Drowsy Cache Implementation Using 5T Port-less SRAM Cell

Lei Qiao, Jiasheng Shi, Min Xu
Department of Electrical and Computer Engineering
University of Rochester

***Abstract*—With the decrease of the transistors' feature size, the leakage power of the integrated circuits has become a more and more significant issue. Drowsy cache [1], one of the modern techniques used to limit the power consumption of memory circuits, provides an easy way to achieve the goal through some modifications on the architecture level. The newly introduced five-transistor (5T) port-less SRAM cell [2][3] seems to be attractive since its small area occupation, low power consumption and the same or even better performance comparing to the conventional 6T SRAM cell. It would be interesting to see the application of this 5T cell to the modern memory circuit techniques. In this paper, we present a simple implementation of drowsy cache using the 5T port-less SRAM cell with 90nm technology, including the organization of the cell array and the peripheral circuits such as the voltage switcher, drowsy set circuit, the decoders, the current sense amplifier and so on.**

I. INTRODUCTION

The most significant difference of the 5T cell compared to the conventional ones is that there is no port transistors to access the cell. Therefore, the bit-line and power-line of the cell array are merged together. As a result, we save the area for every single cell as well as introduce much less metal lines. The basic structure is shown in Fig.1 [2][3].

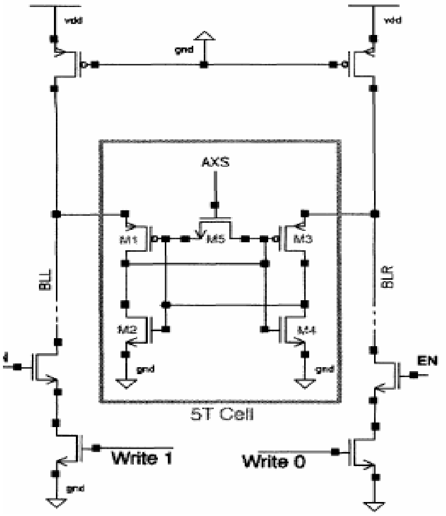

Fig. 1 Basic structure of 5T SRAM cell

The main idea of the standard drowsy cache [1] is to periodically switch the whole cache into drowsy mode, which keeps the data safe under a much lower supply voltage for the purpose of reducing the leakage power of the inactive cells. It works at the granularity of every single cache line; say the smallest unit of switching between normal mode and drowsy mode is a cache line. Every time a cell is accessed, it is necessary to identify whether it is in drowsy mode. If so, the cell has to be waken up first, otherwise, it can be accessed the same way as before.

The rest of the paper is organized as follows: In section 2, the new architecture design of the drowsy cache especially for the 5T cell structure is introduced. Section 3 presents the basic analysis of the 5T cell and the determination of the drowsy mode voltage considering the stability and leakage power reduction. The wake-up penalty is also discussed in this section. In section 4, the drowsy control circuits and read circuits are introduced to meet the requirement of this drowsy cache design. We conclude and discuss the current and future work of the drowsy cache design based on 5T cell structure.

II. ARCHITECTURE LEVEL DESIGN

Since the bit-line and power-line are merged together in the new 5T design, this property prevents us from implementing the drowsy cache in a fine-grain fashion such as at the granularity of a cache line. One remedy for this is to change the arrangement of the data from a horizontal way to a vertical way so as to adapt the fine-grain fashion; however, doing this means give up the potential ability to access several bit-lines at the same time.

Therefore, the drowsy cache is implemented in a coarse-grain way by dividing the cell array into several blocks and the voltage switching works at the granularity of one block. The coarse-grain implementation at the same time provide us the opportunity to keep track of the cache behaviors of every block respectively while it is not practical to trace the cache behavior of every cache line regarding the fine-grain implementation.

Another advantage of 5T cell structure is to solve the leakage problem on the bit-line without extra efforts because the power line and bit-line are merged. The main architecture of the design is shown in Fig.2.

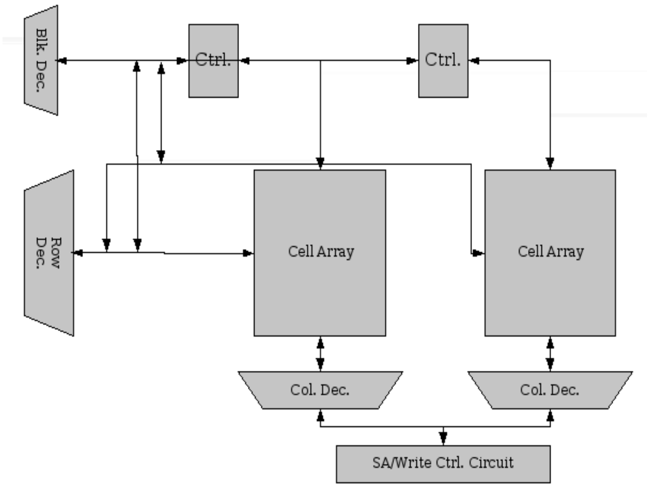

Fig.2 Main Architecture of Drowsy Cache

We take advantage of the existed multi-block cache architecture and add one control circuit for every cache block to trace the behaviors and switch between drowsy and normal mode. Currently, we are simulating a 16x16 cell array divided into four 4x16 blocks since we use a 4-bit word in the design.

III. ANALYSIS OF CELL AND DETERMINATION OF DROWSY VOLTAGE

In the 5T cell design, the SNM of the cell is strongly dependent on the length of AXS transistor [3]. We assigned the AXS transistor to be 800nm long based on the testing results to ensure a safe SNM for the cell.

Speaking of the determination of drowsy voltage, we mainly take two factors into account, the data stability and leakage power consumption. The SNM is measured by DC and AC analysis based on the loop gain model [4] for different supply voltage, and the SNM of over 10% of the Vdd is obtained when power supply voltage is 0.3V (1V for active mode), which is shown in Fig. 3.

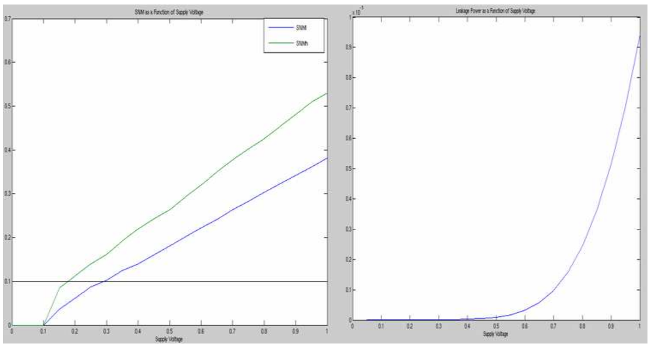

Fig. 3 SNM and leakage power as the function of supply voltage

Another observation is that leakage power reduces to 0.1% of that in the active mode when the supply voltage drops below 0.35V (Fig. 3). Therefore, we decide to let the drowsy voltage be 0.3V.

Since the drowsy cache design is a trade off between the system performance and the leakage power reduction, the wake-up delay (or penalty) is a critical parameter to consider the wake-up policy [1]. We do the simulation by writing a '1' firstly, then drop the voltage of both bit-line to drowsy mode voltage and then wake the cell up to the active mode, shown in Fig. 4.

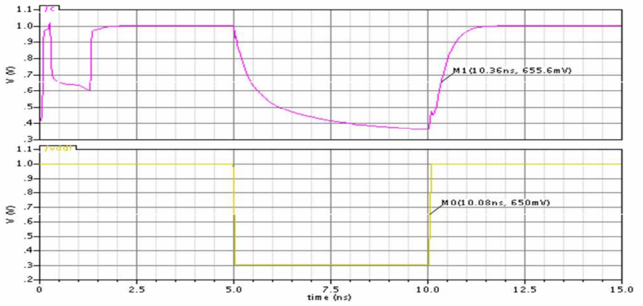

Fig. 4 Basic cell operation and wake-up penalty test

IV. PERIPHERAL CIRCUITS DESIGN

Most of the peripheral circuits are conventional ones including the NAND-based row decoder with divided word-line decoding [7], tree-based column decoder [7], the write control circuit and the address transition detection circuit [8]. In this section, we mainly focus on the current sense amplifier and the drowsy control circuit.

A. *Read Circuit*

The drawback of the conventional voltage sense amplifier is large bit-line capacitance. This results in large delay, low speed, large power consumption and small output signal swing. In addition, due to the characteristic of 5T port-less SRAM cell, the bit-line voltage differential is a function of the SRAM cell current, the bit-line capacitance and the pre-charge transistor width. Therefore, a current sense amplifier rather than a conventional voltage-based sense amplifier are employed which provides high performance and low power consumption.

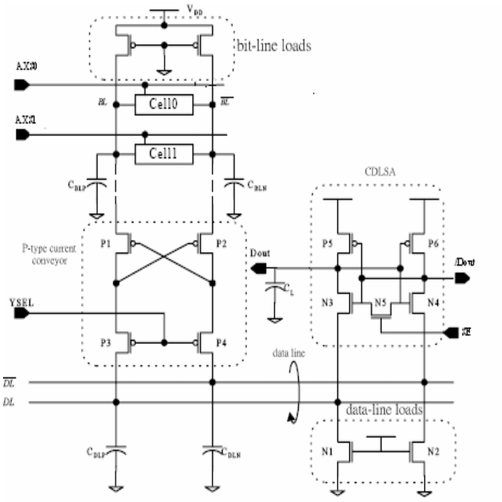

Fig. 5 Read circuit of current-mode 5T port-less SRAM

We implemented a hybrid current sense amplifier, a combination of a current conveyor and current sense amplifier, shown in Fig. 5 [5]. The sensing delay is relatively insensitive to the bit-line capacitance since no differential discharging of the large bit-line capacitance is needed to sense the cell data. Therefore, this scheme increases the speed of the sense amplifier further due to the fact that there is a flow of bias current before the sense-amplifier is actually enabled [6].

B. Drowsy Control Circuit

One complicated part of the drowsy cache is its drowsy policy, namely when to put which cell into drowsy mode. This policy can be either done in software or hardware and either simple or complex. The standard drowsy cache put all the cells into drowsy mode periodically according to the CPU cycles by setting the drowsy bit of the cache lines [1]. This method may need the assistance from the software part and some modification to the MMU. In our design, we implement a counter with the input from CPU cycle as CLK and the output of block decoder as RESET. In other words, when N CPU cycles passed and the corresponding block is never accessed, the counter can trigger one pulse signal to set the block into drowsy mode, where N is exactly the upper limit of the counter. Implementing this in hardware makes the drowsy cache usable in the system without any modification to other parts or in the software. The area overhead introduced, however, may be an issue. It is not as flexible as to do this in software at the same time.

It is essential to keep the data safe when the cell is in drowsy mode, so the access to a drowsy cell should be delayed for a while until it is waken up and safe to be accessed. The circuit for this protection is shown in Fig. 6 along with the switch circuit to change the supply voltage between 1V and 0.3V. The circuit is very similar to the published one [1] while we made some corrections and change the gating part from only an AND gate to a RC delay circuit. The main idea to do this change is we doubt that just eliminating the part of the access signal when the cell is in drowsy mode is not enough. Even if the power supply is changed to 1V, the bit-line still needs time to pre-charge and the cell needs even more time to restore the data. Therefore, we consider a delay circuit here could be safer.

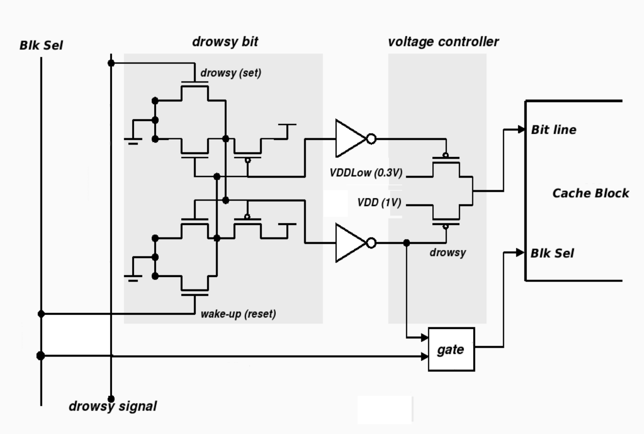

Fig. 6 Voltage switching and gating circuits

V. CONCLUSIONS

A different way of drowsy cache implementation at circuit level has been presented. Comparing to the original implementation, this method adapts a coarse-grain fashion and chooses to get most implementation done in the hardware, without modifications to other parts of the system. Unfortunately, we are not able to provide much detailed performance data here as there are still some issues when getting the whole circuitry to work properly although we have simulated the R/W and switching operations on one single cell as well as verified the correct function of every peripheral circuit implemented. In addition to collecting these data, there are a lot of things can be done in the continuous work. Our main targets here include reporting the performance data based on the simulation of SPEC benchmarks and comparing with the standard design if possible; improving the peripheral circuits either for the sake of high performance or low power; searching a way to better use the electrons discharged when switching from active mode to drowsy mode.

ACKNOWLEDGMENT

Thanks for the great help from Michael Wieckowski, John Liobe and Professor Martin Margala during the whole process of our design.

REFERENCES

[1] K. Flautner, et. al. “Drowsy Caches: Simple Techniques for Reducing Leakage Power,” *Proceedings of 2002 Annual International Symposium on Computer Architecture*, pp.148-157, Jun 2002.
[2] M. Wieckowski and M. Margala, “A Novel Five-Transistor (5T) SRAM Cell for High Performance Cache,” *Proceedings of 2005 IEEE International SOC Conference*, pp.25-28, Sep 2005.
[3] M. Wieckowski and M. Margala, “Portless SRAM – A High-Performance Replacement for the 6T Methodology,” 2007 (to be published)
[4] E. Seevinck, F.J. List, and J. Lohstroh, “Static Noise Margin Analysis of MOS SRAM Cells,” *IEEE Journal of Solid-State Circuits*, vol.22, pp.748-754, Oct 1987.
[5] J. Wang, P. H. Yang, and W. Tseng, “Low-Power Embedded SRAM Macros with Current-Mode Read/Write Operations,” *Proceedings of 1998 International Symposium on Low Power Electronics and Design*, pp.282 – 287, Aug 1998.
[6] M. Sinha et. al. “High-Performance and Low-Voltage Sense-Amplifier Techniques for sub-90nm SRAM,” *Proceeding of 2003 IEEE International SOC Conference*, pp.113-116, Sep 2003.
[7] J. M. Rabaey, A. Chandrakasan, B. Nikolic, “Digital Intergrated Circuits: A Design Perspective”, 2003, pp. 672 – 679
[8] A. Bellaouar and M. I. Elmasry, “Low Power Digital VLSI Design: Circuits and Systems,” 1995, pp.332-337.