

Novel Approaches to Low Leakage and Area Efficient VLSI Design

A Thesis Submitted

to

the Department of Electrical and Electronic Engineering
in Partial Fulfillment of the Requirements for the Degree of
Bachelor of Science in Electrical and Electronic Engineering

By

Tajrian Izma	Student Id: 09221088
Parag Barua	Student Id: 09221082
Md. Rejaur Rahman	Student Id: 09221157
Prianka Sengupta	Student Id: 09221092

Thesis Supervisor:

Professor Md. Shafiqul Islam, *PhD*



**Department of Electrical and Electronic Engineering
BRAC University, Dhaka
August 2011**

DECLARATION

We hereby declare that this thesis or any part of the thesis has not been submitted elsewhere for any degree or any award.

Tajrian Izma

Parag Barua

Md. Rejaur Rahman

Prianka Sengupta

ACKNOWLEDGEMENT

We would like to express our sincere gratitude and appreciation to everyone who made this thesis possible. Most of all, we would like to thank our supervisor, Professor Dr. Md. Shafiqul Islam for his guidance of our thesis and his patience during thesis study at BRAC University. With his knowledge and experience, he has guided us to successfully achieve our thesis objective.

Second, we would like to thank all the faculties of Electrical and Electronic Engineering Department of BRAC University for their support and valuable suggestions.

Third, we would like to thank our teacher Mehdi Zahid Sadi for his heartiest help and invaluable feedback for our thesis.

Last, but most importantly, we would like to specially thank to our classmates for their support and encouragements.

Table of Contents

DECLARATION.....	i
ACKNOWLEDGEMENT.....	ii
Table of Contents.....	iii
LIST OF TABLES.....	vi
LIST OF FIGURES.....	vi
ABSTRACT.....	viii
CHAPTER 1 INTRODUCTION.....	1
1.1 Problem statement.....	2
1.2 Motivation.....	3
1.3 Thesis organization.....	4
CHAPTER 2 DESIGN CRITERIA.....	5
2.1 Leakage power.....	5
2.2 Switching power and delay tradeoffs.....	8
2.3 SRAM cell leakage path.....	9
CHAPTER 3 PREVIOUS APPROACHES.....	10
3.1 Base Case.....	10
3.2 Sleep Transistor Technique.....	11
3.3 Forced Stack Technique.....	11
3.4 Sleepy Stack Approaches.....	12
3.5 Sleepy Keeper Approaches.....	14
3.6 Dual Sleep Approaches.....	15
CHAPTER 4 PROPOSED METHODS.....	17
4.1 Forced sleep method.....	17
4.1.1 Structure of Forced sleep method.....	17

4.1.2 Operation of Forced sleep method.....	18
4.2 Stacked Sleep Approach.....	19
4.2.1 Structure of Stacked Sleep Approach.....	19
4.2.2 Operation of Stacked Sleep Approach.....	20
4.3 Variable Body Biasing.....	20
4.3.1 Structure of Variable Body Biasing.....	21
4.3.2 Operation of Variable Body Biasing.....	22
CHAPTER 5 APPLICATION OF PROPOSED METHODS	23
5.1 Application in logic circuits.....	23
5.1.1 Forced sleep method using a chain of 4 inverters.....	23
5.1.2 Stacked sleep approach using a chain of 4 inverters.....	24
5.1.3 Variable body biasing technique using a chain of 4 inverters.....	25
5.2 Application in SRAM cell.....	26
5.2.1 Forced sleep application in SRAM cell.....	26
5.2.2 Stacked sleep application in SRAM cell.....	27
5.2.3 Variable body biasing application in SRAM cell.....	28
CHAPTER 6 SIMULATION RESULTS	29
6.1 Simulation results for logic circuits	30
6.1.1 Simulation results for a chain of four inverters	30
6.1.2 Simulation results for SRAM cell	35
CHAPTER 7 CONCLUSIONS AND SUGGESTIONS FOR FUTURE	
RESEARCH	40
7.1 Conclusion.....	40
7.2 Suggestions for future work.....	41
REFERENCES	42

APPENDICES

APPENDIX-A DESIGN LAYOUTS	A.1
A1: Layout of a chain of four inverters using forced sleep method	A.1
A2: Layout of an SRAM cell using forced sleep method	A.2
A3: Layout of a chain of four inverters using stacked sleep approach	A.3
A4: Layout of an SRAM cell using stacked sleep approach	A.4
A5: Layout of a chain of four inverters using variable body biasing Technique	A.5
A6: Layout of an SRAM cell using variable body biasing technique	A.6
APPENDIX-B SIMULATION DATA (a chain of four inverters)	B.1
B1: 130 nm technology	B.1
B2: 90 nm technology	B.1
B3: 45 nm technology	B.2
B4: 32 nm technology	B.2
B5: 22 nm technology	B.3
APPENDIX-C SIMULATION DATA (SRAM cell)	C.1
C1: 130 nm technology	C.1
C2: 90 nm technology	C.1
C3: 45 nm technology	C.2
C4: 32 nm technology	C.2
C5: 22 nm technology	C.3

Figure 6.1.1(d): Area of a chain of four inverters.....	34
Figure 6.1.2(a): Static power dissipation of an SRAM cell.....	36
Figure 6.1.2(b): Dynamic power dissipation of an SRAM cell.....	37
Figure 6.1.2(c): Propagation delay of an SRAM cell.....	38
Figure 6.1.2(d): Area of an SRAM cell	39

ABSTRACT

The development of digital integrated circuits is challenged by higher power consumption. The combination of higher clock speeds, greater functional integration, and smaller process geometries has contributed to significant growth in power density. Scaling improves transistor density and functionality on a chip. Scaling helps to increase speed and frequency of operation and hence higher performance. As voltages scale downward with the geometries threshold voltages must also decrease to gain the performance advantages of the new technology but leakage current increases exponentially. Thinner gate oxides have led to an increase in gate leakage current. Today leakage power has become an increasingly important issue in processor hardware and software design. With the main component of leakage, the sub-threshold current, exponentially increasing with decreasing device dimensions, leakage commands an ever increasing share in the processor power consumption. In 65 nm and below technologies, leakage accounts for 30-40% of processor power. According to the International Technology Roadmap for Semiconductors (ITRS) [1], leakage power dissipation may eventually dominate total power consumption as technology feature sizes shrink. While there are several process technology and circuit-level solutions to reduce leakage in processors, we propose novel approaches for reducing both leakage and dynamic power with minimum possible area and delay trade off.

CHAPTER 1

INTRODUCTION

For the most recent CMOS feature sizes (e.g., 90nm and 65nm), leakage power dissipation has become an overriding concern for VLSI circuit designers. International technology roadmap for semiconductors (ITRS) [1] reports that leakage power dissipation may come to dominate total power consumption. Power consumption of CMOS consists of dynamic and static components. Dynamic power is consumed when transistors are switching, and static power is consumed regardless of transistor switching. Dynamic power consumption was previously (at 0.18 μ technology and above) the single largest concern for low-power chip designers since dynamic power accounted for 90% or more of the total chip power. Therefore, many previously proposed techniques, such as voltage and frequency scaling, focused on dynamic power reduction. However, as the feature size shrinks, e.g., to 0.09 μ and 0.065 μ , static power has become a great challenge for current and future technologies. There are many reasons for which power losses occur in CMOS circuit. Figure 1 shows different types of leakage components. They are:

1. Sub-threshold leakage (weak inversion current)
2. Gate oxide leakage (Tunneling current)
3. Channel punch through
4. Drain induced barrier lowering

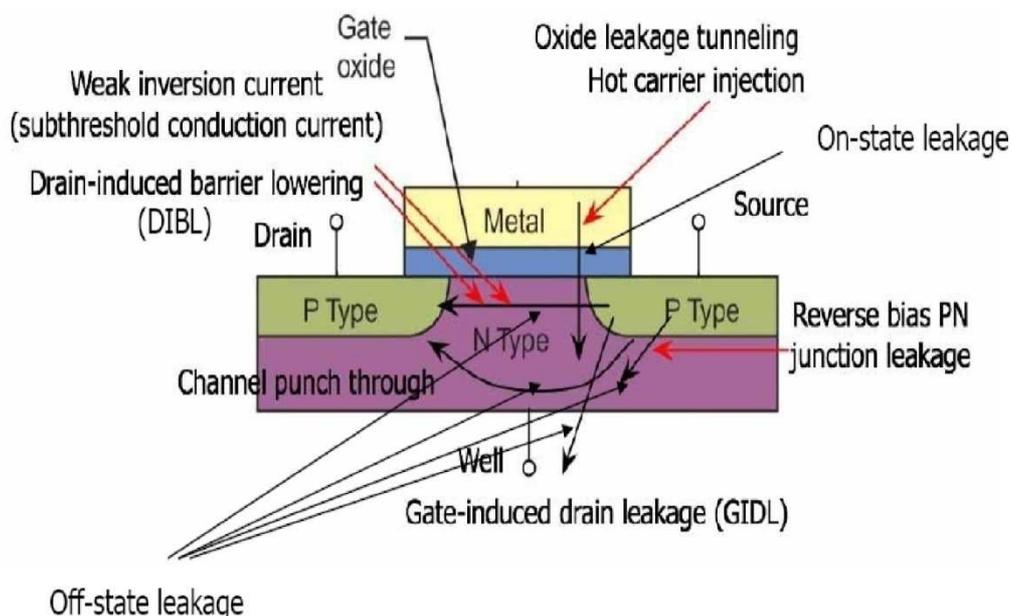


Figure 1: Leakage power components in CMOS

One of the main reasons causing the leakage power increase is increase of sub-threshold leakage power. The Sub-threshold conduction or the sub-threshold leakage or the sub-threshold drain current is the current that flows between the source and drain of a MOSFET when the transistor is in sub-threshold region, or weak-inversion region, that is, for gate-to-source voltages below the threshold voltage. The sub-threshold region is often referred to as the weak inversion region. When technology feature size scales down, supply voltage and threshold voltage also scale down. Sub-threshold leakage power increases exponentially as threshold voltage decreases which increases the sub-threshold leakage power. Next the gate oxide leakage, the gate oxide, which serves as insulator between the gate and channel, should be made as thin as possible to increase the channel conductivity and performance. But as the gate oxide is made thinner the barrier voltage of the oxide changes. For the positive gate voltage thus some positive charges get stuck in the oxide. Therefore, current flows through the oxide. This is also known as tunneling current.

Punch through in a MOSFET is an extreme case of channel length modulation where the depletion layers around the drain and source regions merge into a single depletion region. The field underneath the gate then becomes strongly dependent on the drain-source voltage, as is the drain current. Punch through causes a rapidly increasing current with increasing drain-source voltage. This effect is undesirable as it increases the output conductance and limits the maximum operating voltage of the device.

Drain induced barrier lowering or DIBL is referred to the reduction of threshold voltage of the transistor at higher drain voltages. The combined charge in the depletion region of the device and that in the channel of the device is balanced by three electrode charges: the gate, the source and the drain. As drain voltage is increased, the depletion region of the p-n junction between the drain and body increases in size and extends under the gate, so the drain assumes a greater portion of the burden of balancing depletion region charge, leaving a smaller burden for the gate. As a result, the charge present on the gate retains charge balance by attracting more carriers into the channel, an effect equivalent to lowering the threshold voltage of the device.

1.1 Problem statement

This research work is titled new low power approaches for VLSI logic and memory. While designing a VLSI system power dissipation is one of the major concerns. Up to a certain

time dynamic power was the single largest concern; however as the technology feature size shrinks static power has become an important issue as dynamic power. A well-known previous technique called the sleep transistor technique cuts off V_{dd} and/or Gnd connections of transistors to save leakage power consumption. However, when transistors are allowed to float, a system may have to wait a long time to reliably restore lost state and thus may experience seriously degraded performance. Therefore, retaining state is crucial for a system that requires fast response even while in an inactive state. The two common approaches are sleepy stack and sleepy keeper. Both methods are excellent in this regard. The static and dynamic power of sleepy stack is considerably low. But it has a delay penalty and its area requirement is maximum compared with other processes. Again the sleepy keeper process possesses excellent speed criteria but it requires more static and dynamic power than sleepy stack. Our goal is to trade off between these limitations and thus propose new methods which reduce both leakage and dynamic power with minimum possible area and delay trade off.

1.2 Motivation

CMOS Technology is one of the mainstreams of VLSI Design. In 0.18 μ m and above technology Dynamic power is one of the main factors of total power consumption. But when technology feature size shrinks static (Leakage) power dominates the dynamic power. So however, the designers proposed several methods to reduce the leakage. In Base Technique there is a no method for leakage reduction but it saves the state as well as minimum area and delay. Sleep Transistor Technique is most common method for achieving ultra-low leakage but it destroy the state and as well as increasing delay and area. Forced stack technique is another method and it can save the state. But in this technique Dynamic Power consumption is increases and it cannot use high V_{th} without incurring the Delay. By combing this two prior technique Sleepy Stack approaches is proposed. It reduces the leakage similarly like sleep transistor technique but the main advantage over sleep transistor technique is save the logic state. Moreover, Sleepy Stack approach comes with area and delay overhead and slower method than other technique. However, Sleepy Keeper approaches are really appreciable for propagation delay and static power performances. Although, Sleepy Keeper approaches incurs area and dynamic power dissipation. But still now this two common technique is used. Hence we sought a new method which can have excellent tradeoff between power, area, and delay.

1.3 Thesis organization

The thesis is organized into seven chapters.

CHAPTER 1: INTRODUCTION: This chapter contains power consumption issues in VLSI. This chapter also summarizes the contributions of this thesis. Finally, this chapter explains organization of the thesis.

CHAPTER 2: DESIGN CRITERIA: This chapter explains design criteria used throughout this disquisition.

CHAPTER 3: PREVIOUS PPROACHES: This chapter describes previous works in power reduction research and explains the main differences between our solutions and previous works.

CHAPTER 4: PROPOSED METHODS: This chapter includes the methods that we have proposed. The structures of the methods are described followed by detailed explanation of its operation.

CHAPTER 5: APPLYING PROPOSED METHODS. This chapter explores various applications of our proposed methods. The applications include generic logic circuits and memory circuits.

CHAPTER 6: SIMULATION RESULTS: This chapter shows different simulation results which was done during the whole thesis work.

CHAPTER 7: CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK: This chapter discusses about the results of the thesis in brief and gives some information which can be helpful in future research.

CHAPTER 2

DESIGN CRITERIA

For widely using of CMOS technology is basically for consuming less power. But as the technology feature size shrink sub-threshold leakage current is increases as the decrease of threshold voltage. In this design criterion it focuses on sub threshold leakage power consumption and it also focuses on body biasing effect and stack effect. Finally we explain the switching power and delay tradeoff of generic CMOS circuit.

2.1 Leakage Power

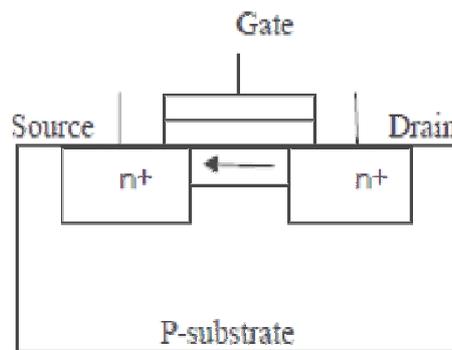


Figure 2.1(a): Sub-threshold Leakage of an NMOS

In 0.18 μ m and above technology dynamic power is the dominant factor but 0.13 μ m and below static power is another dominant factor for power consumption. One of the main contributors for the static power consumption is sub-threshold leakage current which is shown in the Figure 2.1(a) i.e., the drain to source current when the gate voltage is smaller than the threshold voltage. As the technology feature size shrink sub-threshold leakage current is increases exponentially as the decrease of threshold voltage.

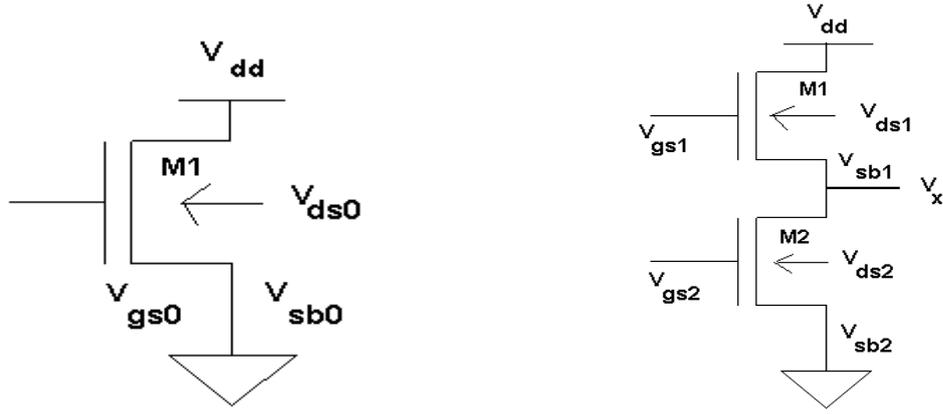


Figure 2.1(b): (i) Single Transistor

(ii) Stacked transistor

Stacking transistor can reduce sub-threshold leakage [2]. So it is called stacked effect. Where two or more stacked transistor is turned off together, the result can reduce the leakage power.

For turned off the single transistor in Figure 2.1(b) (i), leakage current I_{sub0} can be expressed as follows:

$$I_{sub0} = A e^{\frac{1}{nV_{\theta}}(V_{gs0} - V_{th0} - \gamma V_{sb0} + \eta V_{ds0})} \left(1 - e^{-\frac{V_{ds0}}{V_{\theta}}} \right) \quad (1)$$

$$= A e^{\frac{1}{nV_{\theta}}(-V_{th0} + \eta V_{dd})} \quad (2)$$

Where,

$$A = \mu_0 C_{ox} (W/L_{eff}) V_{\theta}^2 e^{1.8}$$

n =sub-threshold coefficient V_{θ} =thermal voltage

V_{gs0} , V_{th0} , V_{sb0} and V_{ds0} are the gate-to-source voltage, the zero-bias threshold voltage, the base -to-source voltage and the drain-to-source voltage respectively. γ is the body-bias effect coefficient, and η is the Drain Induced Barrier Lowering (DIBL) coefficient. μ is zero-bias mobility, C_{ox} is the gate-oxide capacitance, W is the width of the transistor, and L_{eff} is the effective channel length. (Note that throughout this project we assume $\mu_n = 2\mu_p$, i.e., NMOS carrier mobility is twice PMOS carrier mobility). Also note that we use a W/L

ratio based on a actual transistor size, in which way a W/L ratio properly characterizes circuit models used in this case.

In Figure 2.1 (b) (ii) two transistor are turned off together (M1=M2). So,

$$I_{sub1} = Ae^{\frac{1}{nV_{\theta}}(V_{gs1}-V_{th0}-\gamma V_{sb1}+\eta V_{ds1})} \left(1 - e^{-\frac{V_{ds1}}{V_{\theta}}}\right) \quad (3)$$

$$= Ae^{\frac{1}{nV_{\theta}}(-V_x-V_{th0}-\gamma V_x+\eta(V_{dd}-V_x))} \quad (4)$$

$$I_{sub2} = Ae^{\frac{1}{nV_{\theta}}(V_{gs2}-V_{th0}-\gamma V_{sb2}+\eta V_{ds2})} \left(1 - e^{-\frac{V_{ds2}}{V_{\theta}}}\right) \quad (5)$$

$$= Ae^{\frac{1}{nV_{\theta}}(-V_{th0}+\gamma V_x)} (1 - e^{-V_x/V_{\theta}}) \quad (6)$$

Where V_x is the voltage at the node between M1 and M2, and we assume

$$1 \gg e^{-\frac{V_{ds1}}{V}}$$

Now consider X is the factor of I_{sub0} and I_{sub1} ($=I_{sub2}$)

$$X = \frac{I_{sub0}}{I_{sub1}} = \frac{Ae^{\frac{1}{nV_{\theta}}(-V_{th0}+\eta V_{dd})}}{Ae^{\frac{1}{nV_{\theta}}(-V_x-V_{th0}-\gamma V_x+\eta(V_{dd}-V_x))}} = e^{\frac{V_x}{nV_{\theta}}(1+\gamma+\eta)} \quad (7)$$

If $I_{sub1}=I_{sub2}$ then from equation (7) we can write,

$$1 = e^{\frac{1}{nV_{\theta}}(\eta V_{dd}-V_x(1+2\eta+\gamma))} + e^{-V_x/V_{\theta}}$$

n coefficient will be increase as the technology feature size shrinks.

Threshold voltage can be controlled by body bias effect.

$$V_{th} = V_{t0} + \gamma(\sqrt{V_{sb}})$$

Changing the substrate voltage causes the threshold voltage to change. So the different kind of effect is arises for changing the substrate voltage like Zero-Body Bias, Reverse-Body Bias and Forward-Body Bias. This Phenomenon is frequently used for controlling the threshold voltage.

γ constant dependent on the transistor parameter and the technology feature size. By controlling body biasing effect with changing the constant term we can easily control the leakage power [3].

2.2 Switching power and delay tradeoffs

We explain tradeoffs between switching power and delay in this section. In CMOS, power consumption consists of leakage power and dynamic power Dynamic power includes both switching power and short circuit power. Switching power is consumed when the transistors are in active mode and short circuit power is consume when a pull-up and pull-down network are on turning on and off. For 0.18u and above leakage power is small compared to dynamic power but 0.13u and below leakage power is dominant. $P_{switching}$ equation is for 0.18u and above are given below,

$$P_{switching} = p_t C_L V_{dd}^2 f \quad (8)$$

If the V_{dd} is small, then the $P_{switching}$ is small but it increases the Time delay of the gate.

$$T_d \propto \frac{V_{dd}}{(V_{dd} - V_{th})^\alpha} \quad (9)$$

α is the velocity saturation index of a transistor and it is depend on the technology feature size. Key note is that lowering V_{dd} can reduce the switching power but it increases the delay time.

2.3 SRAM cell leakage path

In this section, we explain the major sub-threshold leakage components in a 6-T SRAM cell. The sub-threshold leakage current in an SRAM cell is typically categorized into two kinds [4] as shown in Figure 2.3, (i) cell leakage current that flows from V_{dd} to Gnd internal to the cell and (ii) bitline leakage current that flows from bitline (or bitline') to Gnd. Although an SRAM cell has two bitline (BL) leakage paths, the bitline leakage current and bitline' (BL') leakage current differs according to the value stored in the SRAM bit. If an SRAM cell holds '1' as shown in Figure 2.3, the bitline leakage current passing through N3 and N2 is effectively suppressed due to two reasons. First, after precharging bitline and bitline' both to '1,' the source voltage and the drain voltage of N3 are the same, and thus potentially no current flows through N3. Second, two stacked and turned off transistors (N2 and N3) induce the stack effect.

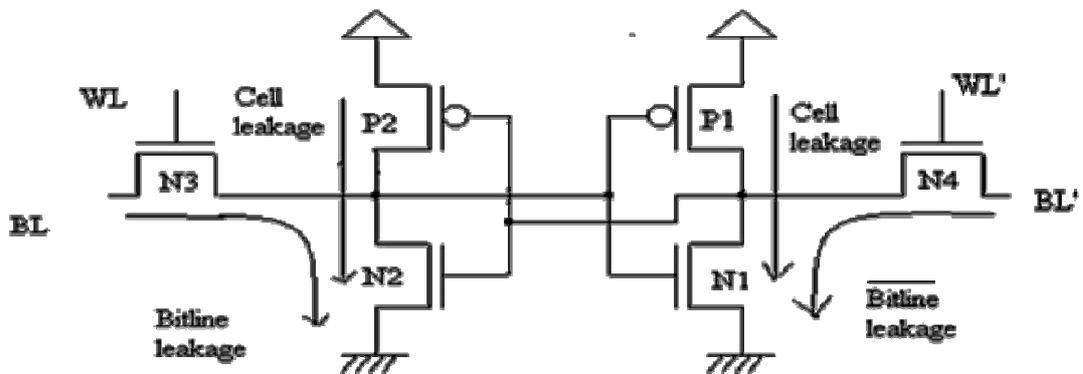


Figure 2.3: SRAM cell leakage paths

Meanwhile, for this case where the SRAM bit holds value '1', a large bitline' leakage current flows passing through N4 and N1. If, on the other hand, the SRAM cell holds '0,' a large bitline leakage current flows while bitline' leakage current is suppressed.

Meanwhile, for this case where the SRAM bit holds value '1', a large bitline' leakage current flows passing through N4 and N1. If, on the other hand, the SRAM cell holds '0,' a large bitline leakage current flows while bitline' leakage current is suppressed.

CHAPTER 3

PREVIOUS APPROACHES

In this section we analyze the previous approaches which are closely related to our research. Here we discuss previous low power technique that primarily target for reducing leakage. This technique for leakage reduction can be grouped into two categories:

(i) State saving (ii) state destructive [5]. State save can have an advantage over the state destructive.

The approaches that are adopted in VLSI design are:

3.1 Base Case

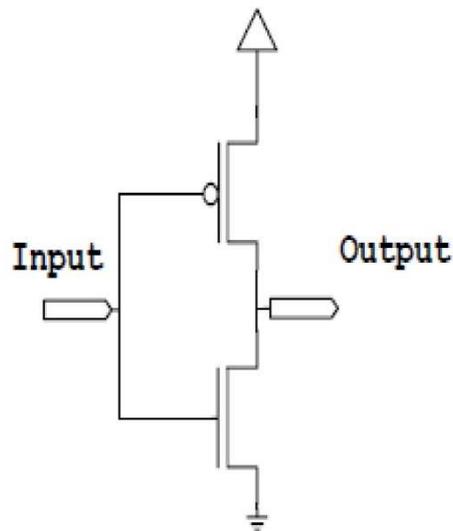


Figure 3.1: Base Case

The base case circuit contains only the PMOS network and the NMOS network and there exists no method to reduce leakage. A base case inverter is shown Figure 3.1. It is a state-saving technique and minimum area requirement.

3.2 Sleep Transistor Technique

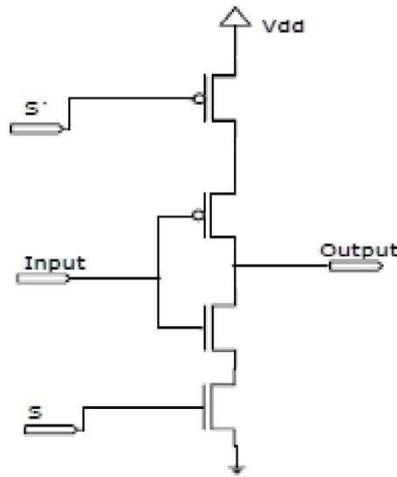


Figure 3.2: Sleep Transistor Technique

State -destructive techniques cut off transistor (pull-up or pull-down or both) networks from supply voltage or ground using sleep transistors. These types of techniques are also called gated V_{dd} and gated-GND (note that a gated clock is generally used for dynamic power reduction). Mutoh et al. propose a technique they call Multi-Threshold- Voltage CMOS (MTCMOS) [6], which adds high- V_{th} sleep transistors between pull-up networks and V_{dd} and between pull-down networks and ground as shown in Figure 3.2 while logic circuits use low- V_{th} transistors in order to maintain fast logic switching speeds. The sleep transistors are turned off when the logic circuits are not in use. By isolating the logic networks using sleep transistors, the sleep transistor technique dramatically reduces leakage power during sleep mode. However, the additional sleep transistors increase area and delay. Furthermore, the pull-up and pull-down networks will have floating values and thus will lose state during sleep mode. These floating values significantly impact the wakeup time and energy of the sleep technique due to the requirement to recharge transistors which lost state during sleep.

3.3 Forced stack

Another technique to reduce leakage power is to stack the transistors. Figure 3.3 shows a forced stack inverter. The effect of stacking the transistor results in the reduction of sub-threshold leakage current when two or more transistors are turned off together.

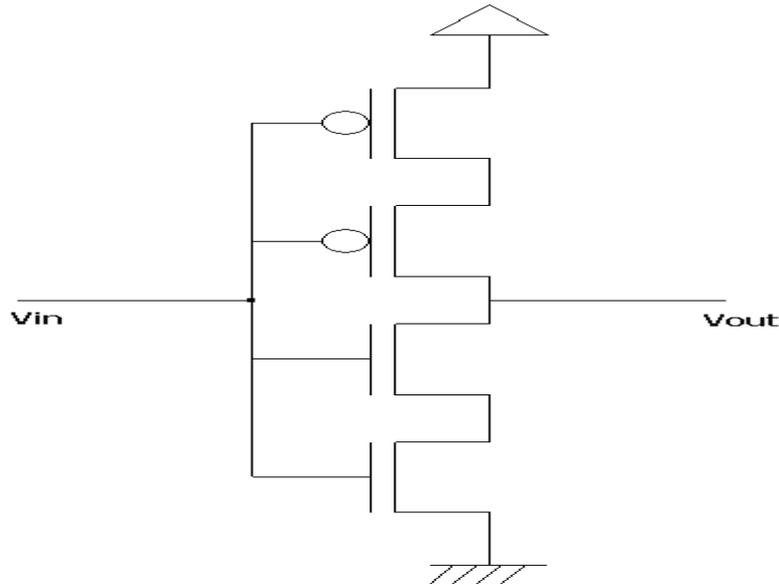


Figure 3.3: Forced stack inverter

The stacking effect [7] can be understood from the forced stack inverter shown in figure 3.3. In the generic inverter there are only two transistors. But here in case of forced stack inverter two pull up transistors and two pull down transistors are used. All inputs share the same input in the forced stack circuit. If the input is '0', then both transistor M1 and M2 are turned off. Here V_x is the intermediate node voltage. Transistor M2 has its internal resistance. Due to this resistance V_x is greater than the ground potential. This positive V_x results in a negative gate-source (V_{gs}) for the M1 transistor and the negative source-base voltage (V_{sb}) for M1. Here M1 also has a reduced drain-source voltage (V_{ds}), which lower the drain induced barrier lowering (DIBL) effect. These three effects together reduced the factor X in equation (7) and hence the leakage power. All transistors are getting the same input. So this forced stack technique is a state saving technique. That means when the circuit is in OFF mode it saves the current state. The main drawback of this forced stack inverter is that it can not use the high V_{th} transistor. Because if it use the high V_{th} transistor than there is a dramatic increase of delay. This delay increase is 5X larger than the conventional CMOS.

3.4 Sleepy stack approach

In the sleepy stack structure the forced stack and the sleep transistor techniques are combined together [8]. Hence the names sleepy stack. Figure 3.4 shows a sleepy stack inverter.

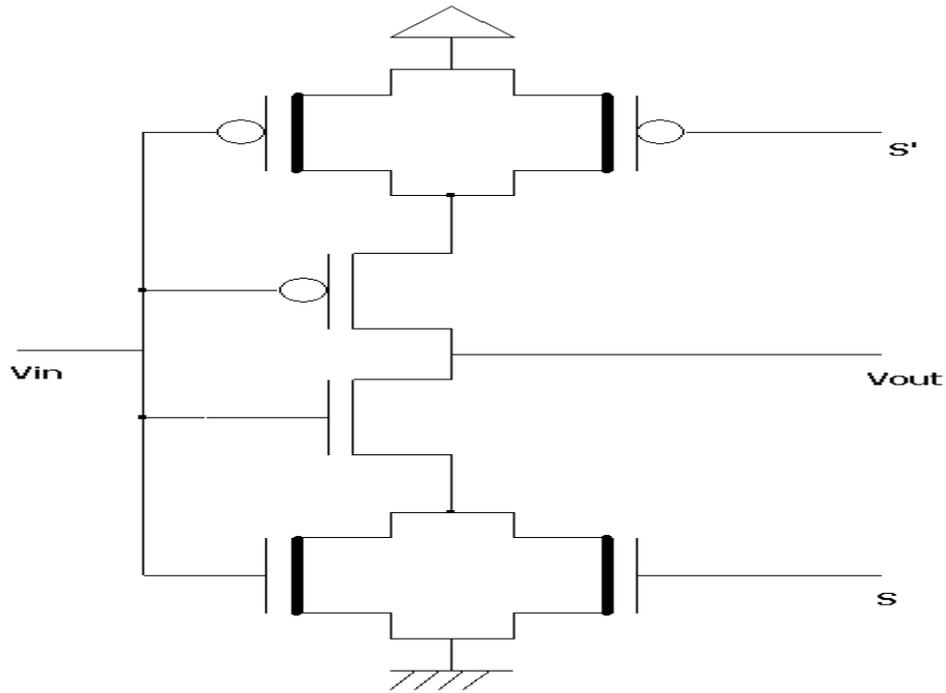


Figure 3.4: Sleepy stack inverter

The sleepy stack inverter in figure 3.4 uses the aspect ratio $W/L = 3$ for the pull up transistors and $W/L = 1.5$ for the pull down transistors. At the same time the conventional inverter with the same input capacitance uses the aspect ratio $W/L = 6$ for the pull up transistors and $W/L = 3$ for the pull down transistors. Here $\mu_n = 2\mu_p$ is assumed. The sleep transistor and the stacked transistor in each network are made parallel. Here the width of the sleep transistors is reduced. Changing the width of the sleep transistors may provide additional tradeoffs between delay, power and area. The activity of the sleep transistors in sleepy stack is same as the activity of the sleep transistors in the sleep transistor technique. The sleep transistors are turned on during active mode and turned off during sleep mode. The sleepy stack structure can reduce the circuit delay in two ways. First, since the sleep transistors are always on during active mode so there is always a current flow through the circuit. That's why it gives a faster switching time than the forced stack structure. The high V_{th} transistors are used for the sleep transistor and the transistors parallel to the sleep transistor without incurring large delay increase. The delay time is increasing here but it gives low leakage. During sleep mode both the sleep transistors are turned off. But the sleepy stack structure maintains exact logic state. As high V_{th} transistor is used here so the leakage power is suppressed. The stacked transistors also suppressed the leakage power

consumption. So sleepy stack structure achieves ultra low leakage power consumption during sleep mode while retaining the exact logic state. But the main drawback of this sleepy stack technique, however, is increasing area a lot.

3.5 Sleepy keeper approach

In the traditional CMOS design the NMOS are placed always at the pull down network because it is well known that NMOS transistors are not efficient at passing V_{dd} shown in Figure 3.5. On the other hand PMOS transistors are placed at the pull up network because PMOS transistors are not efficient at passing GND.

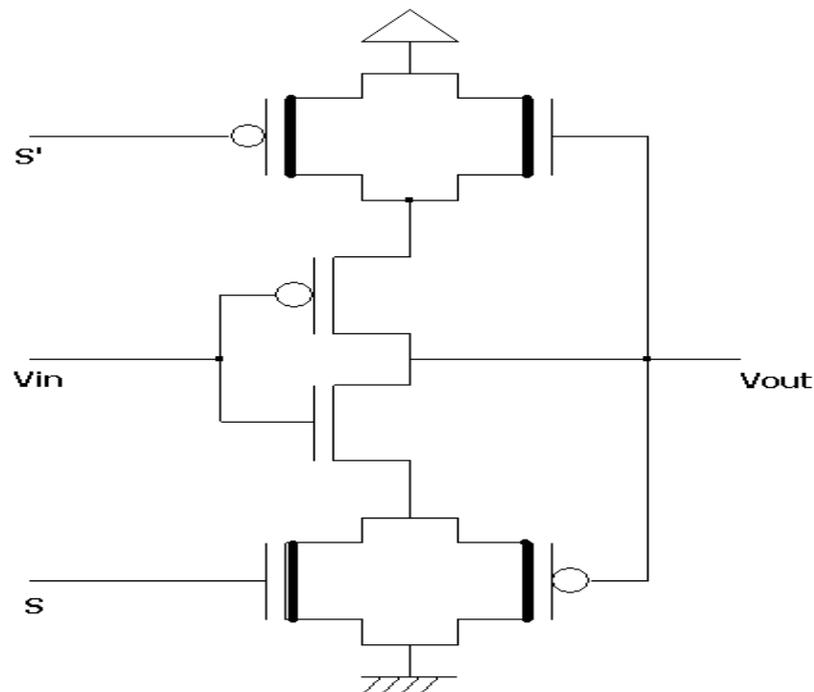


Figure 3.5: Sleepy keeper approach

Let us maintain a value of '1' in sleep mode and assume that the value has already been calculated. The sleepy keeper [9] circuit in figure 3.5 uses this output value of '1' and an NMOS transistor maintains this value during sleep mode. An additional NMOS transistor is added in parallel to the pull up sleep transistor connected to V_{dd} . At sleep mode this NMOS transistor is the only source of V_{dd} to the pull-up network since the sleep transistor is off. Similarly, to maintain a '0' value, assume that the value is already calculated. The sleepy keeper approach [10] uses this output value of '0' and a PMOS transistor maintains the value

during sleep mode.

An additional PMOS transistor is added in parallel to the pull down sleep transistor connected to GND. At sleep mode this PMOS transistor is only source of GND the pull down network since the sleep transistor is off. In 0.07 μm technology the sleepy keeper approach (with dual V_{th}) achieves 2290X leakage reduction over the base case and 175X leakage reduction over the stack approach. The result is similar to the best previous leakage reduction technique with state saving, sleepy stack, but in sleepy keeper the delay is less than the sleepy stack. The drawbacks [11] of sleepy keeper is that it consumes 31% more dynamic power than the sleepy stack with single V_{th} and 41% more dynamic power with dual V_{th} . The dynamic power increases around 15% over the base case and 10% over the sleep transistor approach. The area usage by sleepy keeper increase 93% over the base case but 49% smaller area usage than sleepy stack.

3.6 Dual Sleep approach

Sleep transistors are crucial part in any low leakage power design. In dual sleep method [12] shown in figure 3.6, two sleep transistors in each NMOS or PMOS block are used. One sleep transistor is used to turn on in ON state and the other one is used to turn on in OFF state. Again in OFF state a block containing both PMOS and NMOS transistors are used in order to reduce the leakage power.

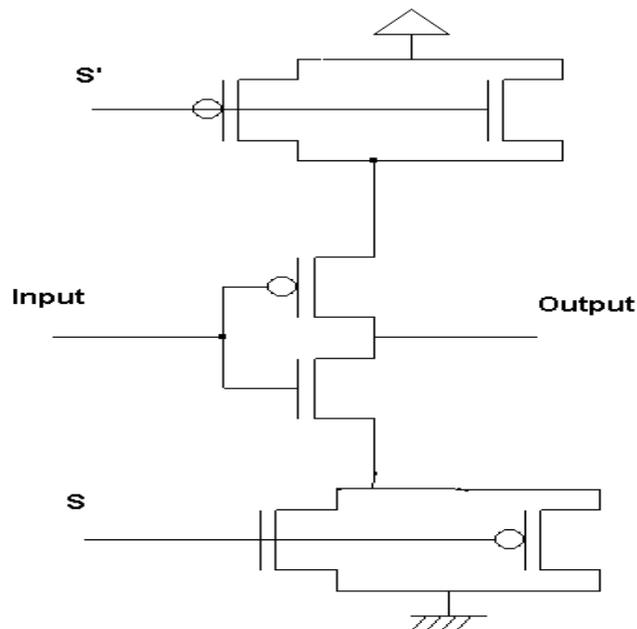


Figure 3.6: Dual sleep method

Like the sleep, sleepy stack and sleepy keeper approaches, dual V_{th} technology can be applied in dual sleep approach to obtain greater leakage power reduction [13]. Since high- V_{th} results in less leakage but lowers performance, high- V_{th} is applied only to leakage reduction transistors, which are sleep transistors, and any transistors in parallel to the sleep transistors.

Dual sleep approach uses the advantage [14] of using the two extra pull-up and two extra pull-down transistors in sleep mode either in OFF state or in ON state. It uses two pull-up sleep transistors and two pull-down sleep transistors. When $S=1$ the pull down NMOS transistor is ON and the pull-up PMOS transistor is ON since $S'=0$. So the arrangement works as a normal device in ON state. During OFF state S is forced to 0 and hence the pull-down NMOS transistor is OFF and PMOS transistor is ON and the pull-up PMOS transistor is OFF while NMOS transistor is ON. So in OFF state a PMOS is in series with an NMOS both in pull-up and pull-down circuits which is liable to reduce power.

CHAPTER 4

PROPOSED METHODS

In this section, we introduce our new leakage power reduction techniques. At first we will discuss the structure of the techniques then we will show the operations of these techniques.

4.1 Forced sleep method

We introduce our new leakage power reduction technique. In this technique we use the concept of the forced stack technique and the sleep transistor technique. Though this new method has a delay penalty greater than the forced stack and sleep transistor technique, this technique is far better than any prior approach known to us. The forced sleep method can achieve ultra-low leakage power consumption.

4.1.1 Structure of forced sleep Method

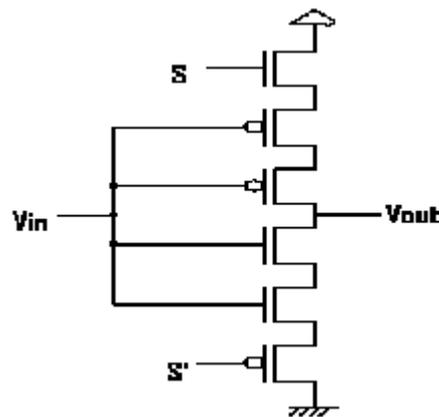


Figure 4.1: Structure of forced sleep Method

The forced sleep method has a structure merging the forced stack technique and the sleep transistor technique. Figure 4.1 shows a forced sleep inverter. The forced sleep inverter in Figure 4.1 uses $W/L = 3$ for the pmos transistors and $W/L = 1.5$ for the nmos transistors, while a conventional inverter with the same input capacitance would use $W/L = 6$ for the

pull-up transistor and $W/L = 3$ for the pull-down transistor (assuming $\mu_n = 2\mu_p$). Then sleep transistors are added in series to each set of two stacked transistors. We use two sleep transistors here, the nmos sleep transistor with V_{dd} and the pmos sleep transistor with ground. Conventionally the nmos transistor is connected to ground because it is very efficient passing ground voltage and the pmos transistor is connected to V_{dd} because it is efficient passing V_{dd} [15]. In forced sleep method we just reverse the connection. That's why we have some delay penalty in our method.

We use same W/L for all the pmos and nmos transistors in this method. However, changing the sleep transistor width may provide additional tradeoffs between delay, power and area.

4.1.2 Operation of forced sleep method

Now we explain how the forced sleep works during active mode and during sleep mode.

Also, we explain leakage power saving using forced sleep structure. The sleep transistors of forced sleep method operate similar to the sleep transistors used in the sleep transistor technique in which sleep transistors are turned on during active mode and turned off during sleep mode. Figure 4.1 depicts the forced sleep operation using a forced sleep inverter. During active mode (Figure 4.1), $S = 1$ and $S' = 0$ are asserted, and thus all sleep transistors are turned on. As we said before that the nmos transistor is connected in V_{dd} . But it is not efficient in passing V_{dd} . So when the input is low output voltage is reduced to $V_{dd} - V_{th}$ [15]. The drain terminal of the nmos sleep transistor is connected to V_{dd} . It is getting always a high voltage which is greater than the source terminal voltage of this nmos sleep transistor. Current I_{ds} will flow through the transistor and the stacked transistors also. This nmos sleep transistor has its own internal resistance for which the leakage current will be less than that of a forced stack inverter circuit when the input voltage is low. The pmos is connected to ground. When the input signal is high voltage the output is $0 - V_{th}$. The source terminal voltage is higher than the drain terminal voltage of the pmos sleep transistor. Current I_{ds} will flow through the transistors. For an internal resistance of the pmos sleep transistor the leakage current will be much less than any other approaches.

During sleep mode (Figure 4.1), $S = 0$ and $S' = 1$ are asserted, and so both of the sleep transistors are turned off. The leakage reduction of forced sleep structure occurs in two ways. First, leakage power is suppressed by the two sleep transistors which are not efficient in passing V_{dd} (nmos) and ground (pmos) potential. They will be in pure sleep mode at sleep

mode operation. Second, two stacked and turned off transistors induce the stack effect, which also suppresses leakage power consumption. By combining these two effects, the forced sleep technique achieves ultra-low leakage power consumption during sleep mode. The price for this, however, is increased delay.

4.2 Stacked sleep approach

This is another new leakage reduction technique. In this technique the sleep transistors are getting stacked. That's why we call it "Stacked sleep" approach.

4.2.1 Structure of Stacked sleep approach

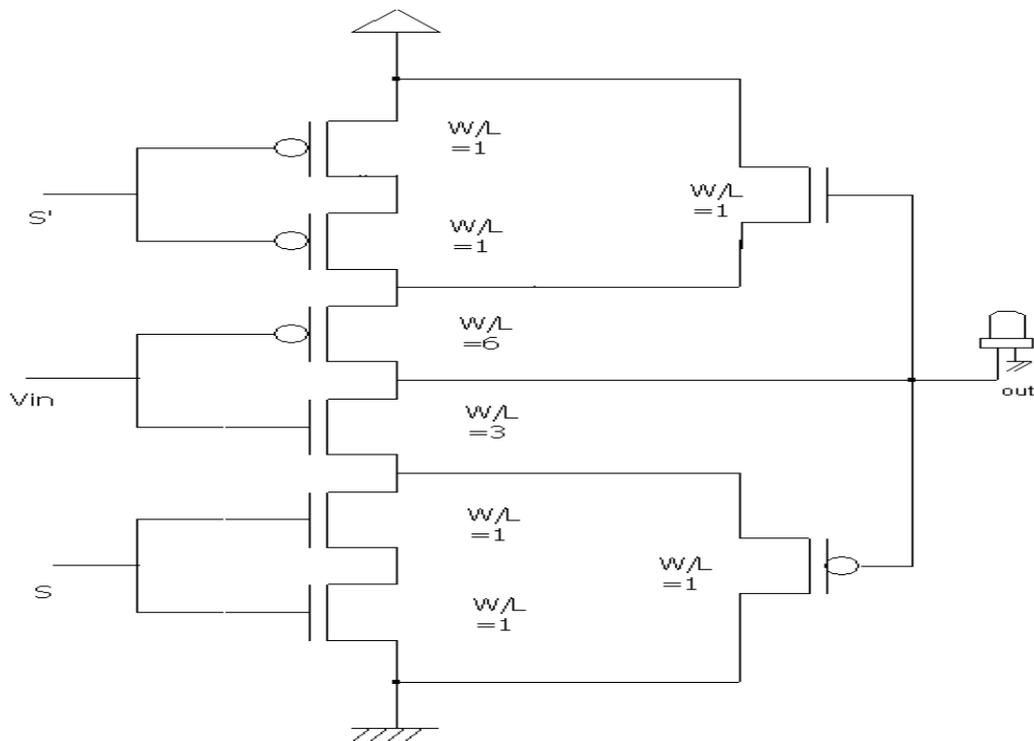


Figure 4.2: Structure of Stacked sleep approach

This technique uses two stacked sleep transistor in V_{dd} and two stacked sleep transistor in ground. So, leakage reduction in this technique occurs in two ways. First, the stack effect [16] of sleep transistors and second, the sleep transistor effect. It is well known that pmos transistors are not efficient at passing GND; similarly, it is well known that nmos transistors are not efficient at passing V_{dd} . But this stacked sleep technique uses pmos transistor in GND and nmos transistor in V_{dd} for maintaining the exact logic state during sleep mode.

This stacked sleep transistor uses aspect ratio $W/L=3$ for nmos transistor and $W/L=6$ for pmos transistor in the main inverter portion. For the stacked sleep transistor this technique uses aspect ratio $W/L=1$ for both the nmos and pmos transistors. The extra two transistors of the design for maintaining the logic state during sleep mode also use aspect ratio $W/L=1$.

4.2.2 Operation of stacked sleep approach

Let us to maintain a value of '1' in sleep mode. During sleep mode $S=0$ and $S'=1$ are asserted. Assume that the '1' value has already been calculated. The stacked sleep approach uses this output value of '1' and an nmos transistor connected to V_{dd} to maintain output value equal to '1' when in sleep mode. As shown in the figure, an additional nmos transistor placed in parallel to the pull up sleep transistors connects V_{dd} to the pull up network. When in sleep mode this nmos transistor is the only source of V_{dd} to the pull up network since the sleep transistors are turned off. Similarly to maintain a value of '0' in sleep mode, assume that the '0' value has already been calculated, the stacked sleep approach uses this output value of '0' and a pmos transistor connected to GND to maintain output value equal to '0' when in sleep mode. Figure 4.2 shows an additional pmos transistor placed in parallel to the pull down sleep transistors. When in sleep mode, this pmos transistor is the only source of GND to the pull down network since the sleep transistors are turned off.

4.3 Variable body biasing technique

This is another new leakage reduction technique, which we call the "Variable body biasing" technique.

4.3.1 Structure of variable body biasing technique

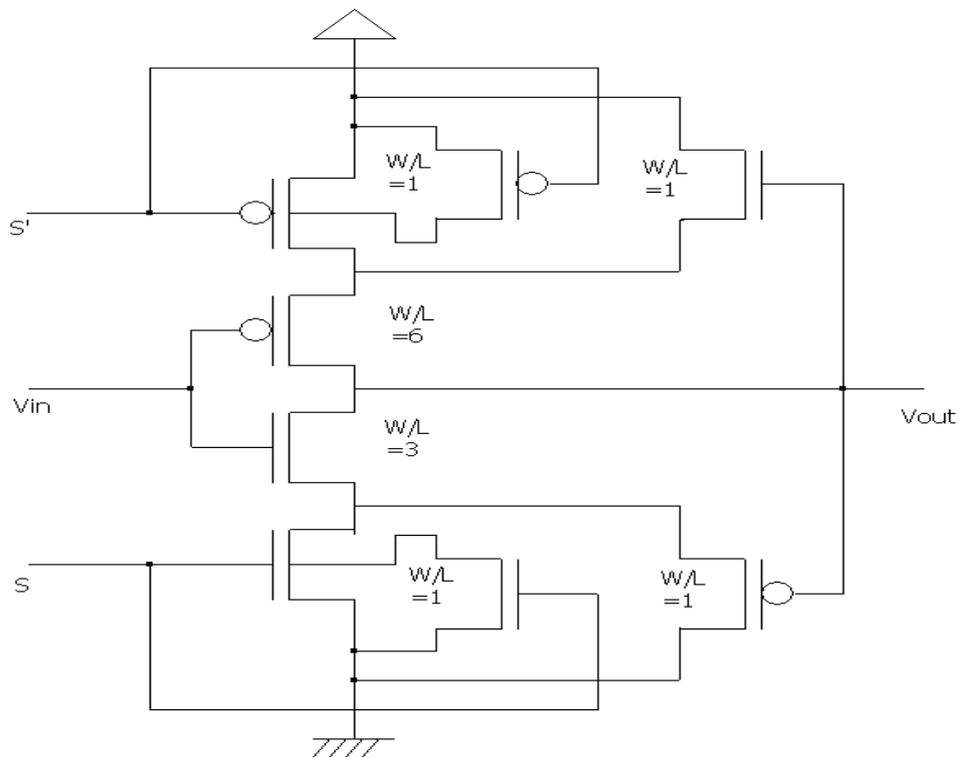


Figure 4.3: Structure of variable body biasing technique

This technique in figure 4.3 uses two parallel connected sleep transistors in V_{dd} and two parallel connected sleep transistors in GND. The source of one of the pmos sleep transistor is connected to the body of other pmos sleep transistor for having so called body biasing effect. Similarly the source of one of the nmos sleep transistor is connected to the body of other nmos sleep transistor for having the same effect as for pmos sleep transistors. So, leakage reduction in this technique occurs in two ways. Firstly, the sleep transistor effect and secondly, the variable body biasing effect. It is well known that pmos transistors are not efficient at passing GND; similarly, it is well known that nmos transistors are not efficient at passing V_{dd} . But this variable body biasing technique uses pmos transistor in GND and nmos transistor in V_{dd} , both are in paralleled to the sleep transistors, for maintaining exact logic state during sleep mode. This technique uses aspect ratio $W/L=3$ for nmos transistor and $W/L=6$ for pmos transistor in the main inverter portion. For the sleep transistors this technique uses aspect ratio $W/L=1$ for both the nmos and pmos transistors. The extra two transistors of the design for maintaining the logic state during sleep mode also use aspect ratio $W/L=1$. Due to the minimum aspect ratio the sub-threshold current reduces.

4.3.2 Operation of Variable body biasing technique

During active mode $S=1$ and $S'=0$ is asserted. Then the entire sleep transistors are ON and the inverter is in active mode. For different input signal desired output signal is caught. During sleep mode or inactive mode $S=0$ and $S'=1$ is asserted. Then the entire sleep transistor is turned OFF. Since the sources of the nmos and pmos sleep transistors are connected to the body of similar transistor as shown in figure 4.3, the threshold voltage of the sleep transistors increases due to the body biasing effect [17] during sleep mode. This increase of threshold voltage of the transistors reduces the leakage current. That's why the static power consumption is also lowers. This variable body biasing technique also has two extra transistors in parallel to the sleep transistors. The purpose of this extra transistor is to save data during inactive mode of the circuit.

Let us to save the value '1' in sleep mode. Assume that the '1' value has already been calculated. This technique uses this output value of '1' and an nmos transistor connected to V_{dd} to maintain output value equal to '1' when in sleep mode. When in sleep mode this nmos transistor is the only source of V_{dd} to the pull up network since the sleep transistors are turned off. Similarly to maintain a value of '0' in sleep mode, assume that the '0' value has already been calculated, this technique uses this output value of '0' and a pmos transistor connected to GND to maintain output value equal to '0' when in sleep mode. When in sleep mode, this pmos transistor is the only source of GND to the pull down network since the sleep transistors are turned off.

CHAPTER 5

APPLYING PROPOSED METHODS

Forced sleep approach, Stack sleep approach and Variable body biasing techniques can successfully be implemented in logic design. To verify this statement here these methods are applied in 2 ways.

1. A chain of 4 inverters
2. SRAM cell

5.1 Application in logic circuit

A chain of four inverters is chosen as a logic circuit.

5.1.1 Forced sleep method using a chain of 4 inverters

A chain of four inverters is chosen because an inverter is the most basic logic circuit in CMOS technology. Figure 5.1.1 depicts a chain of four inverters using forced sleep method. We size the transistor of the inverter to have equal rise and fall times in each state. In this method we use aspect ratio $W/L = 1.5$ for all the nmos transistors and $W/L = 3$ for all the pmos transistors.

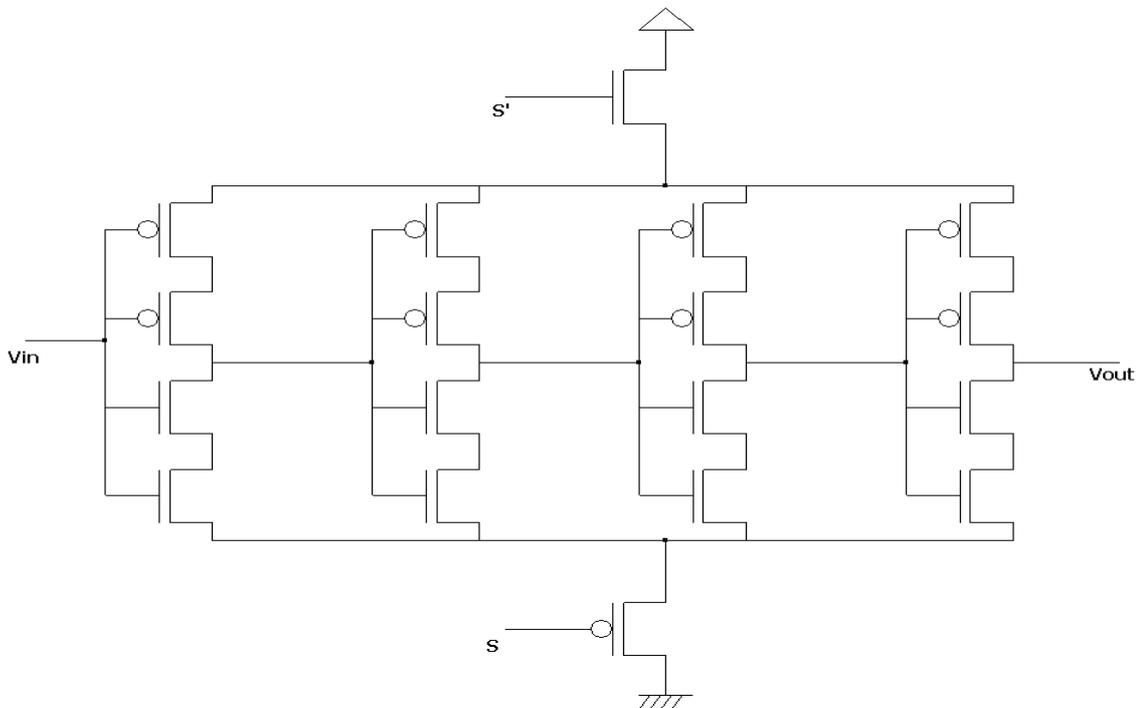


Figure 5.1.1: Forced sleep method using a chain of 4 inverters

5.1.2 Stacked sleep approach using a chain of 4 inverters

In this stacked sleep approach we size the transistors of the inverter to have equal rise and fall times in each state. Figure 5.1.2 depicts a chain of four inverters using stacked sleep approach. This method uses aspect ratio $W/L = 3$ for all nmos transistors and $W/L = 6$ for all pmos transistors in the main inverter circuit. For the sleep transistors and extra transistors it uses aspect ratio $W/L = 1$.

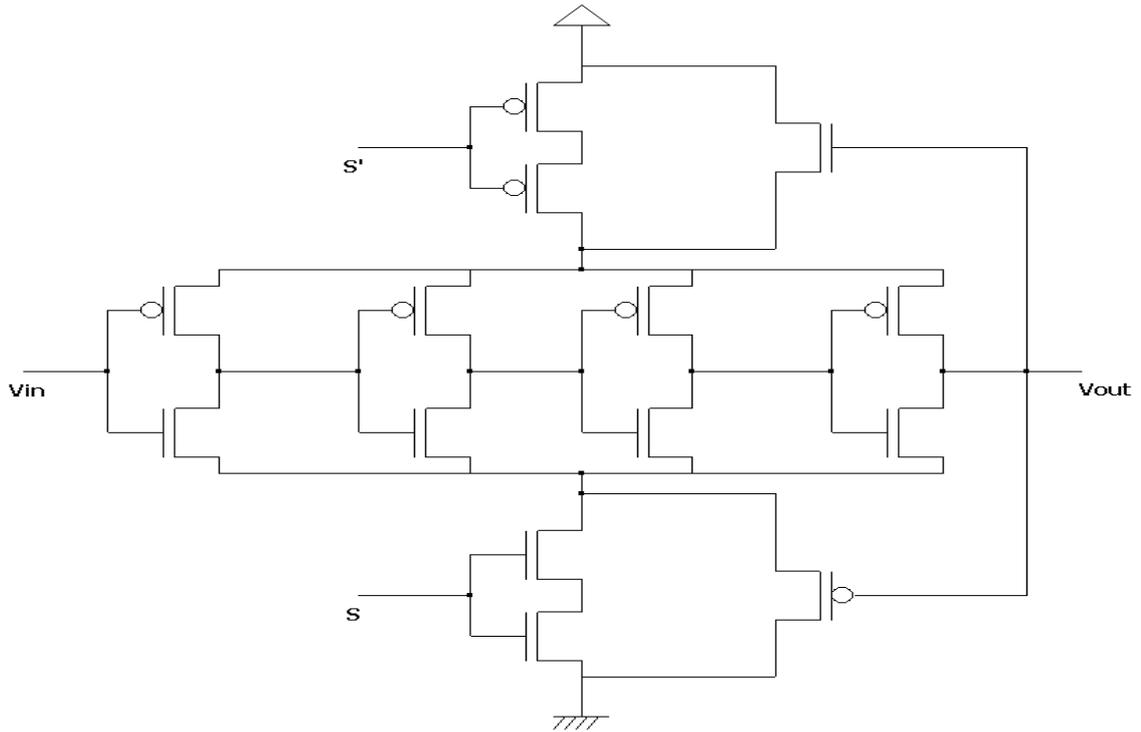


Figure 5.1.2: Stacked sleep approach using a chain of 4 inverters

5.2 Application in SRAM cell

We implement the techniques in SRAM cell.

5.2.1 Forced sleep application in SRAM cell

Figure 5.2.1 depicts the forced sleep application in SRAM cell.

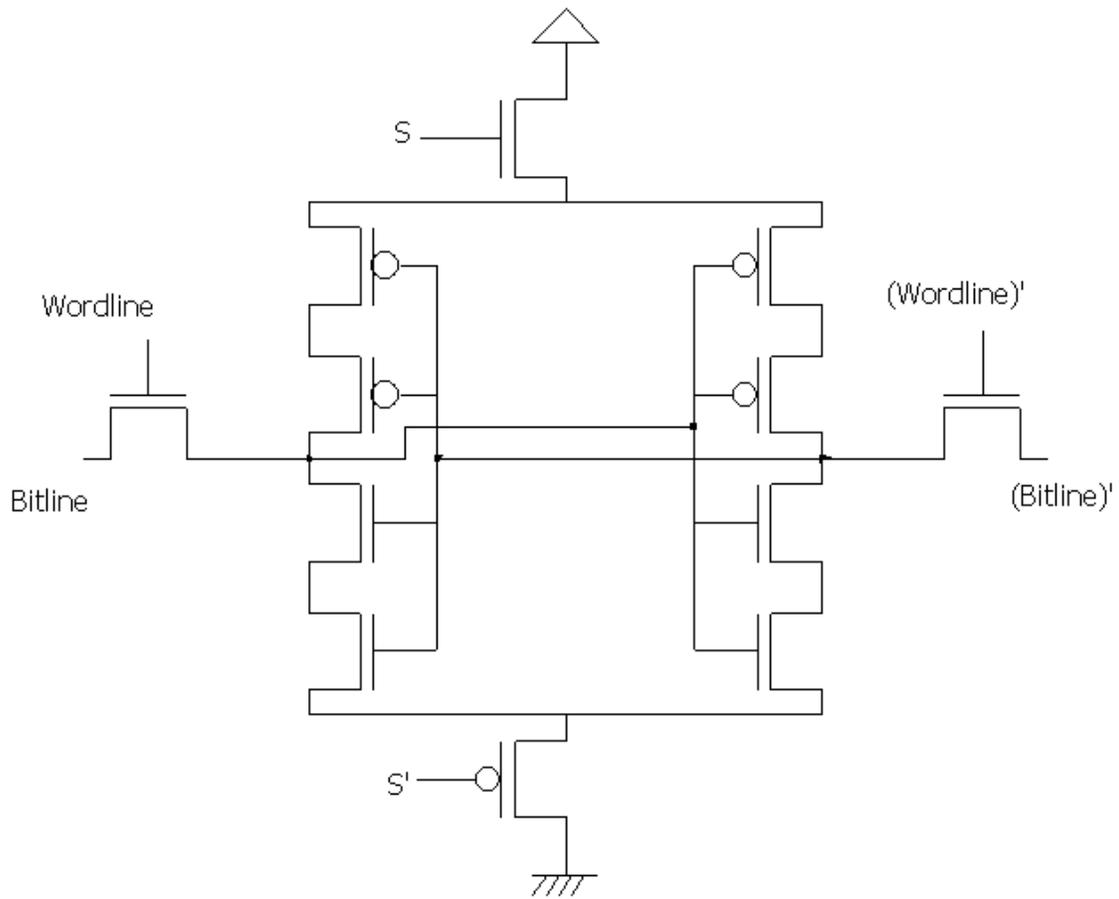


Figure 5.2.1: SRAM cell using forced sleep method

CHAPTER 6

SIMULATION RESULTS

We compare the stacked sleep approach, variable body biasing technique and forced sleep technique to a base case (Figure 3.1) and five other previous approaches, namely sleep transistor, forced stack, sleepy stack, sleepy keeper and dual sleep. Thus, we compare nine design approaches in terms of power consumption (dynamic and static), delay and area. To show that the approaches are applicable to general logic design, we choose a chain of four inverters. We use Synopsis HSPICE [18] for simulation purpose to estimate delay and power consumption. Area is calculated with the help of MICROWIND. All considered approaches are evaluated for performance by using a single, low- V_{th} for all transistors.

The inverter chain uses three inverters each with $W/L=6$ for PMOS and $W/L=3$ for NMOS for the base case. For example, sleep transistors (Figure 3.2) used in the pull-up and pull-down networks of the base case inverter chain have $W/L=6$ and $W/L=3$, respectively. Transistors in the forced stack approach (Figure 3.3) are sized to half of the size of the base case transistors, e.g., transistors used in pull-up and pull-down of the base case inverter chain have $W/L=3$ and $W/L=1.5$, respectively. Similarly, transistors, including sleep transistors, in the sleepy stack approach are sized to half of the size of the base case transistors. Variable body biasing technique (Figure 4.3) uses aspect ratio $W/L=3$ for NMOS transistor and $W/L=6$ for PMOS transistor in the main inverter portion. For the sleep transistors this technique uses aspect ratio $W/L=1$ for both the NMOS and PMOS transistors. The extra two transistors of the design for maintaining the logic state during sleep mode also use aspect ratio $W/L=1$. Stacked sleep approach also uses the same aspect ratio that is used in variable body biasing technique. In the forced sleep approach aspect ratio $W/L=1.5$ for NMOS and $W/L=3$ for PMOS is used.

For layout purpose we have used DSCH software where we have got a verilog while designing the circuit. Then after the compilation of the verilog file in MICROWIND software we got the accurate layout of the design. In this way we got the layout of each method. We have used BSIM4 PTM [19] technologies and adopted 130nm, 90nm, 45nm, 32nm and 22nm processes. The chosen technologies and their supply voltages are given in Table 1.

Table 1: Power supply voltage for different technologies

180n	130n	90n	65n	45n	32n	22n
1.8V	1.3V	1.2V	1.1V	1.0V	0.9V	0.8V

6.1 Simulation results for logic circuits

The simulation results are given in tabulated and figure form below:

6.1.1 Simulation results for a chain of four inverters

Table 2 shows the result of different methods using 32 nm technologies.

Table 2: Data for 32 nm technology:

Method	Prop. Delay (s)	Static power (w)	Dynamic power (w)	Area (μm^2)
Base case	2.4484E-11	8.7649E-08	3.6138E-06	1.20
Sleep	3.6201E-11	1.6272E-09	2.7774E-06	2.45
Forced stack	1.3511E-10	3.9920E-10	8.0436E-07	1.38
Sleepy stack	5.8477E-11	7.0533E-10	1.4503E-06	2.07
Sleepy keeper	4.0711E-11	1.4976E-09	2.8465E-06	1.83
Dual sleep	3.8831E-11	1.1840E-09	2.0870E-06	1.28
Stacked sleep	1.3128E-10	1.9640E-10	7.2014E-07	1.54
Variable body biasing	7.8629E-11	3.6360E-10	1.0344E-06	1.54
Forced sleep	3.3781E-09	5.9908E-12	1.3517E-07	1.78

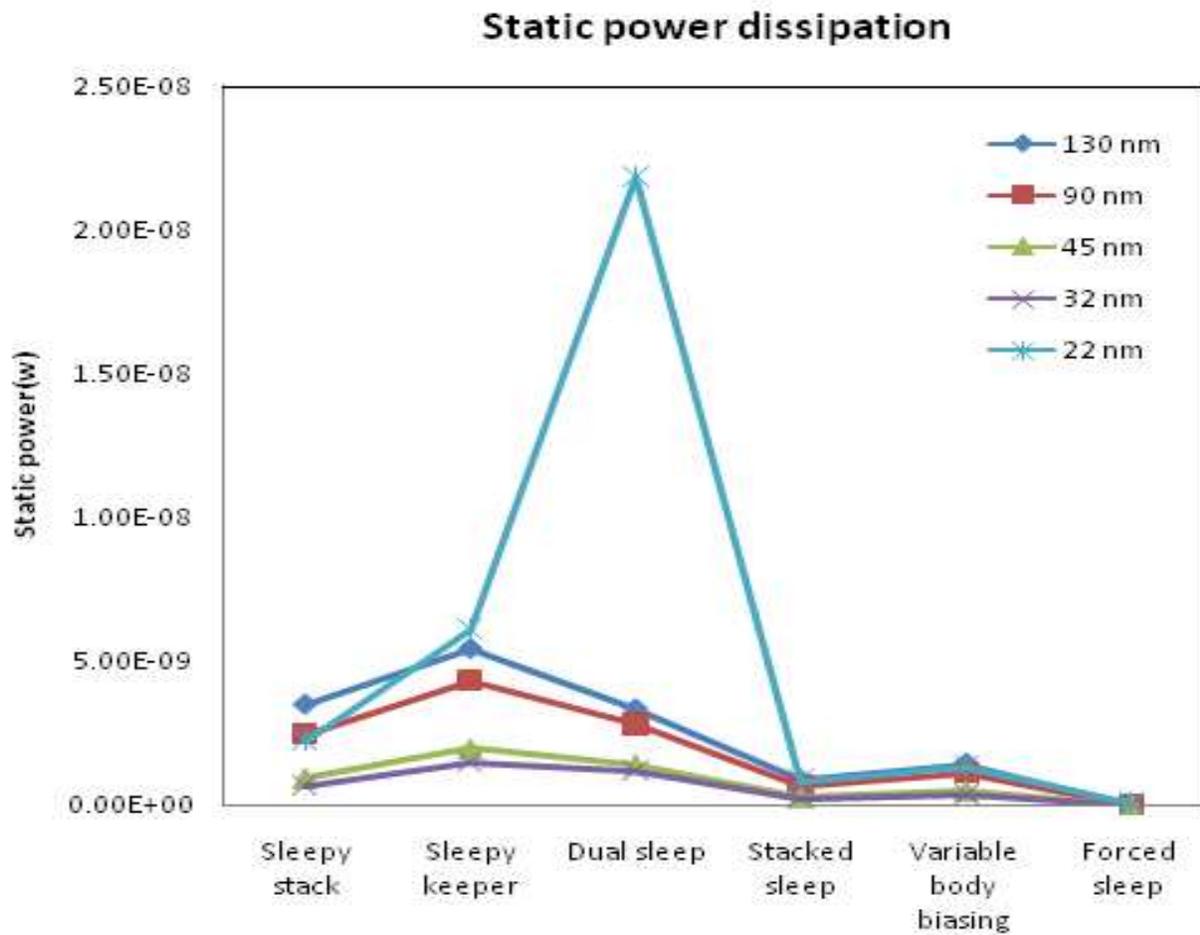


Figure 6.1.1(a): Static power dissipation of a chain of four inverters

Figure 6.1.1(a) shows the static power dissipation of a chain of four inverters in different technologies. In case of 32 nm technology stacked sleep approach gives 83.41% and variable body biasing technique gives 69.29% good results than the dual sleep approach. The forced sleep approach shows a 99.49% satisfactory result rather than the dual sleep approach.

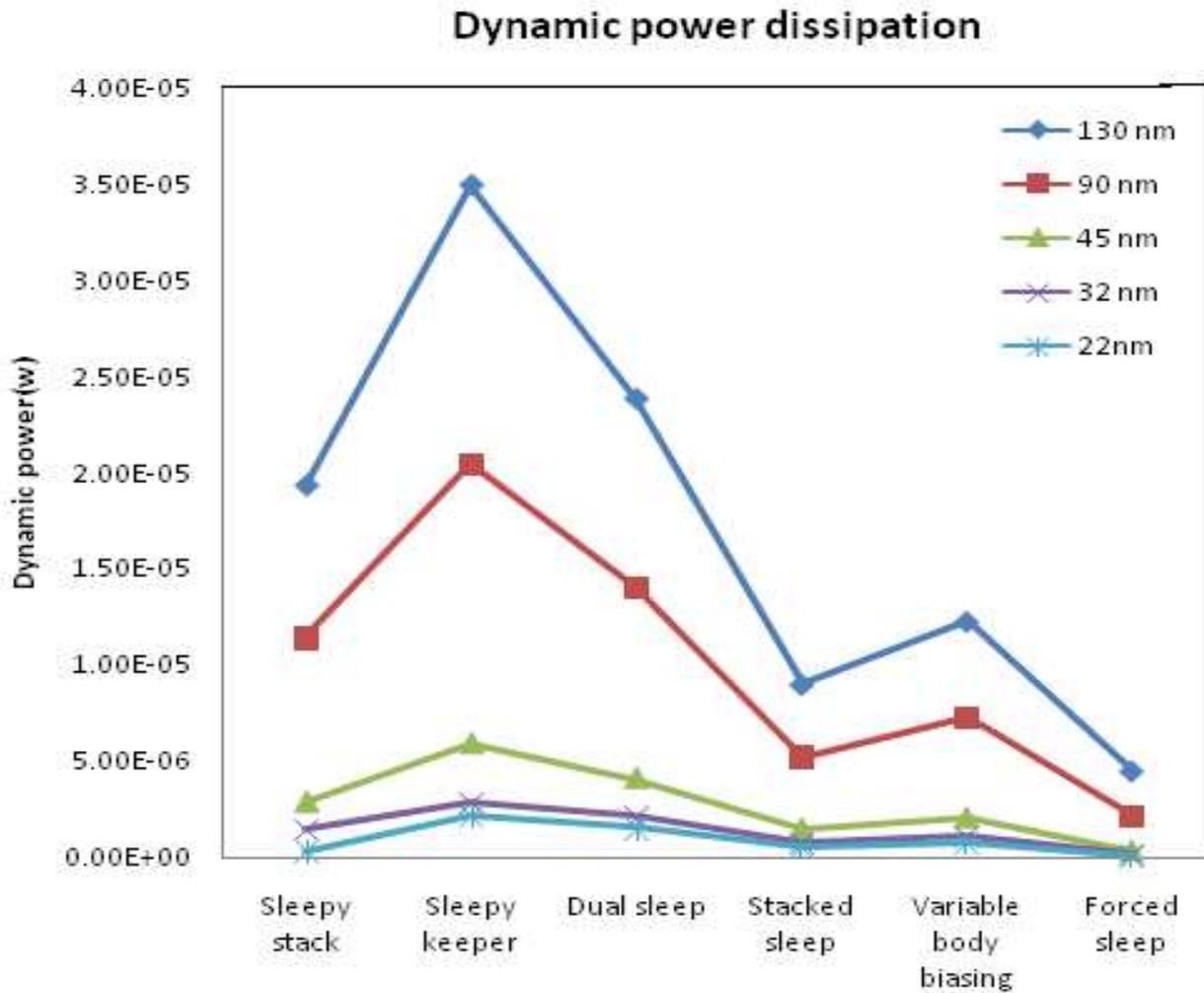


Figure 6.1.1(b): Dynamic power dissipation of a chain of four inverters

Figure 6.1.1(b) shows the dynamic power dissipation of a chain of four inverters in different technologies. In case of 32 nm technology stacked sleep approach gives 65.49% and variable body biasing technique gives 50.44% good results than the dual sleep approach. Forced sleep approach also shows a 93.53% good result than the dual sleep approach.

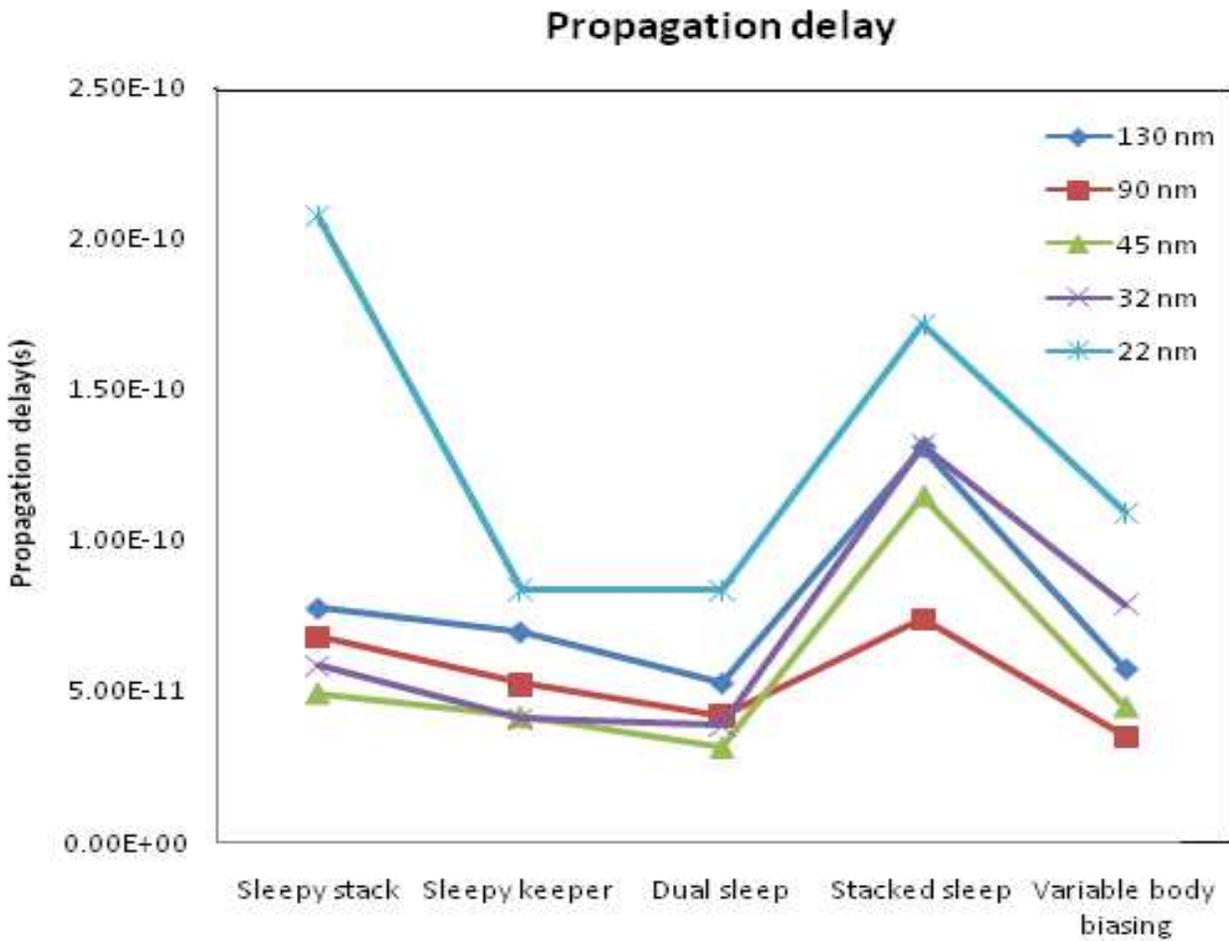


Figure 6.1.1(c): Propagation delay of a chain of four inverters

Figure 6.1.1(c) shows the propagation delay of a chain of four inverters in different technologies. In case of 32 nm technology stacked sleep approach gives -238.08% worst and variable body biasing technique gives -102.49% worst results compare to the dual sleep approach.

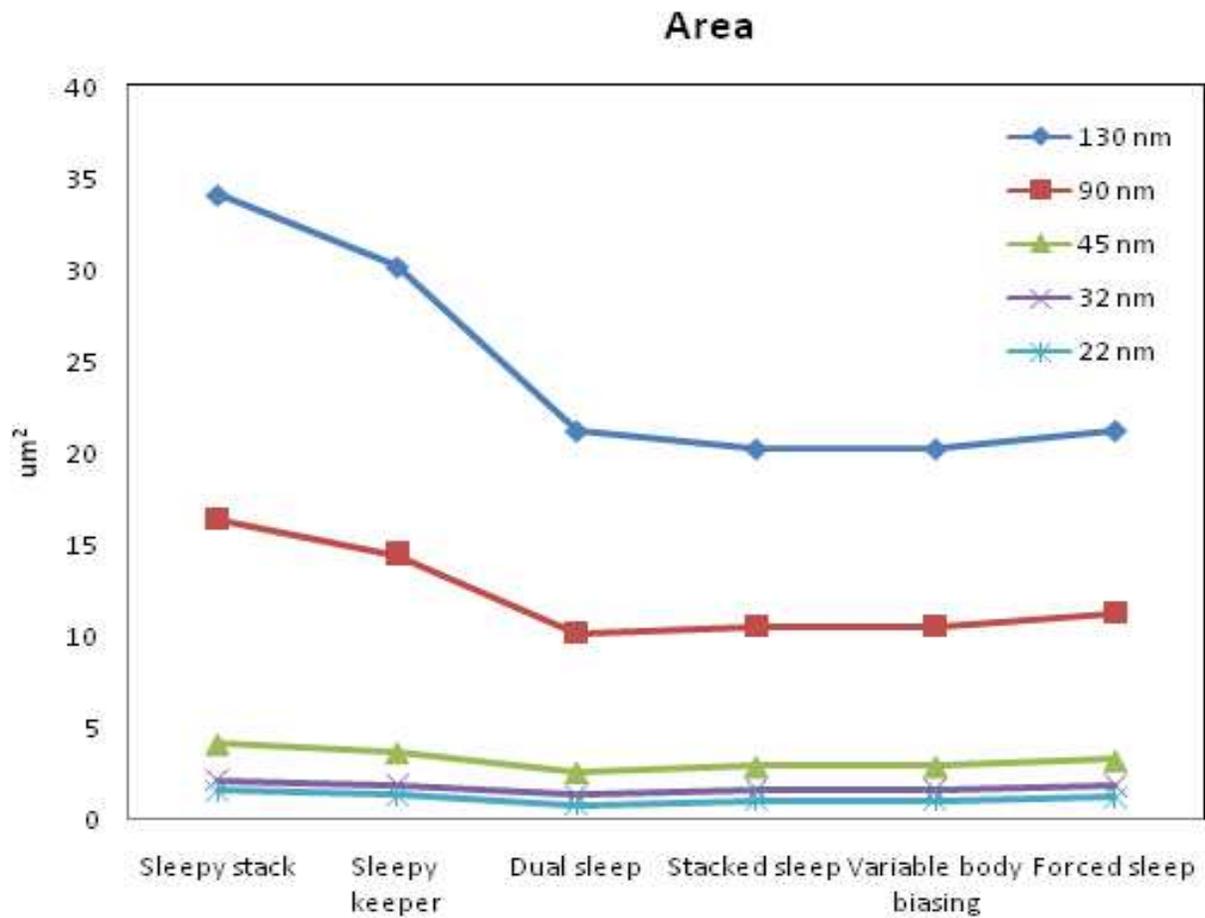


Figure 6.1.1(d): Area of a chain of four inverters

Figure 6.1.1(d) shows the area of a chain of four inverters in different technologies. In case of 32 nm technology both the stacked sleep approach and the variable body biasing technique gives -20.31% worst result than the dual sleep approach. Forced sleep approach gives -39% worst results than the dual sleep method.

6.1.2 Simulation results for SRAM cell

The simulation result for SRAM cell with 32 nm technology is given below in tabulated form.

Table 3: Data for 32 nm technology:

Method	Propagation delay (s)	Static power (w)	Dynamic power (w)	Area (um ²)
Sleep	1.4421E-10	6.0426E-06	4.8164E-05	14.2
Sleepy stack	1.1246E-11	1.7501E-06	2.9430E-05	30.1
Sleepy keeper	1.1019E-11	4.0419E-06	5.8129E-05	22.2
Dual sleep	1.1143E-11	2.5241E-06	4.2409E-05	10.1
Stacked sleep	1.3672E-10	3.7760E-06	1.2236E-06	10.0
Variable body biasing	1.2966E-10	2.0018E-06	1.9816E-06	10.0
Forced sleep	1.3284E-10	7.5519E-07	4.5671E-08	9.97

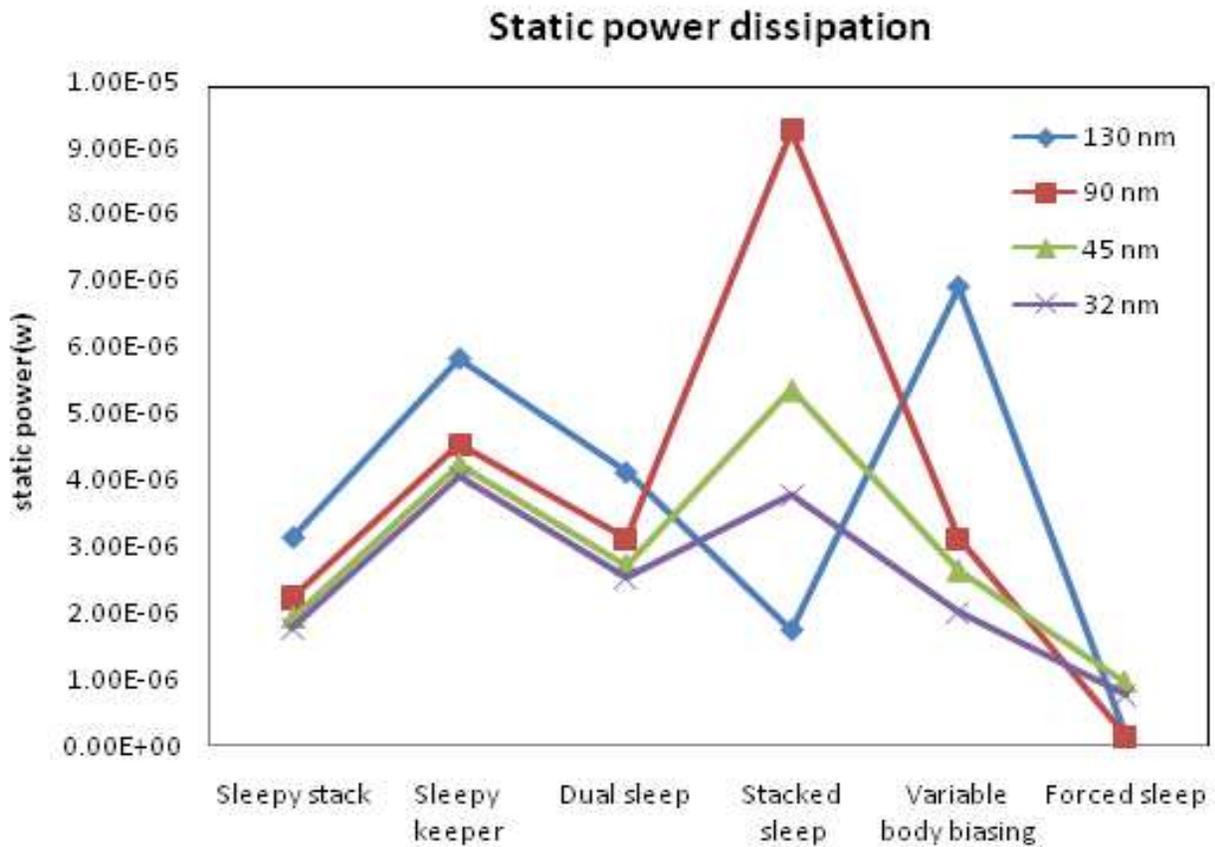


Figure 6.1.2(a): Static power dissipation of an SRAM cell

Figure 6.1.2(a) shows the static power dissipation of an SRAM cell in different technologies. In case of 32 nm technology stacked sleep approach gives -49.59% worst and variable body biasing technique gives 20.69% good results than the dual sleep approach. The forced sleep approach shows a 70.08% satisfactory result rather than the dual sleep method.

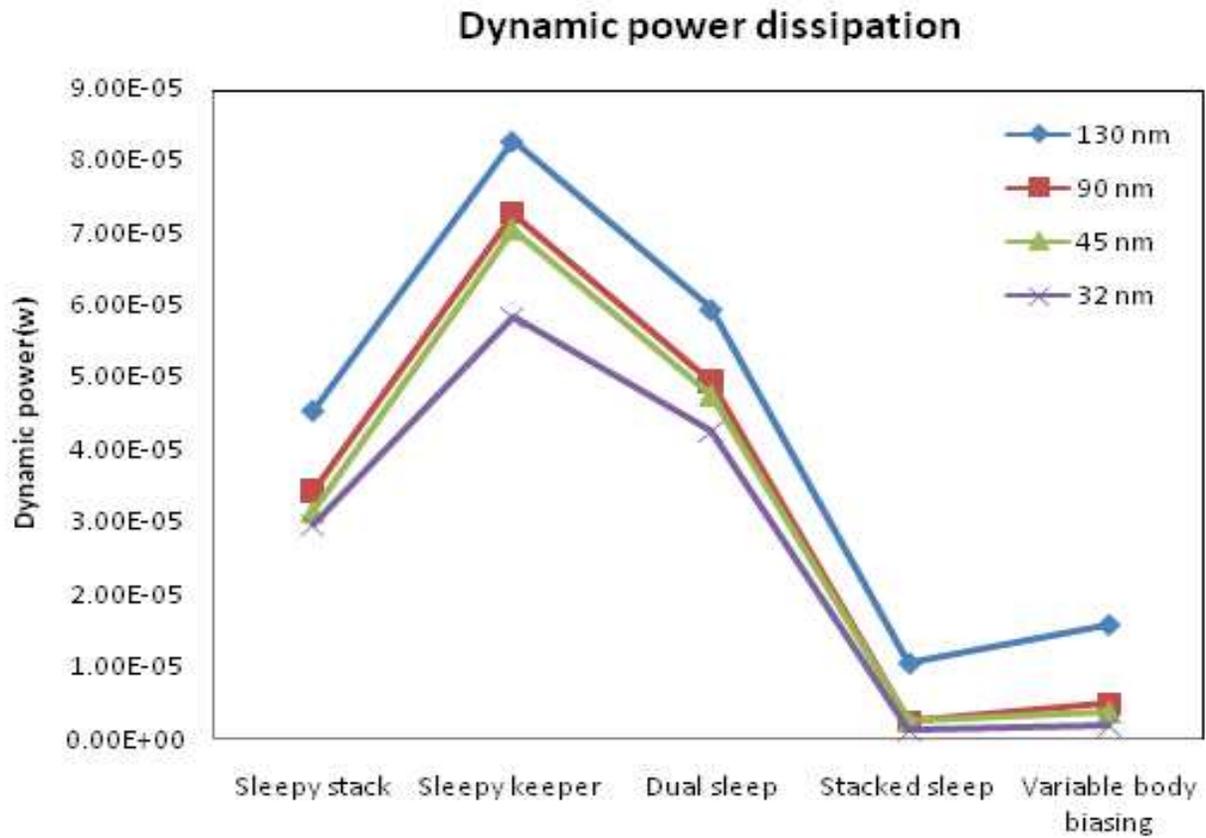


Figure 6.1.2(b): Dynamic power dissipation of an SRAM cell

Figure 6.1.2(b) shows the dynamic power dissipation of an SRAM cell in different technologies. In case of 32 nm technology stacked sleep approach gives 97.11% and variable body biasing technique gives 95.33% good results than the dual sleep approach.

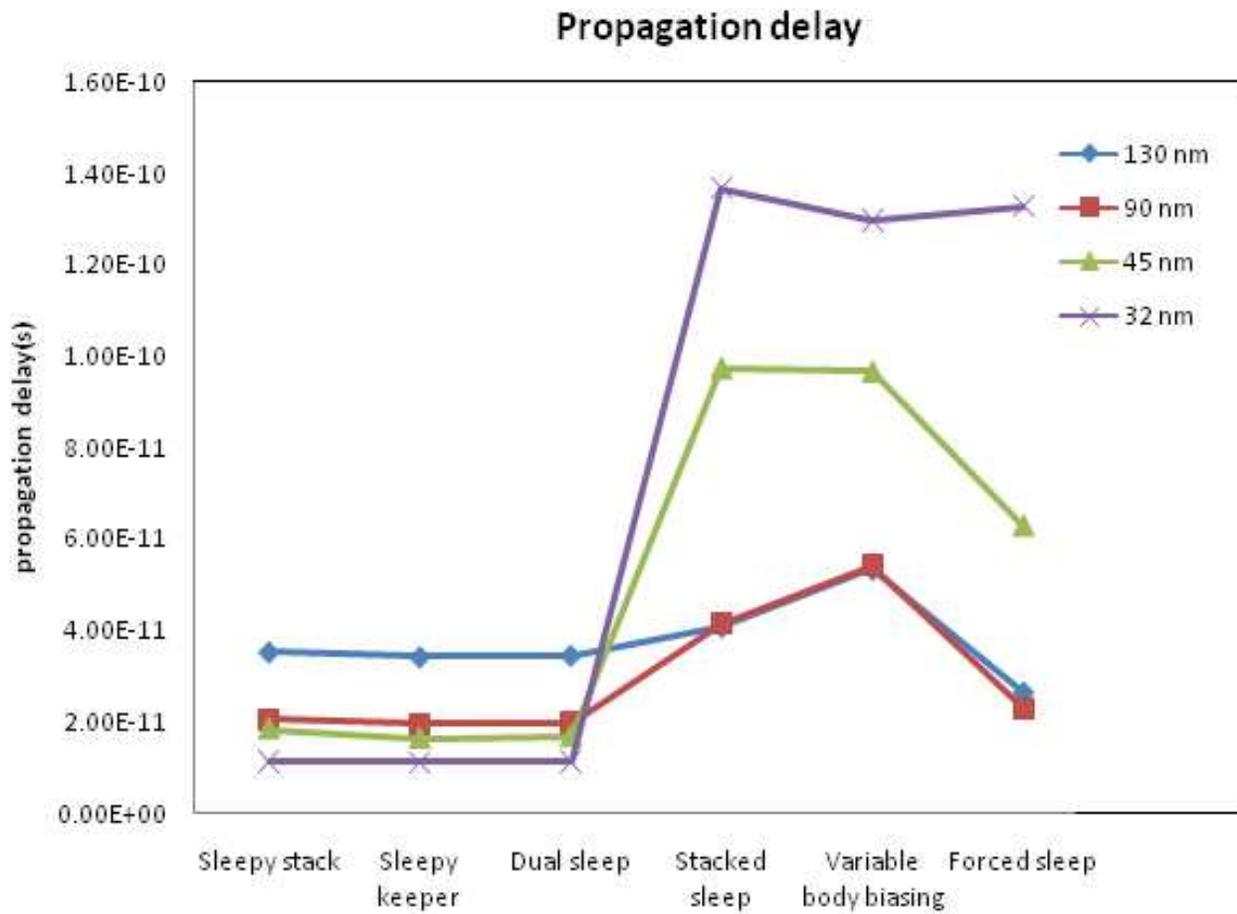


Figure 6.1.2(c): Propagation delay of an SRAM cell

Figure 6.1.2(c) shows the propagation delay of an SRAM cell in different technologies. In case of 32 nm technology stacked sleep approach gives -112.6% worst and variable body biasing technique gives -102.9% worst results compare to the dual sleep approach. Forced sleep approach also gives -109.5% worst results than the dual sleep method.

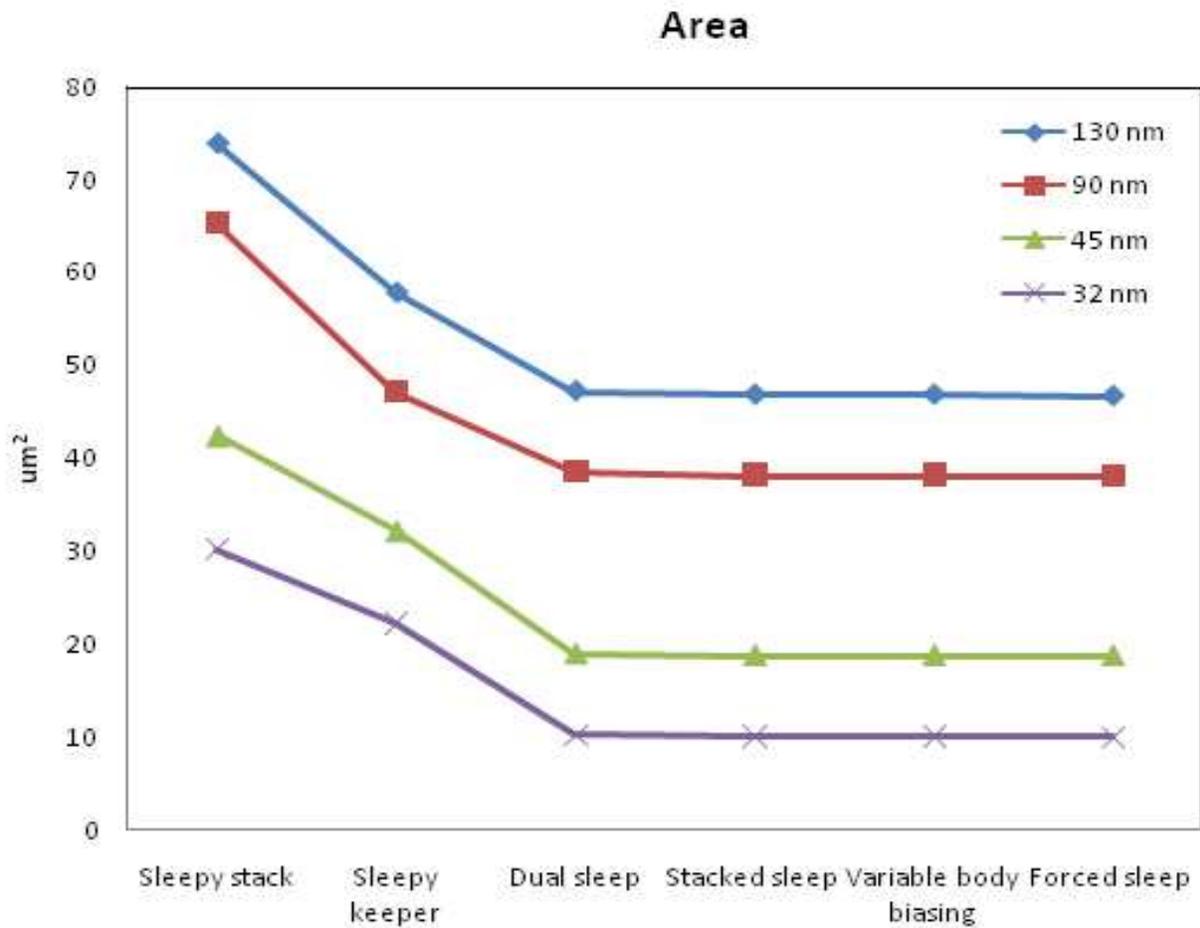


Figure 6.1.2(d): Area of an SRAM cell

Figure 6.1.2(d) shows the area of an SRAM cell in different technologies. In case of 32 nm technology both the stacked sleep approach and the variable body biasing technique gives 1% satisfactory result than the dual sleep approach. Forced sleep approach gives 1.29% good results than the dual sleep method.

CHAPTER 7

CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

7.1 Conclusion

In case of a chain of four inverters, sleepy stack method shows 7.5802% improvement and 92.8275%, 58.8095% degraded performance than stacked sleep method in propagation delay, dynamic power and static power, respectively. When compared to variable body biasing technique, sleepy stack method shows 49.1254%, 87.6638%&, 42.7778% degraded performance in propagation delay, static power and dynamic power, respectively. Again in comparison with sleepy stack approach, forced sleep technique gives 93.35% degraded performance in propagation delay, but gives comparatively good performance in static and dynamic power. In comparison with dual sleep approach, stacked sleep shows 64.4665% degraded and 582.81%, 195.96% improved performance in propagation delay, static and dynamic power respectively. Variable body bias technique shows 40.0423% degraded but 283.66% and 114.240% improved performance in propagation delay, static and dynamic power respectively. Forced sleep technique also gives good performance in static and dynamic power but degraded performance in propagation delay. So the novel stacked sleep, variable body bias and forced sleep technique shows better performance when compared to the sleepy stack and dual sleep method. Finally, these proposed methods show the least speed power product among all methods. Moreover, the novel stacked sleep, variable body bias and forced sleep method shows best performance as far as area requirement and speed are concerned.

CMOS technology in nanometer scale faces great challenge due to sub-threshold leakage power consumption. The earlier approaches and our proposed approaches can be effective in some ways, but no one exactly knows the real solution for the reduction of power consumption. So, based on different technology and design parameters the techniques are chosen by the designers. The earlier approaches are discussed in brief in this paper and two novel approaches are proposed for generic logic and memory circuit. The methods can be applied to single and multi-threshold voltages. The proposed methods are unique in area saving and faster than any other approaches. Tradeoff between power and delay is occurred in excellent way in our

methods. So, these stacked sleep, variable body biasing and forced sleep techniques represent a new way in the VLSI designer's working area.

7.2 Suggestions for future work

We used HSPICE for simulation and obtaining static power, dynamic power and propagation delay. We used MICROWIND for area calculation. Dependence of static power, dynamic power and propagation delay on threshold voltage scaling and temperature variation can be estimated in these methods. Our proposed new methods can be implemented in 1 bit adder, low power pipelined cache etc in future. Also, static noise margin determination as well as noise analysis of this method can be done for further improvement.

REFERENCES

- [1] International Technology Roadmap for Semiconductors by Semiconductor Industry Association, 2002. [Online] Available <http://public.itrs.net>
- [2] J. Park, "Sleepy Stack: a New Approach to Low Power VLSI and Memory," Ph.D. Dissertation, School of Electrical and Computer Engineering, Georgia Institute of Technology, 2005. [Online].
- [3] Linda E. M. Brackenbury, "Design of VLSI Systems- A Practical Introduction", Department of Computer Science, Manchester University.
- [4] KIM, C. and ROY, K., "Dynamic Vt SRAM: a Leakage Tolerant Cache Memory for Low Voltage Microprocessors," *Proceedings of the International Symposium on Low Power Electronics and Design*, pp. 251–254, August 2002.
- [5] KIM, N., AUSTIN, T., BAAUW, D., MUDGE, T., FLAUTNER, K., HU, J., IRWIN, M., KANDEMIR, M., and NARAYANAN, V., "Leakage Current: Moore's Law Meets Static Power," *IEEE Computer*, vol. 36, pp. 68–75, December 2003.
- [6] S. Mutoh et al., "1-V Power Supply High-speed Digital Circuit Technology with Multithreshold-Voltage CMOS," *IEEE Journal of Solid-State Circuits*, Vol. 30, No. 8, pp. 847-854, August 1995.
- [7] NARENDRA, S., S. BORKAR, V. D., ANTONIADIS, D., and CHANDRAKASAN, A., "Scaling of Stack Effect and its Application for Leakage Reduction," *Proceedings of 148th International Symposium on Low Power Electronics and Design*, pp. 195–200, August 2001.
- [8] J.C. Park, V. J. Mooney III and P. Pfeifferberger, "Sleepy Stack Reduction of Leakage Power," *Proceeding of the International Workshop on Power and Timing*

- Modeling, Optimization and Simulation, pp. 148-158, September 2004.
- [9] J. Kao and A. Chandrakasan, "MTCMOS sequential circuits, Proceedings of European Solid-State Circuits Conference, pp 332-335, September 2001.
- [10] Se Hun Kim, V.J. Mooney, "Sleepy Keeper: a New Approach to Low-leakage Power VLSI Design" Proceeding of the 2006 IFIP International Conference on Very Large Scale Integration, pp367-372, Oct. 2006.
- [11] Sleepy Keeper: a New Approach to Low-leakage Power VLSI Design Vincent J. Mooney III, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA USA (mooney@ece.gatech.edu)
- [12] Nittaranjan Karmakar, Mehdi Z. Sadi, Md. Khorshed Alam and M. S. Islam1, "A Novel Dual Sleep Approach to Low Leakage and Area Efficient VLSI Design" IEEE-RSM Proc. 2009, Kota Bahru, Malaysia
- [13] NOSE, K. and SAKURAI, T., "Analysis and Future Trend of Short-Circuit Power," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 9, pp. 1023–1030, September 2000.
- [14] CHANDRAKASAN, A. P., SHENG, S., and BRODERSEN, R. W., "Low-Power CMOS Digital Design," *IEEE Journal of Solid-State Circuits*, vol. 27, no. 4, pp. 473–484, April 1992.
- [15] <http://www.edaboard.com/thread133596.html>
- [16] JOHNSON, M., SOMASEKHAR, D., CHIOU, L.-Y., and ROY, K., "Leakage Control with Efficient Use of Transistor Stacks in Single Threshold CMOS," *IEEE Transactions on VLSI Systems*, vol. 10, no. 1, pp. 1–5, February 2002.

[17] <http://www.edaboard.com/thread108793.html>

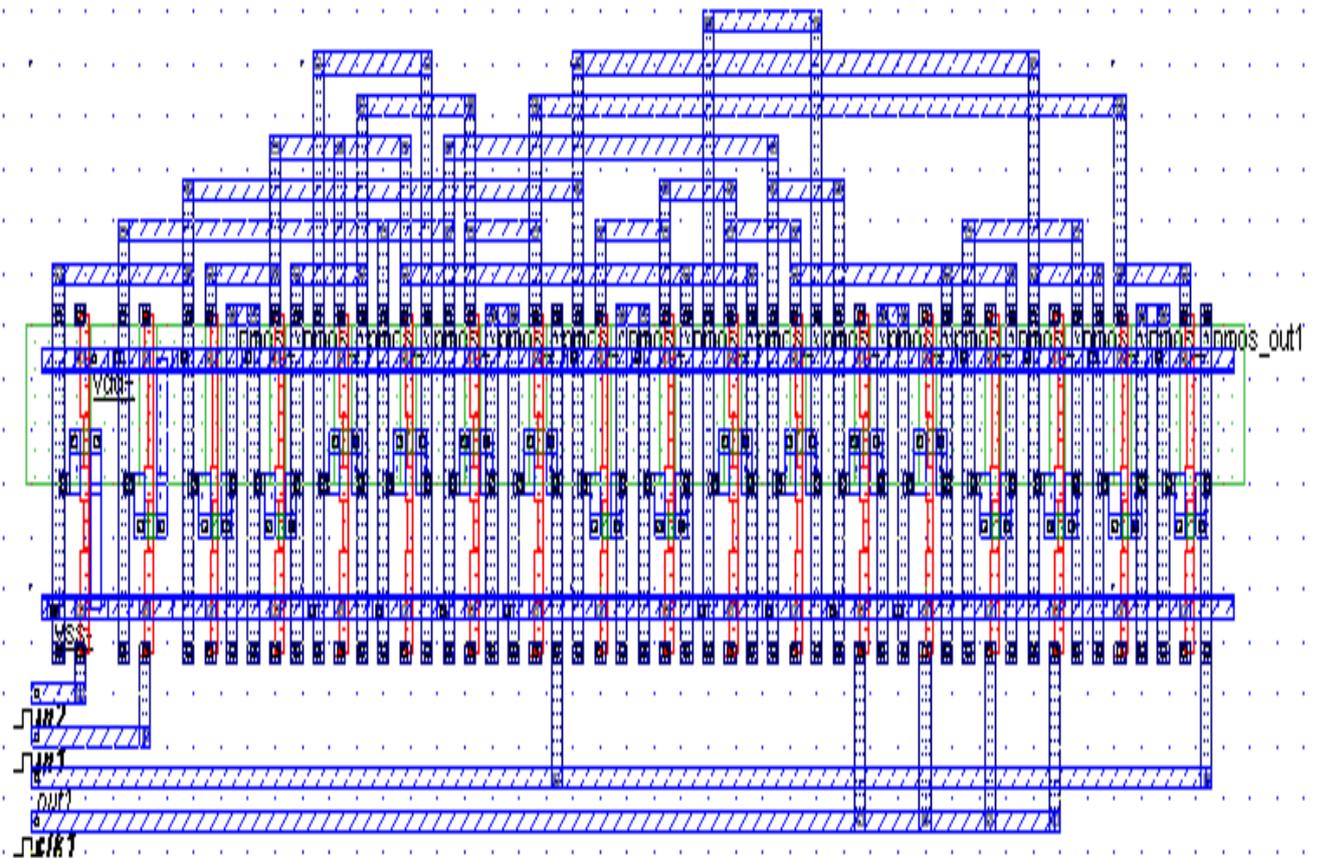
[18] Avant! Corporation, <http://www.avanticorp.com>

[19] PTM Model [<http://ptm.asu.edu/latest.html>]

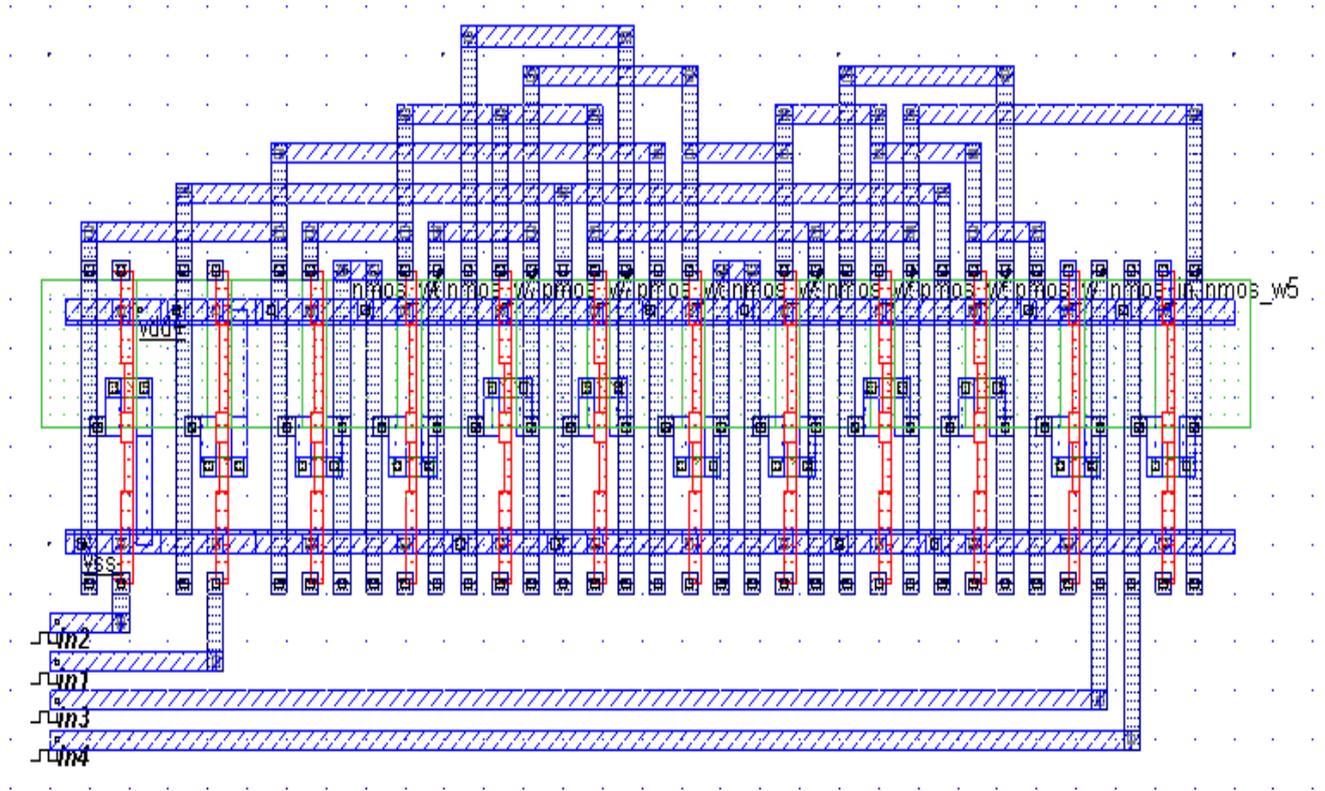
APPENDICES

APPENDIX A: DESIGN LAYOUTS

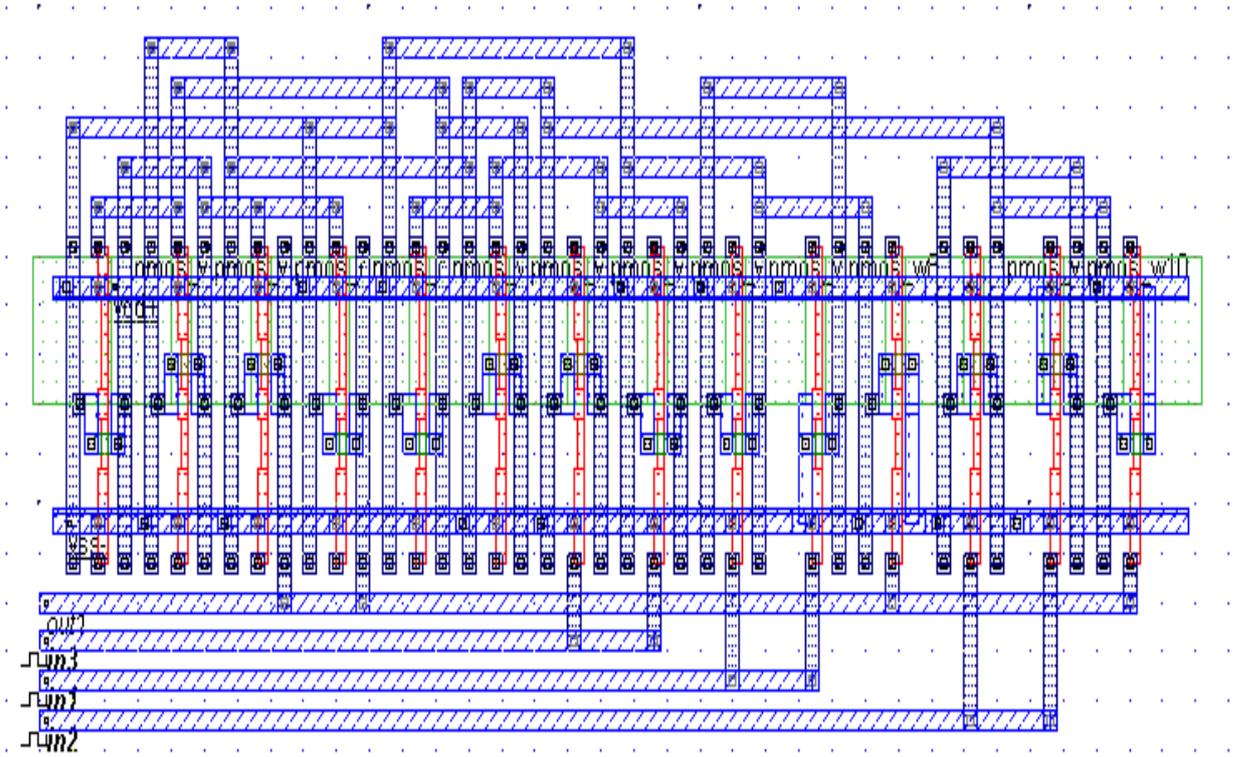
A1: Layout of a chain of four inverters using forced sleep method



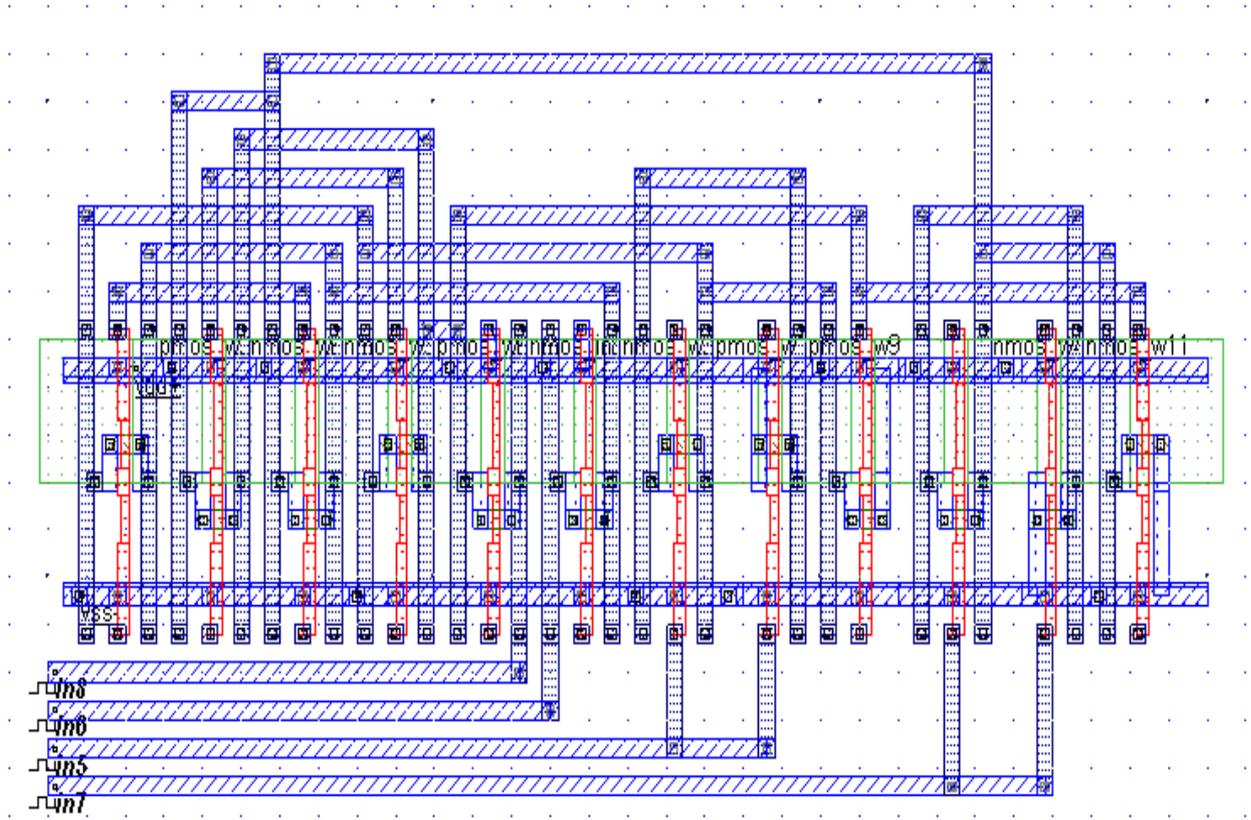
A2: Layout of an SRAM cell using forced sleep method



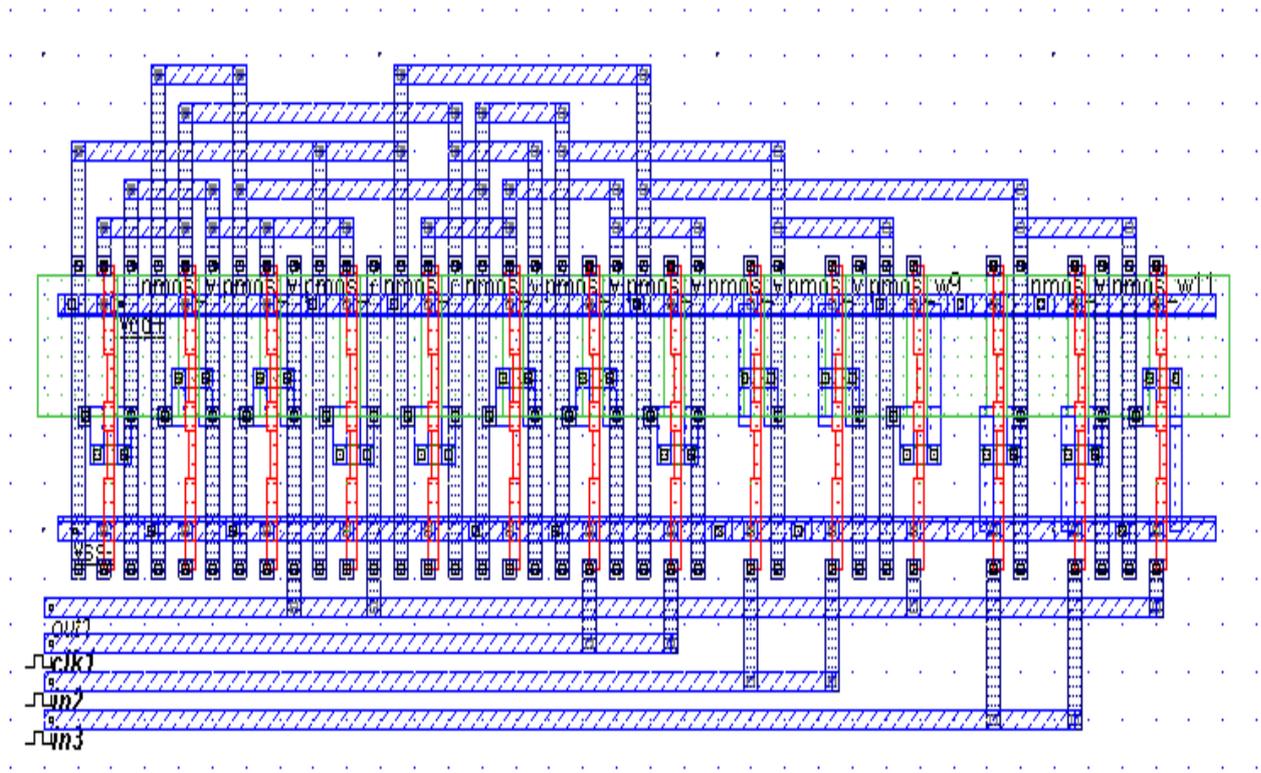
A3: Layout of a chain of four inverters using stacked sleep approach



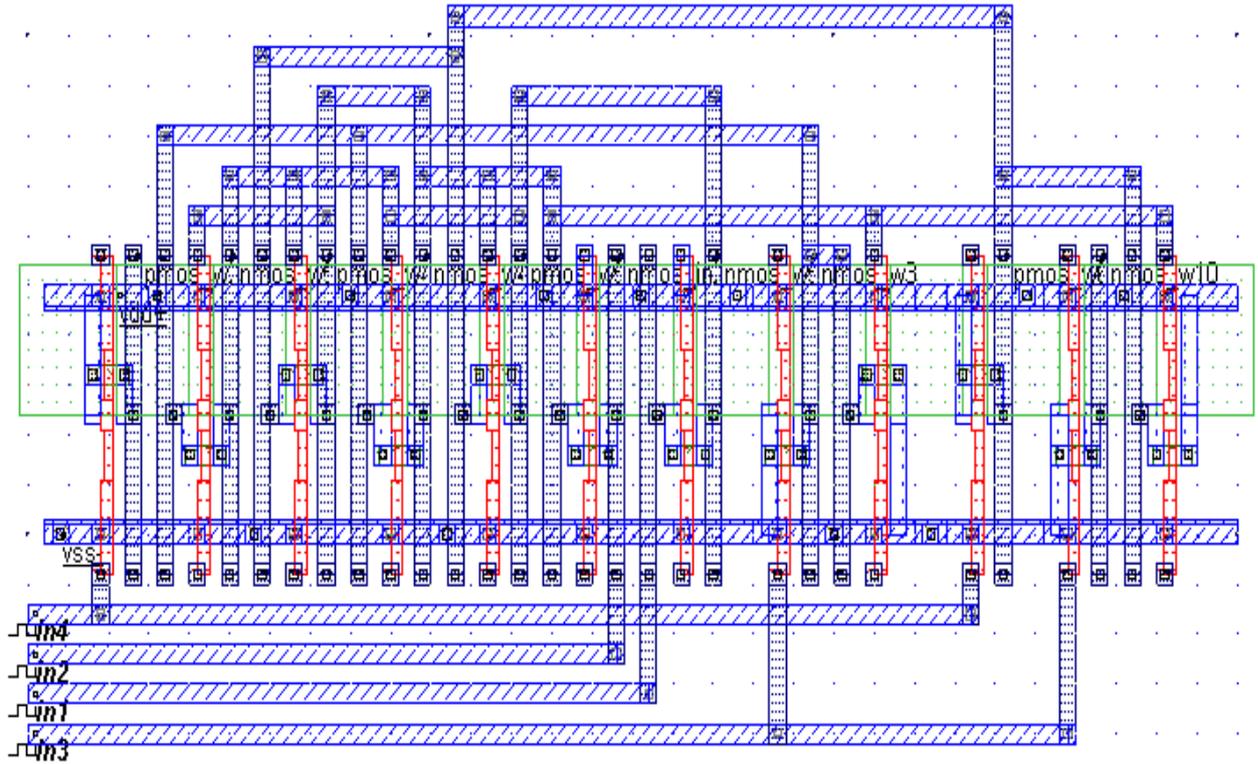
A4: Layout of an SRAM cell using stacked sleep approach



A5: Layout of a chain of four inverters using variable body biasing technique



A6: Layout of an SRAM cell using variable body biasing technique



APPENDIX B

Simulation data (A chain of four inverters)

B1: 130 nm technology

Method	Prop. Delay (s)	Static power (w)	Dynamic power (w)	Area (μm^2)
Base case	4.7115E-11	7.6409E-08	4.4100E-05	19.86
Sleep	6.1844E-11	3.2616E-09	3.3640E-05	40.48
Forced stack	1.2386E-10	1.8794E-09	1.3656E-05	22.76
Sleepy stack	7.7467E-11	3.4858E-09	1.9299E-05	34.14
Sleepy keeper	6.9639E-11	5.4257E-09	3.4876E-05	30.19
Dual sleep	5.2810E-11	3.3280E-09	2.3799E-05	21.18
Stacked sleep	1.3061E-10	8.6053E-10	8.9493E-06	20.20
Variable body biasing	5.7529E-11	1.4386E-09	1.2219E-05	20.20
Forced sleep	9.6417E-10	4.4478E-12	4.4622E-06	21.20

B2: 90 nm technology

Method	Prop. Delay (s)	Static power (w)	Dynamic power (w)	Area (μm^2)
Base case	3.4727E-11	7.0288E-08	2.7160E-05	9.52
Sleep	4.7792E-11	4.9303E-09	2.0274E-05	19.40
Forced stack	1.0014E-10	1.3404E-09	7.7115E-06	10.91
Sleepy stack	6.7790E-11	2.4768E-09	1.1395E-05	16.36
Sleepy keeper	5.2241E-11	4.3106E-09	2.0414E-05	14.47
Dual sleep	4.1823E-11	2.7981E-09	1.4023E-05	10.15
Stacked sleep	7.3762E-11	6.5702E-10	5.1863E-06	10.50
Variable body biasing	3.4948E-11	1.1297E-09	7.2141E-06	10.50
Forced sleep	1.0194E-09	4.5516E-12	2.1236E-06	11.20

B3: 45 nm technology

Method	Prop. Delay (s)	Static power (w)	Dynamic power (w)	Area (μm^2)
Base case	2.2987E-11	6.1374E-08	7.2937E-06	2.38
Sleep	3.4156E-11	2.1831E-09	5.8161E-06	4.85
Forced stack	1.0190E-10	5.1468E-10	1.7567E-06	2.73
Sleepy stack	4.9286E-11	9.2945E-10	2.8937E-06	4.09
Sleepy keeper	4.1588E-11	1.9550E-09	5.9554E-06	3.62
Dual sleep	3.1395E-11	1.4113E-09	4.1342E-06	2.54
Stacked sleep	1.1429E-10	2.6841E-10	1.4861E-06	2.87
Variable body biasing	4.4924E-11	4.9402E-10	2.0614E-06	2.87
Forced sleep	2.1201E-09	5.3652E-12	3.4213E-07	3.20

B4: 32 nm technology

Method	Prop. Delay (s)	Static power (w)	Dynamic power (w)	Area (μm^2)
Base case	2.4484E-11	8.7649E-08	3.6138E-06	1.20
Sleep	3.6201E-11	1.6272E-09	2.7774E-06	2.45
Forced stack	1.3511E-10	3.9920E-10	8.0436E-07	1.38
Sleepy stack	5.8477E-11	7.0533E-10	1.4503E-06	2.07
Sleepy keeper	4.0711E-11	1.4976E-09	2.8465E-06	1.83
Dual sleep	3.8831E-11	1.1840E-09	2.0870E-06	1.28
Stacked sleep	1.3128E-10	1.9640E-10	7.2014E-07	1.54
Variable body biasing	7.8629E-11	3.6360E-10	1.0344E-06	1.54
Forced sleep	3.3781E-09	5.9908E-12	1.3517E-07	1.78

B5: 22 nm technology

Method	Prop. Delay (s)	Static power (w)	Dynamic power (w)	Area (μm^2)
Base case	6.3791E-11	3.4845E-07	2.8959E-06	0.68
Sleep	7.9615E-11	6.0288E-09	2.1075E-06	1.93
Forced stack	2.0591E-10	1.3204E-09	3.5722E-06	0.86
Sleepy stack	2.0703E-10	2.2767E-09	3.5258E-07	1.55
Sleepy keeper	8.3391E-11	6.1068E-09	2.1433E-06	1.31
Dual sleep	8.3266E-11	2.1830E-08	1.5364E-06	0.76
Stacked sleep	1.7133E-10	8.4722E-10	5.2741E-07	1.02
Variable body biasing	1.0887E-10	1.3597E-09	7.4509E-07	1.02
Forced sleep	2.4907E-09	7.9637E-11	7.7169E-08	1.19

APPENDIX C

Simulation data (SRAM cell)

C1: 130 nm technology

Method	Propagation delay (s)	Static power (w)	Dynamic power (w)	Area (um ²)
Sleep	2.1206E-10	7.9013E-06	6.1393E-05	52.1
Sleepy stack	3.5213E-11	3.1326E-06	4.5315E-05	73.9
Sleepy keeper	3.4139E-11	5.8216E-06	8.2496E-05	57.8
Dual sleep	3.4434E-11	4.1089E-06	5.9265E-05	47.2
Stacked sleep	4.0760E-11	1.7412E-06	1.0388E-05	46.8
Variable body biasing	5.3268E-11	6.9011E-06	1.5723E-05	46.8
Forced sleep	2.6381E-11	1.3504E-07	9.2194E-07	46.66

C2: 90 nm technology

Method	Propagation delay (s)	Static power (w)	Dynamic power (w)	Area (um ²)
Sleep	1.9249E-10	6.9236E-06	5.2415E-05	42.1
Sleepy stack	2.0398E-11	2.2123E-06	3.4191E-05	65.2
Sleepy keeper	1.9426E-11	4.5210E-06	7.2423E-05	47.1
Dual sleep	1.9647E-11	3.1021E-06	4.9254E-05	38.4
Stacked sleep	4.1341E-11	9.2446E-06	2.4230E-06	38.13
Variable body biasing	5.3959E-11	3.0964E-06	4.8405E-06	38.13
Forced sleep	2.2880E-11	1.2825E-07	3.5644E-07	38.03

C3: 45 nm technology

Method	Propagation delay (s)	Static power (w)	Dynamic power (w)	Area (um ²)
Sleep	1.6245E-10	6.5124E-06	5.0213E-05	25.2
Sleepy stack	1.8460E-11	1.9215E-06	3.1246E-05	42.4
Sleepy keeper	1.6321E-11	4.2316E-06	7.0314E-05	32.1
Dual sleep	1.6809E-11	2.7113E-06	4.7390E-05	18.9
Stacked sleep	9.7186E-11	5.3533E-06	2.3571E-06	18.76
Variable body biasing	9.6452E-11	2.6273E-06	3.6438E-06	18.76
Forced sleep	6.2797E-11	9.4451E-07	8.4688E-08	18.72

C4: 32 nm technology

Method	Propagation delay (s)	Static power (w)	Dynamic power (w)	Area (um ²)
Sleep	1.4421E-10	6.0426E-06	4.8164E-05	14.2
Sleepy stack	1.1246E-11	1.7501E-06	2.9430E-05	30.1
Sleepy keeper	1.1019E-11	4.0419E-06	5.8129E-05	22.2
Dual sleep	1.1143E-11	2.5241E-06	4.2409E-05	10.1
Stacked sleep	1.3672E-10	3.7760E-06	1.2236E-06	10.0
Variable body biasing	1.2966E-10	2.0018E-06	1.9816E-06	10.0
Forced sleep	1.3284E-10	7.5519E-07	4.5671E-08	9.97

C5: 22 nm technology

Method	Propagation delay (s)	Static power (w)	Dynamic power (w)	Area (um ²)
Sleep	2.5961E-10	5.1466E-06	3.3264E-05	13.7
Sleepy stack	2.2262E-11	3.2641E-06	1.2930E-05	29.6
Sleepy keeper	2.3693E-11	6.0421E-06	2.7209E-05	21.7
Dual sleep	2.6325E-12	3.2416E-06	2.3125E-05	9.6
Stacked sleep	2.0898E-11	1.1594E-06	3.2581E-06	9.5
Variable body biasing	2.6483E-11	2.2526E-06	5.8098E-06	9.5
Forced sleep	2.8777E-11	3.2621E-07	3.4391E-07	9.4