

# Modeling and Evaluation of Multi-Bank SRAM Design for Leakage Power Reduction

Byunghyun Jang and Yong-Bin Kim  
Department of Electrical and Computer Engineering  
Northeastern University, Boston, MA 02115  
Email: {bjang,ybk}@ece.neu.edu

**Abstract**—In this paper, the modeling and evaluation of multi-bank SRAM design with dynamic threshold and supply voltage control is presented to reduce leakage power. The bank of SRAM, the unit of control, is put in sleep mode (high threshold voltage and low supply voltage) from active mode (low threshold voltage and high supply voltage) whenever it is not frequently used. The change of modes is based on the characteristics of the temporal and spatial locality of memory accesses. The simulation results show that significant leakage reduction can be achieved through combined implementation of spatial locality and temporal locality while minimizing the re-synchronization penalties when the size of superbank is optimized based on the characteristics of application program.

## I. INTRODUCTION

As process technology advances under 100nm, leakage power consumption increases at much faster rate than dynamic power and is expected to dominate total power consumption. Leakage power is a big issue in memory design where large portions of memory is just waiting for activation while holding data, causing significant amounts of energy to leak. This problem increases as memory arrays increase in size with time.

Typical applications running on any computer system inherently exhibit a significant degree of locality to memory accesses [5]. Locality of memory access is targeted by many researchers for performance and energy optimization. There are two kinds of locality: *temporal* and *spatial*. Having been aware of its importance, there are many software techniques to increase the locality of memory access patterns [1][3][4] but unfortunately there are few hardware approaches.

This paper proposes multi-bank dynamic threshold and supply voltage SRAM (DTSSRAM) as a means of leakage power reduction. Especially, we propose DTSSRAM design utilizing temporal and spatial locality. In this paper, its pros and cons are discussed in terms of performance metrics: *leakage energy consumption* and *re-synchronization cost*. The intention of the proposed approach is to achieve significant leakage reduction through combined implementation of spatial locality and temporal locality along with dynamic threshold and supply voltage control technique. The re-synchronization penalties can be minimized if the size of superbank(the control unit) is optimized based on the characteristics of the application program.

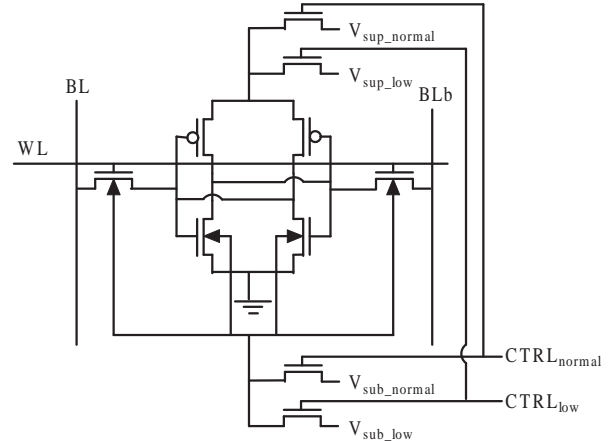


Fig. 1. Schematic of Dynamic Threshold & Supply Voltage SRAM Cell

## II. MEMORY ARCHITECTURE AND ITS LEAKAGE REDUCTION

Figure 1 shows the schematic of a dynamic threshold voltage and supply voltage SRAM (DTSSRAM). Both the substrate of NMOS and source of PMOS are connected to the control transistors. Since most leakage paths are through NMOS, significant leakage reduction can be achieved through substrate biasing of NMOS [2]. When the block of SRAM is not used its substrate bias voltage decreases to a negative voltage and supply voltage decreases to reduce the leakage power(sleep state) but returns to the normal to maximize the performance when it is invoked to be accessed(active state). The states are changed with control signals through a pass transistor. There exists a significant penalty when changing states from sleep to active due to the delay charging all capacitances and the time required for stability.

In order to implement temporal locality a capacitor discharging scheme is used as shown in Figures 2, where some amount of charge is stored whenever the block of memory array is accessed and checks its level to decide when to transit to sleep mode. This scheme is effective because it eliminates the overhead caused by the frequent state transitions between active and sleep. It is shown that the energy required for one state transition is much larger than the leakage saved during one clock cycle [2]. Therefore, special care should be taken to minimize the overhead while maximizing the energy save.

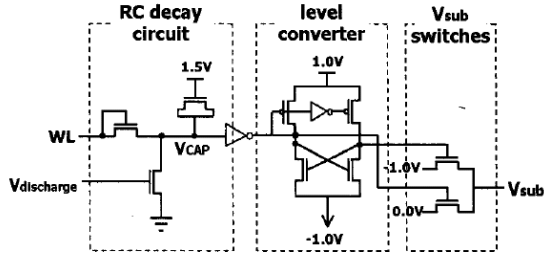


Fig. 2. Schematic of bias control circuit for temporal locality [4]

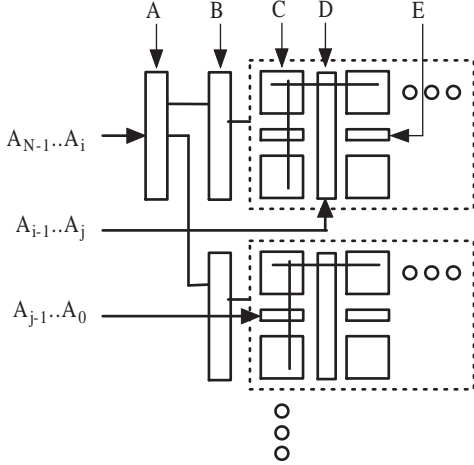


Fig. 3. Simplified Block Diagram of Multi-Bank SRAM: (A) Superbank predecoder, (B) Biasing Control Unit, (C) Memory bank, (D) Row decoder, (E) Column decoder

In order to implement spatial locality a multi-bank memory system instead of a monolithic memory system is adopted. In multi-bank memory systems, each bank can be controlled separately. The goal of the implementation of spatial locality is to limit memory access to as few banks as possible so that the rest of the banks can remain in sleep mode (power saving mode). Figure 3 shows the simplified block diagram of multi-bank SRAM. Multiple banks are grouped together to form *superbank*.

### III. SIMULATION AND EVALUATION

The simulation is performed extensively using high level C++ language with its hardware information extracted from SPICE simulation. The result shows that as decay time increases the leakage power increases while re-synchronization cost decreases (effect of temporal locality). It also shows that the effect of spatial locality is strongly dependent on the program size and superbank size as well as the characteristics of program's memory access pattern (effect of spatial locality).

Table I shows the summary of leakage energy consumption and re-synchronization cost per clock cycle in three cases: temporal only, spatial only, and both case. As shown in the table, when only temporal implementation is used, its leakage energy consumption is 298% more than the case where both

TABLE I

LEAKAGE ENERGY CONSUMPTION AND RE-SYNCHRONIZATION COSTS PER CLOCK CYCLE: TEMPORAL ONLY, SPATIAL ONLY, AND BOTH EMPLOYED

Metrics	Leakage Energy	Re-synchronization Cost
Temporal only	107.086598	0.001123
Spatial only	29.295742	0.049964
Both employed	35.913858	0.000548

temporal and spatial implementations are employed. The re-synchronization cost is also 205% more than the other cases. Spatial only case, by the way, has lower leakage energy consumption but its re-synchronization cost (9118% more than both cases) is not tolerable. It can be easily understood that the case without temporal implementation (zero decay time) lets the superbank change its state instantly whenever needed, resulting in minimum leakage power consumption while its re-synchronization cost reaches maximum due to too frequent state changes. Consequently, it is demonstrated that employing both temporal and spatial implementation in multi-bank SRAM design after characterization of program behavior, especially with dynamic threshold and supply voltage technique is very desirable for leakage power reduction while minimizing delay overhead.

### IV. CONCLUSION

In this work, we proposed multi-bank dynamic threshold and supply voltage SRAM (DTSSRAM) for leakage power reduction. The contribution of this work is that this work initiated a basis for leakage aware SRAM design considering the application program behavior. The proposed method can be applied at early power estimation and memory system architecture design stage, especially in embedded system where the number of application programs is limited. Along with software driven locality enhancement techniques proposed in pervious literatures, the proposed SRAM design technique can reduce leakage power synergistically.

### REFERENCES

- [1] M. Kandemir, J. Ramanujam, M.J. Irwin, N. Vijaykrishnan, I. Kadayif, and A. Parikh, *Dynamic management of scratch-pad memory space*, Design Automation Conference, 2001. Proceedings 2001 Page(s):690 - 695
- [2] C.H. Kim and K. Roy, *Dynamic  $V_t$  SRAM: a leakage tolerant cache memory for low voltage microprocessors*, Low Power Electronics and Design, 2002. ISLPED '02. Proceedings of the 2002 International Symposium on 2002 Page(s):251 - 254
- [3] A. Macii, E. Macii, and M. Poncino, *Increasing the locality of memory access patterns by low-overhead hardware address relocation*, Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on Volume 5, 25-28 May 2003 Page(s):V-385 - V-388 vol.5
- [4] V. De La Luz, M. Kandemir, and I. Kolcu, *Automatic data migration for reducing energy consumption in multi-bank memory systems*, Design Automation Conference, 2002. Proceedings. 39th 10-14 June 2002 Page(s):213 - 218
- [5] D. A. Patterson and J. L. Hennessy, *Computer architecture a quantitative approach*, Morgan Kaufmann Publishers, 1996