipelining to boost the performance of the examined FFT. Also, two of the reviewed subthreshold SRAMs are used to help simulate the power/performance of another implementation of the FFT processor.

Before proceeding to a final conclusion, though, for the sake of completeness of this review, a few other studies are also mentioned. Recent studies show that one should certainly consider the effect of sizing in addition to BB and stacking. The Body Biasing techniques are usually chip level whilst sizing can be applied at both chip or block level and also along gate width or length. For selection of the most appropriate one (chip/block, Width/Length), [100] [102] investigate some experiments at V_{DD}=300mV. It was shown that a processor with W sizing (Proc B) and another one with both W+L sizing (Proc C), along critical paths, are 22% and 85% faster, respectively, than a processor with minimum sizing (Proc A). But for Proc C this improvement came at a 14% energy penalty with respect to Proc A that could alternatively be achieved by a ~7% energy penalty with increasing V_{DD} by 20-30mV in Proc A. This suggests that although L sizing is superior to W sizing, it is only appropriate for block-level performance tuning, and not as an entire chip performance variations solution.

Beside sizing, it has to be pointed out that a processor implemented in a 0.18μm technology is 7.7 times larger than a similar processor in a 65nm technology, but analysis shows that total energy is reduced by 647 times [69]. This is a very desirable trade-off, especially when the size of a product is determined by the battery size. Moreover, in [90] the authors verified the same idea by stating that minimum energy level is 30% higher in 45nm technology (@30MHz) than in 130nm technology (@0.7MHz).

2.3 Conclusion

The extraordinary speed of semiconductor industry growth has been more or less the same since 1965 resulting in consequential challenges in circuit power consumption and process variations, demanding low power variation-aware techniques to deal with cooling, battery life time, reliability, and variation issues. DVS was introduced as a technique for low power processors and all recent sub/nearthreshold techniques,

therefore, were reviewed to find the best solutions in SRAMs and pipeline architecture. Body Biasing was identified as the best solution for addressing the DVS consequences. Based on this literature review, a technique is proposed in Chapter 4 for a body bias generator working in aggressive dynamic voltage scaling and severe PVT variations. The SRAM and pipeline review will help in Chapter 6 in the practical simulation of the proposed technique.

3 Design of a Standard Cell Library

3.1 Introduction

In this chapter, the design flow used for creating the standard cell library as well as the test circuits used in Chapters 5 and 6 is briefly explained. Unified Power Format (UPF) standard is briefly explained and the recent idea of variation aware standard cell design is also examined. This chapter gives a concise insight into what a low power design and implementation may involve and how to approach it.

3.2 CAD tools and Design Flow

Figure 3.1 illustrates the design flow used in this study to design, simulate and verify the test circuits. The first step in digital designing is specifying the electrical and architectural specifications and describing them in an RTL (register-transfer level) abstraction by coding the intended circuit in a hardware language (here Verilog) and subsequently testing its functionality (name of CAD tools mentioned in parentheses). The verified design and the standard cell library, whose design flow is explained in section 3.3.4, together with design constraints are fed into the synthesis and timing tool.

If the initial block level and pre-layout static timing and power requirements are met, the produced netlist is placed and routed using the foundry provided technology files and the standard cell library. To help save time, timing constraints can be forward annotated to the place & route tool as well. A better estimation of timing and power requirements is gained at this stage but a few more steps need to be taken for more accurate (and time consuming) HSPICE simulations.

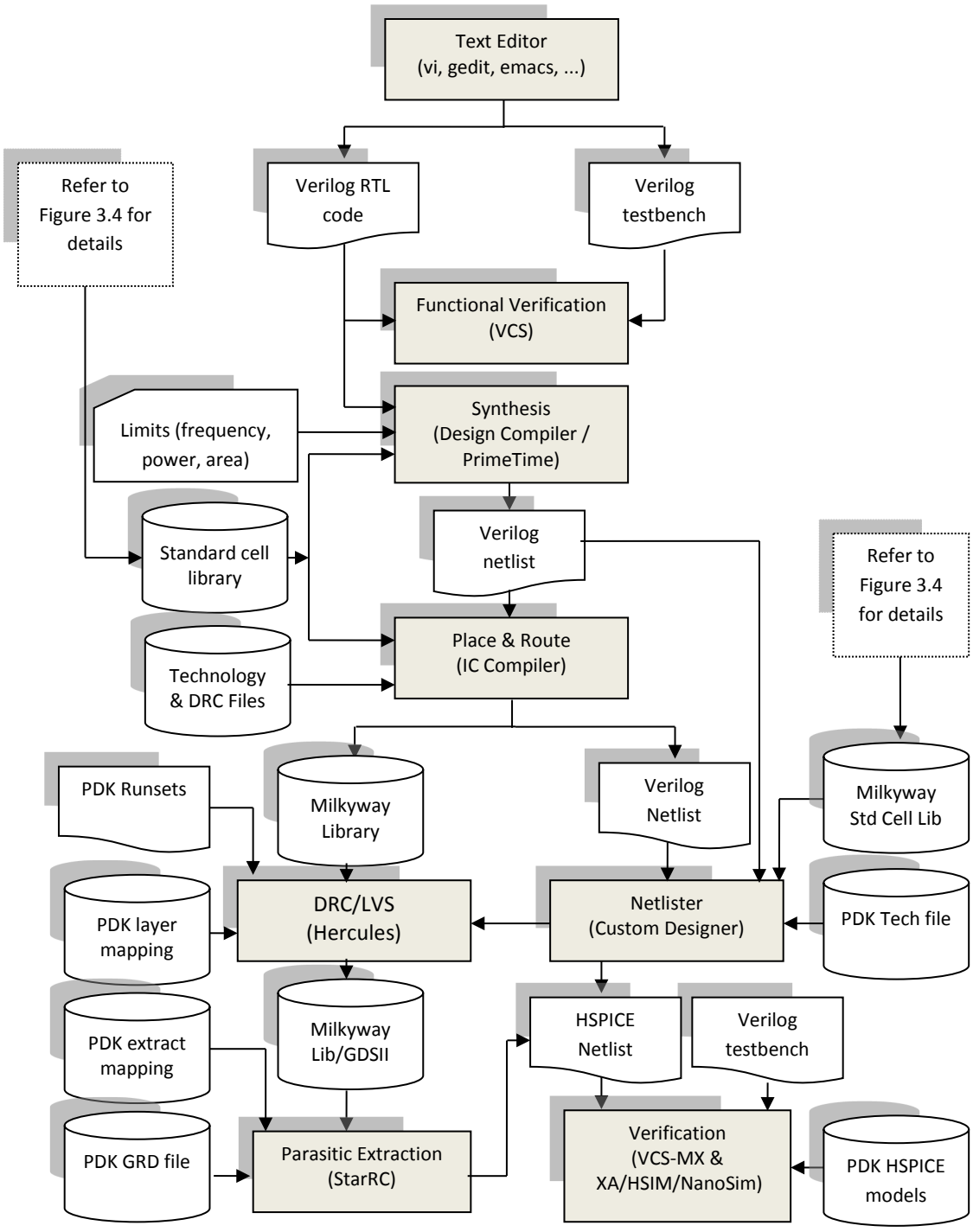


Figure 3.1. Design Flow used for implementing the test circuits

After design rule check (DRC) and layout versus schematic (LVS) verifications, HSPICE code can be extracted and mixed digital/analogue simulation tools employed to verify the final functionality and examine the exact power and timing requirements in all possible working conditions. If the simulations results are satisfactory (maybe after a few adjustments), the layout is ready for finalising and tape out.

3.3 Standard Cell Library

3.3.1 Timing and Power Information

As the chart in Figure 3.1 showed, CAD tools heavily depend on standard cell library as the main source of power and timing information of the available cells. This information is produced during the characterisation process of the cells which has its own flow (discussed in section 3.3.4). During this flow, cells are examined in all process corners of FAST, SLOW and TYPICAL by choosing the respective temperature and process corner and then a wide variety of signal conditions are applied to cells' input and output pins to characterise the delay and power reaction of cells at each voltage supplied by DVS. The gathered information help synthesis and place & route tools choose the appropriate cell from the available cells, depending on their driving strength and power consumptions, to meet the design constraints.

While the above mentioned process is performed on combinational logic, sequential cells also have to be verified for their setup and hold times. As a result, the latch and flip-flop characterisation process has many more steps to identify these timings across all conditions.

3.3.2 Cell Library Characterisation

In this section, the characterisation process, which was used in this study for combinational and sequential gates of the standard cell library, is briefly reviewed.

As this standard cell library should be able to cope with ultra voltage scaling (that is subthreshold supply voltages) which results in less robust cells, noise susceptibility should be minimised during characterisation takes place. For this library, the low power 65nm TSMC iPDK was exploited during all design steps such as schematic drawing, simulating, laying out, extracting, etc. The feature size NMOS of 60nm length and 200nm width was chosen as the basic component of the iPDK for the rest of the characterisation. It should be pointed out that narrower transistor widths of down to 120nm are also possible in this iPDK, but in expense of $0.002\mu m^2$ overhead in area (due to manufacturing rules). This means that the smallest transistor in terms of area (which is $0.082\mu m^2$) will have a 60nm by 200nm channel.

In the process of creating this library, the main goal would be to minimise the power consumption and satisfy the noise immunity and DVS requirements. Hence, performance is not a concern at this stage as it will be compensated by the proposed body bias technique. Therefore, using methods such as multi-finger transistors to improve performance is not practiced in creating the standard cell library. However, such methods will be utilised in designing the body bias generator (Chapter 4).

Table 3.1. Output symmetry analysis for a 2:1 aspect ratio inverter at 25°C

V_{DD} (V)	V_{in} (V)	V_{out} (V)	V_{in}-V_{out} (V)
0.3	0.15	0.1144	0.0356
0.4	0.2	0.133	0.0670
0.5	0.25	0.1506	0.0994
0.6	0.3	0.1704	0.1296
0.7	0.35	0.1978	0.1522
0.8	0.4	0.24	0.1600
0.9	0.45	0.3019	0.1481
1	0.5	0.3784	0.1216
1.1	0.55	0.4559	0.0941
1.2	0.6	0.5256	0.0744

First, an optimal aspect ratio (ratio of width of PMOS to width of NMOS transistor in an inverter) is worked out to meet a satisfying noise margin for combinational cells. Table 3.1 shows the result of a DC sweep simulation on a 2:1 aspect ratio inverter[1], when V_{DD} is swept from 0.3V to 1.2V and V_{out} is measured when V_{in} is at $V_{DD}/2$. With HSPICE optimisations across the required range of voltages in DVS, an aspect ratio of 2:1 for the inverter was chosen which shows a reasonable symmetry at both superthreshold and subthreshold voltages ($V_{in} - V_{out}$ is minimum at these two regions). The same procedure is used for aspect ratio identification of the rest of the combinational cells.

For Flip-Flops, however, the optimum noise margin is found based on the butterfly diagram. As a latch is the fundamental data storing block found in almost every flip-flop, it should be designed such that the data are stored in, at the most noise immune way possible.

Figure 3.2 illustrates a schematic of a basic latch. The ellipse in fact shows where data are latched and stored for later use and emphasises where noise immunity should be of

[1] PMOS width and length are 400nm and 60nm, respectively, when feature size NMOS transistor is chosen

concern. Any connection entering or leaving this ellipse can be a source of noise (entering body bias connections are not shown).

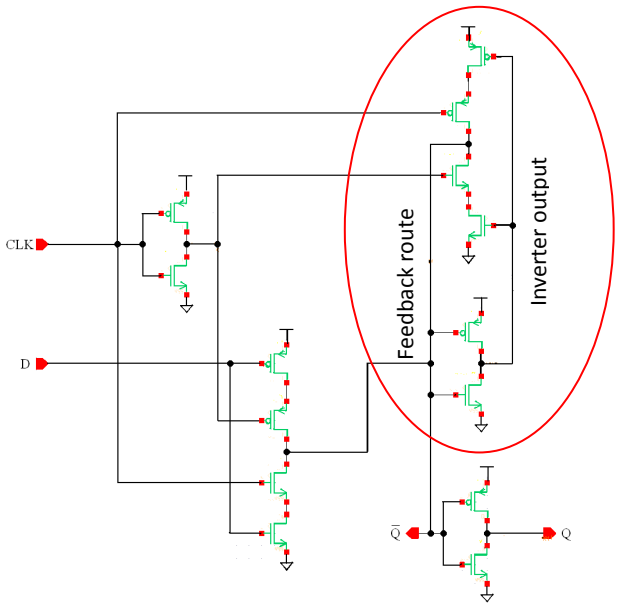


Figure 3.2. Schematic diagram of a Latch used in Flip-Flops

Through HSPICE DC optimisations and at the subthreshold voltage of interest, appropriate aspect ratios are chosen so that the squares sketched in Figure 3.3 are maximised in area with a cap on overall cell size.

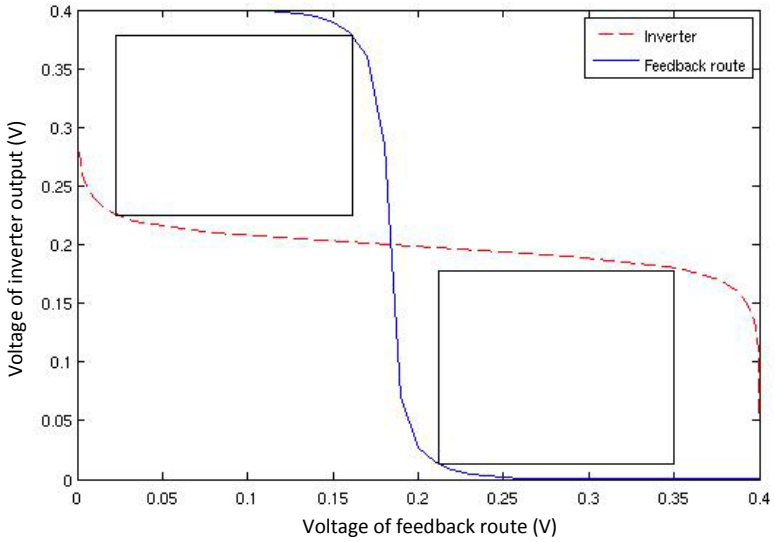


Figure 3.3. Butterfly plot of two cross-coupled inverters in the latch of Figure 3.2

Unlike combinational circuits, symmetry in Figure 3.3 is not a goal in these HSPICE optimisations but clearly the more symmetric the design is, the less the rise and fall times become and the larger the squares will be. However, this causes bulky PMOS transistors

and large area overhead in subthreshold voltages (refer to [97] for more explanation on how noise immunity is defined through a butterfly plot of two cross-coupled inverters).

3.3.3 Level Shifter Design

The iterative version of the FFT processors, implemented in Chapter 6, has superthreshold SRAMs and ROMs, while the supply voltage in the rest of the chip is scaled to subthreshold voltages. As a result, an interface is needed between the subthreshold circuits and superthreshold memory domains capable of converting the level of voltage between these two voltage domains and able to work on all DVS voltages. This interface, called level shifter, was adapted from a study [103] (among many others which failed to work in such harsh voltage scaling and PVT varying conditions). Here, level shifters only were used to shift a low voltage level to a high one, as the reverse conversion is unnecessary although possible to do if the high voltage is considered to be damaging.

3.3.4 Standard Cell Library Characterization Flow

As stated before, design tools mentioned in Figure 3.1 need standard cell libraries with specific formats describing timing and power behaviours as well as physical properties of cells. As a result, the output data generated from characterising cells have to follow these standard formats.

The library characterisation flow in Synopsys, which complies with these predefined formats, is shown in Figure 3.4. After optimising the transistor sizes using the above mentioned techniques and through HSPICE simulations, the final gates with different drive strengths are ready to be laid out. When layout is complete and DRC/LVS error free, then the Milkyway library (a physical library specific to Synopsys physical design tools) and GDSII (Graphic Database Format−version II) are extracted together with LEF (Library Exchange Format) description of the layout. These physical formats and descriptions are used to produce the physical side of the standard cell library. The parasitic extraction (resistors/capacitors as well as transistors) of all cells are formatted as HSPICE files. Starting with these HSPICE netlists of the standard cells, the rest of this process includes the use of Liberty NCX, HSPICE, and Library Compiler to create the

characterised library files. Several additional input files are also used with the Liberty NCX tool (as Figure 3.4 shows).

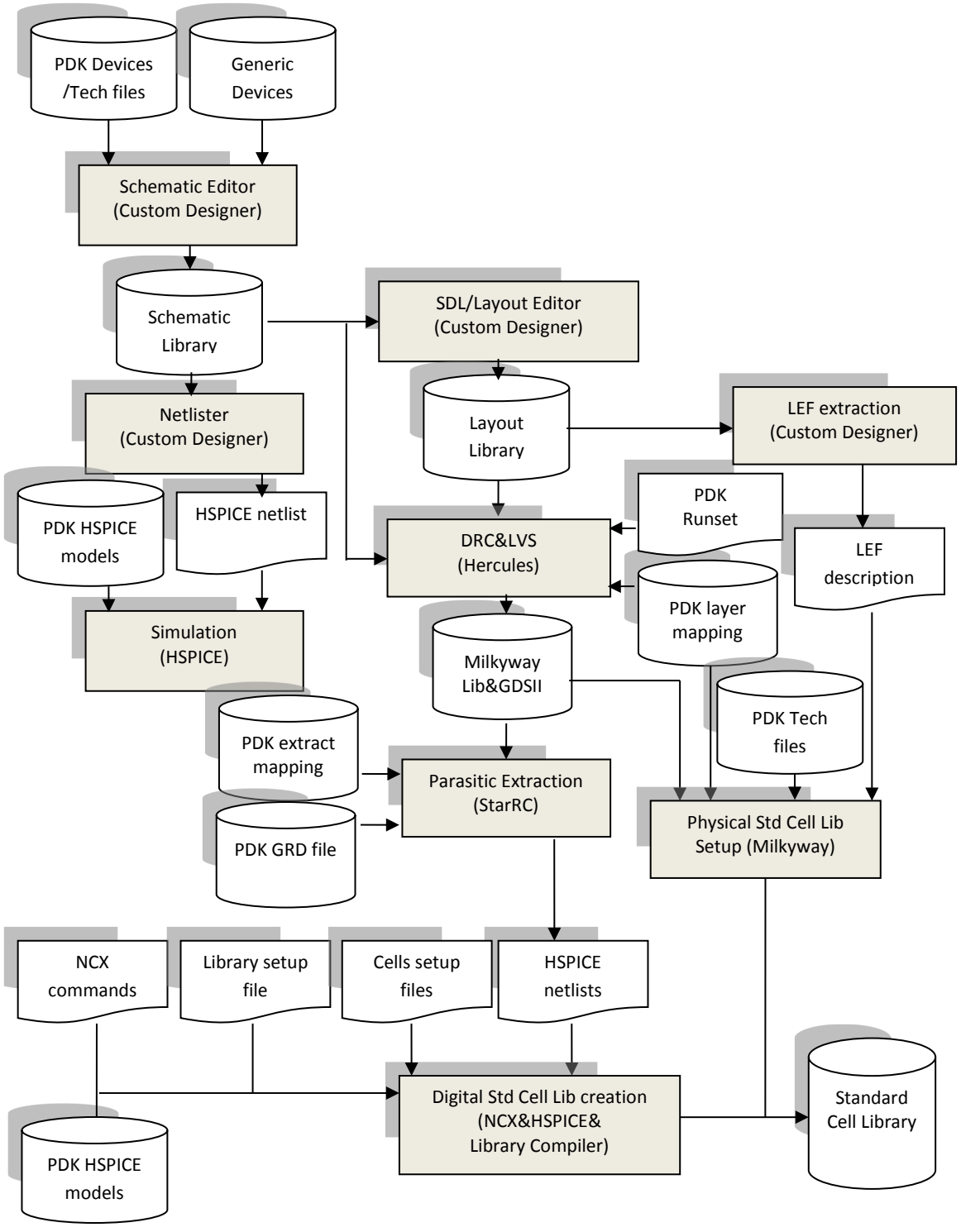


Figure 3.4. Design Flow used for implementing the Standard Cell Library

These files include the NCX command file which defines the characterisation choices such as the type of timing model to be generated (as synthesis and place & route tools accept different types, each having their own advantages and disadvantages). Library

setup files also initialise different characterisation parameters such as transition times in the input signals and capacitance in the output signals of the cells. In addition to its physical design information, each cell needs a logic associated with it, which is defined in the cell setup files. Other cell specific parameters are also defined in cell setup files. One of these parameters is the body bias amount, which makes characterisation more delicate as measures need to be taken to supply the correct body bias that has been generated by the proposed technique and according to the situation under which the characterisation simulations are executing. All this information plus the extracted HSPICE code of the cell are used in various PVT conditions as well as subthreshold to superthreshold voltage domains to generate the timing and power information embedded in the logic part of the standard cell library.

3.4 IEEE 1801 Standard (UPF)

Designing multi-voltage domain chips has become a common practice between VLSI designers and, hence, commercial tools have implemented procedures to support this technique. Released in 2009 [104] (and later revised in 2013 [105]) a standard was devised to unify the format used by different tools, called the "IEEE 1801 Standard for Design and Verification of Low Power Integrated Circuits", also known as the Unified Power Format (UPF). This format basically defined what commands are required for multi-voltage designing and how they should be formatted. Through these commands, designers can portray where they want power management cells (level shifters being one of them) to be placed, how to distinguish between voltage domains, how to supply voltage domains, etc. Synopsys synthesis and place & route tools, as well as many others, support UPF which was, as a result, employed to create the iterative FFT processor.

3.5 Variation aware standard cell library

As mentioned in section 2.1.2, corner analysis is not a feasible technique for subthreshold designing as it leads to drastic measures and crosses out many otherwise absolutely practical ultra-low power designs [38]. Therefore, statistical analysis through Monte Carlo simulations was suggested. Another nascent method of statistical analysis is utilising variation-aware libraries in the timing tools to determine the variation behaviour of the parameters, such as delay, without the need to model the variation sources and propagate the effects throughout the circuit so as to be able to examine the outcome on, for instance, the overall delay [106].

To verify this new technique, a variation-aware standard cell was also produced whose design flow is almost the same as Figure 3.4 except for the Liberty NCX part. To start, the parameters identified and modelled by the foundry to be used in Monte Carlo analysis are found in the HSPICE model of the provided iPDK. Five fundamental parameters, for instance, were found in the 65nm TSMC iPDK which are varied, during the Monte Carlo simulations, following a Gaussian distribution. Then, all cells are characterised at mean value (typical corner) and ±σ (fast and slow corners) of all identified parameters (which obviously takes a lot longer than overall typical and worst case corners).

The idea is to calculate the distribution of, for instance, delay by means of linear interpolation and extrapolation of delay of mean value and ±σ points of the identified parameters M_i (presented in Figure 3.5).

Using this idea, all variation-aware libraries at 0.3V to 1.2V supply voltages and -15°C to 75°C temperatures were created to verify the accuracy of results. Results, however, showed a significant discrepancy between Monte Carlo simulation outcome and timing analysis tool outcome on the distribution of the final delay. This discrepancy was expected, however, to some extent as this technique is at its early stages and therefore future versions and updates may improve its reliability.

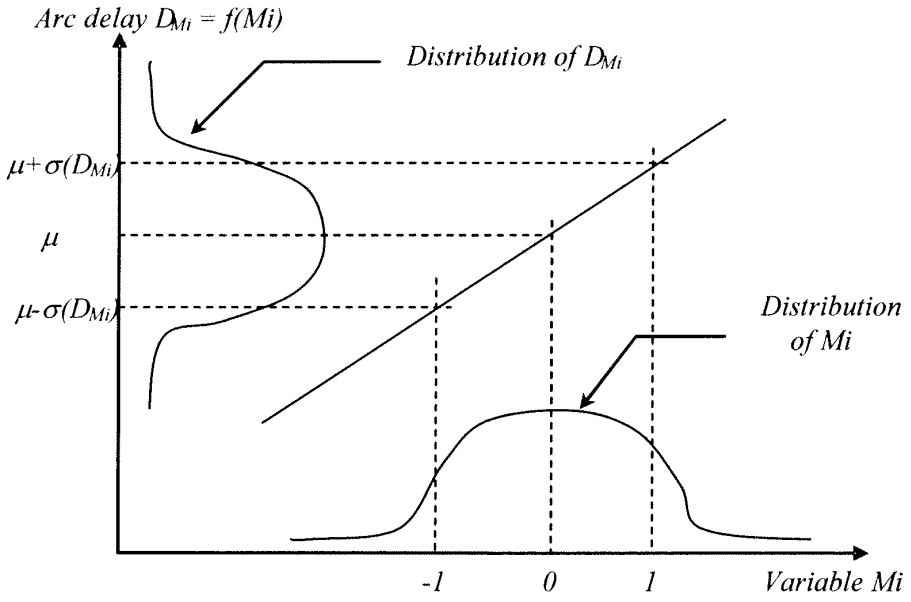


Figure 3.5. Interpolation and extrapolation of delay of different parameters M_i

3.6 Conclusion

In this chapter, by having a brief look at the design flow and the tools used, the main focus was on the standard cell library creation and its characterisation to support sub/superthreshold voltage domains in DVS. A few basic cells were reviewed to give a brief idea on the main steps of characterisation and finally the characterisation flow was described. UPF standard was explained as an essential for multi-voltage designing and ultimately a critique on the recent idea of variation-aware libraries was delivered.

Using this standard cell library, two test circuits are implemented, in Chapters 5 and 6, to verify the proposed technique of Chapter 4.

4 Design and Analysis of SULP FBB

4.1 Introduction

In this chapter, the challenge of effective mitigation of PVT variations is addressed. An extreme process variation sensitive and ultra-low power (SULP) forward body bias (FBB) circuit is proposed, capable of addressing PVT variations by tuning PMOS to NMOS ratios in different process corners while at the same time supply voltage can be scaled too. This technique helps improve the performance of the system by:

- Applying FBB to both NMOS and PMOS networks at lower voltages and/or lower temperatures
- Applying appropriate FBB to slower devices depending on the process variation

On the other hand, this technique enormously improves delay variation and hence performance yield. In addition, compared to a zero body biased (ZBB) system, this technique also improves the energy delay product (EDP) of the system to which the FBB is applied. Simplicity of the design and also its low power operation incurs low energy and area overhead to the static CMOS system to which this FBB is applied. The rest of this chapter describes how the SULP FBB circuit can achieve this by mathematically analysing how the circuit works and explaining why it maintains the EDP of the system.

4.2 Proposed Forward Body Bias Circuit

Figure 4.1 shows the schematic of the proposed SULP FBB generator [107]. The buffered output of these two circuits is forward body biased to the PMOS and NMOS devices throughout a digital system to address PVT variations while the system is working in subthreshold voltages.

Here, the focus is on the PMOS network body bias (PBB) generator and, because the NMOS network body bias (NBB) generator works the same way that the PBB generator does, discussions and results are valid for NBB too.

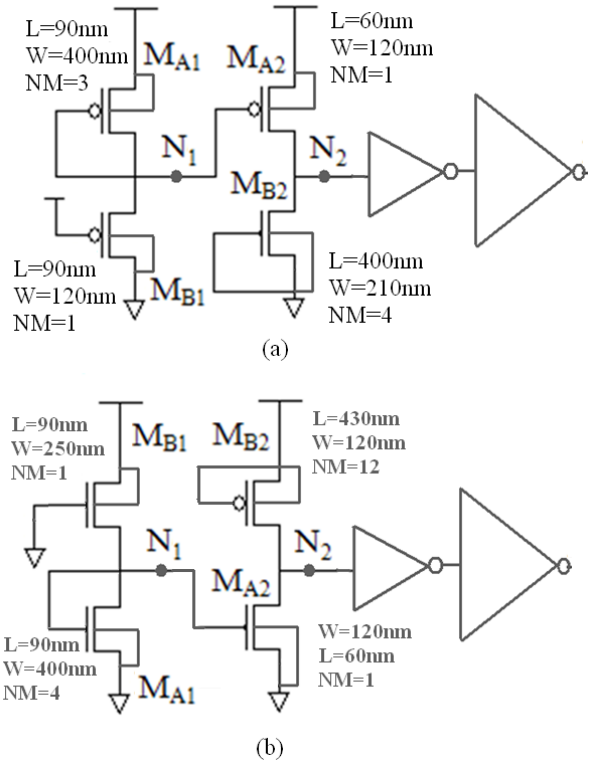


Figure 4.1. Body Bias generators for a) PMOS network and b) NMOS network

This technique comprises two main stages plus process variation independent buffers at the output (which are basically two inverters with large length and width sized devices). Firstly, it is explained, in words, how this circuit is capable of addressing PVT variation while DVS. Then, an exact mathematical analysis is put forward to back the claims.

Two PMOS transistors in the first stage form a voltage reference generator adapted from [108]. Referring to equation details of this generator discussed in [109-110] provides clarification that this adaptation is not exactly what the authors intended to achieve which is a very largely sized (L=60µm, W_{A1}=1.5µm, W_{A2}=3.3µm) supply voltage insensitive generator. In addition to having different (and rather opposite) sizings, the voltage dependence is of primary interest in this thesis. A new idea was also employed to make this voltage reference generator a process variation resistant one.

It can be seen that transistor M_{B1} is off but at the same time forward body biased. This, as a result, causes a leakage current passing through its channel, and creates a very small current sink from M_{A1}. On the other hand, M_{A1} is always saturated and with this configuration, (gate connected to drain), it acts as a diode and results in a voltage drop. Figure 4.2 depicts the reference voltage output (V_{N1} for NBB or V_{DD}-V_{N1} for PBB) in different corners and temperatures. In subthreshold voltages, it can be seen that the

reference voltage is independent of temperature and process variations. When approaching superthreshold voltage, reference voltage is increased rapidly, which leads to leakier M_{A2} and the effect of PVT or DVS is cancelled on M_{A2}.

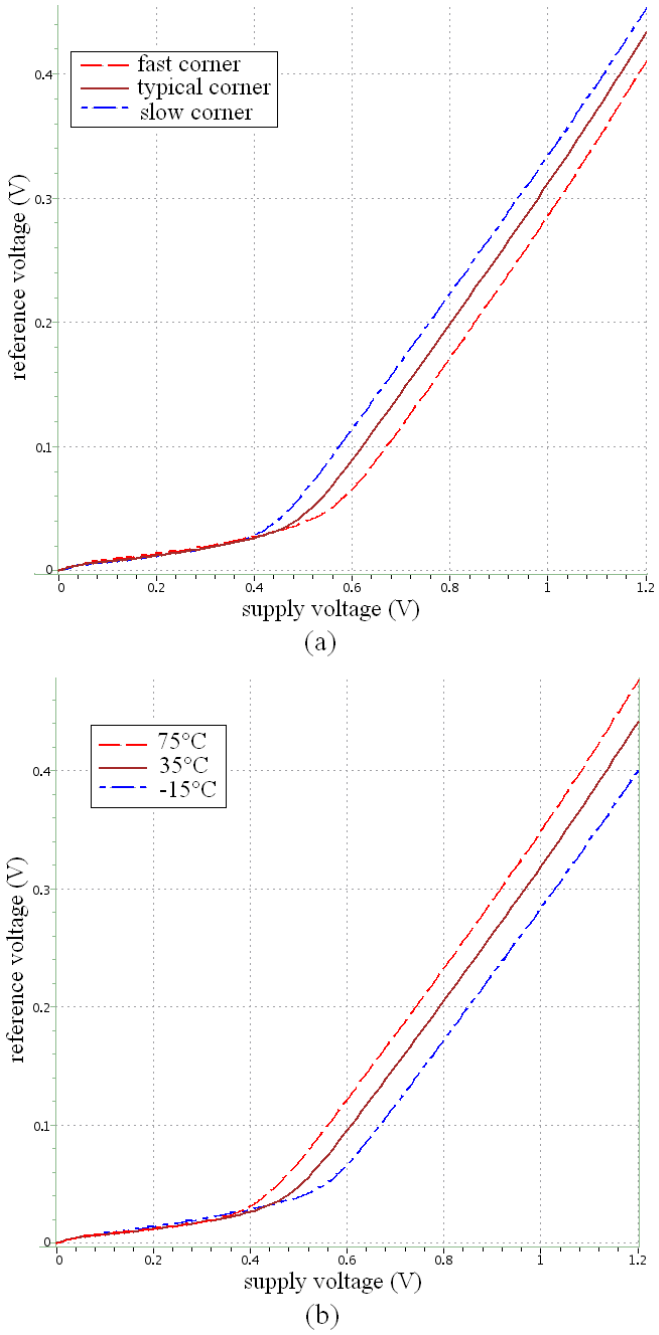


Figure 4.2. Reference voltage at a) different corners (35°C) and b) different temperatures (typical corner)

This stage, therefore, provides M_{A2} with a process/temperature independent V_{GS} proportional to V_{DD} at subthreshold voltages; that is, as V_{DD} increases, V_{GS} also increases correspondingly. In second stage, with a feature size, M_{A2} becomes a PVT dependent device, and with a large channel length and width, M_{B2} becomes PVT invariant. Sizes can

be tuned to determine when, in the buffered output, V_{N2} should cause an FBB (which will be discussed later). It should be noted that FBB voltage is different depending on what type of device it is applied to. For example, a V_{SS} voltage applied to PMOS network or a V_{DD} voltage applied to NMOS network leads to an FBB (when T=25°C and V_{DD}=0.4V in here). Increasing supply voltage and subsequently V_{GS} of M_{A2}, raises V_{N2} which in return cancels FBB at superthreshold domain. As M_{B2} and first stage are process/temperature variation independent, a fast PMOS network or a high temperature at a subthreshold domain leads to a leaky and hence strong M_{A2} which withdraws the FBB as it is no longer needed.

Exact sizing of buffers can be achieved by both simulations and formulations (through equation (4.3)). As channel length has a major role in handling process variations [37, 111], M_{B2} is also sized to have a small threshold voltage deviation when facing process variations (and it also satisfies equation (4.9)). Multipliers are used to make transistors M_{A1} and M_{B2} leaky and therefore strong in subthreshold voltages. As PMOS devices have lower mobility of the majority carriers with respect to NMOS devices, more multiplications are needed to make a leaky M_{B2} in an NBB generator than is needed for M_{B2} of the PBB generator. A strong M_{A1} in subthreshold, on the other hand, will reduce V_{GS} of M_{A2} which in return leads to a weak M_{A2} and enables a now stronger M_{B2} to be able to reduce V_{N2} and causes an FBB in the subthreshold domain.

Considering the fact that all devices of these two stages are always kept off, the large channel length in M_{B2} together with large number of multipliers will have no effect on the performance of the overall circuit and only bring about an area overhead whose cost will be discussed later.

As mentioned, transistors in first and second stages of these body bias generators are always kept off and hence the current passing through them is a subthreshold current. Focusing on the first stage, which is a reference voltage generator [108] and using equation (2.2), equation (4.1) equals the subthreshold currents formulae of M_{A1} and M_{B1} to work out the voltage reference V_{NI} [25].

$$I_{0B1}\mathrm{e}^{\frac{(V_{N1}-V_{DD})-V_{thB1}+\gamma V_{N1}}{m_{B1}v_T}}\left(1-\mathrm{e}^{-\frac{V_{N1}}{v_T}}\right)=I_{0A1}\mathrm{e}^{\frac{V_{DD}-V_{N1}-V_{thA1}}{m_{A1}v_T}}(1-\mathrm{e}^{-\frac{V_{DD}-V_{N1}}{v_T}}) \tag{4.1}$$

where γ is only applicable to M_{B1}, as it is the only transistor with the body forwarded to ground.

If both M_{A1} and M_{B1} are set to be PMOS transistors with different widths and multipliers, then equation (4.1) is reduced to equation (4.2).

$$W_{B1}\mathrm{e}^{\frac{(V_{N1}-V_{DD})+\gamma V_{N1}}{m_p v_T}}\left(1-\mathrm{e}^{-\frac{V_{N1}}{v_T}}\right)=mul_{A1}W_{A1}\mathrm{e}^{\frac{V_{DD}-V_{N1}}{m_p v_T}}\left(1-\mathrm{e}^{-\frac{V_{DD}-V_{N1}}{v_T}}\right) \quad (4.2)$$

where mul_{A1} is the number of multipliers in device M_{A1}.

This shows that process variations, mainly affecting channel length and V_{th}, do not control the reference voltage V_{N1}. This reference voltage can also be designed to be temperature variation independent as stated in [108] and shown in Figure 4.2. Considering the authors' suggestion in [108] to use near-zero threshold voltage MOSFETs for M_{B1} to keep it in the weak inversion mode at negative V_{GS} (in here equal to $V_{N1} - V_{DD}$), an FBB has been applied to this transistor, as mentioned before, to reduce its threshold voltage as much as possible using a nominal threshold voltage MOSFET device instead of, as suggested by the authors, a low-threshold voltage device.

The main well-known drawback of using low-threshold devices (or a multi-V_{th} process) is the incurred overhead in process costs caused by extra masks and steps. Besides, two nominal threshold voltage MOSFETs minimise the process variation effect on reference voltage. Two different threshold voltage MOSFETs have different processes which can result in uncertainty in the output reference voltage. Although applying the FBB eliminates this cost and uncertainty in M_{B1} transistors, a PMOS (NMOS) device biased to V_{SS} (V_{DD}) needs a separate n-well (deep n-well) which leads to about 6 times area overhead in this body bias generator cell (which is discussed later in the test circuit layout of Figure 5.2). This is as a result of DRC spaces required between V_{DD} biased NBB M_{B1}'s deep n-well and the rest of the circuit's deep n-well and between V_{SS} biased PBB M_{B1}'s n-wells and the n-well rings required for separating p-wells (presented in Figure 4.3). The deep n-well is needed for the two differently body biased p-wells and has a large area overhead to meet the DRC requirements.

Although body biasing needs triple-well process, which is optional for 90nm CMOS process, it is required for 65nm and lower processes. Despite the mentioned area overhead, this voltage reference generator is still smaller than the original one proposed in [109] (which has other functionalities not required here).

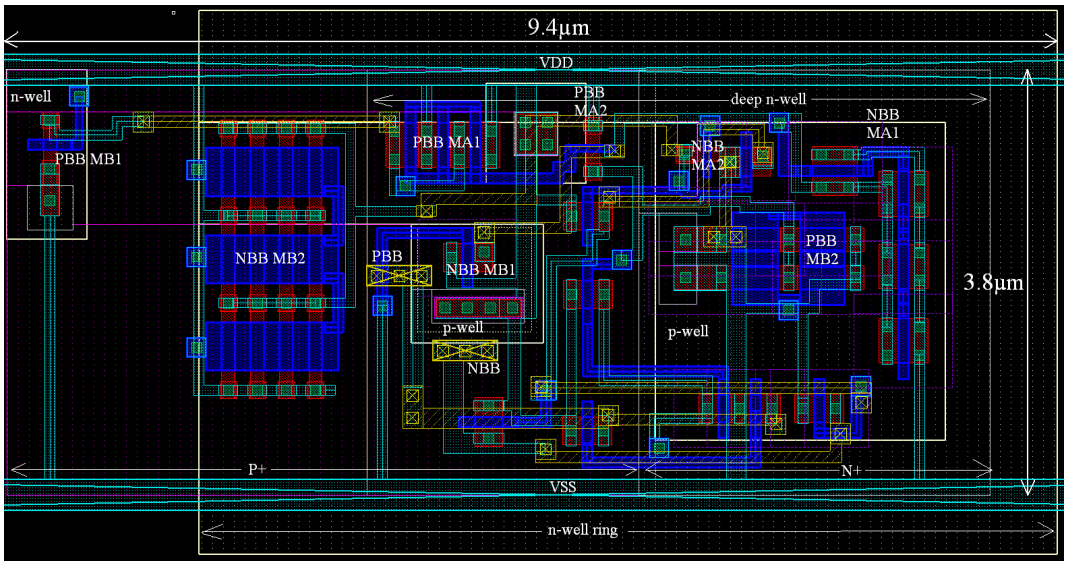


Figure 4.3. Layout of FBB generator with two separate p-wells and deep n-wells for M_{B1} transistors

Nevertheless, this cost is reduced as the system size is raised because the current needed for body biasing is very insignificant compared to the current drained from the power supply and hence this generator can provide body bias for the whole design despite being employed only once. For example, in the examined 8-bit Kogge-Stone adder (Chapter 5), the area overhead was 14% compared to a ZBB implementation. In an implementation of a 1024pt, radix 4, 32x32bit complex FFT (Chapter 6), the SULP FBB generator is 70,000 smaller than the entire chip which brings about almost no area overhead.

As process variations affect many parameters such as channel length and the device's threshold, it is necessary to inspect how final buffered output is going to overcome these variations. However, a mathematical representation is needed to model the behaviour of the proposed circuit and its potential effect on a device under test.

As mentioned in the literature review, it is usually assumed that all process variations can be abstracted in V_{th} variations [33]. By considering this fact and through sizing, the buffer stages of PBB circuit (two inverters that follow the second stage) can be designed using equation (4.3).

$$VBSP = \begin{cases} 0 & \frac{V_{DD}}{2} \le VN2 < V_{DD} \ (or\ VTHP < V_{thp0}) \\ V_{DD} & 0 < VN2 \le \frac{V_{DD}}{2} \ (or\ VTHP \ge V_{thp0}) \end{cases} \tag{4.3}$$

where *VBSP* is a random variable representing V_{BSP} that is the body bias voltage applied to PMOS network (potential difference between V_{DD} and final buffered output).V_{thp0} is a specific threshold voltage of M_{A2} in which V_{N2} is equal to $V_{DD}/2$. *VN2* and *VTHP* also

represent random variables for variable V_{N2} (output voltage of the second stage) and variable V_{thp} (threshold voltage of M_{A2}), respectively.

Note that larger buffers can be used to form larger fanouts, if required, but it is always the first two buffers that determine the final output voltage. This is because, V_{N2} is converted to V_{SS} or V_{DD} through the first two buffers and therefore the final output will be determined before being amplified any further. Besides, even huge circuits' body biases can be driven by the first two buffers, if designed properly, because a small proportion of the current is needed for body biasing, compared to power supplying. As opposed to power distribution in an integrated circuit, the small current requirement for body biasing also means that no extra designing and routing considerations are required, which results in almost no design time spent on body biasing and having negligible impact on design turnaround.

So far just *VTHP* has a known probability density function (PDF) as variations in V_{thp} presumably follow a normal distribution.

It is clear that V_{BSP} has a PDF as described in equation (4.4).

$$f_{VBSP}(V_{BSP}) = \begin{cases} p & V_{BSP} = V_{DD} \\ 0 & V_{BSP} = 0 \end{cases} \tag{4.4}$$

where p is the probability of $V_{BSP}=V_{DD}$ or $V_{N2} \leq V_{DD}/2$. In other words, with the probability of p, an FBB of V_{DD} is applied (which is the potential difference between V_{DD} and V_{SS} making the final body bias output equal to V_{SS} for the PMOS network), otherwise no FBB is applied (which means V_{DD} body bias for the PMOS network).

By defining $VBSP=V_{DD}X_B$, it can be seen that X_B has a Bernoulli distribution [112] with the mean value $\mu_{X_B} = p$. Hence, the mean value of the body bias voltage applied to PMOS network can be worked out by equation (4.5).

$$\mu_{VBSP} = V_{DD}\mu_{X_B} = V_{DD}p = V_{DD}P\left(VN2 \leq \frac{V_{DD}}{2}\right) = V_{DD}P(VTHP \geq V_{thp0}) \tag{4.5}$$

where *P (X)* is the probability of the event *X*. The first stage reference voltage (V_{N1}) is designed to be very close to V_{DD} so that if applied to PMOS transistor M_{A2}, it keeps this transistor off but at the same time increases its subthreshold leakage current which is

needed for creating the forward body bias voltage (V_{N2}). Therefore, a similar equation to (4.2) can again be formed for the second stage by equation (4.6).

$$I_{0p}e^{\frac{V_{DD}-V_{N1}-V_{thp}}{m_p v_T}}\left(1-e^{-\frac{V_{DD}-V_{N2}}{v_T}}\right)=mul_n\, I_{0n}e^{\frac{-V_{thn}}{m_n v_T}}\left(1-e^{\frac{-V_{N2}}{v_T}}\right) \tag{4.6}$$

V_{thp0} can be found by letting $V_{N2} \leq \frac{V_{DD}}{2}$ in equation (4.6) resulting in equation (4.7).

$$\begin{aligned} mul_n\, I_{0n}e^{\frac{-V_{thn}}{m_n v_T}}\left(1-e^{\frac{-\frac{V_{DD}}{2}}{v_T}}\right) &\geq mul_n\, I_{0n}e^{\frac{-V_{thn}}{m_n v_T}}\left(1-e^{\frac{-V_{N2}}{v_T}}\right) \\ = I_{0p}e^{\frac{V_{DD}-V_{N1}-V_{thp}}{m_p v_T}}\left(1-e^{-\frac{V_{DD}-V_{N2}}{v_T}}\right) &\geq I_{0p}e^{\frac{V_{DD}-V_{N1}-V_{thp}}{m_p v_T}}\left(1-e^{-\frac{V_{DD}-\frac{V_{DD}}{2}}{v_T}}\right) \end{aligned} \tag{4.7}$$

Solving with respect to V_{thp}, equation (4.7) summarises to equation (4.8).

$$V_{thp} \geq V_{DD} - V_{N1} + \frac{V_{thn}\, m_p}{m_n} + m_p v_T \ln\left(\frac{I_{0p}}{I_{0n}\, mul_n}\right) \tag{4.8}$$

Sized to its feature size, M_{A2} is prone to process variations while M_{B2} is sized large enough (especially in channel length) to be resistant to these variations [37, 111]. For example, standard deviation of *VTHN* in PBB M_{B2} has been decreased to 0.01V, by up-sizing, while *VTHP* standard deviation in PBB M_{A2} is 0.04V by being feature sized. As a result, M_{A2} is 4 times more prone to process variations than M_{B2}; however, more accuracy can be achieved if M_{B2} is up-sized even more. Ultimately, it can be assumed that *VTHN* is constant with respect to *VTHP* variations and, by exploiting this fact and substituting the mean value of *VTHN* in equation (4.8), it can be defined by equation (4.9).

$$V_{thp0} = V_{DD} - V_{N1} + \frac{\mu_{VTHN} m_p}{m_n} + m_p v_T \ln\left(\frac{I_{0p}}{I_{0n}\, mul_n}\right) \tag{4.9}$$

Since PDF of *VTHP* is already known (which is a normal distribution), then the expected value of *VBSP* can be simply found using equation (4.5) and by equation (4.10):

$$\mu_{VBSP} = V_{DD}P\left(VN2 \leq \frac{V_{DD}}{2}\right) = V_{DD}P\left(VTHP \geq V_{thp0}\right) = \frac{V_{DD}}{2}Erfc\left(\frac{V_{thp0}-\mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) \tag{4.10}$$

where *Erfc(x)* is the complementary error function (presented in Figure 4.4) [36] and V_{thp0} is found via equation (4.9).

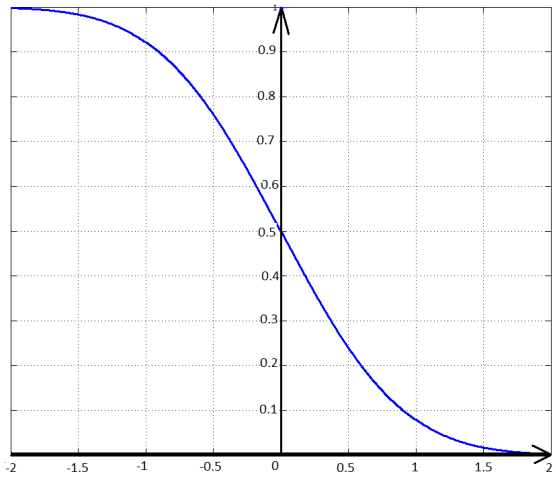


Figure 4.4. Error function *Erfc(x)*.

For NBB circuit, similarly, V_{thn0} can be defined by equation (4.11).

$$V_{thn0} = V_{N1} + \frac{\mu_{VTHP} m_n}{m_p} + m_n v_T \, ln\left(\frac{I_{0n}}{I_{0p}\, mul_p}\right) \tag{4.11}$$

Therefore, the mean value of the body bias voltage applied to NMOS network can be worked out by equation (4.12):

$$\mu_{VBSN} = \frac{V_{DD}}{2} Erfc\left(\frac{V_{thn0} - \mu_{VTHN}}{\sqrt{2\sigma^2_{VTHN}}}\right) \tag{4.12}$$

Using equation (4.10), it can be explained how this circuit addresses PVT variations as it clearly shows that mean of V_{BSP} (or FBB) follows variations occurring in transistor M_{A2} (which has the random variable *VTHP*). For example, if M_{A2} is a fast (leaky) transistor, which makes the mean value of its random variable, μ_{VTHP}, lower than its nominal value of V_{th} (i.e. threshold voltage of M_{A2}), then μ_{VBSP} reduces, which means that it becomes less likely for FBB to be applied in these circumstances (Figure 4.5).

In order to evaluate equation (4.10), V_{thp0} and subsequently V_{DD}-V_{N1} in equation (4.9) needs to be determined. Equation (4.12), similarly, needs an evaluation of V_{N1} in equation (4.11). As equation (4.2) cannot be analytically solved to find V_{N1}, a numerical solution is sought for V_{DD}-V_{N1} (V_{N1}) by fitting the most appropriate curve to the typical corner of Figure 4.2 at 25°C with respect to V_{DD}. Since V_{N1} is insensitive to temperature and process variations at the subthreshold region, this fitted curve exhibits the exact V_{N1}'s

behaviour at subthreshold voltages. As will be seen, the sudden increase of V_{DD}-V_{N1} (V_{N1}) in the superthreshold region together with the *Erfc(x)*'s drop (presented in the Figure 4.5) leads to zero μ_{VBSP} which detracts from the influence of temperature and process variations on V_{N1}.

By replacing V_{thp0} with the above mentioned fitted curve in the equation (4.10), μ_{VBSP} can be sketched in Figure 4.5.

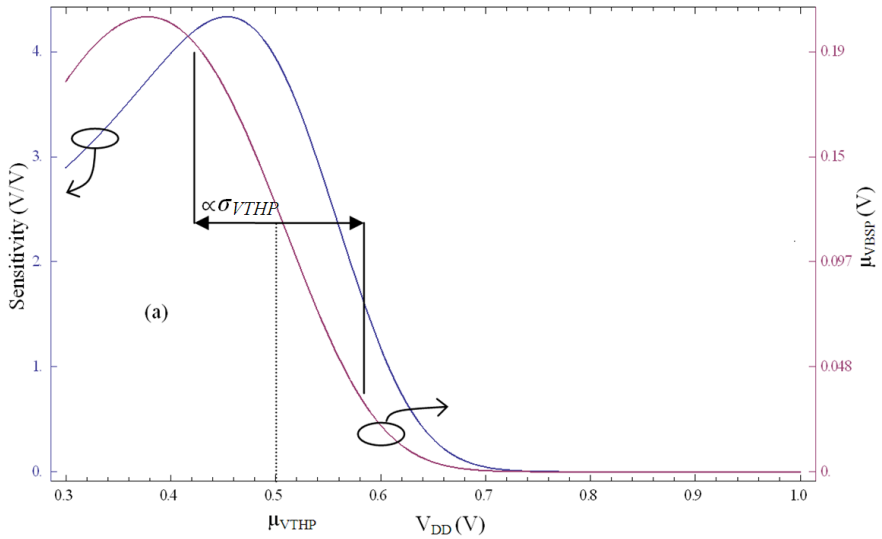


Figure 4.5. μ_{VBSP} and its sensitivity to μ_{VTHP} variations (at μ_{VTHP}~0.5V) for m_p=1.7, m_n=1.48, v_T=0.026V, σ_{VTHP}=0.04V, μ_{VTHP}=0.5V, μ_{VTHN}=0.45V, *T=25°C,* and multipliers and sizes of Figure 4.1.

Figure 4.5 demonstrates that moving toward superthreshold voltages reduces μ_{VBSP} to zero, leading to FBB cancellation (as was predicted in earlier discussions). In subthreshold voltages, however, the mean of V_{BSP} for PMOS network tends toward V_{DD} (and thus as V_{DD} reduces, the mean value of V_{BSP} also decreases). This also describes how SULP FBB handles the voltage variations as well as voltage scaling.

Finally, based on temperature independence of V_{N1} at subthreshold voltages, thermal voltage (v_T) is the only remaining variable in equation (4.9) depending on temperature. If coefficient of v_T, i.e. $m_p \ln\left(\frac{I_{0p}}{I_{0n}\, mul_n}\right)$, was a value near zero, then the thermal voltage would not be an influential parameter. However, this is avoided through choosing a higher number of multipliers, mul_n, and appropriate sizing. Thus, a temperature rise increases V_{thp0} and therefore reduces μ_{VTHP} (presented in Figure 4.5). This means that at higher temperatures and higher subthreshold voltages, where transistors become faster, SULP FBB generator starts cancelling FBB to prevent more leakage energy consumption.

FBB cancellation in higher voltages also suggests that, when using this technique, if a system goes to standby at times, it should be placed in standby mode at higher voltages

which has no FBB applied and thus no leakage. This is because, not only a subthreshold leakage is prevented due to not being in subthreshold voltage domain, but also FBB application is cancelled resulting in no leakage due to forward body biasing.

It should be noted that SULP FBB behaviour, shown in Figure 4.5, is dependent on V_{thp0}, hence, it can be altered by tuning parameters in equation (4.9). For example, in case of experiencing very slow devices, higher FBB mean values might be needed, that is, μ_{VBSP} in Figure 4.5 should continue rising to higher V_{DD} voltages before it is cancelled. This can be achieved by adding $mult_n$ or number of multipliers in M_{B2}, for example, which decreases V_{thp0}, in return. In equations (4.9) and (4.10), hence, V_{DD} can increase more to compensate the effect of adding multipliers before FBB is cancelled. In this way, SULP FBB circuit is tuned to continue providing FBB even in nearthreshold voltages. This means SULP FBB can be exploited in a variety of systems. Furthermore, when the number of multipliers creates very asymmetrical and uneven shapes, (e.g. a gate with very big NMOS devices and very small PMOS ones,) this problem can be mitigated by skewed buffers.

Another useful observation from equation (4.10) is that if variance of process variations (σ^2_{VTHP}) increases, (for example if a system is implemented by narrower channel technologies,) then the chance of FBB being applied in higher voltages increases too, which is a reasonable action to take. Otherwise, if a technology has less variance in threshold voltage of its feature size transistors by its nature, then the chance of having FBB applied at near/superthreshold voltages reduces. This can be clearly understood using Figure 4.5 in which σ_{VTHP} is shown.

Sensitivity to process variations in the output of this circuit is also a parameter which also needs specific design as it determines how responsive to the variations the circuit is and when this sensitivity maximises or minimises. Output sensitivity to process variations can be defined by the rate of variation in the voltage of V_{N2} when threshold voltage of M_{A2} changes. In other words, the rate μ_{VBSP} varies, while μ_{VTHP} is varying, is considered the sensitivity [113] and can be calculated by differentiating the equation (4.10) with respect to μ_{VTHP} which results in equation (4.13).

$$\frac{\partial \mu_{VBSP}}{\partial \mu_{VTHP}} = \frac{V_{DD}}{\sqrt{2\pi\sigma^2_{VTHP}}} \mathrm{e}^{\left(\frac{V_{thp0}-\mu_{VTHP}}{2\sigma^2_{VTHP}}\right)^2} \tag{4.13}$$

Equation (4.13) has been sketched for μ_{VTHP}≈0.5V in Figure 4.5 which maximises to ~4.3V/V on V_{DD}≈0.45V. This sensitivity is directly proportional to threshold voltage. For example, at μ_{VTHP}=0.6V, maximum sensitivity of ~6V/V is obtained at V_{DD}≈0.64. Once again, sensitivity to process variations is cancelled at superthreshold voltages. Sensitivity to process variations at nearthreshold voltages in SULP FBB circuit equals the latest sensitivity that has so far been achieved (for superthreshold applications) in [113]. Likewise [113], SULP FBB is also a bias free circuit as its first stage provides a reference voltage which is process variation independent.

4.3 Effect of SULP FBB Circuit on Energy and Delay

This section focuses on application of this FBB to a typical inverter in a data-path. The goal of this section is to find the effect of FBB on delay and energy in terms of improvement or degradations without calculating their exact amount, as the exact amount will be determined through simulations. This helps make sure that the technique, in theory and before the implementation, is satisfying and gives an insight into how it affects the functionality of the whole system.

As FBB is cancelled at superthreshold voltages, it can be assumed that no improvement or degradation happens while operating in superthreshold region. Therefore, results of the following equations, which are sub/nearthreshold equations, are valid for superthreshold voltages as long as there is no impact on energy and delay at superthreshold voltages, which will be proved true later on. A mathematical method for analysis of the energy and EDP of the proposed circuit is developed and introduced which can further be used for analysis of any other circuits under PVT variations.

When output of an inverter, in subthreshold region, switches from 0 to 1, PMOS is the device which is playing the main role in the delay of the gate. Therefore, its delay can represent the gate's delay and can be calculated by equation (4.14) [33]:

$$t_d = \frac{\frac{1}{2}\eta C_s V_{DD}}{I_{on_p}} \quad where \quad I_{on_p} = I_{0p} e^{\frac{V_{DD}-V_{thp}+\gamma V_{BSP}}{m_p v_T}} \left(1 - e^{-\frac{V_{DD}}{v_T}}\right) \tag{4.14}$$

where η is the delay factor of the inverter's non-step input, C_S is the inverter's switching load capacitance, and I_{on_p} is the current passing through PMOS when it is switched on (at the subthreshold region).

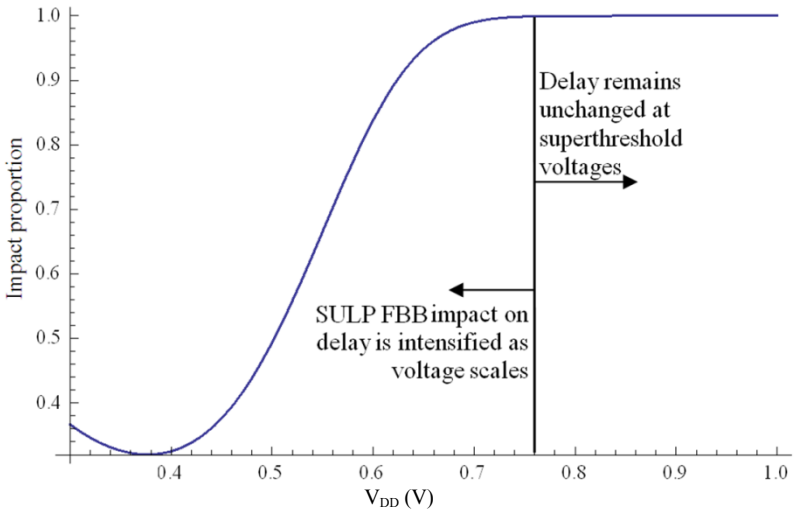


Figure 4.6. $f_{Dp1}(V_{thp0})$ for m_p=1.7, m_n=1.48, v_T=0.026V, σ_{VTHP}=0.04V, μ_{VTHP}=0.518V, μ_{VTHN}=0.45V and multipliers and sizes of Figure 4.1.

The mean value of delay can be worked out by equation (4.15) (for details of derivations refer to equations (A.3) and (A.4) in section A.2 of Appendix A in which, by assuming a normal PDF for *VTHP*, equation (4.14) is used to calculate the mean value).

$$E_{T_d}(t_d) = \mu_{T_{dZBB}} \cdot f_{Dp1}(V_{thp0}) \tag{4.15}$$

where μ_{TdZBB} is the mean value of the delay of the inverter when zero body bias (ZBB) is applied and $f_{Dp1}(V_{thp0})$ is the impact of SULP FBB on the inverter's mean delay (refer to equation (A.1) section A.1 of Appendix for its definition).

Figure 4.6 shows how $f_{Dp1}(V_{thp0})$ influences delay's mean value. As V_{DD} and hence V_{thp0} are small in subthreshold voltages, inverter's delay reduces after SULP FBB application to a ZBB inverter. As V_{DD} rises, impact of FBB is dropped and, as a result, inverter's delay at higher voltages is no longer affected by this technique. It has to be noticed that Figure 4.6 only shows the impact on the mean value of delay of a ZBB inverter after SULP FBB is applied and not the actual delay values as this is only for comparison purposes.

Since the impact of this technique on delay showed an improvement when the inverter is in sub/nearthreshold regions, it is now necessary to identify how expensive the cost of the delay reduction is, in terms of the increased energy consumption. If an FBB is applied to a device, then threshold voltage of that device is reduced resulting in an elevated subthreshold current (presented in section 2.1.3 the Body Bias Effect). This, hence, has a direct impact on increasing the leakage energy consumption. The rest of this section, therefore, explores how energy-delay product (EDP) alters after SULP FBB technique is

put into practice and determines whether the negative impact on energy is compensated by the above mentioned positive impact on the delay.

Before determining EDP equations, the equation of leakage current is formulated for a stable inverter gate (an inverter that is not switching). If it is assumed that PMOS device is off and NMOS is on, then PMOS has a leaking current using equation (2.2) which is shown in equation (4.16).

$$I_{leak_p} \approx I_{0p} e^{\frac{-V_{thp}+\gamma V_{BSP}}{m_p v_T}} \text{ (assuming } 1 - e^{-\frac{V_{DD}}{v_T}} \approx 1) \tag{4.16}$$

where V_{BSP} is the body bias applied to PMOS network.

In Appendix A (section A.3), equation (4.16) is used for leakage energy calculation, formula $½\alpha C_S V_{DD}^2$ for active energy calculation, and using the calculated t_d in equation (4.14), EDP can be formulated thoroughly.

Equation (4.17) shows an abbreviated form of the impact of SULP FBB on EDP for an inverter in a data-path. It has been assumed in equation (4.17) that the expected value of EDP, when ZBB is applied, equals $E(EDPZBB)=A+B+C+D+E$ where B and C are related to leakage energy-delay product ($E_{leak}DP$) and D and E are related to active energy-delay product ($E_{act}DP$) (refer to equation (A.17) and (A.18) in section A.3 of Appendix A for the extended version and section A.1 and List of Variable Definitions for definitions). Equation (4.17), in fact, shows that application of SULP FBB on a ZBB data-path is equivalent to applying coefficients to the expected value of EDP for ZBB.

$$E(EDPSULP) =$$
$$A + B\, f_{Ep}(V_{thp0}) f_{Dn1}(V_{thn0}) + C f_{En}(V_{thn0})\, f_{Dp1}(V_{thp0}) + D f_{Dp1}(V_{thp0}) + E f_{Dn1}(V_{thn0}) \tag{4.17}$$

Figure 4.7 shows how $f_{Ep}(V_{thp0})$ in equation (4.17) affects leakage energy, which as expected, causes static energy to rise at subthreshold voltages. On the other hand and as previously shown in Figure 4.6, $f_{Dn1}(V_{thn0})$ reduces the inverter's delay and, eventually, the final impact defined by EDP of leakage energy (or $E_{leak}DP$), can be worked out by $f_{Ep}(V_{thp0})\, f_{Dn1}(V_{thn0})$ and $f_{En}(V_{thn0})\, f_{Dp1}(V_{thp0})$) in equation (4.17) (which is demonstrated in Figure 4.7).

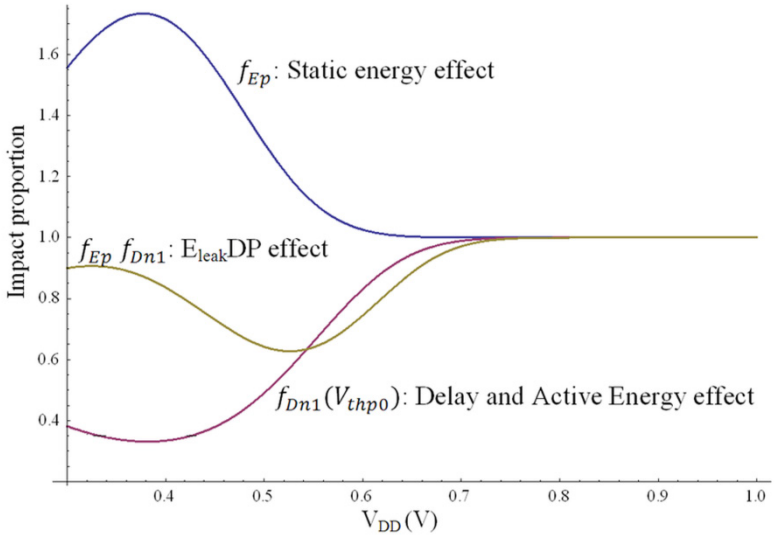


Figure 4.7. Effect of SULP FBB on energy and delay of the examined inverter for m_p=1.7, m_n=1.48, v_T=0.026V, σ_{VTHN}=0.034V, μ_{VTHP}=0.518V, μ_{VTHN}=0.493V, *T=25°C,* and multipliers and sizes of Figure 4.1.

$E_{leak}DP$ shows that, despite having a leakage energy rise, the final impact of SULP FBB application is a decreased energy-delay product in both NMOS and PMOS transistors at subthreshold voltages. The effect on active energy is an absolute reduction as $f_{Dp1}(V_{thp0})$ (and $f_{Dn1}(V_{thn0})$) for active energy acts the same as illustrated in Figure 4.6 which is a reduction in subthreshold voltages and no change in superthreshold voltages. This is because, active energy spent while switching remains the same, no matter how fast the inverter switches, which as a result, when the final active energy-delay product is calculated, an absolute reduction is observed in subthreshold voltage domain. Figure 4.7 shows that the overall EDP remains unchanged (or even reduces), as V_{DD} scales to subthreshold voltages, that makes this technique not only beneficial, in terms of performance increase, but also conservative regarding the consumed energy. Moreover, EDP stays unchanged in superthreshold domain, too, because of FBB withdrawal and the near zero energy overhead of SULP FBB circuit (which can be seen later using simulations). All of the above mentioned results are also proved by simulations as discussed in the next chapters.

4.4 Modelling the Error Rate Reduction

In this section, it is shown how the SULP FBB generator can take advantage of PVT variations in order to significantly reduce the error rate.

The probability of an erroneous event in a data-path can be approximated by the probability of error in an inverter. Although delay variation analysis for a single gate does

not represent the exact effect of variations on a data-path comprised of many various gates, a data-path made up from various series gates deals with variations better than a single gate [33], which means the above approximation of improvement will be experienced even more in a data-path as simulations prove later on.

Once again, by defining T_d as the random variable of t_d, it can be assumed that an inverter has to have a delay of less than t_0 to meet the required timing constraint. By means of equation (A.3), the probability of error or simply error rate can be acquired by the equation (4.18).

$$P(T_d > t_0) = P\left(D_0 e^{\frac{VTHP-\gamma VBSP}{m_p v_T}} > t_0\right) = P\left(VTHP > m_p v_T \ln\left(\frac{t_0}{D_0}\right) + \gamma VBSP\right) \tag{4.18}$$

Using the law of total probability, equation (4.19) is calculated from equation (4.18).

$$P(T_d > t_0) = P\big(B|\, VTHP > V_{thp0}\big)P\big(VTHP > V_{thp0}\big) + P\big(B|\, VTHP \le V_{thp0}\big)P\big(VTHP \le V_{thp0}\big) \tag{4.19}$$

where $B = m_p v_T \ln\left(\frac{t_0}{D_0}\right) + \gamma VBSP$

Using equation (4.3), it can be seen that *VBSP* (i.e. the PBB FBB output) is V_{DD} when *VTHP* is larger than V_{thp0}, otherwise it is zero. By exploiting this fact, equation (4.20) can be inferred from equation (4.19).

$$\begin{aligned} P(T_d > t_0) = {} & P\left(VTHP > m_p v_T \ln\left(\frac{t_0}{D_0}\right) + \gamma V_{DD}\right) P\big(VTHP > V_{thp0}\big) \\ & + P\left(VTHP > m_p v_T \ln\left(\frac{t_0}{D_0}\right)\right) P\big(VTHP \le V_{thp0}\big) \end{aligned} \tag{4.20}$$

Assuming normal distribution for *VTHP* and using equation (4.9), equation (4.21) is formed.

$$\begin{aligned} P(T_d > t_0) = {} & \frac{1}{2} Erfc\left(\frac{m_p v_T \ln\left(\frac{t_0}{D_0}\right) + \gamma V_{DD} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) . \frac{1}{2} Erfc\left(\frac{V_{thp0} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) + \\ & \frac{1}{2} Erfc\left(\frac{m_p v_T \ln\left(\frac{t_0}{D_0}\right) - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) . \left(1 - \frac{1}{2} Erfc\left(\frac{V_{thp0} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right)\right) \end{aligned} \tag{4.21}$$

A ZBB inverter, with T_{dZBB} as the random variable for output delay, has no FBB and hence a zero *VBSP* and, therefore, the error rate can be found by equation (4.22).

$$P(T_{dZBB} > t_0) = \frac{1}{2} Erfc\left(\frac{m_p v_T \ln\left(\frac{t_0}{D_0}\right) - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) \tag{4.22}$$

Since $1 - e^{-\frac{V_{DD}}{v_T}} \approx 1$ for V_{DD}>0.3V, D_0 can be substituted in equation (4.22) resulting in equation (4.23).

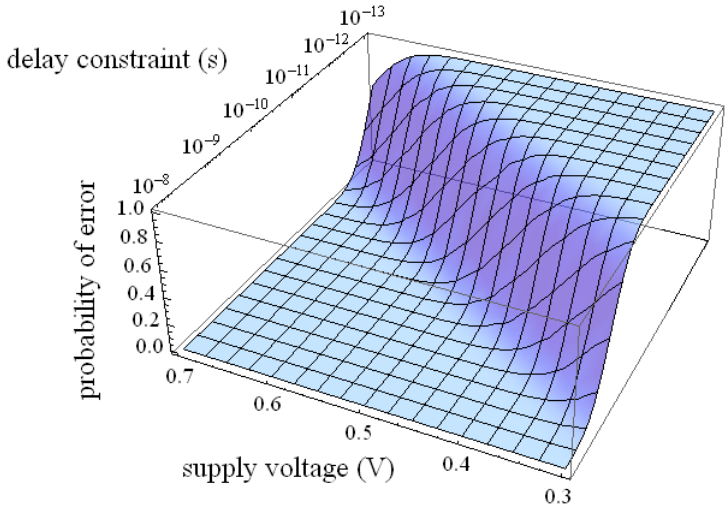
$$P(T_{dZBB} > t_0) = \frac{1}{2} Erfc\left(\frac{m_p v_T \ln\left(2\frac{t_0 I_{0p}}{\eta C_s V_{DD}}\right) + V_{DD} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) \tag{4.23}$$

The same substitution in equation (4.21) leads to equation (4.24).

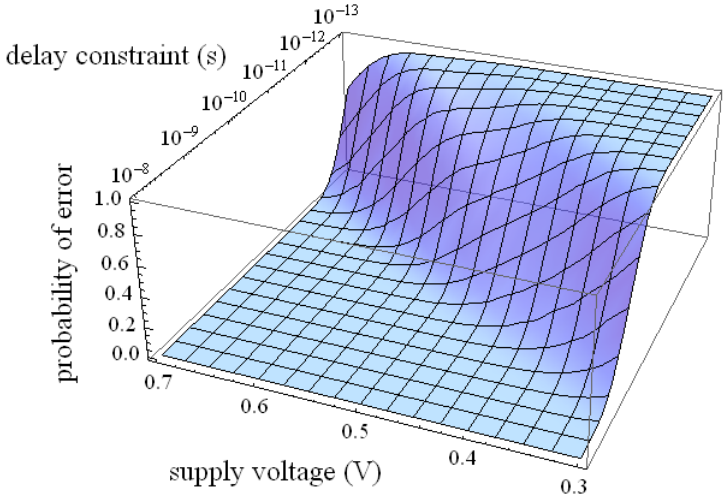
$$\begin{aligned} &P(T_d > t_0) \\ &= \frac{1}{2} Erfc\left(\frac{m_p v_T \ln\left(2\frac{t_0 I_{0p}}{\eta C_s V_{DD}}\right) + V_{DD}(1+\gamma) - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) . \frac{1}{2} Erfc\left(\frac{V_{thp0} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) \\ &+ \frac{1}{2} Erfc\left(\frac{m_p v_T \ln\left(2\frac{t_0 I_{0p}}{\eta C_s V_{DD}}\right) + V_{DD} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) . \frac{1}{2} Erfc\left(-\frac{V_{thp0} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right) \end{aligned} \tag{4.24}$$

Equations (4.23) and (4.24) have been plotted in Figure 4.8.a and Figure 4.8.b, respectively, by substituting coefficients with numbers extracted by simulations and the curve fitting procedure explained earlier (values have also been brought into Figure 4.8). Figure 4.8 shows that higher supply voltages or looser delay constraints lead to zero probability of error. Delay constraints, in here, refer to the time needed for a combinational circuit to meet the required setup and hold times.

It can be observed that there is a concavity in the error probability of SULP FBB inverter shown in Figure 4.8.b, when in subthreshold voltage domain. This means that tighter delay constraints can be tolerated by an inverter with SULP FBB applied to it with respect to a ZBB inverter.



(a)



(b)

Figure 4.8 Probability of error for m_p=1.7, m_n=1.48, v_T=0.026V, σ_{VTHP}=0.04V, μ_{VTHP}=0.5V, μ_{VTHN}=0.45V, η=2.1, and C_s=1pF in a) ZBB inverter and b) SULP FBB inverter

If error probability of ZBB inverter is subtracted from error probability of SULP FBB inverter (using expression $P(T_{dZBB}>t_0)$-$P(T_d>t_0)$ or equation (4.23) – equation (4.24)), then the outcome has the form of Figure 4.9.

Figure 4.9 shows the improvement (reduction) in probability of error in a ZBB inverter after SULP FBB technique is applied. As it can be seen in Figure 4.9, improvement in the probability of error in superthreshold voltage region tends to zero. This is again due to FBB cancellation in this region.

In subthreshold voltage region, however, when t_0 or the delay constraint on inverter output is very tight and close to zero (presented in Figure 4.8), the probability of inverter

delay being larger than the expected constraint for both ZBB and SULP FBB designs are the same and equal to one which results in zero probability improvement. This is true for very loose delay constraints as both ZBB and SULP FBB inverters can meet the constraint and, having the same probability of error equal to zero (presented in Figure 4.8), results in zero improvement too.

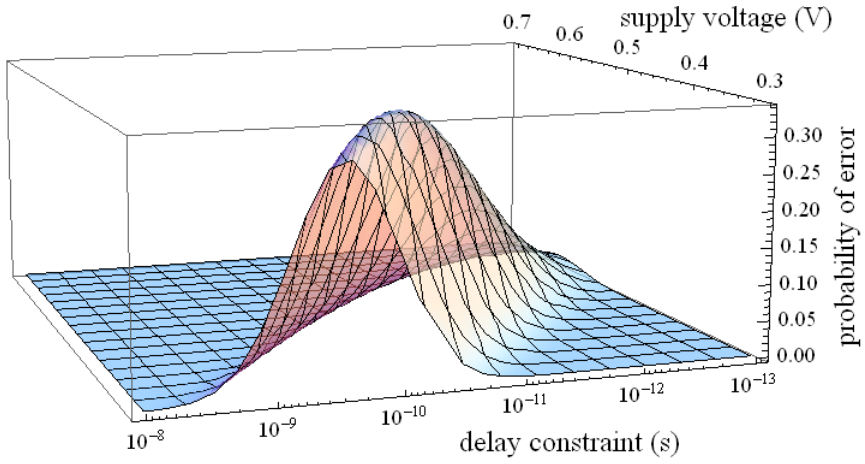


Figure 4.9 Improvement (reduction) in probability of error when SULP FBB is applied

But delay constraints, determined by clock frequency, are not too tight or too loose but set to the tightest error free condition. For example, in Figure 4.8, this delay constraint (or clock frequency for a data-path) is just set to the points where the surface is about to rise from zero to one. The SULP FBB resulted concavity in Figure 4.8.(b), therefore, can either help tighten the delay constraint even more and gain the same amount of error probability as ZBB case or simply benefit from the error probability improvement, as Figure 4.9 suggests, with the same delay constraint as ZBB case. Here maximum error rate reduction is 0.35 for a theoretical inverter. However, as will be seen later through simulations, more error reduction is achieved in the datapath which consists of, usually, several consecutive gates.

4.5 Conclusion

By the aid of the probability theory, this chapter formulated the delay and energy variations, and using them, a comparison was made between SULP FBB and ZBB effects. It is predicted that, in subthreshold voltages, SULP FBB reduces delay compared to the ZBB with the cost of increased energy consumption. However, overall EDP is expected to remain unchanged or even improve after application of the proposed technique. As V_{DD}

rises, impact of FBB is anticipated to drop and, as a result, delay and energy in higher voltages will no longer be affected by this technique. Additionally, SULP FBB technique is expected to reduce the error probability which, as a result, enhances the timing yield. Chapters 5 and 6 will provide simulation results backing the theory discussed and the predictions made in this chapter.

5 An 8-bit Kogge-Stone Adder Using SULP FBB

5.1 Introduction

Using a variation sensitive and ultra low-power design, the previous chapter proposed a novel technique capable of sensing and responding to process, voltage and temperature variations as well as dynamic voltage scaling by providing an appropriate forward body bias so that energy delay product and timing yield of the whole system was improved. In this chapter, the theoretical analysis for process variation probability in Chapter 4 will be confirmed by post-layout HSPICE simulations on an 8-bit pipelined Kogge-Stone adder. For this adder, for example, assuming a voltage scaling form 0.8V to 0.3V and temperature changes of -15°C to 75°C, the proposed technique brings about a 7 times less delay variation while energy delay product improves by 23% compared to a zero body biased adder. Moreover, error probability is decreased from 50% to 1% at 0.4V as a result of this technique. Results prove that the exponential sensitivity of devices to variations in subthreshold voltages can still be exploited to an extent that can compensate the diminishing FBB effectiveness caused by technology scaling.

5.2 Test Circuit Design

In this chapter, an 8-bit Kogge-Stone [114] adder was chosen to verify the SULP FBB technique as it is the most fundamental block found in any processor. Figure 5.1.a illustrates the 8-bit Kogge-Stone tree adder [24] implemented in the test circuit. Gray and black square cells are implemented using basic AND-OR-INV gates while more complicated XOR gates are needed just after input and before output pins (as explained in Figure 5.1.b. Two sets of register files, implemented by flip-flops, are used to clock in and out the input arrays of ($a_{0..7}$,$b_{0..7}$, c_{in}) and output array of ($s_{0..7}$, c_{out}), respectively, simulating a pipelined behaviour. Delay is measured in the critical path (pin s_7 in here). Sequential and combinational parts of the circuit are included in the power consumption measurements to resemble the effect of both dynamic and static energy.

This test circuit is simulated separately with two configurations of SULP FBB and ZBB and results are provided and compared in the next section.

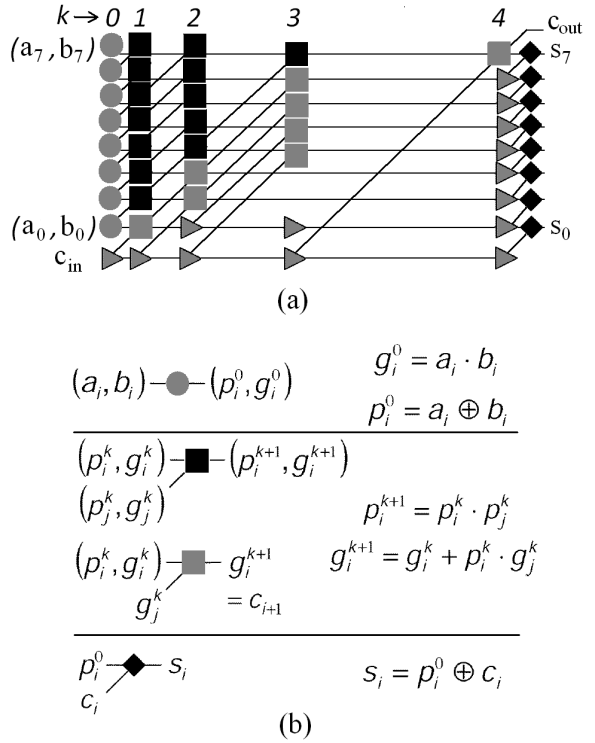


Figure 5.1. Kogge-Stone adder tree

5.3 Simulation Results and Discussions

5.3.1 Simulations

All simulations were performed using a Low Power 65nm TSMC technology model. 1000 Monte Carlo (MC) runs were executed on this adder to simulate the process variations. In addition, 100 MC runs were also performed on the large post-layout HSPICE codes to verify the results. For this purpose, (as explained in Chapter 3) a Standard Cell library was created with PBB and NBB connections for all cells. Then Design Compiler (Synopsys synthesis tool) and IC Compiler (Synopsys place & route tool) were utilised to create the final GDSII file. The temperature sensitive RC extraction was carried out on this file to obtain the final post-layout HSPICE codes.

It should be noted that no subthreshold specific tool is required in the entire design, simulation and implementation process in here. Only the standard cell library and SULP FBB circuits have sub/superthreshold voltage domain. Tools can be simply configured to work in different voltage domains, including subthreshold domain, as nothing but supply voltage is changed. CAD tools support different supply voltages as long as input library has been characterised to support it.

As mentioned in section 2.2.3.2.2, random intra-die variations do not have spatial correlation. Hence this cannot be addressed by SULP FBB unless a calibration method is exploited [113]. These random variations, however, are more damaging in SRAMs than in data-path architectures [113].

As a result, one main duty of the SULP FBB technique is to address inter-die process variations by applying the appropriate body bias to the whole system. Hence, the foundry provided global process variation model was exploited, as explained in Chapter 3, to produce the most accurate possible results. This model contained all foundry proven Gaussian distributions which led to the most practical and realistic Monte Carlo simulation results. Temperature and voltage were also swept to resemble both temperature variations and V_{DD} scaling. In this study, temperature varies from -15°C to 75°C (90°C variation), and if higher or lower working temperatures are required then SULP FBB circuit has to be simply tuned to support this (tuning explained in Chapter 4).

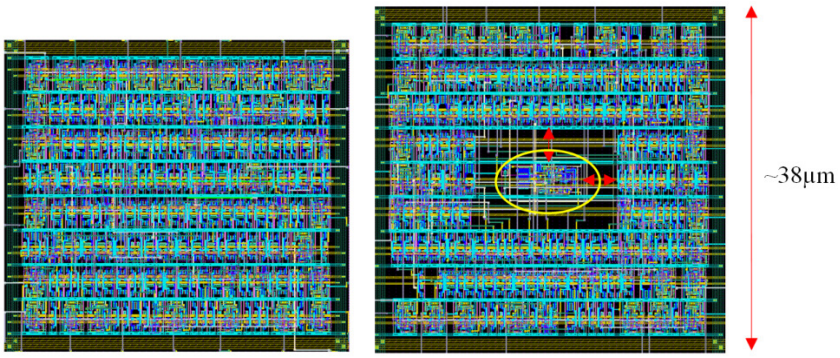


Figure 5.2. Layout drawing for the 8-bit Kogge-Stone adder with a) ZBB design and b) SULP FBB generator located inside the yellow ellipse.

Figure 5.2 shows the layouts of the above mentioned adder in two different schemes. Figure 5.2.a has a ZBB implementation while Figure 5.2.b has the SULP FBB generator in the heart of system in which arrows indicate the required space between two deep n-well layers (FBB generator has its own deep n-well layer for the V_{DD} biased NMOS while the rest of NMOS devices in the chip are biased to the generated FBB).

After extracting the temperature sensitive post-layout HSPICE code using the flow described in section 3.2, the Monte Carlo simulations can be executed in order to find out the circuit's reaction to the PVT variations and DVS in two different layouts shown in Figure 5.2.

The first important simulation is run to examine if the SULP FBB generator's output is actually responding as predicted by mathematical models. Figure 5.3 exhibits the mean

value of 1000 FBB voltages applied to PMOS network at different voltages and temperatures.

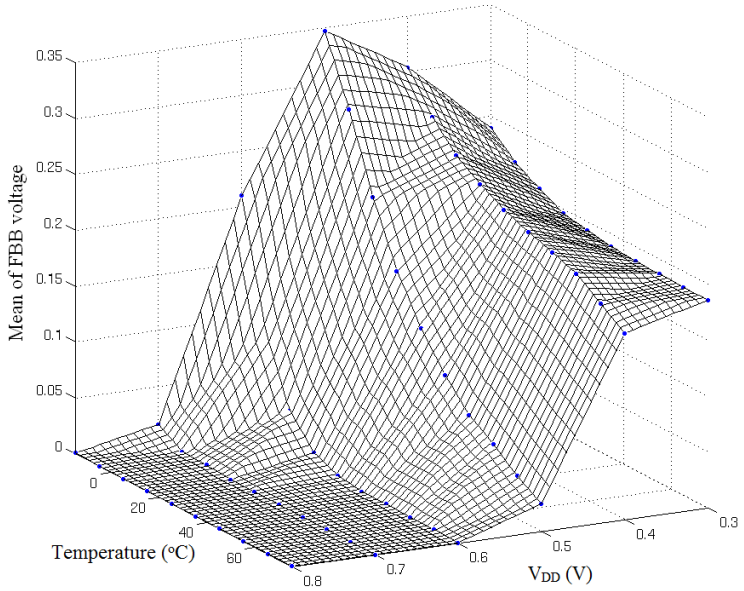


Figure 5.3. Simulation results for mean of PBB circuit output at different temperatures and voltages

Referring to Figure 4.5, which was sketched for T=25°C, it can be seen that Figure 5.3 actually follows the prediction made by equation (4.10) showing its usefulness as a model for studying SULP FBB behaviour.

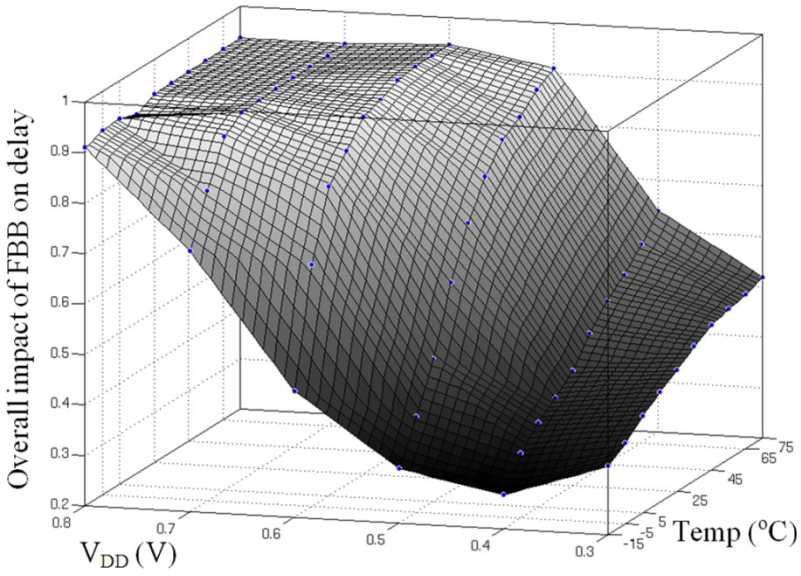


Figure 5.4. Impact on ZBB Mean Delay after introduction of the SULP FBB technique.

To verify the impact on delay, the critical path's delay is also recorded before (ZBB) and after (SULP FBB) body bias application during MC runs and temperature/voltage sweeps.

Figure 5.4 is the outcome that depicts the effect of the proposed technique on the delay of the examined adder. This also acknowledges Figure 4.6 which predicted the same behaviour by mathematical analysis.

Monte Carlo simulations on pre and post-layout HSPICE codes also have negligible differences, in terms of circuit delay and energy consumption, as timing and power requirements were kept satisfied through out the place & route procedures. For higher accuracy, which is gained by a higher number of MC iterations, the distribution of delays in the adder is achieved by pre-layout, instead of post-layout, MC runs for the ZBB and the SULP FBB cases (presented in Figure 5.5), which would have needed much longer time to be completed on a post-layout code. If sketched with respect to t_d (delay of the examined inverter in section 4.3), the PDF of the analysed inverter shown in equation (A.7) (presented in section A.2) would again be similar to Figure 5.5.

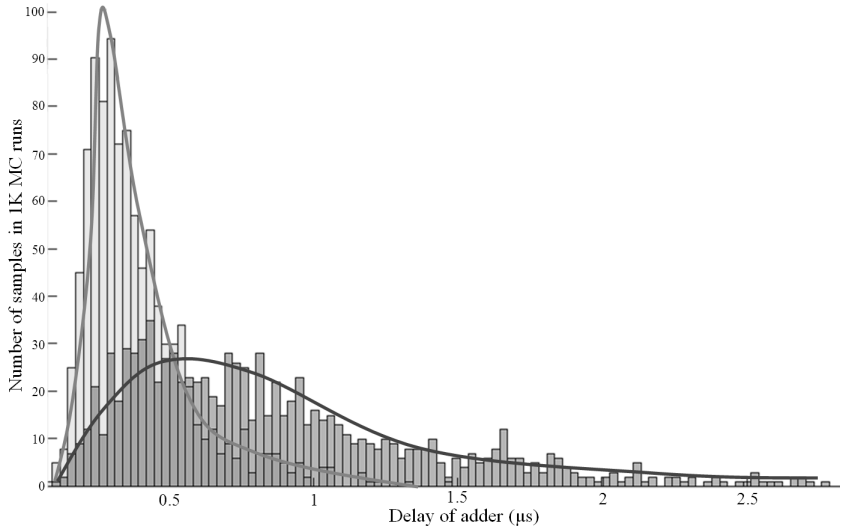


Figure 5.5. Variations in Delay resulting from 1K Monte Carlo simulations for SULP FBB and ZBB cases for T=25°C and V_{DD}=0.3V. Lighter bars represent SULP FBB and darker ones signify ZBB case.

Equation (A.7) also specifies that a lognormal distribution has to be expected for delay as is evident in the Figure 5.5. Figure 5.6 also shows that, as equation (A.10) implies, increasing voltage makes the SULP FBB ineffective and therefore impact on variance gradually starts to subside. This, in return, makes SULP FBB similar to the ZBB case at higher voltages, that is, delay variation of SULP FBB and ZBB become more similar as V_{DD} increases toward the superthreshold voltages.

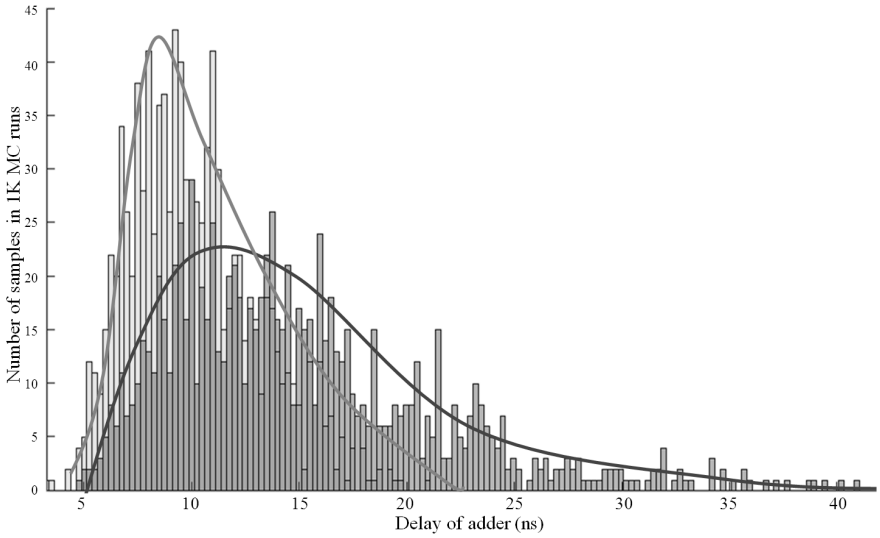


Figure 5.6. Variations in Delay similar to Figure 5.5 but with V_{DD}=0.5V

Figure 5.7 examines the energy overhead of the SULP FBB circuit. On average, the body bias generator has about 0.4% energy overhead across the temperature and voltage range for this adder. The 8-bit Kogee-Stone adder was solely chosen based on manageability of the circuit's overall size in order to execute Monte Carlo simulations at a reasonable run time. This small 0.4% energy overhead of SULP FBB leakage current becomes even more insignificant if the SULP FBB generator's size becomes negligible with respect to the chip's size.

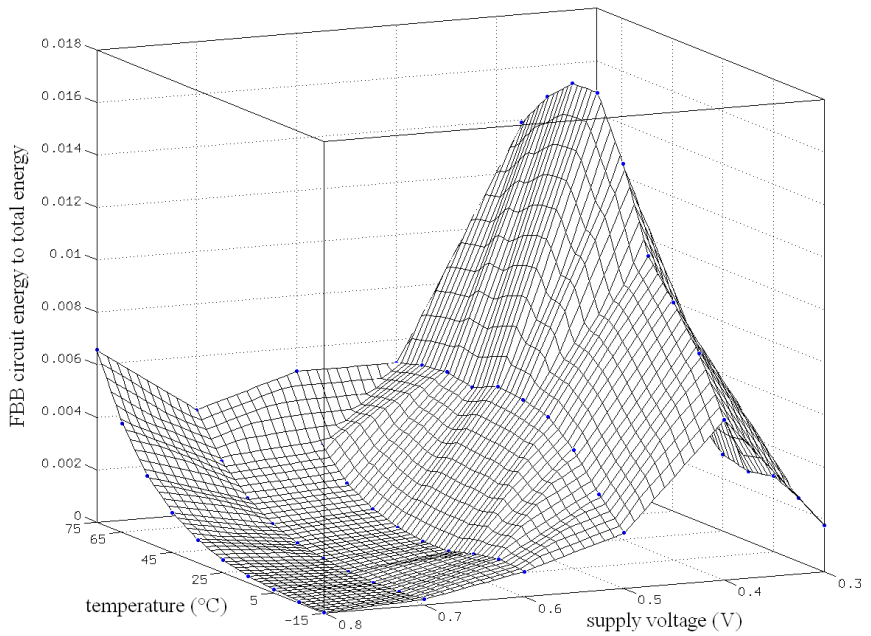


Figure 5.7. SULP Body bias generator's total energy to adder's total energy

It can be seen from Figure 5.2 that this is not the case for the 8-bit Kogee-Stone adder and, therefore, a small energy overhead is caused by the SULP FBB generator's leakage at higher temperatures. Simulations in Chapter 6 will exhibit that with more complicated systems this overhead (and especially the 14% area overhead) will become entirely negligible.

Finally, Figure 5.8 shows the impact of SULP FBB on EDP of the adder in which the overall (static and dynamic as well as adder plus SULP FBB circuit) energy consumption has been measured. By comparing Figure 5.8 and Figure 4.7, it can be seen that overall impact of SULP FBB on EDP, in Figure 5.8 and at 25°C, reacts almost the same to voltage scaling as Figure 4.7 predicted.

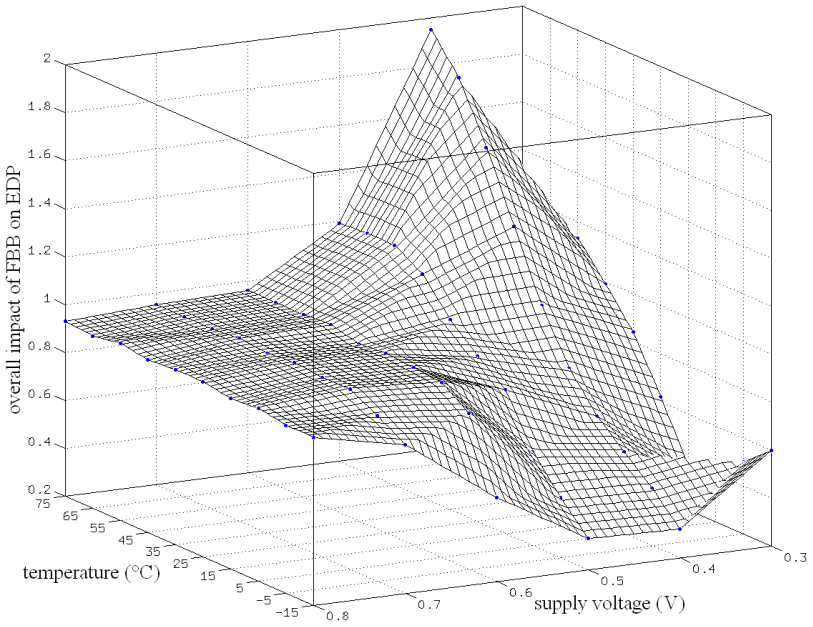


Figure 5.8. Overall EDP effect for SULP FBB technique with respect to ZBB case

Table 5.1 compares the result of post-layout simulations for ZBB and SULP FBB cases measured across 0.3V to 0.8V. It should be noted that raising voltage above 0.8V has no consequences in terms of SULP FBB functionality as FBB will be withdrawn above 0.8V in all corners and temperatures of this design and accordingly all achieved results will be identical. Improvements in Table 5.1 are calculated based on the ratio of the measured parameter after SULP FBB implementation to before SULP FBB implementation (or ZBB). Delay is measured for the critical path (average of rise and fall times), and so is frequency (maximum of fall and rise times reversed). Energy is determined using the integral of power over the period of simulation, and delay variations (measured in pre-layout simulations) based on 1K Monte Carlo runs. Improvement of delay variations by 7

to 9 times, depending on temperature interval and when EDP is reduced, indicates the power of this technique in addressing PVT variations as well as voltage scaling in a very efficient and well adaptive way.

Table 5.1 SULP FBB and ZBB Simulation results comparison

SULP Compared to ZBB	-15°C to 75°C	-15°C to 45°C
EDP improvement	23%	42%
Delay reduction	63%	75%
Frequency increase	45%	54%
Energy increase	33%	23%
Delay Variation reduction	730%	900%

Applying absolute FBB in all conditions, no matter what PVT situation the circuit is in, brings about a leaky system which delivers 7 times less delay compared to SULP FBB technique while consuming 100 times more energy. It has to be noted too that, at superthreshold voltages when performance is high enough to be sacrificed for reducing the leakage power dissipation, reverse body bias can be applied (instead of ZBB) by simply changing the buffering stages to voltage converters.

5.4 Error Rate Simulations and Results

Figure 5.9 shows a range of fast to slow process corners in a scatter graph of 1K MC runs on the adder's critical path. The outcome of 1K MC simulations have been spread by each run's process corner effect on the delay of PMOS and NMOS transistors of a typical ZBB inverter. It has to be pointed out that both axes and black lines are in logarithmic scale. Straight lines demonstrate the normalised delay reduction in the data-path at a specific process corner. When both PMOS and NMOS networks are slow (dots), SULP FBB automatically applies FBB to both networks whereas in SF or FS corners, FBB is only applied to NMOS or PMOS networks, respectively. This, in fact, indicates how SULP FBB manages to improve EDP as FBB is avoided at fast corners and high temperatures which lead to leaky, power consuming and, therefore, speedy transistors in subthreshold region. By cancelling FBB in such circumstances, power is not wasted on circuits that are fast already.

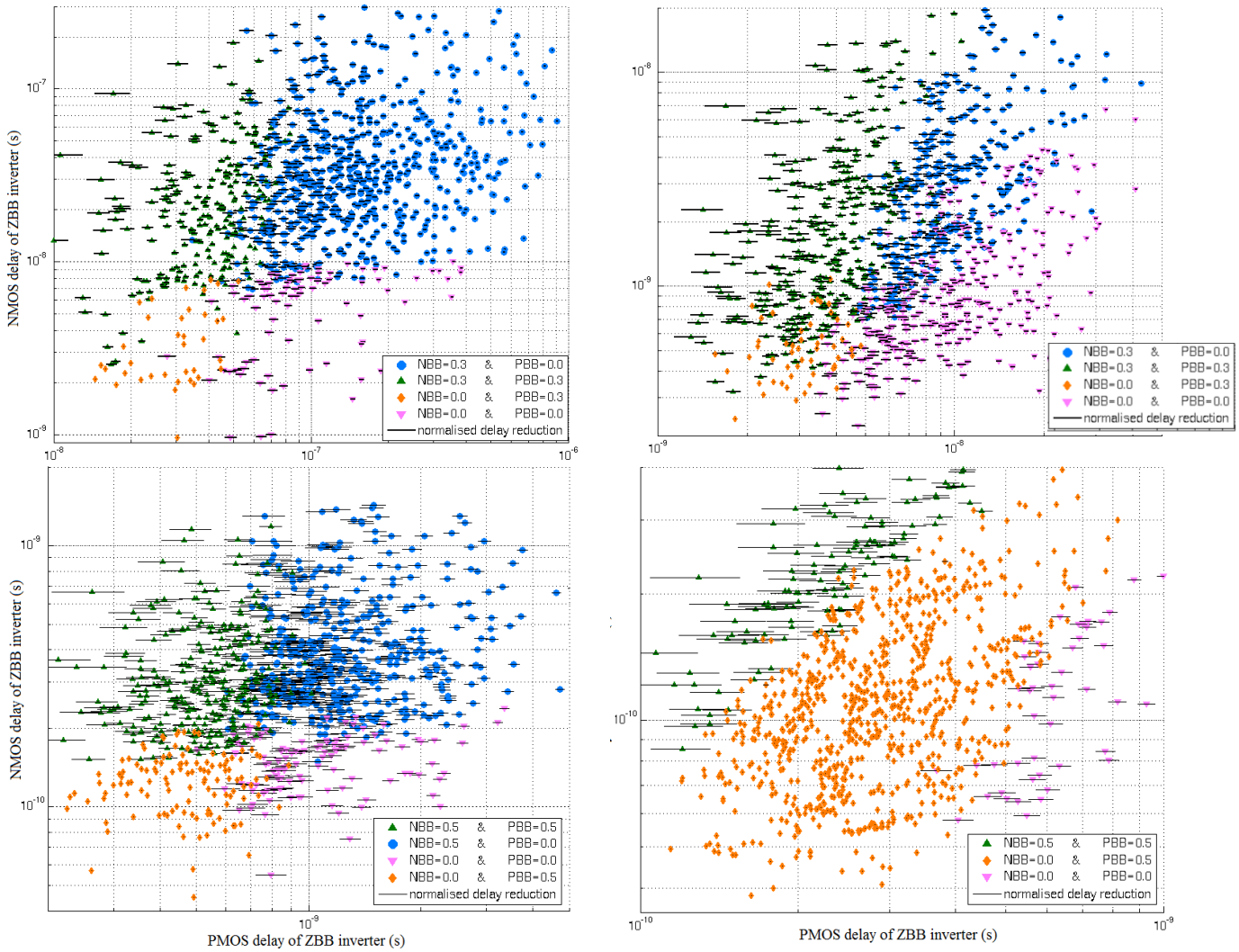


Figure 5.9. The applied FBB for different process variations and its effect on delay (black lines) at: 0.3V (two upper figures), 0.5V (two lower figures), -5°C (two left figures) and 75°C (two right figures). The longer black lines, the higher delay improvement as a result of SULP FBB.

In an FF corner (diamonds), neither NMOS nor PMOS networks are forward body biased hence resulting in no delay improvement as it is not required. Straight lines, indicating delay reduction, therefore, only exist when SULP FBB is applied to either or both networks.

It can be observed that straight lines stretch exponentially when PMOS network is fast (PMOS delay of ZBB inverter is low). That is due to the nature of PMOS devices being slower than NMOS counterparts and this has a great impact in subthreshold region to such an extent that even the forward body biased PMOS network is not as fast as forward body biased NMOS network. This imbalance can be addressed by libraries which are designed for both sub and superthreshold regions [115] and can improve the achieved results in this study even further.

In Figure 5.9, when temperature/voltage increases, SULP FBB generator reduces the number of FBB applications because

- Higher temperature leads to higher subthreshold leakage and faster devices in subthreshold domain and
- Higher supply voltage leads to stronger and hence faster devices

In this way, by sensing the temperature and voltage, the SULP FBB technique avoids applying FBB to devices which are fast enough already due to process, temperature, or voltage variations.

Simulations for the adder also verify the predictions of equations (4.23) and (4.24) abstracted in Table 5.2 and Figure 5.10. As shown by Figure 4.9, there is a maximum improvement curve for Kogge-Stone adder data-path which, as explained, is expected to be higher than what was predicted by the theoretical inverter. This curve can be observed along the maximum improvement points in Figure 5.10.

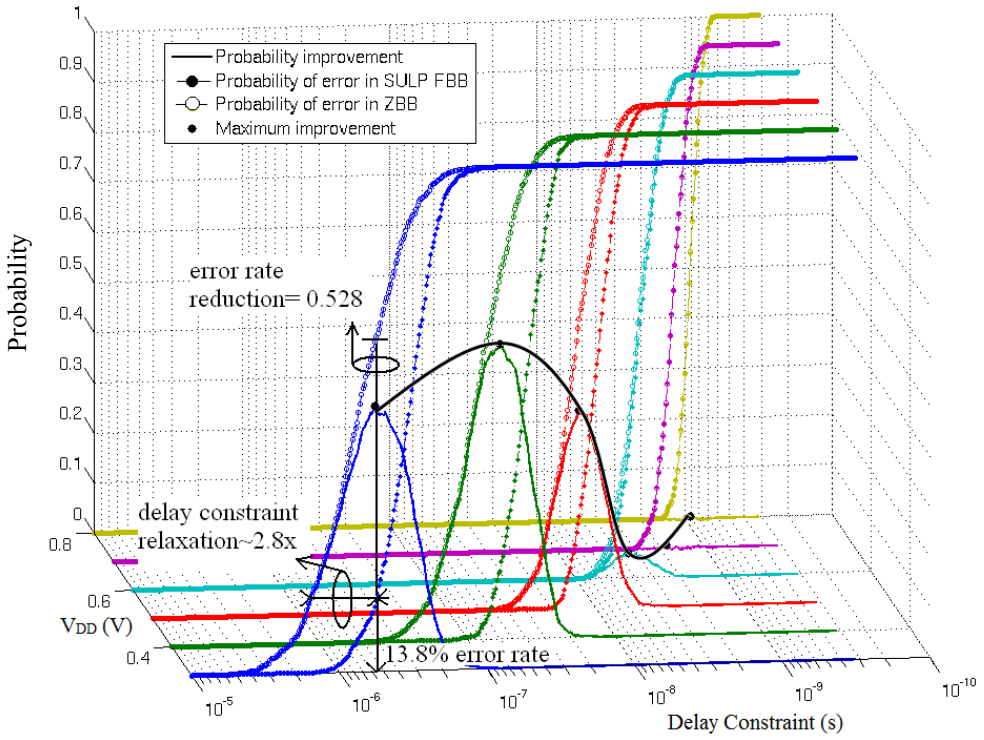


Figure 5.10 Probability of error for SULP FBB and ZBB data-path and the maximum improvement gained for 1K MC simulations at 25°C

Table 5.2 shows the error probability of 0.138 (error rate of 13.8%) for V_{DD} = 0.3V when maximum error probability reduction is sought, which will be equal to 0.528. That is, at this maximum error reduction point, if the delay constraint of both SULP FBB and ZBB inverters were to be the same, then SULP FBB would have 13.8% error rate while ZBB had 13.8%+52.8% error rate.

Table 5.2 Error rate and delay constraint improvements after SULP FBB application

Design objective→	**1% error rate**	**5% error rate**	**maximum improvement**	
Supply Voltage (V)↓	**error rate reduction/delay constraint relaxation(x)**			**error rate (%)**
0.3	0.296/3.3113	0.422/2.8184	0.528/2.8184	13.8
0.4	0.495/3.3884	0.571/2.9512	0.589/2.6303	13.9
0.5	0.256/1.9953	0.335/1.7378	0.394/1.5849	18.8
0.6	0.012/1.0471	0.033/1.0965	0.048/1.0233	26.3
0.7	0/0.9772	0/0.9772	0/1	29
0.8	0/0.9772	0/0.9772	0/1	53.2

On the other hand, if the error rates of SULP FBB and ZBB data-path were to be the same (in this case 13.8%), then Table 5.2 and Figure 5.10 show that a ZBB data-path would have the benefit of ~2.8 times delay constraint relaxation after SULP FBB application.

Although error rate reduction is maximum along this curve, the SULP FBB error rate (probability of error) is not satisfactory along it even for the subthreshold voltage domain. In this case, the choice of maximum error rate improvement leads to an error rate of 13.8% or higher (across different voltages) in data-path which is very yield damaging and, in terms of error rate reduction, it has no benefit in superthreshold region.

As explained in section 4.4, it is error rate that is often set as an objective, in which case Table 5.2 shows examinations of two error rate constraints 1% and 5%. As before, depending on the cap on either error rate or delay constraint, Table 5.2 shows two values for each error rate. In 0.4V for example, if cap is on delay constraint, then SULP FBB improves error rate of ZBB data-path from 49.5% to 1% (around 50% improvement in timing yield). If cap should be on the error rate of 1%, then SULP FBB technique can relax delay constraint by ~3.4 times.

As limit on the final error rate (probability of error) in SULP FBB data-path is tightened, error rate reduction declines as well, due to distancing from the maximum reduction point which is experienced at higher error rates. Despite this decline, the error reduction is still significant even for the error rate of 1%. With this error rate, when SULP FBB is applied, the delay constraint can be relaxed even more, compared to the case of maximum error rate reduction.

This is as a result of two previously discussed characteristics of the SULP FBB technique:

- Performance improvement in the data-path by providing FBB when required and
- Variations reduction in data-path delay

The former feature can be observed in Figure 5.10, in which, as delay constraint reduces (tightens), probability of error raises to 1 in ZBB earlier than SULP FBB, demonstrating SULP FBB's supremacy in performance as it now enables the data-path to perform faster because delay constraints can be tougher.

The latter feature is apparent in the slope of error rate curve in Figure 5.10 when rising up from 0 to 1, with steeper curves belonging to SULP FBB due to its lower delay variation. By referring to equations (4.23) and (4.24), cumulative distribution functions can be found for random variables T_{dZBB} and T_d, respectively. These functions, in fact, show cumulative lognormal distributions with the associated tails. As Figure 5.5 showed, longer tails belong to ZBB data-path because of the effect of extreme corner variations. As a result, when error rate is reduced to lower percentages (like 1% here), a ZBB data-

path will fail more and more as extreme corner variations play an important role in this case, and satisfying such extreme process corners will need looser delay constraints and hence SULP FBB will be more beneficial in this situation as is shown in both Table 5.2 and Figure 5.10.

Another significant problem in the subthreshold region is severe temperature dependency. As in Figure 5.11 asterisks illustrate for a ZBB data-path, when supply voltage approaches the subthreshold voltage domain, variance in the delay leading to a 10% error rate or less, increases across different temperatures. As temperature or voltage increases, applied delay constraint can be dropped dramatically (z axis is in logarithmic scale) which suggests a decline in the rate of FBB application in higher temperatures or voltages if an FBB were to be used.

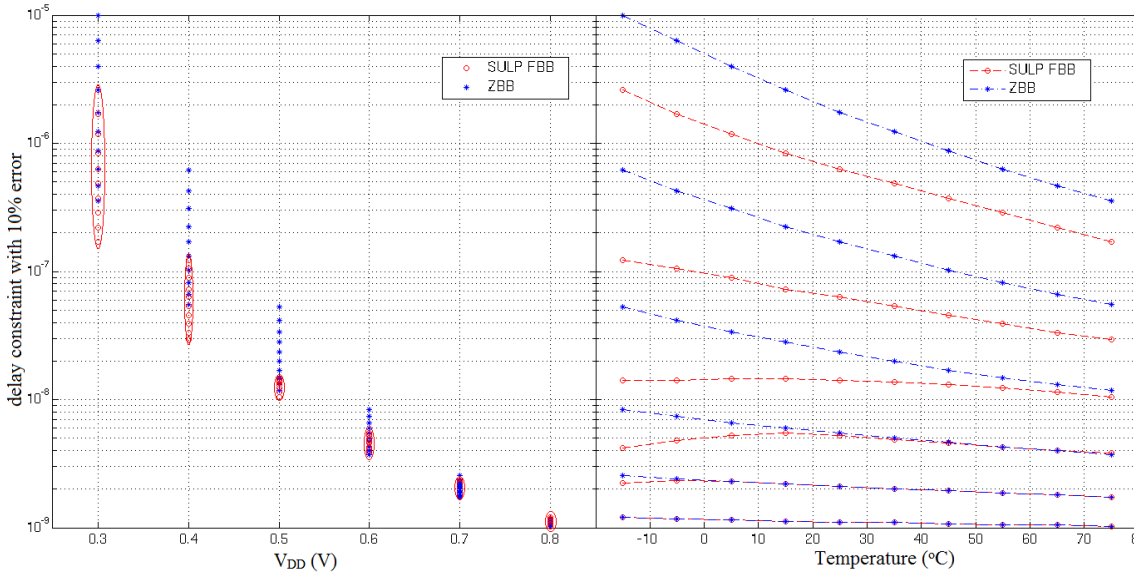


Figure 5.11. X-Z and Y-Z views of the 3D plot showing minimum delay constraint which leads to 10% error rate (Z) for different voltages (X) and temperatures (Y).

Figure 5.3 showed that SULP FBB has been designed considering this decline and points in Figure 5.11, signifying SULP FBB impact, represent this consideration. Although delay constraints can be reduced hugely after SULP FBB application, especially in subthreshold domain, the best temperature independent voltage at sub/nearthreshold domain (the highest temperature yield or the smallest ellipse in Figure 5.11, except ones in superthreshold voltages of 0.7V and 0.8V) occurs at 0.5V which indicates how important the priorities (timing yield or temperature yield) are while tuning the SULP FBB generator.

If higher timing yields at subthreshold voltages are more important than temperature yield (which was the case in this thesis), then mean of FBB voltage has to be higher at the subthreshold region with respect to nearthreshold domain (presented in Figure 5.3). For example, Figure 5.12, in which the error rate improvement curve in Figure 5.10 has been extended to different temperatures, demonstrates that higher error rate improvement (which leads to higher timing yield) is achieved at subthreshold domain, with less regard to temperature, but at nearthreshold domain this improvement will be largely temperature dependant as timing yield is less significant in higher voltages in this design.

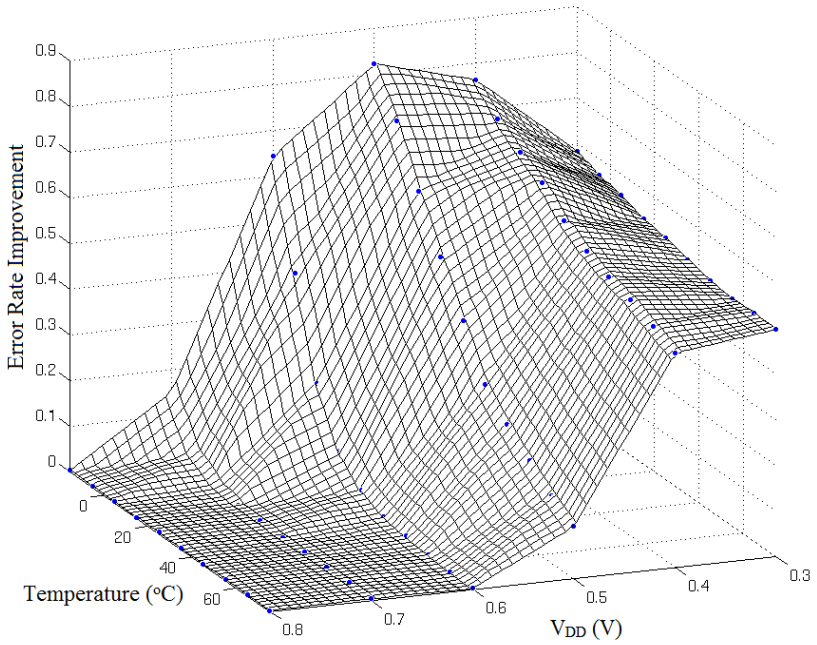


Figure 5.12. Extension of error rate improvement curve (black curve) in Figure 5.10 to different temperatures from -15°C to 75°C

5.5 Conclusion

In this chapter, by examining a Kogge-Stone adder, all theoretical claims made in Chapter 4 were confirmed by means of Monte Carlo post-layout simulations. EDP of the adder was examined and the high temperature influence on EDP deterioration was discussed. Results showed that, assuming a voltage scaling form 0.8V to 0.3V and temperature changes of -15°C to 75°C, the proposed technique leads to a 7 times less delay variation while energy delay product improves by 23% compared to a zero body biased adder. Furthermore, error probability was decreased from ~50% to 1% at 0.4V as a result of this technique.

The next chapter studies the effect of SULP FBB generator at much larger scales to ensure the sustainability of the scalability and reliability of the proposed technique in the most demanding circumstances.

6 An FFT Design Using SULP FBB

6.1 Introduction

The literature review in Chapter 2, pointed out the necessity to consider the effect of pipelining when performance is degraded as a result of aggressive DVS. On the other hand, Chapter 5 only tested the SULP FBB technique on a simple Kogge-Stone adder, in order to be able to handle a multitude of Monte Carlo simulations run to compare with statistical predictions made in Chapter 4. To examine the proposed technique at a larger scale and on a circuit considerably more sizable with respect to the SULP FBB generator, and in order to determine more realistic outcomes, this chapter thoroughly analyses the impact of SULP FBB technique using two different FFT processors and by means of corner simulations. After briefly explaining the algorithm of the FFT processor and choosing the appropriate architectures for the implementation, simulation results are examined and comparisons made with previous works.

6.2 Implementing the FFT Processor

FFT processors are widely used in many signal processing and communication applications such as WLAN and ADSL networks and digital audio/video broadcasting devices.

A linear transform, on a vector of n points, is defined as multiplication of an $n \times n$ matrix by the vector. Discrete Fourier Transform on input vector x is, as a result, defined by $y=DFT_n\ x$, where y is the output vector and DFT_n is the transform defined by equation (6.1) [116].

$$DFT_n = [\omega_n^{kl}]_{0 \le k,l < n}\ ,\ \omega_n = e^{-\frac{2\pi i}{n}} \tag{6.1}$$

where $[\ ]_{0 \le k,l < n}$ is an $n \times n$ matrix with ω_n^{kl} being the element of row k and column l of the matrix, while $0 \le k,l < n$.

For example, the well-known butterfly transform is a *DFT* with $n=2$ formulated by (6.2):

$$DFT_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{6.2}$$

As DFT_n is an $n \times n$ matrix, calculating $y=DFT_n\ x$ has $O(n^2)$ computation complexity. There are many algorithms, called Fast Fourier Transforms (FFT), that take advantage of factorisation of DFT_n into consecutive multiplications of sparse matrices which leads to $O(nlog(n))$ arithmetic complexity [116].

Employed to generate the FFT processors examined in this thesis, the Pease algorithm is one example of such FFT algorithms, which is defined by equation (6.3):

$$DFT_{2^k} = R_{2^k}\left(\prod_{i=0}^{k-1} T_i\left(I_{2^{k-1}} \otimes DFT_2\right) L^{2^k}_{2^{k-1}}\right) \tag{6.3}$$

where L^n_m is stride permutation defined by equation (6.4) [117]:

$$L^n_m : i \cdot \left(\frac{n}{m}\right) + j \mapsto j \cdot m + i, 0 \le i < m, 0 \le j < \frac{n}{m} \tag{6.4}$$

Operator $\otimes$ in equation (6.3) is called a tensor (or Kronecker) product and defined by equation (6.5).

$$B \otimes A = \left[b_{k,l}A\right] \ where\ B = \left[b_{k,l}\right] \tag{6.5}$$

T_i in equation (6.3) is a diagonal matrix composed of complex numbers called twiddle factors [117] which are usually stored in ROMs.

Finally, R_{2^k} in equation (6.3) is called bit reversal permutation and defined as:

$$R_{2^k} = \prod_{i=0}^{k-1}\left(I_{2^{k-i-1}} \otimes L_2^{2^{i+1}}\right) \tag{6.6}$$

For example, if L^8_4 is applied on [0,1,2,3,4,5,6,7], permutation $2i+j \rightarrow 4j+i, 0 \le i < 4,\ 0 \le j < 2$ implies that element 2×2+1 in the output array is actually element 4×1+2 in the input array:

$$\begin{bmatrix}0\\4\\1\\5\\2\\6\\3\\7\end{bmatrix} = \begin{bmatrix}1&.&.&.&.&.&.&.\\.&.&.&.&1&.&.&.\\.&1&.&.&.&.&.&.\\.&.&.&.&.&1&.&.\\.&.&1&.&.&.&.&.\\.&.&.&.&.&.&1&.\\.&.&.&1&.&.&.&.\\.&.&.&.&.&.&.&1\end{bmatrix}\begin{bmatrix}0\\1\\2\\3\\4\\5\\6\\7\end{bmatrix} \tag{6.7}$$

As an example for tensor product, $I_n \otimes DFT_2$ has been shown in equation (6.8) which is a $2n \times 2n$ matrix with DFT_2 matrices along the diagonal and zeros everywhere else.

$$I_n \otimes DFT_2 = \begin{bmatrix}DFT_2 & & \\ & \ddots & \\ & & DFT_2\end{bmatrix}_{2n\times 2n} \tag{6.8}$$

The reversal permutation functionality can also be demonstrated by another example. Assuming [0,1,2,3,4,5,6,7] as the input array, applying R_8 changes the input indices according to binary reversal permutation 00→00, 01→10, 10→01, 11→11, resulting in [0,4,2,6,1,5,3,7].

As an "altogether example", DFT_8 has been formulated in equation (6.9) and shown by dataflow in Figure 6.1.

$$DFT_8 = R_8\{T_2(I_4 \otimes DFT_2)L_4^8\}\{T_1(I_4 \otimes DFT_2)L_4^8\}\{T_0(I_4 \otimes DFT_2)L_4^8\} \tag{6.9}$$

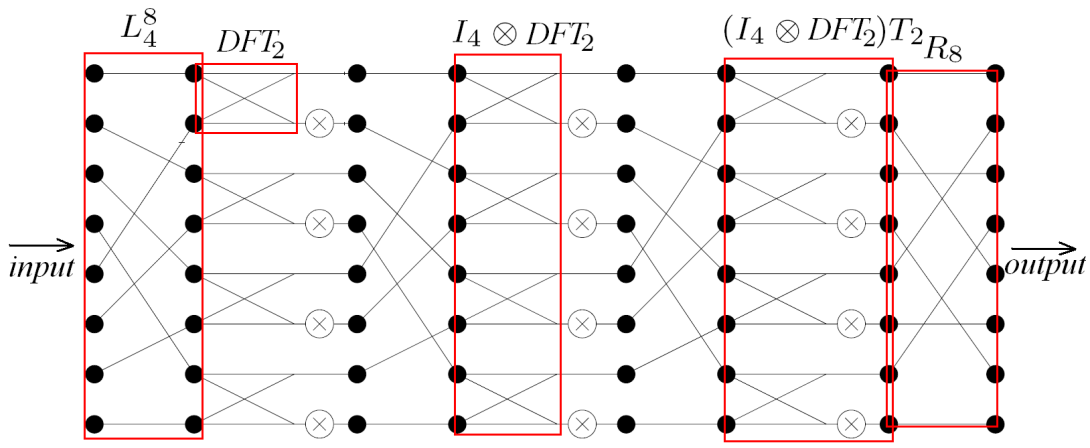


Figure 6.1. DFT_8 dataflow using Pease FFT algorithm

If input vector x is fed into DFT_8, specified in equation (6.9), R_8 would be the last permutation applied before output is calculated, as Figure 6.1 shows. It should be noted

that, there is only one multiplication following each butterfly matrix of DFT_2 because the other twiddle factor is always 1 [118].

As equation (6.9) shows, the only matrices changing in each step are twiddle factors (if each pair of braces is considered one step) and this implies that all three steps can be combined and implemented into one step in hardware while twiddle factors are looked up from ROM memories. As Figure 6.2 exhibits, this is called horizontal folding because it can be assumed that the three steps in Figure 6.1 are horizontally folded to form the repetitive part of Figure 6.2. The horizontal folding is prompted by the fact that the Pease algorithm in equation (6.3) has a repetitive matrix product which forms identical steps creating a platform for space parallelism, i.e. instead of having similar steps consuming space, all units can be integrated and form one repeatable step that results in space and, therefore, power saving. These similar steps can also be configured for pipelining purposes which is discussed later.

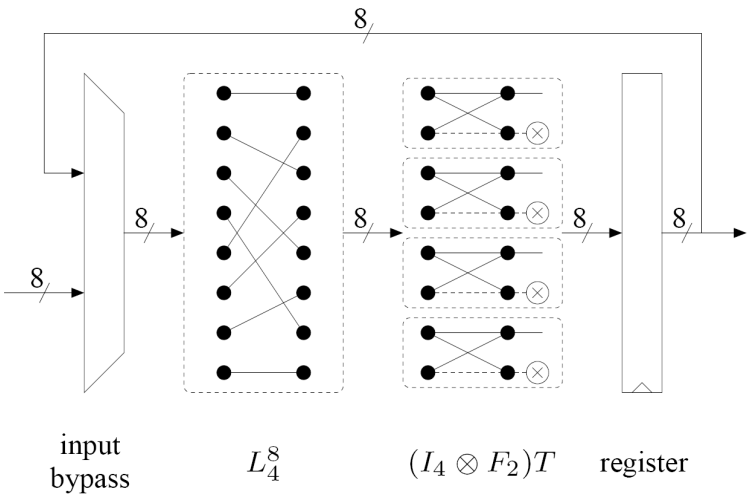


Figure 6.2. A horizontally folded Pease FFT

The register shown in Figure 6.2 can be implemented by latches, flip-flops, register files, or SRAMs, depending on both FFT configuration and how fast and low power the design should be. The iterative structure shown in Figure 6.2 is implemented in this chapter with twiddle factors stored in ROMs and with intermediate SRAMs as registers, to verify the SULP FBB technique in handling such large systems. SRAMs, however, need to be dual-port, that is, they should be read and write accessible at the same time.

If compared with Figure 6.1, the example shown in equation (6.8) indicated that the tensor product results in n times (4 times in Figure 6.1) parallel reuse of DFT_2. These

parallel units can be vertically folded to generate time parallelism, i.e. one unit of DFT_2 can be used 4 consecutive times which results in timing delay and requires buffering.

Horizontal and vertical folding can also be employed at the same time. For example, Figure 6.3 shows the effect of both space and time parallelism on example equation (6.9).

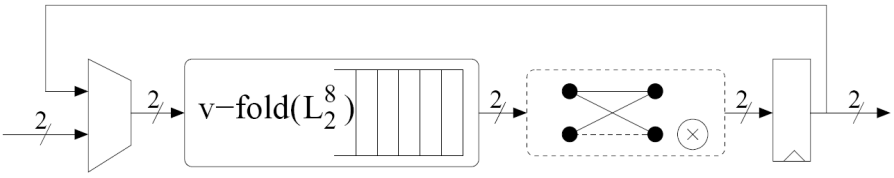


Figure 6.3. A horizontally and vertically folded Pease FFT

If DFT_2 is considered the smallest computational block, then the number of these blocks can be signified by the parameter p, called degree of parallelism. For example, Figure 6.2 and Figure 6.3, both with $n=8$, have $p=n/2=4$ and $p=1$, respectively.

However, vertically folding a permutation matrix can be very complicated as it not only has to buffer and reorder the 8-bit input data and stream them out nonstop, but it has to also guarantee not writing to and reading from the same memory cell at the same time. Without vertical folding, stride permutation L_m^n can be simply implemented by wires. The authors in [118], however, have utilised a technique to address this issue originally proposed in [119]. This technique decomposes $L_{n/2}^n$ into one L_p^{2p} permutation with a simple wiring, $1 \leq p \leq n/2$, and 2-input, 2-output J_m blocks (presented in Figure 6.4). J_m blocks simply have two FIFOs shifters to shift the incoming data and a programmable switch that either act as pass-through for $m/4$ cycles or criss-cross for $m/4$ cycles [120].

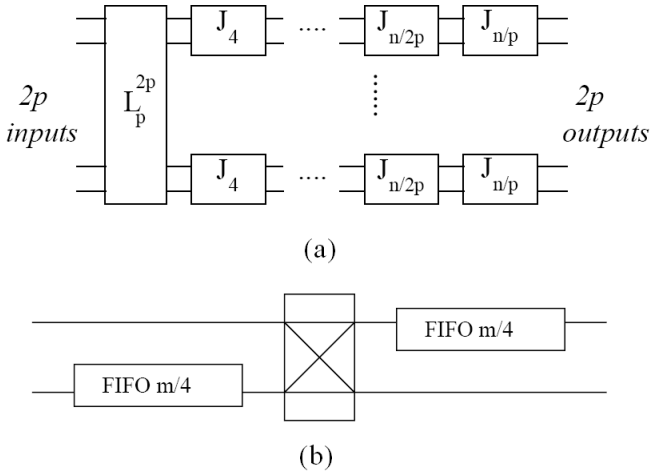


Figure 6.4. a) A vertically folded $L_{n/2}^n$ permutation with a L_p^{2p} permutation and J_m blocks and b) A J_m block

Through folding and by varying parameter *p*, different structures are created by means of a specific tool called Spiral [121], and then simulated in the following sections in order to examine the SULP FBB reaction in different architectures.

6.3 1024 point, Radix 4, 32x32bit Complex FFT

In this section, the proposed technique is examined by being applied to two extensive Fast Fourier Transform (FFT) processors. Two 1024 point, radix 4, 32x32bit complex input FFT processors were generated:

- An iterative version with 64 64x64bit SRAMs and 48 32x1024bit ROMs which takes 241 cycles to perform an FFT,
- A pipelined version with flip-flops which takes 32 cycles per FFT

The first version is used comprehensively to examine SULP FBB effect. However, only corner analysis was performed as practical Monte Carlo simulations were impossible given the scale of the FFTs.

Figure 6.5 shows the layout of the iterative FFT. Two voltage domains of 1.2V and variable voltages (0.3V-1.2V) plus different partitions have been highlighted. As briefly mentioned in Chapter 3, a low power Standard Cell Library was also created in two versions of high voltage (no body bias) and variable voltage (body biased). The required level shifter, capable of shifting voltages in the interval of (0.3V,1.2V) up to 1.2V, was adapted from [103] (presented in section 3.3.3).

As the current sink from the FBB generator is negligible, compared with the current of main power sources, the FBB generators and the connected straps can be designed to be small in the area, despite sourcing numerous numbers of cells. Simulations on this enormous circuit with ~880,000 standard cells were performed using Synopsys Discovery AMS (Analogue Mixed Signal) suite mixing VCS-MX and CustomSim-XA.

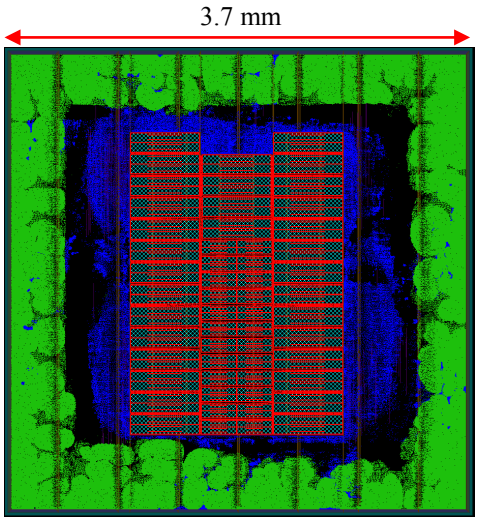


Figure 6.5. Layout of the iterative FFT; red area (middle part) is 1.2V domain and the rest variable voltage domain; green area (bordering accumulation) shows adders/multipliers and the blue area (between two previous ones) is the permutation part in the FFT

Figure 6.6 shows the outcome of simulations (with and without SULP FBB application) in a TT corner and 25°C temperature across different voltages. It is clear from Figure 6.6 that, in this corner and temperature, SULP FBB is applied only at voltages under 0.5V.

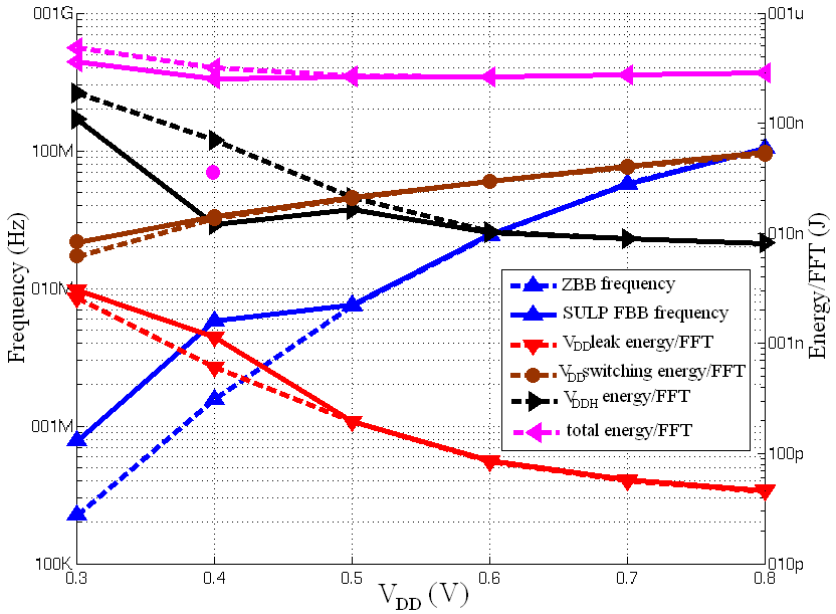


Figure 6.6. Frequency and energy per FFT at temperature 25°C and TT corner for two techniques of SULP FBB and ZBB

As the memory and its related low-to-high level shifters and buffers (inverters) all are placed in a V_{DDH} (1.2V) voltage domain, Figure 6.6 depicts four types of energy consumptions. Black lines (marked ►) show the energy per FFT consumed in V_{DDH} domain except memory consumption, brown lines (marked ●) show switching energy of V_{DD} domain energy (subthreshold domain switching energy per FFT) and magenta lines

(marked ◄) show total energy per FFT. Blue lines (marked ▲) show frequency in the FFT. As presented in [100], energy per instruction does not change in subthreshold voltages, when V_{th} is changed by body biasing, and subthreshold switching energy in Figure 6.6 illustrates this fact. This is because the same amount of energy is needed to switch a gate in both scenarios no matter how fast the gates switch.

Although leakage current is higher after SULP FBB application, the frequency is higher too, which in return increases throughput and reduces working times resulting in reduced total leakage energy and, therefore, compensates the effect of increased leakage current. Besides, overall delay is reduced by ~3.6 times (presented in Table 6.1) when FFT is in subthreshold voltage domain and SULP FBB is applied. However, consumed power remains the same for V_{DDH} domain (as voltage and therefore currents remain the same) at subthreshold voltages, and together with delay reduction, the SULP FBB application decreases the V_{DDH} domain energy and in return total consumed energy.

It is clear that, at subthreshold voltages, V_{DDH} domain has comparable power consumption to V_{DD} domain because V_{DD} domain consumes less switching power. V_{DDH} domain also needs to wait most of the time for the subthreshold part which results in huge leakage power consumption as well (especially in memory). This difference in energy consumption has been presented in Table 6.1.

Table 6.1. Improvements in ZBB FFT after SULP FBB technique is applied for TT corner and 25°C and 75°C temperatures.

SULP FBB effect→	**delay improvement**		**energy reduction per FFT**		**EDP improvement**		**V_{DDH} energy reduction per FFT**		**V_{DD} energy increase per FFT**	
voltage↓	**25°C**	**75°C**	**25°C**	**75°C**	**25°C**	**75°C**	**25°C**	**75°C**	**25°C**	**75°C**
0.3V	3.45x	2.8x	32%	40%	4.55x	4x	1.70x	2.3x	29%	0%
0.4V	3.72x	-	25%	-	4.64x	-	5.82x	-	7%	-
average	3.58x	-	29%	-	4.60x	-	3.76x	-	18%	-

As it approaches higher voltages, V_{DD} domain consumes more of the overall energy. It should be noted that the use of an industrial memory in these simulations gave rise to such high total energy consumption, because the whole V_{DDH} domain had to remain supplied with 1.2V power supply, regardless of V_{DD} domain voltage.

To give an idea of how much this high voltage memory (SRAM, ROM and level shifters and buffers) contributed to the total power consumption, a single dot can be seen in Figure 6.6 showing total consumed energy at 0.4V (for both ZBB and SULP FBB

techniques), if a subthreshold memory (like what proposed in [76] and [73]) were to be used. This subthreshold memory results in 7 times less energy consumption making the total energy ~36nJ per FFT at ~6MHz (~25K FFT/sec).

In this case, consumed V_{DDH} energy in Figure 6.6 is eliminated and now the only energy cost, as a result of SULP FBB application, would be V_{DD} leakage energy which is 0.5nJ per FFT and is negligible compared to the total consumed energy and is hence not distinguishable in Figure 6.6 for these two techniques. Therefore, EDP is improved by the amount of delay improvement which is ~3.6 times, rather than ~4.6 times of the industrial memory case (presented in Table 6.1).

Table 6.1 also shows that, at 75°C, SULP FBB is only applied at 0.3V and is cancelled in higher supply voltages. As explained in Chapters 4 and 5 in detail, this is due to the increased subthreshold current at higher temperatures that leads to a faster subthreshold circuit and, hence, obviates the need for an FBB application. Moreover, 4 times EDP improvement (reduction) at 75°C shown in Table 6.1 emphasises that, despite being at a high temperature and having an increased leakage in SULP FBB circuit (presented in Figure 5.7), this technique's benefit is more pronounced in sizable circuits, whilst the small circuit of the Kogge-Stone adder resulted in almost two times EDP increase (presented in Figure 5.8) at this temperature. The V_{DD} energy consumption shows almost no change, after SULP FBB application and at 75°C, which demonstrates that subthreshold leakage, due to temperature rise, dominates and surpasses the subthreshold leakage, due to FBB application.

Figure 6.7 shows the difference between delays of FF and SS corners signifying ±3σ variation and Table 6.2 compares these values and exhibits the delay variation improvements across different voltages, after SULP FBB is applied. It should be mentioned that all corners (SS, SF, FS, FF and TT) were examined for functionality but corners of interest are FF and SS corners as they are indicative of ±3σ variation. Comparison of Figure 6.7 with Figure 6.6 shows that FBB is applied in an SS corner earlier than in a TT corner and is not applied in an FF corner at all. This is because the SULP FBB generator senses variations in process as well as voltage and applies the FBB depending on how slow the PMOS and/or NMOS networks are.

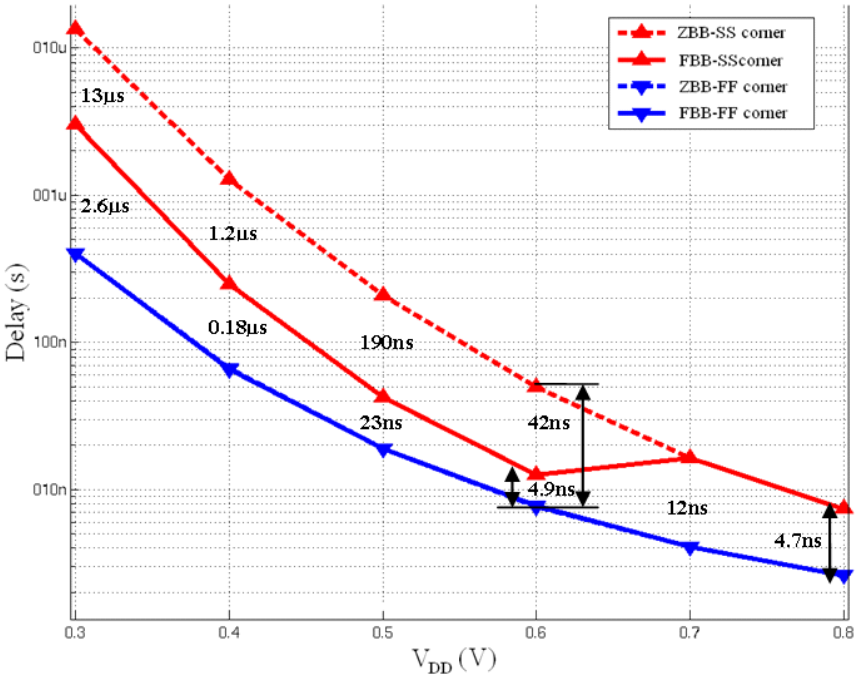


Figure 6.7. Maximum delay in FFT at temperature of 25°C and SS and FF corners for two techniques of SULP FBB and ZBB.

Table 6.2. Variation improvement and V_{DD} domain energy portion when SULP FBB is applied.

voltage (V)	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**
variation reduction (SS to FF)	4.98x	6.63x	8.22x	8.43x	-	-
V_{DD} domain energy to total energy (%)	3.15	5.96	8.05	11.15	14.89	18.79

As supply voltage scales down, delay variations increase; but with SULP FBB application, on average ~7 times improvement is achieved in the delay variations compared to the ZBB case. But this variation improvement deteriorates as V_{DD} scales more towards subthreshold voltages leaving the maximum improvement (in yield) at 0.6V. An SULP FBB design with different supply voltages from the system's supply voltage (in the buffer stages) can solve this problem by applying higher levels of forward body bias with respect to the system's supply voltage. Applying higher FBB voltages, however, leads to increased energy overhead.

Although SULP FBB, on a larger scale such as the simulated FFT, once again helps improve EDP and delay variation (timing yield), designing an FFT with two separate voltage domains for the memory and the logic, turns out to be very energy consuming even with a subthreshold memory. Using high voltage memory together with low voltage

processors has been very popular [72, 122-123] since memories usually start to fail at subthreshold voltages while the logic part is still functional.

However, as was mentioned before, another option in this section's FFT implementation is using flip-flops or latches instead of internal memory which also makes pipelining possible (by means of register files) and eliminates the use of level-shifters and the high voltage domain needed for memory blocks [67].

By exploiting this idea and for the sake of comparison with other works, a pipelined 1024 point, radix 4, complex 32x32b input FFT processor was implemented. Using SULP FBB on this FFT resulted in ~2.5 times delay improvement with 34% energy overhead with respect to the ZBB FFT. This FFT consumed 412pJ/FFT which is ~43 times less than the FFT in [67] while it is ~1.9 times slower (with 125K FFT/sec throughput).

These different FFT structures not only show the flexibility of the SULP FBB technique to be employed across different structures and applications, but also prove its EDP improvement is independent of temperature and application. This makes the SULP FBB technique a reliable and simple solution for the yield and performance issues experienced in sub 65nm technologies.

6.4 Conclusion

In this chapter, after the brief mathematical background into the employed Pease FFT structure and how it can be implemented into iterative and streaming architectures, these two implementations were laid out and simulated in different working situations. Results, once again, showed a 7 times improvement in delay variations. EDP product was also reduced to around 4 times, as a result of SULP FBB technique.

Finally, a pipelined version of FFT showed about 40 times less energy consumption per FFT, compared to the latest low power FFT implementations, while being only 2 times slower.

7 Conclusion and Future Works

7.1 Major Outcomes

This study created a mathematical platform for researching the behaviour of the proposed ultra low power body bias technique when confronting the PVT variations. This platform is able to precisely and accurately predict the functionality, performance, and energy consumption of similar low power techniques in order to analyse and evaluate the final operation. The analysis outcomes show a high level of sensitivity in this technique to process variations (~6V/V at near threshold voltages) and a reduced energy-delay product (EDP) in the analysed inverter.

In addition to sensing the PVT variations at subthreshold voltages, the proposed technique was also capable of reacting to DVS. When combined, these two capabilities meant that if a circuit is working at sub/nearthreshold voltages or any other performance restricting conditions (as a result of process and temperature variations), the proposed technique detects these circumstances and helps the system cope with the PVT variations which are more pronounced at sub/nearthreshold voltages.

On the other hand, the provided support is withdrawn when the system is working in superthreshold voltages or any other high performance circumstances (as a result of process and temperature variations).

The former response results in considerable reduction of functionality failures at slow conditions and the latter response incurs no energy overhead at fast situations. This technique also leads to a significant improvement in production yield as a result of its adaptive approach towards PVT variations as well as voltage reductions.

Assuming a voltage scaling down form 0.8V to 0.3V and temperature changes from -15°C to 75°C, a seven times improvement in the delay variation was observed in the simulated 8-bit Kogge-Stone adder plus 23% improvement in its EDP. Its error probability was also decreased from 50% to 1% at 0.4V as a result of this technique.

The outcome of mixed signal simulations on a 1024 point, radix 4, 32x32bit complex input iterative FFT processor showed not only the same seven times improvement in

delay variations, as before, but EDP was also reduced by around 4 times, as a result of this technique, which showed the real benefit of the technique lying in large scale circuits.

Finally, a pipelined version of the FFT consumed 412pJ/FFT which was ~43 times less than the latest sub/nearthreshold FFT processor but ~1.9 times slower than it (with 125K FFT/sec throughput). SULP FBB also resulted in ~2.5 times delay improvement in this pipelined FFT processor, with respect to a ZBB FFT processor, with a 34% energy overhead.

7.2 Conclusion

In this section, achievements throughout this thesis are summarised and conclusions presented. Before that, however, the beneficial characteristics of the SULP FBB generator are also outlined briefly as follows:

- Sensitivity to sub/nearthreshold voltages as well as performance restricting conditions caused by process and temperature variations, to generate the FBB
- Withdrawing the applied FBB when in high voltage levels or any other high performance circumstances occurring due to process and temperature variations
- Negligible energy and area overhead

The mathematical platform, developed in this study, helped understand and adjust the parameters and factors which have the highest and lowest sensitivity in the output of the SULP FBB circuit. It was also essential in analysing and predicting, with a high accuracy, what effects the proposed technique had on an inverter.

In addition, considerable improvements were achieved in the EDP of the simulated Kogge-Stone adder and the FFT processors. Negligible energy and area overhead of the proposed technique were also confirmed.

In addition, seven times delay variation reduction and subthreshold error probability reduction of 50% to 1% led to massive production yield boost for vulnerable and PVT variation prone ultra low power circuits under test. Results proved the proposed FBB effectiveness for subthreshold voltages despite the decline in impact of FBB in short-channel devices.

Lastly, the comprehensive literature review of recent sub/nearthreshold techniques helped understand all the advantages and disadvantages of the available techniques, and the

standard cell library, designed using the latest design flow standards, was an absolute requirement to obtain the necessary accuracy in simulations needed for studying the results.

7.3 Future Directions

It was mentioned in several instances, previously, that the SULP FBB can be changed and/or tuned in many ways to satisfy different requirements in various systems. For example, applying certain voltages to the SULP FBB buffers, which are different from system's supply voltage, can add the reverse body bias benefits to the body bias generator too, which can be attractive for memories or standby modes where leakage avoidance is essential. It was also mentioned that, SULP FBB can be tuned to change behaviour to meet the minimum temperature yield instead of timing yield.

However, this technique can be taken to the next level by, for example, applying it to an error detection pipeline system. This gives DVS even more opportunity to reduce energy by aggressively scaling the supply voltage and at the same time enjoying the delay variation reduction due to SULP FBB presence.

As a result, a potential novel scenario is proposed in this section but a discussion about time borrowing is necessary before proceeding (readers are referred to [49] and [124] for a complete understanding of error detection techniques in pipelines). It is worth mentioning that, under process variations, the soft edge flip-flop (SEFF) delay should be changed (presented in section 2.2.3.1.1, Pipelining). It means a technique should evaluate variation and apply different post silicon SEFF delays so that variations are compensated. A system which can calculate variations, as a result, should be integrated. Razor [125] is such a technique that has already been used. However, the Razor method might pick up the wrong frequency. For example, if an AND is followed by an ADD, as ADD is more prone to variations and tends to cause error, Razor detects this error and increases clock period. However, the subsequent instruction, which is an AND, does not need this frequency reduction.

One way to find out if a flip-flop (FF) has caught true data, as Razor does, is by comparing the data with delayed clock data. Another way, however, is calculating if an incoming data has violated the setup and hold times of FF and, based on that, latching erroneous data, as done in [126]. This idea is actually similar to Razor II [127] and has the same problem as Razor. Razor also needs a minimum short path delay because when a

clock is triggered, shadow latch at Razor will be waiting for a late coming signal, by means of a delayed clock, but at same time the short path results may change the shadow latch data, before critical path of previous clock discloses its data. There is a probability that metastability propagates through the error detection logic and causes metastability of the restore signal itself, which has been addressed in later versions by adding more circuitry like [128]. In [124] (which is an advanced form of [126]) a new FF is proposed which can handle both short and critical path errors and, in addition, the FF can recover critical path errors, like Razor, and also can predict short path errors. But this technique requires a large area and is suitable for superthreshold voltages and it still has the same problem as discussed before.

Another problem of these in situ monitors is their activity, area and energy overhead. The authors in [129], use the high clock phase as the error-detection window for the short path problem, where min-delay paths must not arrive before the falling clock edge. Latch transparency feature was also taken advantage of and by the above assumption, the extra master latch of FF was eliminated and energy was reduced and metastability was tackled using transparency.

Using instruction isolation and due to instruction specific delays, however, the rate of violation resulting from different instructions delays will be cancelled and just those violations caused by temperature will emerge. The authors in [54] present a comprehensive research by putting all these ideas together along with using error correction (the details of the error correction scheme were not published however) for its pipeline and a look-up table (LUT) for keeping the delay of different instructions.

The requirement of different delays by different instructions means that a transparency window is needed whose size can be changed by different instructions. Employing SEFFs for this purpose is a unique technique that has never been used before. By designing this new FF, a transition detector can detect long path delays during the transparency window and therefore sets an error signal for tuning the size of the window. Furthermore, different instructions also lead to different delays being applied which as a result decrease the rate of errors.

Like the approach in [54], an instruction delay can be predicted, with an LUT, and be applied to SEFF and if timing is violated because of error detection, LUT entry should be updated with an increased delay. This LUT can be implemented in a ROM as it keeps

data permanently after it has been filled once. In another technique, if just process variations are of concern, error detection can be eliminated by just post silicon evaluations and in this way error detection circuitry can be clock gated or totally discarded by an off chip error detection scheme.

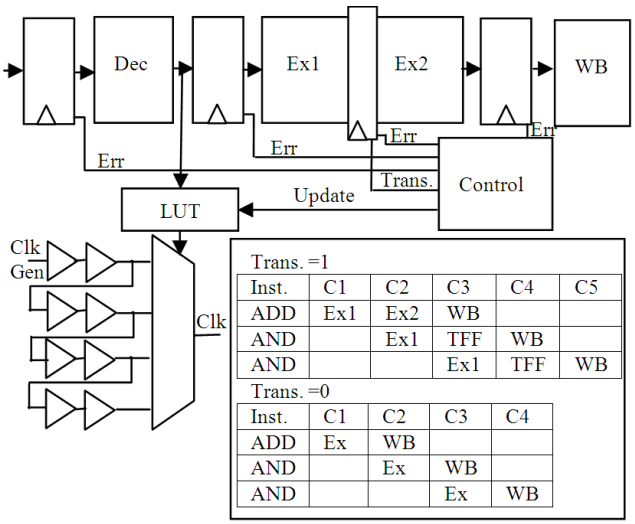


Figure 7.1. A sample architecture with a sample instruction flow. When Trans.=1, pipeline is working in LP mode, otherwise the high speed mode is applied.

Another opportunity for performance improvement can be created by employing instruction isolation using an extra transparent FF (TFF) in execution stage of pipeline, so that long delay instructions can use two stages and short delays use one stage (presented in Figure 7.1) and even more stages are applicable. Depending on the situation, these TFFs can be transparent or functional. This way it is not necessary to stall the pipeline for energy consuming instructions to be completed by two cycles (or more). A short instruction can also be completed quickly, and depending on other instructions on pipeline, they can go through the transparent FFs without clocking (which again saves energy). Having said that, the execution stage design should be changed so that long instructions are split into two (or more) parts in order to implement each part in one stage of pipeline with almost the same delay. Besides, for taking advantage of DVS, the FFs can be one of those with error correction and time borrowing features [129]. When the pipeline needs more speed and can handle higher voltages, the TFF becomes transparent and the pipeline can work with higher frequencies. Once the pipeline needs to consume less power and lower frequencies can be tolerated, by activating TFF, DVS helps the pipeline work with the lowest voltage possible. When the optimum voltage is applied, the error detection scheme helps the pipeline with DFS to handle PVT variations. LUT can also be used to keep a record of appropriate delays for different instructions.

Appendix A

A.1. Definitions

Some definitions are used for simplicity which are later used to abbreviate the subsequent long derivations:

$$f_{Dx\alpha}(V_{thx0}) = \frac{1}{2}\left(1 - Erf\left(\frac{\alpha\sigma_{VTHX}}{\sqrt{2}m_x v_T} + \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma_{VTHX}^2}}\right)\right) + \frac{e^{\frac{-\alpha\gamma V_{DD}}{m_x v_T}}}{2}\left(1 + Erf\left(\frac{\alpha\sigma_{VTHX}}{\sqrt{2}m_x v_T} + \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma_{VTHX}^2}}\right)\right)$$

(A.1)

$$f_{Ex}(V_{thx0}) = \frac{1}{2}\left(1 + Erf\left(\frac{\sigma_{VTHX}}{\sqrt{2}m_x v_T} - \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma_{VTHX}^2}}\right)\right) + \frac{e^{\frac{\gamma V_{DD}}{m_x v_T}}}{2}\left(1 - Erf\left(\frac{\sigma_{VTHX}}{\sqrt{2}m_x v_T} - \frac{\mu_{VTHX} - V_{thx0}}{\sqrt{2\,\sigma_{VTHX}^2}}\right)\right)$$

(A.2)

Note that $f_{Dx\alpha}$ and f_{Ex} are equal to 1 at superthreshold voltages.

A.2. Mean Value and Variance of Delay in a Typical Inverter:

Equation (4.14) can be rewritten as:

$$t_d = D_0 e^{\frac{V_{thp} - \gamma V_{BSP}}{m_p v_T}} \quad where\ D_0 = \frac{\frac{1}{2}\eta C_s V_{DD}}{I_{0p} e^{\frac{V_{DD}}{m_p v_T}}\left(1 - e^{-\frac{V_{DD}}{v_T}}\right)}$$

(A.3)

Using equations (A.3) and (4.3) and, as before, assuming a normal distribution for *VTHP*, the expected value of the random variable T_d can be sought as follows:

$$E(T_d) = E\left(D_0 e^{\frac{VTHP - \gamma VBSP}{m_p v_T}}\right)$$

(A.4)

According to equation (4.3), random variable *VBSP* is dependent on random variable *VTHP*. As a result equation (A.5) can be derived from equation (A.4).

$$E(T_d) = D_0 \int_{-\infty}^{\infty} e^{\frac{V_{thp}-\gamma V_{BSP}}{m_p v_T}} f_{VTHP}(V_{thp}) dV_{thp}$$

$$= \frac{D_0}{\sqrt{2\pi\sigma_{VTHP}^2}} \left(\int_{-\infty}^{V_{thp0}} e^{\frac{V_{thp}}{m_p v_T}} e^{-\frac{(V_{thp}-\mu_{VTHP})^2}{2\sigma_{VTHP}^2}} dV_{thp} + e^{\frac{-\gamma V_{DD}}{m_p v_T}} \int_{V_{thp0}}^{\infty} e^{\frac{V_{thp}}{m_p v_T}} e^{-\frac{(V_{thp}-\mu_{VTHP})^2}{2\sigma_{VTHP}^2}} dV_{thp} \right)$$

(A.5)

After a few derivations and substitution defined by (A.1), equation (A.6) is formed.

$$E(T_d) = D_0 e^{\frac{\mu_{VTHP}}{m_p v_T} + \frac{\sigma_{VTHP}^2}{2 m_p^2 v_T^2}} \left(\frac{1}{2} \left(1 - Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2} m_p v_T} + \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}} \right) \right) + \frac{e^{\frac{-\gamma V_{DD}}{m_p v_T}}}{2} \left(1 + Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2} m_p v_T} + \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}} \right) \right) \right)$$

$$= \mu_{T_{dZBB}} . f_{Dp1}(V_{thp0})$$

(A.6)

By means of equation (A.3) and with some mathematical derivations, PDF of T_d and T_{dZBB} are realised in equations (A.7) and (A.8), respectively.

$$f_{T_d}(t_d) = \left(\frac{m_p v_T}{\sqrt{2\pi\sigma_{VTHP}^2}\ t_d} \right) . \left(e^{-\frac{\left(\lambda V_{DD} + m_p v_T ln\left(\frac{t_d}{D_0}\right) - \mu_{VTHP}\right)^2}{2\sigma_{VTHP}^2}} p + e^{-\frac{\left(m_p v_T ln\left(\frac{t_d}{D_0}\right) - \mu_{VTHP}\right)^2}{2\sigma_{VTHP}^2}} (1-p) \right)$$

(A.7)

$$f_{T_{dZBB}}(t_{dZBB}) = \left(\frac{m_p v_T}{\sqrt{2\pi\sigma_{VTHP}^2}\ t_d} \right) e^{-\frac{\left(m_p v_T ln\left(\frac{t_d}{D_0}\right) - \mu_{VTHP}\right)^2}{2\sigma_{VTHP}^2}}$$

(A.8)

where as shown in equation (4.10) $p = \frac{1}{2} Erfc\left(\frac{V_{thp0} - \mu_{VTHP}}{\sqrt{2\sigma_{VTHP}^2}}\right)$.

Variance of delay can be derived using the following equations:

$$E(T_d^2) = D_0^2 \int_{-\infty}^{\infty} e^{2\frac{V_{thp} - \gamma V_{BSP}}{m_p v_T}} f_{VTHP}(V_{thp}) dV_{thp}$$

$$= \frac{D_0^2}{\sqrt{2\pi\sigma_{VTHP}^2}} \left(\int_{-\infty}^{V_{thp0}} e^{\frac{2V_{thp}}{m_p v_T}} e^{-\frac{(V_{thp} - \mu_{VTHP})^2}{2\sigma_{VTHP}^2}} dV_{thp} + e^{\frac{-2\gamma V_{DD}}{m_p v_T}} \int_{V_{thp0}}^{\infty} e^{\frac{2V_{thp}}{m_p v_T}} e^{-\frac{(V_{thp} - \mu_{VTHP})^2}{2\sigma_{VTHP}^2}} dV_{thp} \right)$$

(A.9)

Using equation (A.9) and substitutions defined by (A.1) and the fact that variance of a random variable, *X*, is calculated using equation $\sigma_X^2 = E(X^2) - E(X)^2 = \mu_{X^2} - \mu_X^2$, variance of delay is determined by equation (A.10).

$$E(T_d^2) = D_0^2 e^{2\left(\frac{\mu_{VTHP}}{m_p v_T} + \frac{\sigma_{VTHP}^2}{m_p^2 v_T^2}\right)} f_{Dp2}(V_{thp0}) = \mu_{T_{dZBB}^2} . f_{Dp2}(V_{thp0})$$

$$hence\ \sigma_{T_d}^2 = \mu_{T_{dZBB}^2} . f_{Dp2}(V_{thp0}) - \mu_{T_{dZBB}}^2 . f_{Dp1}^2(V_{thp0})$$

(A.10)

Variance of ZBB inverter's delay is simply calculated by $\sigma_{T_{dZBB}}^2 = \mu_{T_{dZBB}^2} - \mu_{T_{dZBB}}^2$.

A.3. Energy Delay Product in a Typical Inverter:

The following derivations find EDP of an inverter in the execution stage of the adder in which it is assumed that the probabilities of a gate input being 0 or 1 are equal to ½. Leakage energy is calculated using (4.16) and active energy using the well-known expression $\frac{1}{2}\alpha C_s V_{DD}^2$ and t_d using (4.14):

$$EDP_{SULP} = (E_{act} + E_{leak}).t_d = \left(\frac{1}{2}\alpha C_s V_{DD}^2 + V_{DD}\frac{1}{2}\left(I_{leak_p} + I_{leak_n}\right)t_{total}\right).\eta C_s V_{DD}\left(\frac{1}{I_{on_p}} + \frac{1}{I_{on_n}}\right)$$

$$= \left(\frac{1}{2}\alpha C_s\, V_{DD}^2 + V_{DD}\frac{1}{2}\left(I_{0p}e^{\frac{-V_{thp}+\gamma V_{BSP}}{m_p v_T}} + I_{0n}e^{\frac{-V_{thn}+\gamma V_{BSN}}{m_n v_T}}\right)t_{total}\right).\eta C_s V_{DD}.\left(\frac{1}{I_{0p}}e^{\frac{-V_{DD}+V_{thp}-\gamma V_{BSp}}{m_p v_T}} + \frac{1}{I_{0n}}\, e^{\frac{-V_{DD}+V_{thn}-\gamma V_{BSn}}{m_n v_T}}\right)$$

(A.11)

where t_{total} is the total time of the examination and, therefore, is constant. First active energy effect is taken into account by equation (A.12).

$$E_{act}DP = \left(\frac{\eta}{2}\alpha C_s^2\, V_{DD}^3\right)\left(\frac{1}{I_{0p}}e^{\frac{-V_{DD}+V_{thp}-\gamma V_{BSP}}{m_p v_T}} + \frac{1}{I_{0n}}\, e^{\frac{-V_{DD}+V_{thn}-\gamma V_{BSN}}{m_n v_T}}\right) \quad \text{(A.12)}$$

Now, by defining $E_{ACT}DP$ as the random variable representing $E_{act}DP$, equation (A.12) is used for calculating the mean value of $E_{ACT}DP$:

$$E(E_{ACT}DP) = \left(\frac{\eta}{2}\alpha C_s^2\, V_{DD}^3\right).\left(\frac{1}{I_{0p}}e^{\frac{-V_{DD}}{m_p v_T}}E\left(e^{\frac{VTHP-\gamma VBSP}{m_p v_T}}\right) + \frac{1}{I_{0n}}\, e^{\frac{-V_{DD}}{m_n v_T}}E\left(\frac{1}{I_{0n}}\, e^{\frac{VTHN-\gamma VBSN}{m_n v_T}}\right)\right)$$

$$= \left(\frac{\eta}{2}\alpha C_s^2\, V_{DD}^3\right).$$

$$\left(\begin{array}{c} \frac{1}{I_{0p}}\, e^{-\frac{V_{DD}}{m_p v_T}}e^{\frac{\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}}. \\ \left\{\frac{e^{\frac{-\gamma V_{DD}}{m_p v_T}}}{2}\left(1 + Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2}m_p v_T} + \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}}\right)\right) + \frac{1}{2}\left(1 - Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2}m_p v_T} + \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}}\right)\right)\right\} \\ + \\ \frac{1}{I_{0n}}\, e^{-\frac{V_{DD}}{m_n v_T}}e^{\frac{\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}}. \\ \left\{\frac{e^{\frac{-\gamma V_{DD}}{m_n v_T}}}{2}\left(1 + Erf\left(\frac{\sigma_{VTHN}}{\sqrt{2}m_n v_T} + \frac{\mu_{VTHN} - V_{thn0}}{\sqrt{2\,\sigma_{VTHN}^2}}\right)\right) + \frac{1}{2}\left(1 - Erf\left(\frac{\sigma_{VTHN}}{\sqrt{2}m_n v_T} + \frac{\mu_{VTHN} - V_{thn0}}{\sqrt{2\,\sigma_{VTHN}^2}}\right)\right)\right\} \end{array}\right)$$

$$= \left(\frac{\eta}{2}\alpha C_s^2\, V_{DD}^3\right).\left(\frac{1}{I_{0p}}e^{-\frac{V_{DD}}{m_p v_T}}e^{\frac{\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}}f_{Dp1}(V_{thp0}) + \frac{1}{I_{0n}}\, e^{-\frac{V_{DD}}{m_n v_T}}e^{\frac{\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}}f_{Dn1}(V_{thn0})\right)$$

(A.13)

Now using the same method, effect of SULP FBB on leakage energy is calculated:

$$E_{leak}DP =$$

$$\frac{1}{2}\eta C_s V_{DD}^2 \cdot \left(I_{0p} e^{\frac{-V_{thp}+\gamma V_{BSP}}{m_p v_T}} + I_{0n} e^{\frac{-V_{thn}+\gamma V_{BSN}}{m_n v_T}} \right) t_{total} \cdot \left(\frac{1}{I_{0p}} e^{\frac{-V_{DD}+V_{thp}-\gamma V_{BSP}}{m_p v_T}} + \frac{1}{I_{0n}} e^{\frac{-V_{DD}+V_{thn}-\gamma V_{BSN}}{m_n v_T}} \right)$$

$$= \frac{1}{2}\eta C_s V_{DD}^2 t_{total} \left(e^{\frac{-V_{DD}}{m_p v_T}} + e^{\frac{-V_{DD}}{m_n v_T}} + \frac{I_{0p}}{I_{0n}} e^{\frac{-V_{thp}+\gamma V_{BSP}}{m_p v_T} + \frac{-V_{DD}+V_{thn}-\gamma V_{BSN}}{m_n v_T}} + \frac{I_{0n}}{I_{0p}} e^{\frac{-V_{thn}+\gamma V_{BSN}}{m_n v_T} + \frac{-V_{DD}+V_{thp}-\gamma V_{BSP}}{m_p v_T}} \right)$$

(A.14)

As *VTHN* and *VTHP* are presumably independent random variables then it can be written:

$$E(E_{LEAK}DP) = \frac{1}{2}\eta C_s V_{DD}^2 t_{total} \left(e^{\frac{-V_{DD}}{m_p v_T}} + e^{\frac{-V_{DD}}{m_n v_T}} + \frac{I_{0p} e^{\frac{-V_{DD}}{m_n v_T}}}{I_{0n}} E\left(e^{\frac{-VTHP+\gamma VBSP}{m_p v_T}} \right) E\left(e^{\frac{VTHN-\gamma VBSN}{m_n v_T}} \right) + \frac{I_{0n} e^{\frac{-V_{DD}}{m_n v_T}}}{I_{0p}} E\left(e^{\frac{-VTHN+\gamma VBSN}{m_n v_T}} \right) E\left(e^{\frac{VTHP-\gamma VBSP}{m_p v_T}} \right) \right)$$

$$= \frac{1}{2}\eta C_s V_{DD}^2 t_{total} \cdot$$

$$\left(\begin{array}{c} e^{\frac{-V_{DD}}{m_p v_T}} + e^{\frac{-V_{DD}}{m_n v_T}} \\ + \frac{I_{0p} e^{\frac{-V_{DD}}{m_n v_T}}}{I_{0n}} e^{\frac{-\mu_{VTHP}}{m_p v_T} + \frac{\sigma_{VTHP}^2}{2 m_p^2 v_T^2}} e^{\frac{\mu_{VTHN}}{m_n v_T} + \frac{\sigma_{VTHN}^2}{2 m_n^2 v_T^2}} \cdot \\ \left\{ \frac{e^{\frac{\gamma V_{DD}}{m_p v_T}}}{2} \left(1 - Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2} m_p v_T} - \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}} \right) \right) + \frac{1}{2} \left(1 + Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2} m_p v_T} - \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}} \right) \right) \right\} \cdot \\ \left\{ \frac{1}{2} \left(1 - Erf\left(\frac{\sigma_{VTHN}}{\sqrt{2} m_n v_T} + \frac{\mu_{VTHN} - V_{thn0}}{\sqrt{2\,\sigma_{VTHN}^2}} \right) \right) + \frac{e^{\frac{-\gamma V_{DD}}{m_n v_T}}}{2} \left(1 + Erf\left(\frac{\sigma_{VTHN}}{\sqrt{2} m_n v_T} + \frac{\mu_{VTHN} - V_{thn0}}{\sqrt{2\,\sigma_{VTHN}^2}} \right) \right) \right\} \\ + \frac{I_{0n} e^{\frac{-V_{DD}}{m_n v_T}}}{I_{0p}} e^{\frac{-\mu_{VTHN}}{m_n v_T} + \frac{\sigma_{VTHN}^2}{2 m_n^2 v_T^2}} e^{\frac{\mu_{VTHP}}{m_p v_T} + \frac{\sigma_{VTHP}^2}{2 m_p^2 v_T^2}} \cdot \\ \left\{ \frac{e^{\frac{\gamma V_{DD}}{m_n v_T}}}{2} \left(1 - Erf\left(\frac{\sigma_{VTHN}}{\sqrt{2} m_n v_T} - \frac{\mu_{VTHN} - V_{thn0}}{\sqrt{2\,\sigma_{VTHN}^2}} \right) \right) + \frac{1}{2} \left(1 + Erf\left(\frac{\sigma_{VTHN}}{\sqrt{2} m_n v_T} - \frac{\mu_{VTHN} - V_{thn0}}{\sqrt{2\,\sigma_{VTHN}^2}} \right) \right) \right\} \cdot \\ \left\{ \frac{1}{2} \left(1 - Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2} m_p v_T} + \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}} \right) \right) + \frac{e^{\frac{-\gamma V_{DD}}{m_p v_T}}}{2} \left(1 + Erf\left(\frac{\sigma_{VTHP}}{\sqrt{2} m_p v_T} + \frac{\mu_{VTHP} - V_{thp0}}{\sqrt{2\,\sigma_{VTHP}^2}} \right) \right) \right\} \end{array} \right)$$

(A.15)

By means of definition (A.2), an abstract version of equation (A.15) can be presented by equation (A.16).

$$E(E_{LEAK}DP) = \frac{1}{2}\eta C_s V_{DD}^2 t_{total}.\begin{pmatrix} e^{\frac{-V_{DD}}{m_p v_T}} + e^{\frac{-V_{DD}}{m_n v_T}} \\ +f_{Ep}(V_{thp0})\, f_{Dn1}(V_{thn0}).\frac{I_{0p}}{I_{0n}} e^{\frac{-V_{DD}}{m_n v_T}} e^{\frac{-\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}+\frac{\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}} \\ +f_{En}(V_{thn0})\, f_{Dp1}(V_{thp0}).\frac{I_{0n}}{I_{0p}} e^{\frac{-V_{DD}}{m_n v_T}} e^{\frac{-\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}+\frac{\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}} \end{pmatrix} \quad \text{(A.16)}$$

Finally, by combining equations (A.13) and (A.16), overall impact of SULP FBB application on EDP in a typical inverter can be extracted in equation (A.17).

$$E(EDPSULP) = \frac{1}{2}\eta C_s V_{DD}^2 t_{total}.\begin{pmatrix} e^{\frac{-V_{DD}}{m_p v_T}} + e^{\frac{-V_{DD}}{m_n v_T}} \\ +f_{Ep}(V_{thp0})\, f_{Dn1}(V_{thn0}).\frac{I_{0p}}{I_{0n}} e^{\frac{-V_{DD}}{m_n v_T}} e^{\frac{-\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}+\frac{\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}} \\ +f_{En}(V_{thn0})\, f_{Dp1}(V_{thp0}).\frac{I_{0n}}{I_{0p}} e^{\frac{-V_{DD}}{m_n v_T}} e^{\frac{-\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}+\frac{\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}} \end{pmatrix} \quad \text{(A.17)}$$

$$+\left(\frac{\eta}{2}\alpha C_s^2\, V_{DD}^3\right).\left(\frac{1}{I_{0p}} e^{-\frac{V_{DD}}{m_p v_T}} e^{\frac{\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}} f_{Dp1}(V_{thp0}) + \frac{1}{I_{0n}}\, e^{-\frac{V_{DD}}{m_n v_T}} e^{\frac{\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}} f_{Dn1}(V_{thn0})\right)$$

It should be noticed that in equation (A.17), t_{total} is constant and it is not the time per instruction, or inverter's delay in here, in which case it would change when body bias changes.

EDP_{ZBB} can also be determined using equation (A.17) by equalling f_{Dp1} , f_{Dn1}, f_{Ep} and f_{En} to 1:

$$E(EDPZBB) = \frac{1}{2}\eta C_s V_{DD}^2 t_{total}.\begin{pmatrix} e^{\frac{-V_{DD}}{m_p v_T}} + e^{\frac{-V_{DD}}{m_n v_T}} \\ +\frac{I_{0p}}{I_{0n}} e^{\frac{-V_{DD}}{m_n v_T}} e^{\frac{-\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}+\frac{\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}} \\ +\frac{I_{0n}}{I_{0p}} e^{\frac{-V_{DD}}{m_n v_T}} e^{\frac{-\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}+\frac{\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}} \end{pmatrix} \quad \text{(A.18)}$$

$$+\left(\frac{\eta}{2}\alpha C_s^2\, V_{DD}^3\right).\left(\frac{1}{I_{0p}}e^{-\frac{V_{DD}}{m_p v_T}}e^{\frac{\mu_{VTHP}}{m_p v_T}+\frac{\sigma_{VTHP}^2}{2m_p^2 v_T^2}}+\frac{1}{I_{0n}}\,e^{-\frac{V_{DD}}{m_n v_T}}e^{\frac{\mu_{VTHN}}{m_n v_T}+\frac{\sigma_{VTHN}^2}{2m_n^2 v_T^2}}\right)$$

References

[1] L. Yu-Shiang, *et al.*, "Low-voltage circuit design for widespread sensing applications," in *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, 2008, pp. 2558-2561.

[2] D. B. Grayden and G. M. Clark, "Implant design and development," in *Cochlear Implants: A Practical Guide*, H. C. L. Craddock, Ed., ed London: Whurr Publishers Limited, 2006, pp. 1-20.

[3] Bionic Vision Australia [Online]. Available: http://www.bionicvision.org.au/

[4] J. Yick, *et al.*, "Wireless sensor network survey," *Computer Networks,* vol. 52, pp. 2292-2330, 2008.

[5] G. E. Moore, "Cramming more components onto integrated circuits," *Electronics,* vol. 8, pp. 114–117, 1965.

[6] I. Ferain, *et al.*, "Multigate transistors as the future of classical metal-oxide-semiconductor field-effect transistors," *Nature,* vol. 479, pp. 310-316, 2011.

[7] D. W. Dobberpuhl, *et al.*, "A 200-MHz 64-b dual-issue CMOS microprocessor," *Solid-State Circuits, IEEE Journal of,* vol. 27, pp. 1555-1567, 1992.

[8] R. Mahajan, *et al.*, "Emerging directions for packaging technologies," *Intel Technology Journal - Semiconductor Technology and Manufacturing,* vol. 06, pp. 62 – 75, 2002.

[9] Microprocessor Quick Reference Guide [Online]. Available: http://www.intel.com/pressroom/kits/quickreffam.htm

[10] S. Gunther, *et al.*, "Managing the Impact of Increasing Microprocessor Power Consumption," *Intel Technology Journal,* vol. Q1, pp. 1 -9, 2001.

[11] E. Pakbaznia and M. Pedram, "Minimizing data center cooling and server power costs," in *Proceedings of the 14th ACM/IEEE International Symposium on Low Power Electronics and Design*, San Fancisco, CA, USA, 2009, pp. 145-150.

[12] C.-H. Lin, *et al.*, "Energy analysis of multimedia video decoding on mobile handheld devices," *Computer Standards & Interfaces,* vol. 32, pp. 10-17, 2010.

[13] J. Lei, *et al.*, "Critical thermal issues in nanoscale IC design," in *2009 IEEE International Reliability Physics Symposium*, 2009, pp. 909-912.

[14] F. Schwierz, "Graphene transistors," *Nature nanotechnology,* vol. 5, pp. 487-496, 2010.

[15] L. K. Scheffer, "Physical CAD changes to incorporate design for lithography and manufacturability," in *Proceedings of the 2004 Asia and South Pacific Design Automation Conference*, Yokohama, Japan, 2004, pp. 768-773.

[16] K. A. Bowman, *et al.*, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *Solid-State Circuits, IEEE Journal of,* vol. 37, pp. 183-190, 2002.

[17] T. Karnik, *et al.*, "Sub-90nm technologies: challenges and opportunities for CAD," in *Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design*, San Jose, California, 2002, pp. 203-206.

[18] K. Kang, *et al.*, "On-chip variability sensor using phase-locked loop for detecting and correcting parametric timing failures," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 18, pp. 270-280, 2010.

[19] O. S. Unsal, *et al.*, "Impact of Parameter Variations on Circuits and Microarchitecture," *Micro, IEEE,* vol. 26, pp. 30-39, 2006.

[20] M. Junxia, *et al.*, "Layout-Aware Pattern Generation for Maximizing Supply Noise Effects on Critical Paths," in *VLSI Test Symposium, 2009. VTS '09. 27th IEEE*, 2009, pp. 221-226.

[21] W. N. HE, *CMOS VLSI design: a circuits and systems perspective*: Pearson Education India, 2006.
[22] W.-K. Chen, *The VLSI handbook*: CRC press, 2007.
[23] M. Alioto, "Understanding DC Behavior of Subthreshold CMOS Logic Through Closed-Form Analysis," *Circuits and Systems I: Regular Papers, IEEE Transactions on,* vol. 57, pp. 1597-1607, 2010.
[24] N. H. E. Weste and D. M. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*. Boston: Addison-Wesley, 2011.
[25] W. Liu*, et al.*, "BSIM4. 6.4 MOSFET Model."
[26] T. Grotjohn and B. Hoefflinger, "A parametric short-channel MOS transistor model for subthreshold and strong inversion current," *Solid-State Circuits, IEEE Journal of,* vol. 19, pp. 100-112, 1984.
[27] K. Tae-Hyoung*, et al.*, "Utilizing Reverse Short-Channel Effect for Optimal Subthreshold Circuit Design," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 15, pp. 821-829, 2007.
[28] E. Vittoz and J. Fellrath, "CMOS analog integrated circuits based on weak inversion operations," *Solid-State Circuits, IEEE Journal of,* vol. 12, pp. 224-231, 1977.
[29] S. Hanson*, et al.*, "Ultralow-voltage, minimum-energy CMOS," *IBM Journal of Research and Development,* vol. 50, pp. 469-490, 2006.
[30] M. Radfar*, et al.*, "Recent Subthreshold Design Techniques," *Active and Passive Electronic Components,* vol. 2012, p. 11, 2012.
[31] K. Bernstein*, et al.*, "High-performance CMOS variability in the 65-nm regime and beyond," *IBM J. Res. Dev.,* vol. 50, pp. 433-449, 2006.
[32] N. Drego*, et al.*, "A Test-Structure to Efficiently Study Threshold-Voltage Variation in Large MOSFET Arrays," in *8th International Symposium on Quality Electronic Design* 2007, pp. 281-286.
[33] B. Zhai*, et al.*, "Analysis and mitigation of variability in subthreshold design," presented at the Proceedings of the 2005 international symposium on Low power electronics and design, San Diego, CA, USA, 2005.
[34] M. Orshansky and K. Keutzer, "A general probabilistic framework for worst case timing analysis," in *Proceedings of the 39th annual Design Automation Conference*, New Orleans, Louisiana, USA, 2002, pp. 556-561.
[35] N. Metropolis and S. Ulam, "The Monte Carlo method," *Journal of the American Statistical Association,* vol. 44, pp. 335-341, 1949.
[36] D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers, (with CD)*. New York: John Wiley & Sons, 2007.
[37] A. Srivastava*, et al.*, "Modeling and analysis of leakage power considering within-die process variations," in *Proceedings of the 2002 International Symposium on Low Power Electronics and Design*, Monterey, California, USA, 2002, pp. 64-67.
[38] R. Rao*, et al.*, "Statistical analysis of subthreshold leakage current for VLSI circuits," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 12, pp. 131-139, 2004.
[39] D. Sylvester*, et al.*, "Variability in nanometer CMOS: Impact, analysis, and minimization," *Integration, the VLSI Journal,* vol. 41, pp. 319-339, 2008.
[40] C. Neau and K. Roy, "Optimal body bias selection for leakage improvement and process compensation over different technology generations," in *Proceedings of the 2003 International Symposium on Low Power Electronics and Design*, Seoul, Korea, 2003, pp. 116-121.
[41] J. T. Kao*, et al.*, "A 175-MV multiply-accumulate unit using an adaptive supply voltage and body bias architecture," *Solid-State Circuits, IEEE Journal of,* vol. 37, pp. 1545-1554, 2002.

[42] B. Zhai, *et al.*, "Theoretical and practical limits of dynamic voltage scaling," presented at the Proceedings of the 41st annual Design Automation Conference, San Diego, CA, USA, 2004.

[43] P. Macken, *et al.*, "A voltage reduction technique for digital systems," in *Solid-State Circuits Conference, 1990. Digest of Technical Papers. 37th ISSCC., 1990 IEEE International*, 1990, pp. 238-239.

[44] B. Zhai, *et al.*, "Energy efficient near-threshold chip multi-processing," in *Proceedings of the 2007 international symposium on Low power electronics and design*, Portland, OR, USA, 2007, pp. 32-37.

[45] J. Kwong and A. P. Chandrakasan, "Advances in Ultra-Low-Voltage Design," *Solid-State Circuits Newsletter, IEEE,* vol. 13, pp. 20-27, 2008.

[46] Y. K. Ramadass and A. P. Chandrakasan, "Voltage Scalable Switched Capacitor DC-DC Converter for Ultra-Low-Power On-Chip Applications," in *Power Electronics Specialists Conference, 2007. PESC 2007. IEEE*, 2007, pp. 2353-2359.

[47] Y. Ramadass, *et al.*, "A 0.16mm^2 completely on-chip switched-capacitor DC-DC converter using digital capacitance modulation for LDO replacement in 45nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, 2010, pp. 208-209.

[48] S. K. Gupta, *et al.*, "Digital Computation in Subthreshold Region for Ultralow-Power Operation: A Device Circuit Architecture Codesign Perspective," *Proceedings of the IEEE,* vol. 98, pp. 160-190, 2010.

[49] R. G. Dreslinski, *et al.*, "Near-Threshold Computing: Reclaiming Moore's Law Through Energy Efficient Integrated Circuits," *Proceedings of the IEEE,* vol. 98, pp. 253-266, 2010.

[50] M. R. Kakoee, *et al.*, "Automatic synthesis of near-threshold circuits with fine-grained performance tunability," in *Proceedings of the 16th ACM/IEEE International Symposium on Low Power Electronics and Design*, Austin, Texas, USA, 2010, pp. 401-406.

[51] Z. Bo, *et al.*, "A 2.60pJ/Inst Subthreshold Sensor Processor for Optimal Energy Efficiency," in *VLSI Circuits, 2006. Digest of Technical Papers. 2006 Symposium on*, 2006, pp. 154-155.

[52] Z. Bo, *et al.*, "Energy-Efficient Subthreshold Processor Design," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 17, pp. 1127-1137, 2009.

[53] B. H. Calhoun and A. Chandrakasan, "Characterizing and Modeling Minimum Energy Operation for Subthreshold Circuits," in *Low Power Electronics and Design, 2004. ISLPED '04. Proceedings of the 2004 International Symposium on*, 2004, pp. 90-95.

[54] L. Seung Eun, *et al.*, "Low power adaptive pipeline based on instruction isolation," in *Quality of Electronic Design*, 2009, pp. 788-793.

[55] S. Ghosh, *et al.*, "O2C: occasional two-cycle operations for dynamic thermal management in high performance in-order microprocessors," in *Proceeding of the 13th International Symposium on Low Power Electronics and Design*, Bangalore, India, 2008, pp. 189-192.

[56] S. Ghosh and K. Roy, "Parameter Variation Tolerance and Error Resiliency: New Design Paradigm for the Nanoscale Era," *Proceedings of the IEEE,* vol. 98, pp. 1718-1751, 2010.

[57] S. Ghosh, *et al.*, "Voltage Scalable High-Speed Robust Hybrid Arithmetic Units Using Adaptive Clocking," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 18, pp. 1301-1309, 2010.

[58] N. Toosizadeh, *et al.*, "VariPipe: Low-overhead variable-clock synchronous pipelines," in *Computer Design, 2009. ICCD 2009. IEEE International Conference on*, 2009, pp. 117-124.

[59] M. Ghasemazar, *et al.*, "A mathematical solution to power optimal pipeline design by utilizing soft edge flip-flops," in *Proceeding of the 13th International Symposium on Low Power Electronics and Design*, Bangalore, India, 2008, pp. 33-38.

[60] L. Xiaoyao, *et al.*, "ReVIVaL: A Variation-Tolerant Architecture Using Voltage Interpolation and Variable Latency," in *Computer Architecture, 2008. ISCA '08. 35th International Symposium on*, 2008, pp. 191-202.

[61] N. Toosizadeh, *et al.*, "Using variable clocking to reduce leakage in synchronous circuits," in *Computer Design (ICCD), 2010 IEEE International Conference on*, 2010, pp. 328-335.

[62] X. Liang, *et al.*, "Process variation tolerant circuit with voltage interpolation and variable latency," ed: Google Patents, 2010.

[63] M. Jeong, *et al.*, "Moebius circuit: dual-rail dynamic logic for logic gate level pipeline with error gate search feature," in *Proceedings of the 19th ACM Great Lakes symposium on VLSI*, Boston Area, MA, USA, 2009, pp. 177-180.

[64] O. Azizi, *et al.*, "Energy-performance tradeoffs in processor architecture and circuit design: a marginal cost analysis," in *Proceedings of the 37th Annual International Symposium on Computer Architecture*, Saint-Malo, France, 2010, pp. 26-36.

[65] S. Mingoo, *et al.*, "Pipeline strategy for improving optimal energy efficiency in ultra-low voltage design," in *Design Automation Conference (DAC), 2011 48th ACM/EDAC/IEEE*, 2011, pp. 990-995.

[66] S. Mingoo, *et al.*, "Extending energy-saving voltage scaling in ultra low voltage integrated circuit designs," in *IC Design & Technology (ICICDT), 2012 IEEE International Conference on*, 2012, pp. 1-4.

[67] S. Mingoo, *et al.*, "A 0.27V 30MHz 17.7nJ/transform 1024-pt complex FFT core with super-pipelining," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, 2011, pp. 342-344.

[68] J. T. Kao, *et al.*, "A 175-mV multiply-accumulate unit using an adaptive supply voltage and body bias architecture," *Solid-State Circuits, IEEE Journal of,* vol. 37, pp. 1545-1554, 2002.

[69] S. Hanson, *et al.*, "A Low-Voltage Processor for Sensing Applications with Picowatt Standby Mode," *Solid-State Circuits, IEEE Journal of,* vol. 44, pp. 1145-1155, 2009.

[70] L. Yoonmyung, *et al.*, "Ultra-low power circuit techniques for a new class of sub-mm^3 sensor nodes," in *Custom Integrated Circuits Conference (CICC), 2010 IEEE*, 2010, pp. 1-8.

[71] M. Ashouei, *et al.*, "A voltage-scalable biomedical signal processor running ECG using 13pJ/cycle at 1MHz and 0.4 V," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, 2011, pp. 332-334.

[72] J. Hulzink, *et al.*, "An Ultra Low Energy Biomedical Signal Processing System Operating at Near-Threshold," *Biomedical Circuits and Systems, IEEE Transactions on,* vol. 5, pp. 546-554, 2011.

[73] M. E. Sinangil, *et al.*, "A Reconfigurable 8T Ultra-Dynamic Voltage Scalable (U-DVS) SRAM in 65 nm CMOS," *Solid-State Circuits, IEEE Journal of,* vol. 44, pp. 3163-3173, 2009.

[74] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8T Subthreshold SRAM Employing Sense-Amplifier Redundancy," *Solid-State Circuits, IEEE Journal of,* vol. 43, pp. 141-149, 2008.

[75] Z. Bo, *et al.*, "A Variation-Tolerant Sub-200 mV 6-T Subthreshold SRAM," *Solid-State Circuits, IEEE Journal of,* vol. 43, pp. 2338-2348, 2008.

[76] M. Sharifkhani and M. Sachdev, "An Energy Efficient 40 Kb SRAM Module With Extended Read/Write Noise Margin in 0.13 um CMOS," *Solid-State Circuits, IEEE Journal of,* vol. 44, pp. 620-630, 2009.

[77] C. Ik Joon*, et al.*, "A 32 kb 10T Sub-Threshold SRAM Array With Bit-Interleaving and Differential Read Scheme in 90 nm CMOS," *Solid-State Circuits, IEEE Journal of,* vol. 44, pp. 650-658, 2009.

[78] K. Roy*, et al.*, "Process-Tolerant Ultralow Voltage Digital Subthreshold Design," in *Silicon Monolithic Integrated Circuits in RF Systems, 2008. SiRF 2008. IEEE Topical Meeting on*, 2008, pp. 42-45.

[79] Z. Bo*, et al.*, "A Sub-200mV 6T SRAM in 0.13um CMOS," in *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, 2007, pp. 332-606.

[80] J. P. Kulkarni*, et al.*, "A 160 mV, fully differential, robust schmitt trigger based sub-threshold SRAM," in *Low Power Electronics and Design (ISLPED), 2007 ACM/IEEE International Symposium on*, 2007, pp. 171-176.

[81] K. Jinmo*, et al.*, "Heterogeneous SRAM Cell Sizing for Low-Power H.264 Applications," *Circuits and Systems I: Regular Papers, IEEE Transactions on,* vol. 59, pp. 2275-2284, 2012.

[82] Y. He*, et al.*, "Xetal-Pro: an ultra-low energy and high throughput SIMD processor," in *Proceedings of the 47th Design Automation Conference*, Anaheim, California, 2010, pp. 543-548.

[83] L. Waeijen*, et al.*, "SIMD made explicit," in *Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS XIII), 2013 International Conference on*, 2013, pp. 330-337.

[84] S. Seo*, et al.*, "Diet SODA: A power-efficient processor for digital cameras," in *Low-Power Electronics and Design (ISLPED), 2010 ACM/IEEE International Symposium on*, 2010, pp. 79-84.

[85] S. Seo*, et al.*, "Process variation in near-threshold wide SIMD architectures," presented at the Proceedings of the 49th Annual Design Automation Conference, San Francisco, California, 2012.

[86] J. Kwong and A. P. Chandrakasan, "Variation-Driven Device Sizing for Minimum Energy Sub-threshold Circuits," in *Low Power Electronics and Design, 2006. ISLPED'06. Proceedings of the 2006 International Symposium on*, 2006, pp. 8-13.

[87] D. Bol*, et al.*, "Robustness-aware sleep transistor engineering for power-gated nanometer subthreshold circuits," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, 2010, pp. 1484-1487.

[88] D. Bol*, et al.*, "Interests and Limitations of Technology Scaling for Subthreshold Logic," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 17, pp. 1508-1519, 2009.

[89] F. Moradi*, et al.*, "New subthreshold concepts in 65nm CMOS technology," in *Quality of Electronic Design*, 2009, pp. 162-166.

[90] D. Bol*, et al.*, "Technology flavor selection and adaptive techniques for timing-constrained 45nm subthreshold circuits," in *Proceedings of the 14th ACM/IEEE international symposium on Low power electronics and design*, San Fancisco, CA, USA, 2009, pp. 21-26.

[91] K. Nose and T. Sakurai, "Optimization of VDD and VTH for low-power and high speed applications," presented at the Proceedings of the 2000 Asia and South Pacific Design Automation Conference, Yokohama, Japan, 2000.

[92] L. Yan*, et al.*, "Joint dynamic voltage scaling and adaptive body biasing for heterogeneous distributed real-time embedded systems," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on,* vol. 24, pp. 1030-1041, 2005.

[93] S. Narendra, *et al.*, "Forward body bias for microprocessors in 130-nm technology generation and beyond," *Solid-State Circuits, IEEE Journal of,* vol. 38, pp. 696-701, 2003.
[94] M. Miyazaki, *et al.*, "A 1.2-GIPS/W microprocessor using speed-adaptive threshold-voltage CMOS with forward bias," *Solid-State Circuits, IEEE Journal of,* vol. 37, pp. 210-217, 2002.
[95] S. Hanson, *et al.*, "Exploring variability and performance in a sub-200-mV processor," *Solid-State Circuits, IEEE Journal of,* vol. 43, pp. 881-891, 2008.
[96] J. Tschanz, *et al.*, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," in *Solid-State Circuits Conference, 2002. Digest of Technical Papers. ISSCC. 2002 IEEE International*, 2002, pp. 422-478 vol.1.
[97] J. M. Rabaey, *Low Power Design Essentials*. Upper Saddle River, NJ: Prentice-Hall, 2009.
[98] Y. Pu, *et al.*, "An Ultra-Low-Energy Multi-Standard JPEG Co-Processor in 65 nm CMOS With Sub/Near Threshold Supply Voltage," *Solid-State Circuits, IEEE Journal of,* vol. 45, pp. 668-680, 2010.
[99] H. Myeong-Eun and K. Roy, "ABRM: Adaptive Beta-Ratio Modulation for Process-Tolerant Ultradynamic Voltage Scaling," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 18, pp. 281-290, 2010.
[100] S. Hanson, *et al.*, "Performance and Variability Optimization Strategies in a Sub-200mV, 3.5pJ/inst, 11nW Subthreshold Processor," in *VLSI Circuits, 2007 IEEE Symposium on*, 2007, pp. 152-153.
[101] P. Yu, *et al.*, "An Ultra-Low-Energy Multi-Standard JPEG Co-Processor in 65 nm CMOS With Sub/Near Threshold Supply Voltage," *Solid-State Circuits, IEEE Journal of,* vol. 45, pp. 668-680, 2010.
[102] S. Hanson, *et al.*, "Exploring Variability and Performance in a Sub-200-mV Processor," *Solid-State Circuits, IEEE Journal of,* vol. 43, pp. 881-891, 2008.
[103] Y. Osaki, *et al.*, "A wide input voltage range level shifter circuit for extremely low-voltage digital LSIs," *IEICE Electronics Express,* vol. 8, pp. 890-896, 2011.
[104] "IEEE Standard for Design and Verification of Low Power Integrated Circuits," *IEEE Std 1801-2009,* pp. 1-218, 2009.
[105] "IEEE Standard for Design and Verification of Low-Power Integrated Circuits," *IEEE Std 1801-2013 (Revision of IEEE Std 1801-2009),* pp. 1-348, 2013.
[106] J. Liu and F. P. Taraporevala, "Generating variation-aware library data with efficient device mismatch characterization," U.S. Patent No. 8,204,730, 19 Jun., 2012.
[107] M. Radfar, *et al.*, "A highly sensitive and ultra low-power forward body biasing circuit to overcome severe process, voltage and temperature variations and extreme voltage scaling," *Int. J. Circ. Theor. Appl.. doi: 10.1002/cta.1935,* 2013.
[108] M. SEOK, *et al.*, "REFERENCE VOLTAGE GENERATOR HAVING A TWO TRANSISTOR DESIGN," ed: WO Patent WO/2010/151,754, 2010.
[109] M. Seok, *et al.*, "A 0.5 v 2.2 pw 2-transistor voltage reference," in *Custom Integrated Circuits Conference, 2009. CICC'09. IEEE*, 2009, pp. 577-580.
[110] S. Mingoo, *et al.*, "A Portable 2-Transistor Picowatt Temperature-Compensated Voltage Reference Operating at 0.5 V," *Solid-State Circuits, IEEE Journal of,* vol. 47, pp. 2534-2545, 2012.
[111] S. Narendra, *et al.*, "Full-chip sub-threshold leakage power prediction model for sub-0.18µm CMOS," presented at the Proceedings of the 2002 international symposium on Low power electronics and design, Monterey, California, USA, 2002.

[112] J. E. Freund and G. A. Simon, *Modern elementary statistics* vol. 12: Prentice-Hall Englewood Cliffs, New Jersey, 1967.

[113] M. Meterelliyoz*, et al.*, "Characterization of Random Process Variations Using Ultralow-Power, High-Sensitivity, Bias-Free Sub-Threshold Process Sensor," *Circuits and Systems I: Regular Papers, IEEE Transactions on,* vol. 57, pp. 1838-1847, 2010.

[114] P. M. Kogge and H. S. Stone, "A parallel algorithm for the efficient solution of a general class of recurrence equations," *Computers, IEEE Transactions on,* vol. 100, pp. 786-793, 1973.

[115] A. Wang and A. Chandrakasan, "A 180-mV subthreshold FFT processor using a minimum energy design methodology," *Solid-State Circuits, IEEE Journal of,* vol. 40, pp. 310-319, 2005.

[116] C. Van Loan, *Computational frameworks for the Fast Fourier Transform* vol. 10. Philadel-phia, PA: Siam, 1992.

[117] P. A. Milder, "A mathematical approach for compiling and optimizing hardware implementations of DSP transforms," Electrical and Computer Engineering, Carnegie-Mellon, Pittsburgh, Pennsylvania, 2010.

[118] G. Nordin*, et al.*, "Automatic generation of customized discrete Fourier transform IPs," in *Proceedings of the 42nd annual Design Automation Conference*, 2005, pp. 471-474.

[119] J. H. Takala*, et al.*, "Multi-port interconnection networks for radix-r algorithms," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, 2001, pp. 1177-1180.

[120] P. A. Milder*, et al.*, "Fast and accurate resource estimation of automatically generated custom DFT IP cores," presented at the Proceedings of the 2006 ACM/SIGDA 14th international symposium on Field programmable gate arrays, Monterey, California, USA, 2006.

[121] M. Püschel*, et al.*, "Spiral: A generator for platform-adapted libraries of signal processing alogorithms," *International Journal of High Performance Computing Applications,* vol. 18, pp. 21-45, 2004.

[122] H. Kaul*, et al.*, "A 320 mV 56 μW 411 GOPS/Watt Ultra-Low Voltage Motion Estimation Accelerator in 65 nm CMOS," *Solid-State Circuits, IEEE Journal of,* vol. 44, pp. 107-114, 2009.

[123] M. Khellah*, et al.*, "A 256-Kb Dual-V_{CC} SRAM Building Block in 65-nm CMOS Process With Actively Clamped Sleep Transistor," *Solid-State Circuits, IEEE Journal of,* vol. 42, pp. 233-242, 2007.

[124] K. Hirose*, et al.*, "Delay-Compensation Flip-Flop with In-situ Error Monitoring for Low-Power and Timing-Error-Tolerant Circuit Design," *Japanese Journal of Applied Physics,* vol. 47, p. 2779, 2008.

[125] D. Ernst*, et al.*, "Razor: a low-power pipeline based on circuit-level timing speculation," in *Microarchitecture, 2003. MICRO-36. Proceedings. 36th Annual IEEE/ACM International Symposium on*, 2003, pp. 7-18.

[126] M. J. Turnquist and L. Koskinen, "Sub-threshold operation of a timing error detection latch," in *Research in Microelectronics and Electronics, 2009. PRIME 2009. Ph.D.*, 2009, pp. 124-127.

[127] D. Blaauw*, et al.*, "Razor II: In Situ Error Detection and Correction for PVT and SER Tolerance," in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, 2008, pp. 400-622.

[128] S. Das*, et al.*, "A self-tuning DVS processor using delay-error detection and correction," in *VLSI Circuits, 2005. Digest of Technical Papers. 2005 Symposium on*, 2005, pp. 258-261.

[129] K. A. Bowman*, et al.*, "Energy-Efficient and Metastability-Immune Timing-Error Detection and Instruction-Replay-Based Recovery Circuits for Dynamic-Variation Tolerance," in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, 2008, pp. 402-623.