A Novel Low Power Design of SRAM cell and its Performance Analysis

Anuja George
P G scholar
Rajagiri School of Engineering & Technology, Kakkanad

Nisha Thankachan
P G scholar
Rajagiri School of Engineering & Technology, Kakkanad

Anita Harapanahalli
Assistant Professor
Rajagiri School of Engineering & Technology, Kakkanad

ABSTRACT
SRAM is a type of semiconductor memory which does not need to be periodically refreshed. With scaling down of the technology, the feature sizes have shrink more and more and miniaturization at chip level has occurred. But as a trade off, the demand for power has also increased. SRAM continues to be a critical component across a gamut of microelectronics applications. Leakage is a serious problem particularly for SRAM. To address sub threshold leakage issue sleepy stack approach is used .The sleepy stack SRAM cell design, is a new technique which involves changing the circuit structure as well as using high-V_{th}. The sleepy stack technique achieves greatly reduced leakage power while maintaining precise logic state in sleep mode. This paper compares performance of SRAM using sleepy stack approach with that of conventional design. The impact of temperature and voltage on the performance of sleepy stack design is also analyzed. Berkeley Predictive Technology Model (BPTM), level 49 targeting 0.18µm technology is used. The design is successfully simulated and analyzed using HSPICE tools.

General Terms
Low power design, SRAM

Keywords
Power consumption, Sleepy Stack technique, Propagation Delay, SRAM cell, Static Power

1. INTRODUCTION
High density is the primary design goal for memories. To continue to improve the performance of the circuits and to integrate more functions into each chip, feature size had to shrink more and more. As a result, the magnitude of power per unit area has kept growing. Dynamic power consumption was the major concern for low-power chip design since dynamic power accounted for 90% or more of the total chip power. However, as the feature size shrinks, e.g., to 0.09 and 0.065 m, static power poses a great challenge for current and future technologies. Based on the International Technology Roadmap for Semiconductors report, the subthreshold leakage power dissipation of a chip may exceed dynamic power dissipation at the 65-nm feature size [4]. Low voltage operation is essential for low power. But V_{dd} cannot be scaled down aggressively for low power consumption. To address sub threshold leakage issue sleepy stack approach is used [1] .The sleepy stack SRAM cell design, is a new technique which involves changing the circuit structure as well as using high-V_{th}. The sleepy stack technique achieves greatly reduced leakage power while maintaining precise logic state in sleep mode. In this paper the performance comparison of SRAM using sleepy stack approach with that of conventional design is done. The impact of temperature and voltage on the performance of sleepy stack designs is also analyzed. HSPICE tools provide an environment for the same. The estimation of power and delay of the two approaches of design is done using BSIM for 0.18 µm technology across PVT conditions. The write operation is illustrated using awaves in the HSPICE Tool. The results of the simulation can be used to prove the effectiveness of the new approach. We can expect the community to meet the power challenge in the next few years.

2. PREVIOUS WORK
Preliminary work on the low power design was concentrated on the logic circuits. Techniques for leakage power reduction can be grouped into the following two categories:1) State-saving techniques where circuit state (present value) is retained 2) State-destructive techniques where the current Boolean output value of the circuit might be lost. Sleep transistor is a state-destructive technique which cut off transistor networks from supply voltage or ground using sleep transistors, thus reducing leakage current. Sleep transistors are inserted between power supply and pull-up network or between ground and pull-down network or both. Sleep transistor can be designed in different ways for different V_{th} based on the trade-off between leakage power and propagation delay.

A state-saving technique has an advantage over a state-destructive technique in that with a state-saving technique the circuitry can immediately resume operation at a point much later in time without having to somehow regenerate state. Another low power technique is zigzag technique [1] which can retain a particular state chosen prior to chip fabrication. Transistor stacking [2] is another technique which can retain state and it exploits stacking effect to reduce subthreshold leakage current since forced stacking increases delay, high V_{th} cannot be applied. Different techniques which focused on reducing power dissipation were studied and the analysis of the different existing techniques is successfully simulated across the PVT corner cases. The efficiency of the sleepy stack along with dual V_{th} proved to be the best among the several solutions as shown in figure 1. There is 99.38% of static power reduction in sleepy stack with dual V_{th} when compared with base case.This trend is shown across the PVT corners i.e. there is 99.34% and 99.01% reduction in static power at slow and fast corners respectively in sleepy stack with dual V_{th} when compared with the base case.

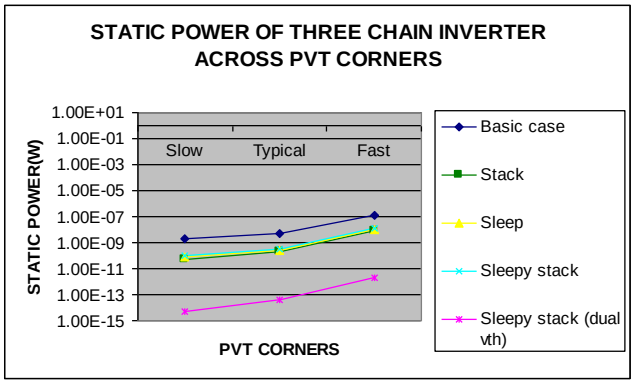


Fig 1: Static power comparison graph for various design cases across PVT condition for three chain inverter

Table 1. Static power across PVT corners

Design Cases	Static power(w)		
	slow	typical	fast
Base case	1.853e-09	4.694e-09	1.244e-07
Stack	5.318e-11	1.787e-10	7.184e-09
Sleep	7.832e-11	2.489e-10	9.345e-09
Sleepy stack	1.003e-10	3.191e-10	1.243e-08
Sleepy stack (dual Vth)	4.913e-15	4.250e-14	1.815e-12

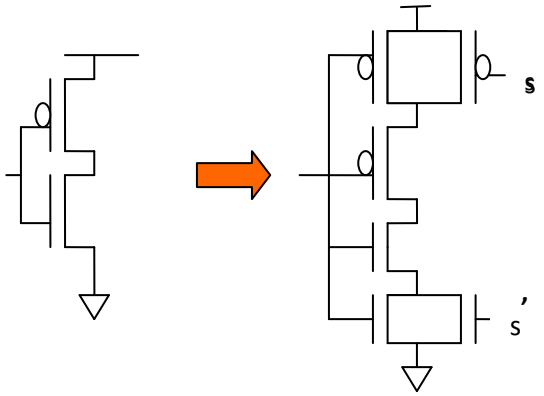
2.1 Sleepy Stack Technique

The sleepy stack technique has combined the advantages of both sleep transistor approach and forced stack. But unlike the sleep transistor technique, the sleepy stack technique retains exact logic state when in sleep mode and also, unlike the forced stack technique, the sleepy stack technique can utilize high-Vth transistors without greater delay penalties. Thus this technique provides better performance than any prior approach known. This technique is applicable to generic VLSI structures as well as SRAM.

2.1.1 Sleepy Stack Structure

The sleepy stack structure has a combined structure of the forced stack and the sleep transistor techniques. Figure 2 shows implementation of sleep an inverter [1].

First, forced Stack structure is obtained by dividing the existing transistor into two transistors of half the width of original transistor. Now a sleep transistor is added in parallel to one of the transistors in each set of two stacked transistors.

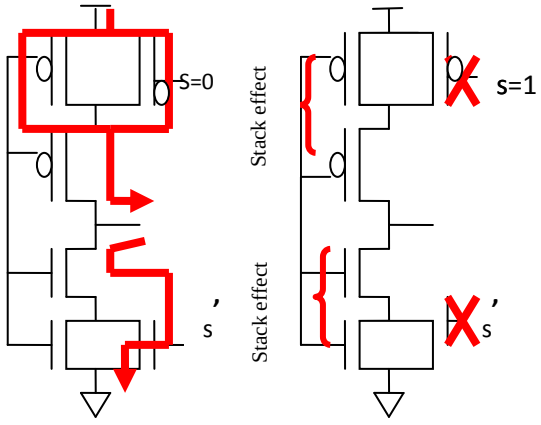


Conventional CMOS Inverter Sleepy Stack Inverter

Fig 2: Sleepy Stack structure implemented in an inverter

2.1.2 Sleepy Stack Operation

When S=0and S'=1 the sleepy stack inverter is in active mode, where all sleep transistors are turned on. Here stacking effect can reduce some amount of dynamic power. This structure can reduce delay in two ways. First, since the sleep transistors are always on during active mode, the sleepy stack structure achieves faster switching time than the forced stack structure. Furthermore, we can use high Vth for the sleep transistors and the transistors parallel to the sleep transistors without incurring large delay increase. During sleep mode S=1 and S'=0, and so both of the sleep transistors are turned off. Even though the sleep transistors are turned off, the sleepy stack structure maintains exact logic state. The leakage reduction of the sleepy stack structure occurs in two ways. First, leakage power is suppressed by high Vth-transistors, which are applied to the sleep transistors and the transistors parallel to the sleep transistors. Second, stacked and turned off transistors induce the stack effect, which also suppresses leakage power consumption. This reduced power is attained at the cost of increased area due to additional transistors used and a slight increase in the delay.



Active Mode Sleep Mode

Fig 3: Sleepy Stack operation in an inverter

3. CONVENTIONAL SRAM CELL

Each bit in an SRAM is stored on four transistors that form two cross-coupled inverters. This storage cell has two stable states which are used to denote 0 and 1. Two additional access transistors serve to control the access to a storage cell during read and write operations. Access to the cell is enabled by the word line which controls the two access transistors M5 and M6 which, in turn, control whether the cell should be connected to the bit lines: BL and BL'. They are used to transfer data for both read and write operations. The two bit lines, both the signal and its inverse are typically provided in order to improve noise margins.

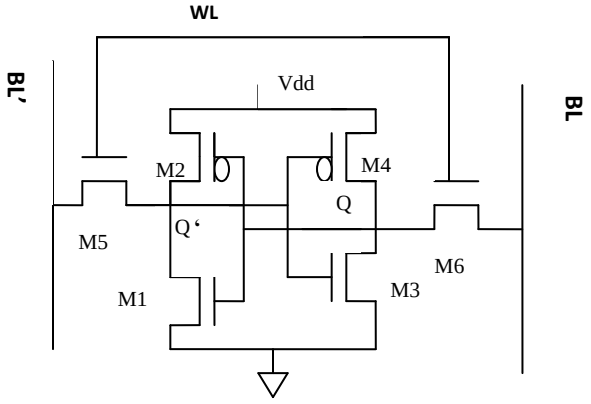


Fig 4: Conventional SRAM Cell

3.1 SRAM Cell Operation

An SRAM cell has three different states. It can be in: standby where the circuit is idle, reading when the data has been requested and writing when updating the contents. The three different states work as follows:

3.1.1 Standby

If the word line is not asserted, the access transistors M_5 and M_6 disconnect the cell from the bit lines. The two cross coupled inverters formed by $M_1 – M_4$ will continue to reinforce each other as long as they are connected to the supply.

3.1.2 Reading

Assume that the content of the memory is 1, stored at Q. The read cycle is started by pre-charging both the bit lines to a logical 1, then asserting the word line WL, enabling both the access transistors. The second step occurs when the values stored in Q and Q' are transferred to the bit lines by leaving BL at its pre-charged value and discharging BL' through M_1 and M_5 to a logical 0. On the BL side, the transistors M_4 and M_6 pull the bit line toward VDD, a logical 1. If the content of the memory were a 0, the opposite would happen and BL' would be pulled toward 1 and BL toward 0. Then these BL and BL' will have a small difference of delta between them and then these lines reach a sense amplifier, which will sense which line has higher voltage and thus will tell whether there was 1 stored or 0. The higher the sensitivity of sense amplifier is faster is the speed of read operation of SRAM.

3.1.3 Writing

The start of a write cycle begins by applying the value to be written to the bit lines. If we wish to write a 0, we would apply a 0 to the bit lines, i.e. setting BL' to 1 and BL to 0. This is similar to applying a reset pulse to a SR-latch, which causes the flip flop to change state. A 1 is written by inverting the values of the bit lines. WL is then asserted and the value that is to be stored is latched in. Note that the reason this works is that the bit line input-drivers are designed to be much stronger than the relatively weak transistors in the cell itself, so that they can easily override the previous state of the cross-coupled inverters. Careful sizing of the transistors in an SRAM cell is needed to ensure proper operation.

4. SLEEPY STACK SRAM CELL

Sleepy stack SRAM Cell: The Sleepy stack technique is applied to a 6-T SRAM cell by replacing each transistor with a combination of three transistors and using high V_{th} for specific transistors. The PU, PD sleepy stack can suppress the majority of the cell leakage while wordline sleepy stack can suppress the bitline leakage [5].

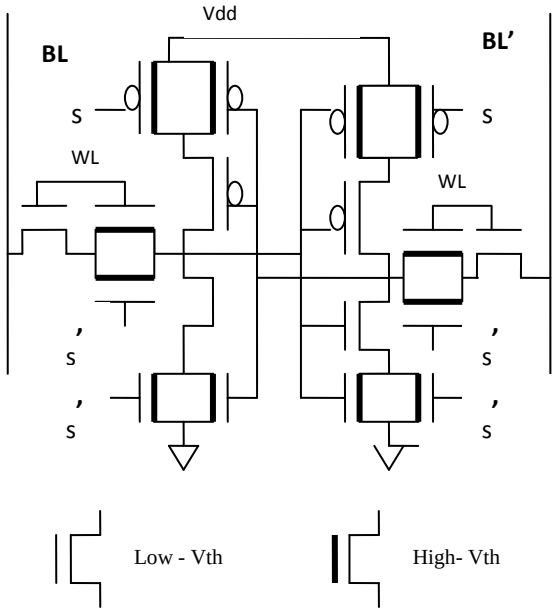


Fig 5: Sleepy stack SRAM cell

5. SIMULATION

To evaluate the sleepy stack SRAM cell, we compare our technique by using high-Vth transistors as direct replacements for low-Vth transistors with the conventional SRAM cell. We estimate static power, dynamic power and write time of both SRAM cell designs using SPICE: Simulation Program with Integrated Circuit Emphasis. The background knowledge for the simulations is obtained from reference [5], which emphasized on read operation of SRAM. In this paper, the write operation of SRAM cell is performed. The impact of temperature and voltage on the performance of sleepy stack designs is also analyzed. HSPICE tools provide an environment for the same. The estimation of power and delay of the two approaches of design is done using BSIM for 0.18u technology across PVT conditions.

The write operation is illustrated using Awaves in the HSPICE tool. Berkeley Predictive Technology Model (BPTM), level 49 targeting 0.18µm technology is used [3].

6. SIMULATION RESULTS

6.1 Performance Analysis across Process Corners

Table 2. Process corners for 180nm Technology

Process Corners	Voltage (v)	Temperature
Slow	1.89	5^{o} C
Typical	1.8	25^{o} C
Fast	1.71	110^{o} C

Absolute power consumption numbers at 110^{o}C show about 6.12X and 1.73X increase of leakage power consumption compared to the results at 25^{o}C for the base case and sleepy stack respectively. Absolute power consumption numbers at 25^{o}C show about 2.06X and 4.38X increase of leakage power consumption compared to the results at 5^{o} C for the base case and sleepy stack respectively. At 25^{o}C the static power consumption reduces by 77.21% of the static power of the base case SRAM cell, in 0.18µm technology according to the simulations.

Table 3. Static power across PVT Corners

Design Cases	Static Power (w)		
	Slow	Typical	Fast
Base case	1.183E-09	2.435E-09	5.479E-08
Sleepy Stack	6.915E-13	4.100E-11	2.613E-10

Dynamic power consumption across slow corner is the highest and is 1.30X of the typical value for base case .Absolute power consumption numbers at 25oC show about 1.130X increase of dynamic power consumption compared to the results at 110oC for the base case. At 25oC the dynamic power consumption of sleepy stack reduces by 30.03% of the dynamic power of the base case.

Table 4. Dynamic power across PVT Corners

Design Cases	Dynamic Power (w)		
	Slow	Typical	Fast
Base case	3.315E-07	2.534E-07	2.239E-07
Sleepy Stack	2.088E-07	1.773E-07	1.565E-07

Delay is highest in fast condition and lowest in the slow condition. Delay from the compared to the typical condition, has increased by .03% in fast corner and decreased by .03% in the slow corner. Across all the PVT corners, the sleepy stack shows slight increase of about 0 .1% in delay from the base case.

Table 5. Propagation Delay across PVT Corners

Design Cases	Propagation Delay (sec)		
	Slow	Typical	Fast
Base case	2.993E-08	2.994E-08	2.995E-08
Sleepy Stack	2.996E-08	2.997E-08	2.998E-08

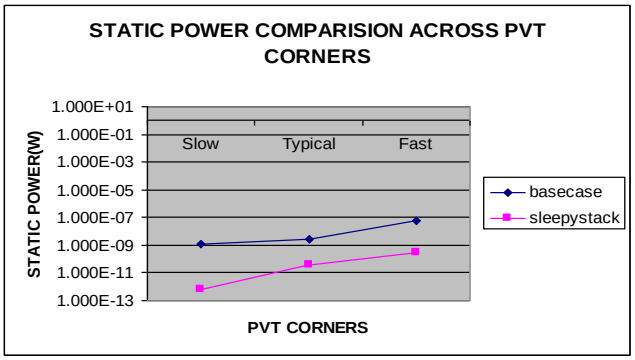


Fig 6: Static power comparison graph across PVT corners for SRAM

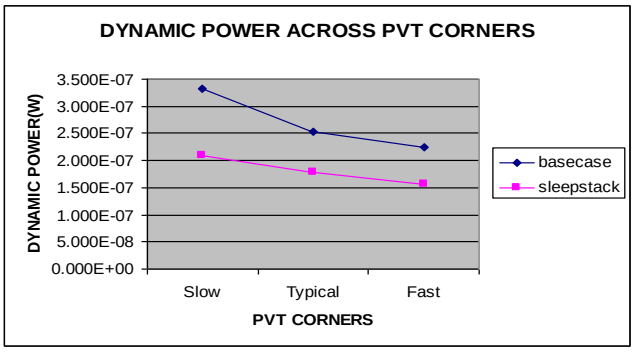


Fig 7: Dynamic power comparison graph across PVT corners for SRAM

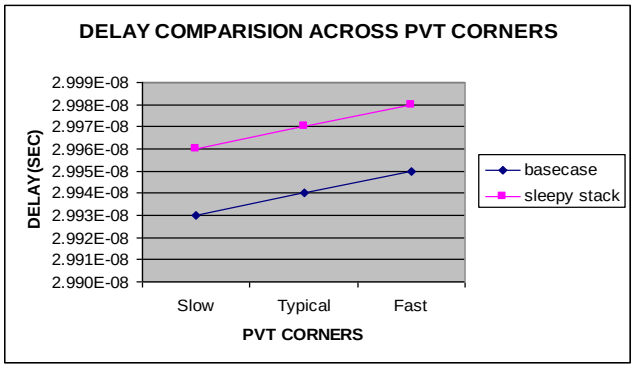


Fig 8: Delay comparison graph across PVT corners for SRAM

6.2 Impact of voltage variations

The simulation results obtained for static, dynamic and delay of 6T SRAM cells are tabulated in Table 6, 7, 8. The saturation current of a cell depends on the power supply. The delay of a cell is dependent on the saturation current. In this way, the power supply affects the propagation delay of a cell. Throughout a chip, the power supply is not constant and hence the propagation delay varies in a chip. The voltage drop is due to nonzero resistance in the supply wires. A higher voltage makes a cell faster and hence the propagation delay is reduced. The decrease is exponential for a wide voltage range.

Table 6. Impact of voltage on static power

VDD (v)	Static Power (w)	
	Base case	Sleepy stack
1.620E+00	1.635E-09	3.203E-11
1.800E+00	2.435E-09	4.100E-11
1.980E+00	3.623E-09	4.946E-11

Table 7. Impact of voltage on dynamic power

VDD (v)	Dynamic Power (w)	
	Base case	Sleepy stack
1.620E+00	1.907E-07	1.446E-07
1.800E+00	2.534E-07	1.773E-07
1.980E+00	3.263E-07	2.184E-07

Table 8. Impact of voltage on write delay

VDD (v)	Propagation Delay (sec)	
	Base case	Sleepy stack
1.620E+00	2.995E-08	2.998E-08
1.800E+00	2.994E-08	2.997E-08
1.980E+00	2.993E-08	2.996E-08

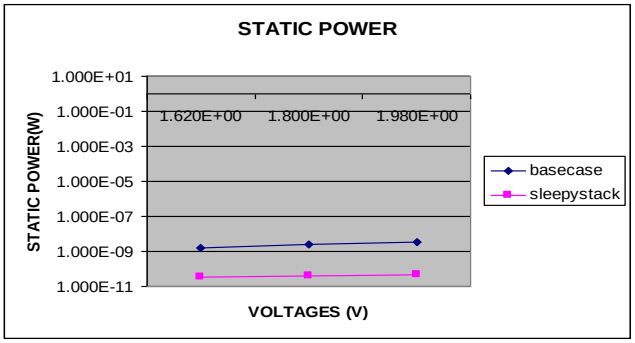


Fig 9: Impact of voltage on static power

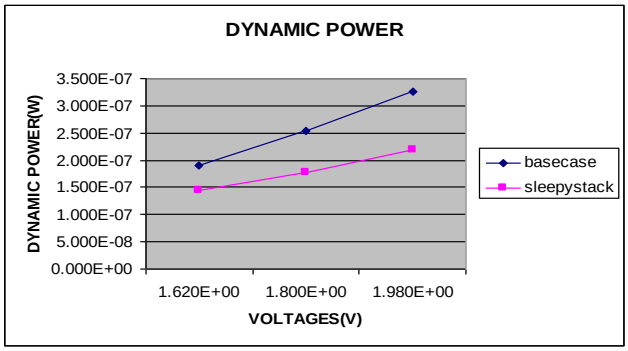


Fig 10: Impact of voltage on dynamic power

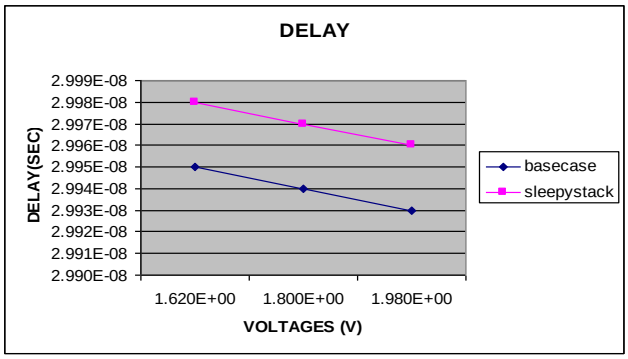


Fig 11: Impact of voltage on write delay

6.3 Impact of Operating Temperature Variations

When a chip is operating, the temperature can vary throughout the chip. This is due to the power dissipation in the MOS-transistors. A higher temperature will decrease the threshold voltage. A lower threshold voltage means a higher off state current. Thus static power of the design increases with temperature.

Table 9. Impact of operating temperature on static power

Temperature (^{o} C)	Static Power (w)	
	Base case	Sleepy stack
-5.000E+00	4.167E-10	3.885E-13
0.000E+00	5.746E-10	4.801E-13
2.500E+01	2.435E-09	4.100E-11
5.000E+01	8.220E-09	4.100E-11
1.000E+02	5.624E-08	2.340E-10

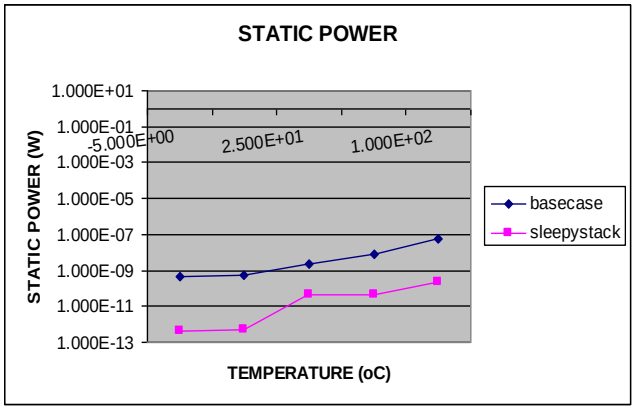


Fig 12: Impact of operating temperature on static power

7. CONCLUSION

In nanometer scale technology, subthreshold leakage power is a great challenge. None of the leakage reduction approaches shows best results in all criteria. Designers need to choose an appropriate technique for a given technology and particular chip. Sleepy stack approach presents a new weapon to VLSI designers. It can attain ultra low static power at the cost of increased area and delay penalty.

8. REFERENCES

[1] J. Park, 2005, "Sleepy Stack: a New Approach to Low Power VLSI and Memory", Ph.D. dissertation, School of Electrical and Computer Engineering, Georgia Institute of Technology.

[2] S. Narendra, V. D. S. Borkar, D. Antoniadis, "Scaling of Stack Effect and its Application for Leakage Reduction,"

[3] Berkeley Predictive Technology Model (BPTM). [Online].

Available http://www-device.eecs.berkeley.edu/˜ptm/.

[4] *"International Technology Roadmap for Semiconductors," Semiconductor* Industry Association, 2007. [Online]. Available: http://public.itrs.net

[5] Jun Cheol Park and Vincent J. Mooney III ,"Pareto Points in SRAM Design Using the Sleepy Stack Approach", School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta

[6] S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S. Shigematsu, and J. Yamada, "1-V power supply high-speed digital circuit technology with multithreshold-voltage CMOS".

[7] N. S. Kim, T. Austin, D. Baauw, T. Mudge, K. Flautner, J. Hu, M. Irwin, M. Kandemir, and V. Narayanan, "Leakage Current: Moore's Law Meets Static Power," *IEEE Computer*, vol. 36, pp.68–75, December 2003.