Advanced Engineering Forum *Online: 2012-09-26*
ISSN: 2234-991X, Vols. 6-7, pp 20-25
doi:10.4028/www.scientific.net/AEF.6-7.20

A Data Drivered Refresh with Multi-bit Error-Correcting Power Optimize Method for Cache Based eDRAM

Guo Yufeng[1, a]

[1] School of Computer Science, National University of Defense Technology

[a]yfguo21@yahoo.com.cn

Keywords: eDRAM; Refresh Power; Multi-bit Error-Correcting; Power;

Abstract. Power problem has been one of most restricting the development barriers of processor. With enhancing of computer performance, there must be large cache to hide memory latency. Large cache can be consisted on one chip based eDRAM which has high density. Unfortunately, eDRAM must be refreshed frequently to maintain data, which would increase cache power. The paper aims at refresh problem of eDRAM, and put forwards a data drivered refresh with multi-bit error-correcting power optimize method. The experimental results show that the method which we put forward can greatly reduce the refresh power of eDRAM.

Introduction

With the increasing of single chip's transistor number and clock frequency, power density of chip raised following with the power Moore law. Power problem has become more and more grimness. On the other hand, more and more cache has been integrated to hide memory latency to solve memory wall problem. Unfortunately, traditional SRAM has low density, and it's very difficult to implement.

Embedded DRAM(eDRAM) memory density increases three to four times compared to SRAM[1]. With the same chip size, much larger cache can be incorporated if use eDRAM, but eDRAM must be periodically refreshed to retain data. And because fast logic transistors has been used, eDRAM has higher leakage current than traditional DRAM, refresh time is thousand times than DRAM[2]. The variations in threshold will cause retention times vary significantly [2,4].

Refreshing paid important contribution to the power of eDRAM, because refresh power cannot be reduced even there has no access to cache or system has been entered into low power mode. As figure1 shown, pfbit curve represents the probability of a retention failure in a single bit cell and pfCache curve represents the failure probability of a 128MB eDRAM cache for different refresh times[5]. The probability of data loss would increase with refresh frequency descending, but high refresh frequency would increase refresh power. Chris Wikerson[6] found a processor with 128MB of eDRAM cache would consume 926mW just refreshing the eDRAM. So reduce refresh power of eDRAM is an important method to reduce the cache power.

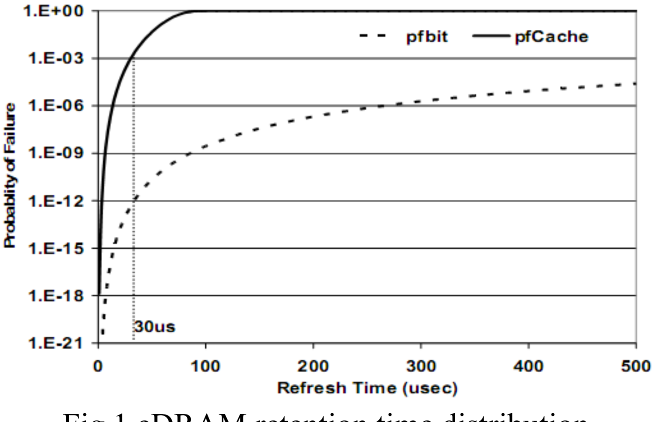

Fig.1 eDRAM retention time distribution

The paper aims at the refresh problem of eDRAM, try to reduce power and keep data reliability. And put forwards a data drivered refresh with multi-bit error-correcting power optimize method.

Data Drivered Refresh with Multi-bit Error-correcting Power Optimize Method

To reduce refresh power of eDRAM and enhance reliability of cache data, we put forwards a data drivered refresh with multi-bit error-correcting power optimize method DDRMC(Data Drivered Refresh with Multi-bit Error-Correcting).

Motivation: we divided data cache array into many sub-array, and each sub-array can be refreshed independently. Only when data of sub-array is valid, refresh would be started, otherwise refresh would be stopped to reduce refresh power. At the same time, to keep data reliability, multi-bit error-correcting code has been used, so even if there are several data error in the cache, data will not be lost. Refresh time can be changed according to the data failure probability. Because cache tag array is very small, and would be regularity accessed, so we used traditional SRAM.

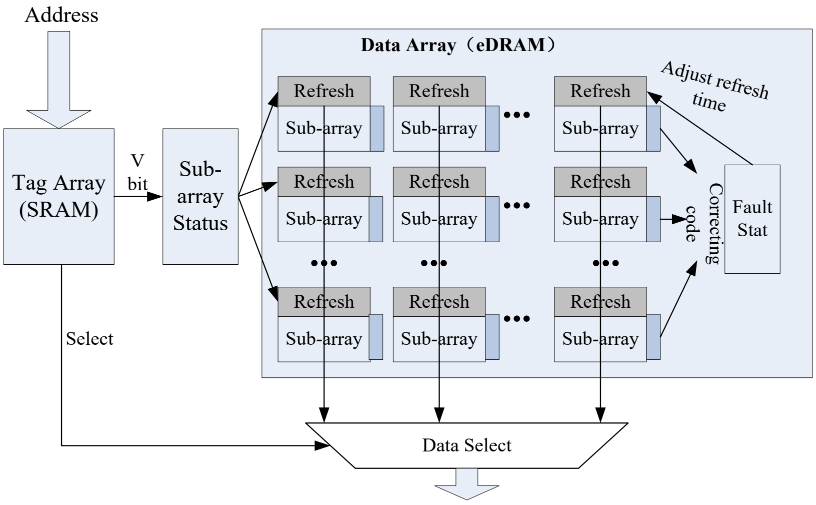

Fig.2 Organization of Cache in DDRMC

Figure2 showed the cache organization of DDRMC. Cache has been divided into many sub-array, each sub-array has refresh control logic. Only when sub-array has valid data(valid bit has been set), the refresh would be started. The correcting code also is saved in the sub-array to check data error and correct error. R.Venkatesan[7] researches show the data retention time is different between different eDRAM cell, and as figure3 showed the difference may be very large. So according the data retention time to set refresh time can enlarge the refresh time of many cell.

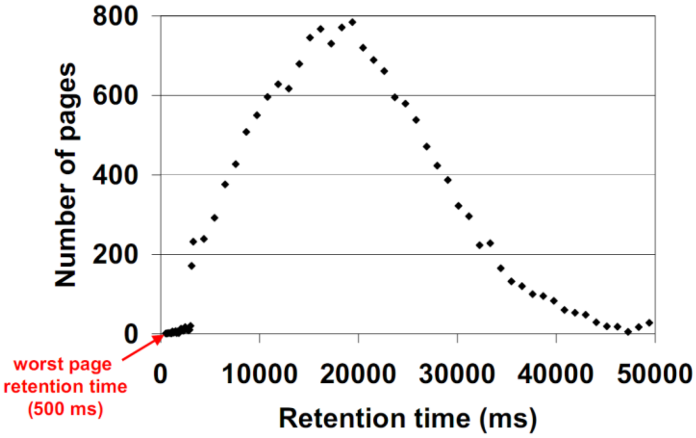

Fig.3 Distribution of different cell retention time

Figure4 is the description of DDREMC arithmetic.

```
DDREMC（cur_addr）{
Input M； //number of data sub-array
Input Req； //Cache requests
For（i=0； i<M； i=i+1）//for each sub-array
Begin
  Ret=Find_Valid_Status（i）； //find sub-array I valid bit
  If（Ret）//data invalid
    Stop_Refresh（i）;//stop refresh
  Else Begin
    Find_bitfail_Info（i）； //find sub-array I failure information
    Adjust_Refresh_Time（i）； //adjust refresh time
    Start_Refresh（i）； //start refresh
  End
  If（Req，i）//hit
  Begin
    If（read）//read
    Begin
      Read_Data； //read data
      Detect_Data_Fail； //detect data error
      Update_bitfail_Info； //update failure information
    End
    Else Begin//Write Request
      Gen_Error_Correcting_Code； //generate correcting code
      Write_Data； //write data and correcting code
    End
  End
End
}
```

Fig.4 DDRMC Arithmetic

We used error-correcting code to enhance data reliability of eDRAM. Different error-correcting code has different correcting power and overhead, refresh time can be enlarged with strong error-correcting code. Figure5 shown how refresh frequency effect data failure rate with different correcting code. Figure6 showed the distribution of data failure with different correcting code.

DDRMC adopted SECDED code [6]. Although SECDED code refresh time is 30% of 5EC6ED code, but storage overhead is very small, for a 1KB line only has 0.18% overhead, and has very low dynamic power for coding/decoding.

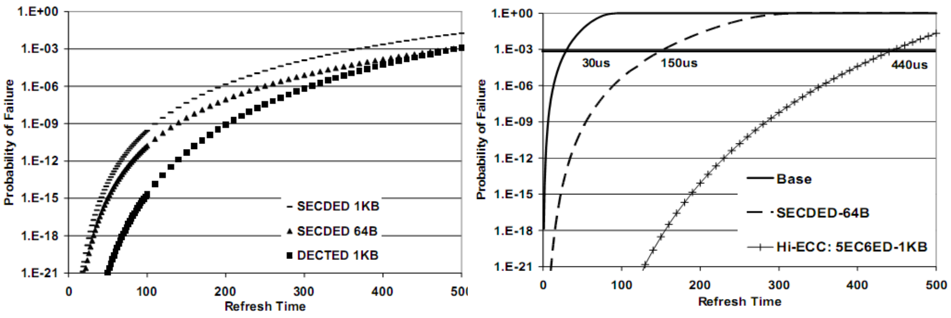

Fig.5 The distribution of data failure with different correcting code

Relative Work

J.Chang[8] used power-gate method to reduce large storage cell power, before enter low power state, store eDRAM data to nonvolatile memory, the overhead is very large, and would decrease performance and power. On 2010 ISCA conference, chirs[6] put forwarded a method to reduce refresh frequency and designed a multi-correcting code Hi-ECC to ensure data valid. Some previous papers used hardware mechanisms to exploit retention time variations by refreshing different DRAM cells at different refresh rates[9,10]. Another promising approach to increase DRAM refresh times is the use of error-correcting codes(ECC) to dynamically detect and correct bits that fail[11]. Venkatesan[12] proposed a software mechanism that allocates DRAM pages with longer retention time before allocating pages with shorter retention time. Ghosh[13] proposed a SmartRefresh technique to reduce refresh power by adding timeout counters in each DRAM row and avoiding unnecessary refreshes.

Experimental Results

In this section we will evaluate the impact of our DDRMC method on eDRAM cache power. Firstly, we stat how eDRAM refresh time distribution when adopted DDRMC method. As Figure6 shown, most refresh time focus on between 100us to 300us.

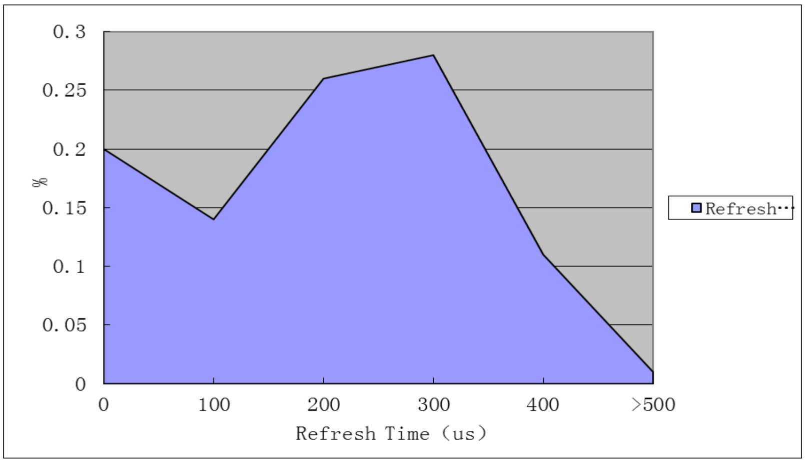

Fig.6 Refresh time distribution of DDRMC

Figure8 shown the refresh power and dynamic power of cache which adopted base strategy without error correcting code, DDRMC and 5EC6ED based on Hi-ECC, the loads are FTP server trace, WEB server trace and MAIL server trace. Dynamic power is the power brought by coding and decoding, for the Base strategy dynamic power is zero. eDRAM refresh power is about 926mW[6]. As Figure7 shown, we can found that the DDRMC and 5EC6ED method reduced refresh power evidently. Because DDRMC can adjust refresh time dynamically, so for different load the refresh power is not same. For MAIL server, because cache usage is very low, many sub-array could be stop refresh, so the refresh power is lower than 5EC6ED. On the other hand, DDREMC adopted SECDED code, overhead and dynamic power is much lower than 5EC6ED.

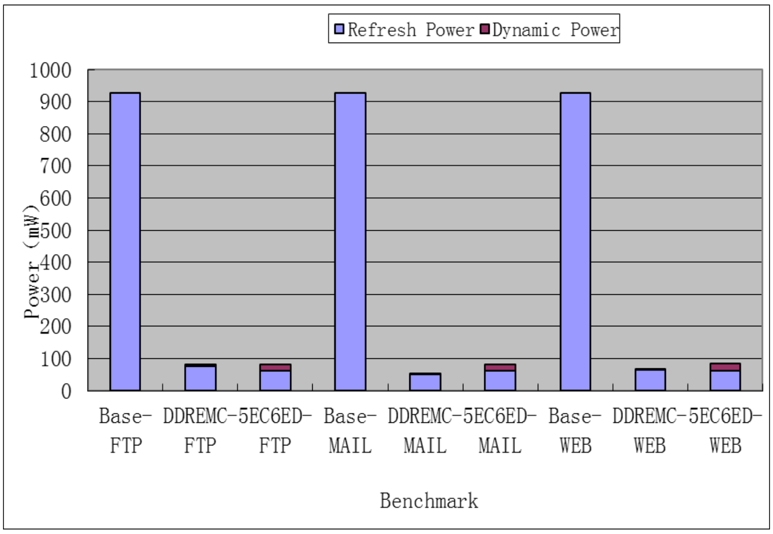

Fig.7 Power result of cache

Conclusion

eDRAM is significantly denser than the traditional SRAMs, which made it’s possible to integrate large on-die cache. Refresh increased cache power, power problem become more and more austerity. The paper adopted enhance multi-error correcting code to keep data reliability when reduce refresh time. Future work will focus on multi-error correcting code, and design a new code to reduce decoder complexity and latency.

References

[1] R. Matick, S. Schuster. Logic based eDRAM: origins and rationale for use [J]. IBM Journal of Research and Development, 2005, 49(1): 145~165.

[2] Inc Micron Technology. TN-41-01 : Calculating memory system power for DDR3. http://download.micron.com/pdf/ technotes/ddr3/TN41_01DDR3%20Power.pdf

[3] M. Ghosh, H. Lee. Smart refresh: An enhanced memory controller design for reducing energy in conventional and 3D die-stacked DRAMs[C]. the 40th International Symposium on Microarchitecture, 2007

[4] Mrinmoy Ghosh, Hsien-Hsin S. Lee. Smart refresh: An enhanced memory controller design for reducing energy in conventional and 3D die-stacked DRAMs[C]. 40th IEEE/ACM International Symposium on Microarchitecture, MICRO 2007, Chicago, IL, United states: Inst. of Elec. and Elec. Eng. Computer Society, 2007: 134~145

[5] W. Kong. Analysis of retention time distribution of embedded DRAM - A new method to characterize across-chip threshold voltage variation[C]. IEEE International Test Conference (ITC 2008), 2008: 1~7

[6] Chris Wilkerson, Alaa R. Alameldeen, Zeshan Chishti, Wei Wu,et al. Reducing cache power with low-cost, multi-bit error-correcting codes[C]. 37th International Symposium on Computer Architecture, ISCA 2010, Saint-Malo, France: Institute of Electrical and Electronics Engineers Inc., 2010: 83~93

[7] R. Venkatesan, S. Herr, E. Rotenberg. Retention aware placement in DRAM (RAPID): Software methods for quasi-non-volatile DRAM[C]. 12th International Symposium on High Performance Computer Architecture (HPCA), 2006: 155~165

[8] Jonathan Chang, Ming Huang, Jonathan Shoemaker, et al. The 65-nm 16-MB shared on-die L3 cache for the Dual-Core Intel Xeon Processor 7100 Series [J]. IEEE Journal of Solid-State Circuits, 2007, 42(Compendex): 846~852

[9] J. Kim, M. Papaefthymiou. Block-based multiperiod dynamic memory design for low data-retention power[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2003, 11(6): 1006~1018.

[10] T. Ohsawa, K. Kai, K. Murakami. Optimizing the DRAM refresh count for merged DRAM/Logic LSIs [C]. Proceedings of the 1998 International Symposium on Low Power Electronics and Design (ISLPED), 1998: 82~87.

[11] P. Emma, W. Reohr, M. Meterelliyoz. Rethinking refresh: Increasing availability and reducing power in DRAM for cache applications [J]. Micro IEEE, 2008, 28(6): 47~56.

[12] Ravi K. Venkatesan, Stephen Herr, Eric Rotenberg. Retention-Aware Placement in DRAM (RAPID): Software methods for quasi-non-volatile DRAM[C]. Twelfth International Symposium on High-Performance Computer Architecture, Austin, TX, United states: Institute of Electrical and Electronics Engineers Computer Society, 2006: 157~167.

[13] Mrinmoy Ghosh, Hsien-Hsin S. Lee. Smart refresh: An enhanced memory controller design for reducing energy in conventional and 3D die-stacked DRAMs[C]. 40th IEEE/ACM International Symposium on Microarchitecture, MICRO 2007, Chicago, IL, United states: Inst. of Elec. and Elec. Eng. Computer Society, 2007: 134~145