www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 10 October,2013 Page No. 2895-2899

THE POWER CONSTRAINT AND REMEDIAL METHOD IN DESIGN OF VARIATION TRAINED DROWSY CACHE (VTD- CACHE) IN VLSI SYSTEM DESIGN

Sura Sreenivasulu [1] ***M.Mahaboob Basha*** [2]
[1]PG Scholar of VLSI System Design in Electronics & Communication Engineering
[2]Professor & HOD in Electronics & Communication Engineering
[1,2]AVR & SVR College of Engineering & Technology, Nandyal, A.P, India
[1]surasreenivasulu7@gmail.com, [2]mmbfasi@gmail.com

Abstract-- ***Power is arguably the critical resource in VLSI system design today. In this paper a brief review is discussed about drowsy cache & also about the "Variation Trained Drowsy Cache" (VTD-Cache) architecture. As process technology scales down, leakage power consumption becomes comparable to dynamic power consumption. The drowsy cache technique is known as one of the most popular techniques for reducing the leakage power consumption in the data cache. However, the drowsy cache is reported to degrade the processor performance significantly. In this paper VTD-Cache allows for a significant reduction of around 50% in power consumption while addressing reliability issues raised by memory cell process variability. By managing voltage scaling at a very fine granularity, each cache way can be sourced at a different voltage where the selection of voltage levels depends on both the vulnerability of the memory cells in that cache way to process variation and the likelihood of access to that cache location. The novel and modular architecture of the VTD-Cache and its associated controller makes it easy to be implemented in memory compilers with a small area and power overhead. This total process is studied with different diagrams ,schematics using Xilinx 14.5 software.***

Keywords: Cache, Drowsy Cache, Leakage Current, Low Power, SRAM, Technology Scaling, Voltage Scaling.

1. INTRODUCTION

As the computing power of the processor[15] is more needed, the power dissipation in the processor inevitably increases. For this reason, reducing power dissipation [1] in the processor becomes one of the most important design considerations. Among the computer components, cache is reported to take account of a significant fraction of total processor power consumption. Thus, power efficiency of the cache should be carefully considered as shown in the figure 1. In the past, dynamic power consumption was larger than leakage power consumption. However, the leakage power consumption becomes comparable to the dynamic power consumption as the number of transistors employed in a processor increases.

To reduce the leakage power consumption of caches, various techniques have been proposed[2][4]. Powell et al. proposed the gated-Vdd[2], a circuit-level technique to gate the supply voltage, resulting in reduced leakage power in unused memory cells. Cache decay[3] reduces the leakage power consumption by invalidating and turning off the cache lines when they hold data that are not likely to be reused, based on the gated-Vdd technique. As described in [3], after the cache line is turned off, the data stored in that cache line cannot be reused. Therefore, when the processor needs the cache line that has been turned off, it has to be fetched from the lower level cache or the memory, resulting in significant performance degradation.

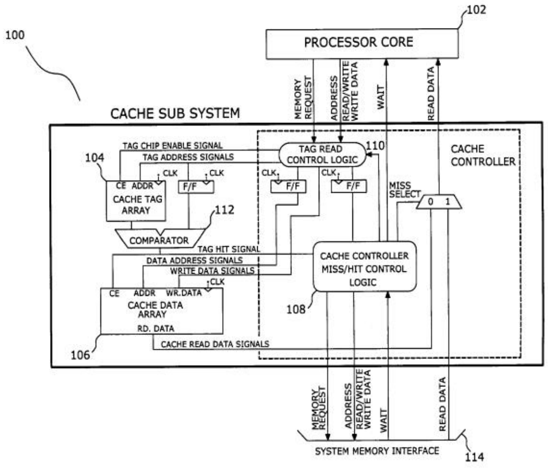


Figure 1: Conventional Cache Memory sub system works with a processor

The drowsy cache scheme[4], which reduces the leakage power consumption with multi-level supply voltage, is reported to be one of the most efficient leakage power reduction techniques. Each cache line has two modes in the drowsy cache scheme: normal mode and drowsy mode. The supply voltage for drowsy mode is lower than that for normal mode to save the leakage power consumption. Drowsy cache technique changes the mode of the cache line which has not been used frequently into low power consumption mode(drowsy mode) instead of turning off the cache line. Contrary to the cache decay technique, when the data stored in the cache line in the drowsy mode is accessed, it just requires extra cycle(s) to wake up the cache line. The wakeup means that the cache line is changed to normal mode from drowsy mode.

The drowsy technique shows better performance than the decay technique, because there is no need to fetch the data from the lower level memory when the data in the cache line in the drowsy mode is required. However, the extra cycle still degrades the performance. Several research groups have focused on the drowsy cache[5][6][8]. Researches dealing with the drowsy cache mainly have focused on the energy reduction, while this work focuses on the performance improvement of the drowsy cache by reducing the extra cycles to wake up the drowsy cache lines. For instruction caches, wakeup prediction technique based on branch prediction was proposed[7], which is not applicable to data caches. Improved Drowsy (ID) technique, which determines the states of cache lines based on the locality, was proposed to improve the efficiency of drowsy instruction cache[8].

2. BASIC OF CONVENTIONAL CACHE

Cache is a high-speed access area that can be either a reserved section of main memory or a storage device. The two main cache types are memory cache and disk cache. Memory cache is a portion on memory of high-speed static RAM (SRAM) and is effective because most programs access the same data or instructions over-and-over. By keeping as much of this information as possible in SRAM, the computer avoids accessing the slower DRAM. Most computers today come with L3 cache or L2 cache as shown in the figure 2, while older computers included only L1 cache.

Like memory caching, disk caching is used to access commonly accessed data. However, instead of using high-speed SRAM, a disk cache uses conventional main memory. The most recently accessed data from a disk is stored in a memory buffer. When a program needs to access data from the disk, it first checks the disk cache to see if the data is there. Disk caching can dramatically improve the performance of applications because accessing a byte of data in RAM can be thousands of times faster than accessing a byte on a hard drive.

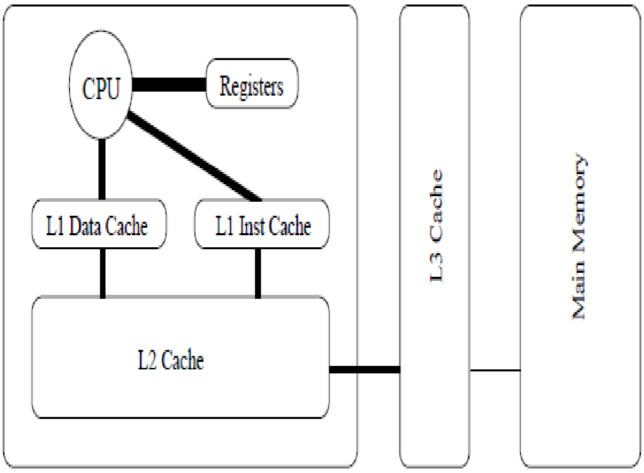


Figure 2: A Typical cache memory hierarchy

Another cache is known as "Internet browser cache" also known as "Temporary Internet Files" in Internet Explorer. Internet cache is used to help improve how fast data is opened while browsing the Internet. In most cases, each time a web page is opened, it is sent to your browser's temporary cache on your hard drive. If that page is accessed again and has not been modified, the browser will open the page from your cache instead of downloading the page again. This saves users a lot of time, especially if that the user is using a modem, and can also help save the web page owner on bandwidth.A cache server is a computer or network device that has been setup to store web pages that have been accessed by users on a network. Any user trying to access a web page that has already been stored on the cache server will be sent the stored version instead of downloading the web page again. This helps reduce network and Internet traffic congestion as well as saves the company on bandwidth costs.

L1 caching

Alternatively referred to as L1 cache, primary cache, internal cache, or system cache. When referring to computer processors, L1 cache is cache that is built into the processor and is the fastest and most expensive cache in the computer. The L1 cache stores the most critical files that need to be executed and is the first thing the processor looks when performing an instruction.

L2 caching:

Short for Level 2 caching, L2 is also commonly referred to as secondary cache or external cache. Unlike Layer 1 cache, L2 cache was located on the motherboard on earlier computers, although with newer processors it is found on the processor chip. When L2 cache is found on the processor, if the cache is also on the motherboard, it is more properly known as L3 cache. The L2 cache is on the same processor chip and uses the same die as the CPU, however, it is still not part of the core of the CPU. L2 cache was first introduced with the Intel Pentium and Pentium Pro computers and has been included with ever process since, with the exception of the early versions of Celeron processor. This cache is not as fast as the L1 cache, but is only slightly slower since it is still located on the same processor chip, and is still faster than the computer memory. The L2 cache is the second thing the computer looks at when performing instructions.

L3 Caching:

L3 Cache is Cache found on the motherboard instead of the processor on earlier computers. With today's computers this type of cache is a cache that is found on the same chip and die as the processors. In the below picture of the Intel Core i7-3960X Processor die, is an example of a processor chip containing six cores (CPUs) and the shared L3 Cache. As can be seen in the picture, the L3 cache is shared between all cores (CPUs) and is very large in comparison to what an L1 or L2 cache would be on the same chip because it is cheaper although slower. L3 or L3 communication is also a supplier of Intelligence, Surveillance, and Reconnaissance systems and products, secure communications systems and products, microwave components, and space and navigation products.

3. THE DROWSY CACHE APPROACH

In Caches, for fix period of time the activity is centered at some cache lines. So, putting rest of cache lines in low power mode can reduce the leakage significantly. This low power mode of cache line is called Drowsy Caches [9][10]. In stand of turning off cache line putting it in to a low power drowsy mode can reduce leakage significantly. In drowsy caches the chance of putting wrong line into drowsy mode is less, for that different policies have been proposed in [9]. When caches are in drowsy mode the data in it are preserved. Drowsy caches can be implemented by adaptive body-biasing with multi-threshold CMOS (ABB-CMOS), dynamic voltage scaling (DVS). Gated-Vdd.

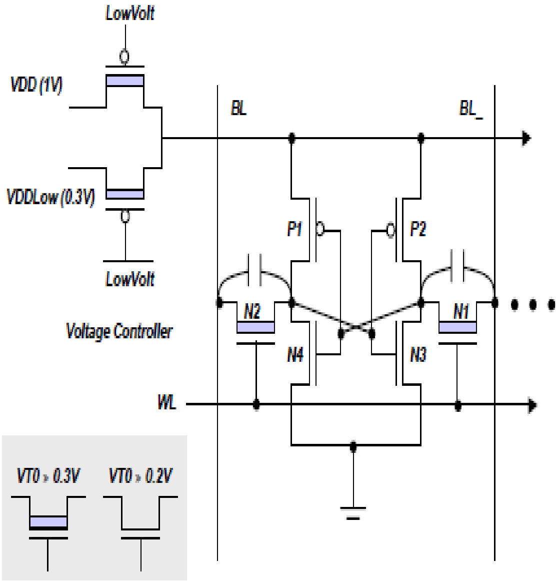


Figure 3: Schematic of Drowsy memory circuit [9].

As shown in figure 4 SRAM cell is connected to voltage-scaling controller. This controller consist of two pMOS pass transistor, one with high threshold voltage while other with low threshold voltage. One pMOS supplies normal supply voltage and other low for drowsy cache lines. Each pass transistor of SRAM cell is of high V_{th} to prevent the leakage current from the normal supply to the low supply through the two pMOS pass gate transistor. For each cache line a separate voltage controller is needed.

ASYMMETRIC SRAM CELL:

As this technique used in cache it is refer to as asymmetric-cell caches (ACCs). Comparing to conventional cache ACCs reduce leakage power even when there are few parts of the cache that are left unused [9]. Traditional SRAM cell transistors are symmetrical with identical leakage and threshold voltage, while asymmetric SRAM cells have low leakage and less impact on performance. In this technique when cell is storing 0, selected transistors are “weakened” to reduce leakage. A weakening can be possible by using higher threshold voltage and also by proper sizing of transistor. In conventional SRAM of symmetrical transistor to reduce leakage current one method can be used that is making all transistors of high V_{th}, but it degrades the performance. This drawback can be overcome by using asymmetric SRAM cell. It works on following principle: selecting a preferred stored value and weaken only those transistors necessary to reduce leakage by increasing the threshold voltage when this value is stored.

Working of Asymmetric SRAM Cell:

In cell most of the leakage is dissipated by transistors that are off and have a voltage differential across their drain and source. This state of transistor can be finding by the value stored in it. When a cell storing a„0‟ value, as shown in fig.4, the leaky transistors will be P1, N4 and N2. If cell was storing ‘1’ value then leakage transistor would be P2, N1 and N3.

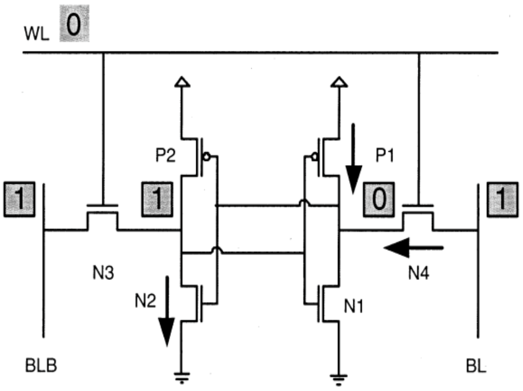


Figure 4: SRAM cell with storing ‘0’ value[16]

To reduce leakage in cell when it storing a ‘0’ value, replace leaky transistor by high V_{th}. The resulted circuit which is called basic asymmetric (BA) SRAM cell. This circuit has the same leakage as conventional SRAM cell when storing 1, but it reduces leakage by 70 times when storing 0, due to longer discharge time.

4. CIRCUIT ISSUES CMOS

Traditionally, three circuit techniques have been used to reduce leakage power in CMOS circuits: gated-V_{dd}, ABB-MTCMOS (Adaptive Body Biasing -Multi Threshold CMOS) and leakage-biased bitlines. In this paper, we instead use dynamic voltage scaling (DVS) for leakage control [11]. While voltage scaling has seen extensive use for dynamic power reduction, short-channel effects also make it very effective for leakage reduction [12]. Furthermore, DVS also reduces gate-oxide leakage, which has increased dramatically with process scaling. Below, we discuss the traditional gated-V_{dd} and ABBMTCMOS techniques for cache leakage reduction, as well as our proposed technique using DVS and compare the different techniques.

4.1 Gated-V_{dd}

The gated-V_{dd} structure was introduced in [13][14].This technique reduces the leakage power by using a high threshold (high-V_t) transistor to turn off the power to the memory cell when the cell is set to lowpower mode. This high-V_t device drastically reduces the leakage of the circuit because of the exponential dependence of leakage on V_t. This method is very effective at reducing leakage, however it has the disadvantage that it loses any information stored in the cell when switched into low-leakage mode. This means that a significant performance penalty is incurred when data in the cell is accessed and more complex and conservative cache policies must be employed.

4.2 ABB-MTCMOS

In this method, the threshold voltages of the transistors in the cell are dynamically increased when the cell is set to drowsy mode by raising the source-to-body voltage of the transistors in the circuit. This higher V_t reduces the leakage current while allowing the memory cell to maintain its state even in drowsy mode. However, to avoid the need for a twin-well process, the dynamic V_t scaling is accomplished by increasing the source of the NMOS devices and by increasing the body voltage of the wells of the PMOS devices when the circuit is in drowsy mode. Although the leakage current through the memory cell is reduced significantly in this scheme, the supply voltage of the circuit is increased, thereby offsetting some of the gain in total leakage power. Also, this leakage reduction technique requires that the voltage of the N-well and of the power and ground supply lines are changed each time the circuit enters or exits drowsy mode. Since the N-well capacitance of the PMOS devices is quite significant, this increases the energy required to switch the cache cell to high-power mode and can also significantly increase the time needed to transition to/from drowsy mode. Similarly to the gated-Vdd technique, ABBMTCMOS also requires special high-Vt devices for the Control logic.

4.3 Dynamic Vdd Scaling (DVS)

The method proposed in this paper utilizes dynamic voltage scaling (DVS) to reduce the leakage power of cache cells [11]. By scaling the voltage of the cell to approximately 1.5 times Vt, the state of the memory cell can be maintained. For a typical 0.07um process, this drowsy voltage is conservatively set to 0.3V. Due to the short-channel effects in high-performance processes, the leakage current will reduce substantially with voltage scaling.

Since both voltage and current are reduced in DVS, a dramatic reduction in leakage power is obtained. Since the capacitance of the power rail is significantly less than the capacitance of the N-wells, the transition between the two power states occurs more quickly in the DVS scheme than the ABB-MTCMOS scheme. Figure 3 illustrates the circuit schematic of memory cells connected to the voltage-selection controller. No high-Vt device is used in the memory cell itself in our proposed technique as opposed to the method in [11] where high-Vt devices were used for the pass transistors that connect the memory's internal inverters to the read/write lines (N1 and N2). Because each cache line in [11] is controlled independently and each bit line is shared by all the cache lines in a sub-bank, all the read/write lines are maintained at high-Vdd, making it necessary to use high-Vdd, making it necessary to use high-Vt transistors for the pass gates in order to maintain acceptable leakage current [11].

However, since for the instruction cache, the entire sub-bank is switched between low-Vdd and high-Vdd, the read/write lines in each sub-bank are included in the DVS and no high-vt pass-transistors are needed. Avoiding the use of high-Vt device for the memory cells has several advantages against the previous approach [11]. First, the access time of the cache is not compromised. High-Vt devices show poor current driving capability at the same gate input voltage, which results in slower caches. Particularly for Icaches, which are critical in determining the cycle time of the processor, it is important to avoid any increase of the access time. This is why a direct-mapped cache is usually employed for an instruction cache since a set-associative cache is slower than a direct-mapped cache. Second, use of low-Vt pass-transistors reduces the dynamic power, since in our previous approach, significantly larger pass transistors are used to compensate the reduced current driving capability which is impaired by high-Vt threshold voltage. In Figure 3, one PMOS pass gate connects the supply line to the normal supply voltage and the other connects it to the low supply voltage for. Each pass gate is a high-Vt device to prevent leakage current from the normal supply to the low supply through the two PMOS pass gate transistors. A separate voltage controller can be implemented for each sub-bank or for each cache line.

A possible disadvantage of the circuit in Figure 3 is that it has increased susceptibility to noise and variation of Vt across process corners. The first problem may be corrected with careful layout because the capacitive coupling of the lines is small. To examine the stability of a memory cell in the low power mode, we simulated a write operation to an adjacent memory cell that shares the same bit lines but whose supply voltage was normal. The coupling capacitance and the large voltage swing across the bit lines would make the bit in the drowsy memory cell vulnerable to flipping if this circuit had a stability problem.

However, our experiments show that the state of the drowsy memory cell is stable. There is just a slight fluctuation in the core node voltage caused by the signal crosstalk between the bit lines and the memory internal nodes. In addition, there is no cross-talk noise between the word line and the internal node voltage, because word line gating prevents accesses to memory cells in drowsy mode. Of course, this voltage scaling technique has less immunity against a single event upset (SEU) from alpha particles, but this problem can be relieved by process techniques such as silicon on insulator (SOI). Other static memory structures also suffer from this problem. making it necessary to implement error correction codes (ECC) even for non-drowsy caches. The second problem, variation of Vt, may be handled by choosing a conservative Vdd value, as we have done in our design. The memory cell layout was done in TSMC 0.18um technology, which is the smallest feature size available to the academic community. The dimensions of our memory cell is 1.84um by 3.66um, and those for the voltage controller are 6.18um by 3.66um.

5.RESULT & DESIGN ANALYSIS

In this result analysis are discussed in terms of architecture , design summary, Internal schematic diagram & output waveforms are also discussed in terms of drowsy line enable/disable whose cycle count will be reduced as 50% as shown in the following diagrams

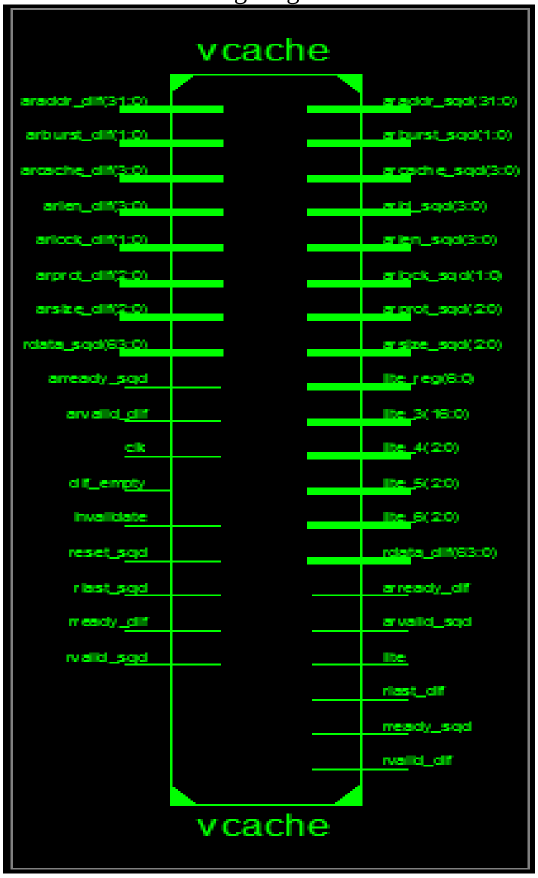
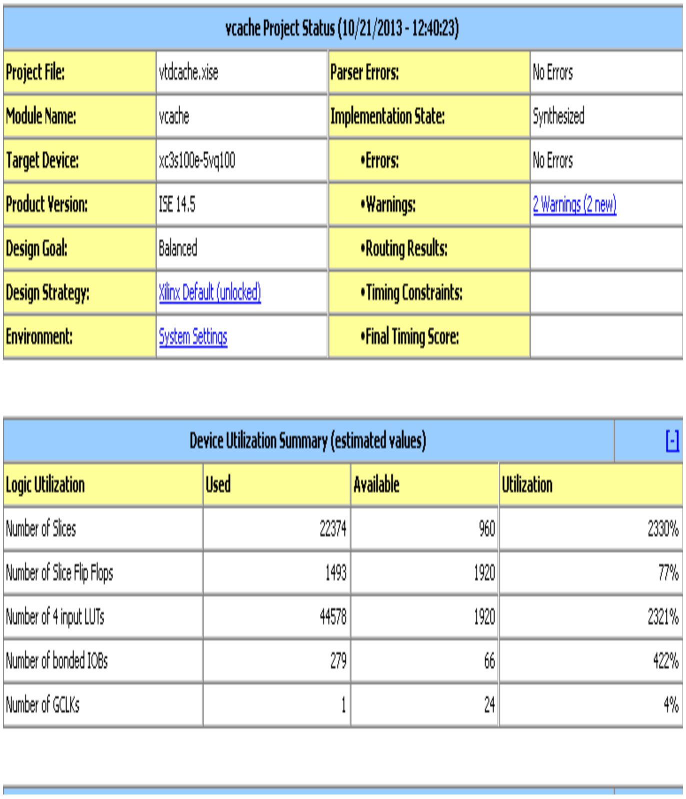


Figure 5:RTL schematic of VTD Cache



vcache Project Status (10/21/2013 - 12:40:23)			
Project File:	vtdcache.xise	Parser Errors:	No Errors
Module Name:	vcache	Implementation State:	Synthesized
Target Device:	xc3s100e-5vq100	•Errors:	No Errors
Product Version:	ISE 14.5	•Warnings:	2 Warnings (2 new)
Design Goal:	Balanced	•Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	•Timing Constraints:	
Environment:	System Settings	•Final Timing Score:	

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	22374	960	2330%
Number of Slice Flip Flops	1493	1920	77%
Number of 4 input LUTs	44578	1920	2321%
Number of bonded IOBs	279	66	422%
Number of GCLKs	1	24	4%

Figure 6: Design Summary of VTD cache

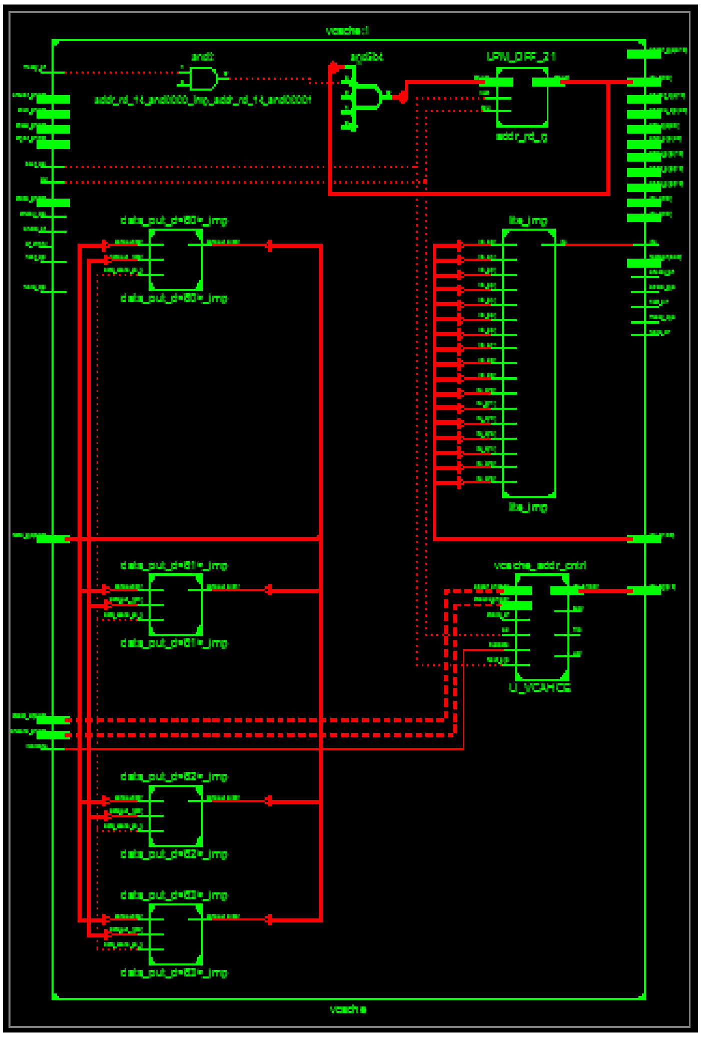
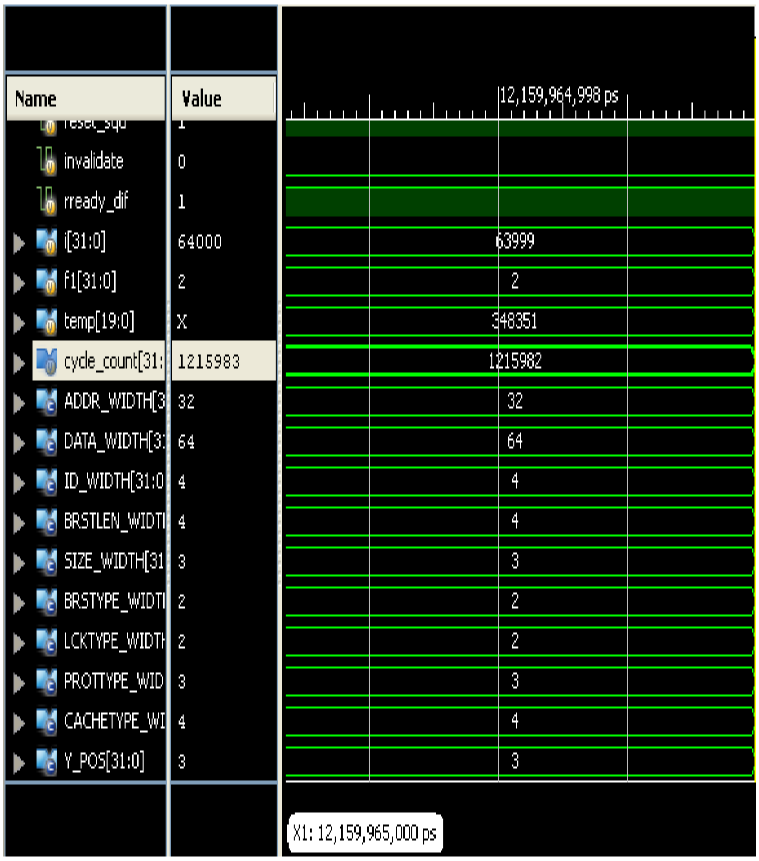


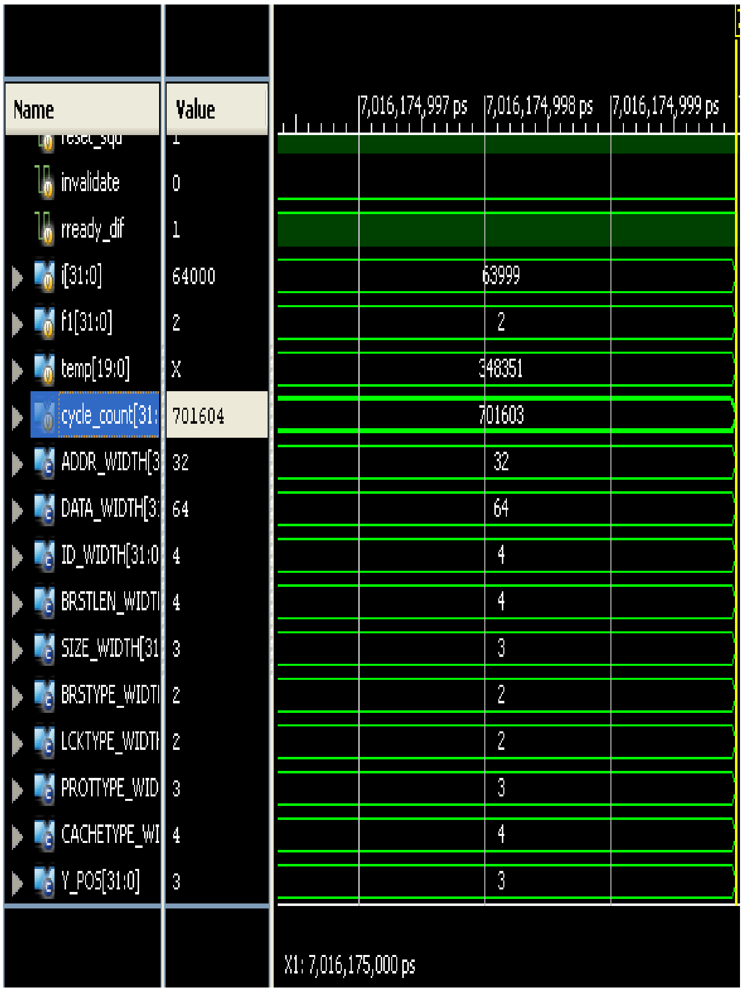
Figure 7: Internal schematic diagram of VTD Cache



Name	Value	12,159,964,998 ps
reset_squ	1	
invalidate	0	
rready_dif	1	
i[31:0]	64000	63999
f1[31:0]	2	2
temp[19:0]	X	348351
cycle_count[31:	1215983	1215982
ADDR_WIDTH[3	32	32
DATA_WIDTH[3	64	64
ID_WIDTH[31:0	4	4
BRSTLEN_WIDTI	4	4
SIZE_WIDTH[31	3	3
BRSTYPE_WIDTI	2	2
LCKTYPE_WIDTI	2	2
PROTTYPE_WID	3	3
CACHETYPE_WI	4	4
Y_POS[31:0]	3	3

X1: 12,159,965,000 ps

Figure 8: VTD Cache Drowsy lines disable cycle count



Name	Value	7,016,174,997 ps / 7,016,174,998 ps / 7,016,174,999 ps
reset_squ	1	
invalidate	0	
rready_dif	1	
i[31:0]	64000	63999
f1[31:0]	2	2
temp[19:0]	X	348351
cycle_count[31:	701604	701603
ADDR_WIDTH[3	32	32
DATA_WIDTH[3	64	64
ID_WIDTH[31:0	4	4
BRSTLEN_WIDTI	4	4
SIZE_WIDTH[31	3	3
BRSTYPE_WIDTI	2	2
LCKTYPE_WIDTI	2	2
PROTTYPE_WID	3	3
CACHETYPE_WI	4	4
Y_POS[31:0]	3	3

X1: 7,016,175,000 ps

Figure 9: VTD Cache Drowsy lines Enable Cycle count

6. CONCLUSIONS

In this paper, we presented the VTD-Cache a novel solution for obtaining 50% reduction in power consumption in low power cache for high performance processors while addressing the reliability issues raised by process variability. We explored the design space of VTD-Cache architecture and its components as shown in the above figure 7. We demonstrated how the VTD-Cache setting is chosen to maximize the improvement in total energy savings. Our simulation results as shown in figure 8 & 9, indicate a significant improvement in total energy consumption across simulated benchmarks.

We consider VTD-Cache as a logical extension to drowsy cache, further improving its dynamic power consumption. While taking into account "weak cells," the VTD-Cache reduces dynamic power consumption of accessing most of the cache ways within CWoE while reducing the static power consumption of cache ways supplied from low voltage between accesses. In future work, we intend to address the problem of enforcing triple voltage supply policy to tag section of the cache as well as dynamic reconfiguration policies and design issues to further improve energy consumption for adapting with changes in the phase of each benchmark execution.

Acknowledgements:

It is a pleasure to recognize the many individuals who have helped me in completing this technical paper. I sincerely express heartiest thanks to my Project guide & HOD of ECE Department **Sri M.Mahaboob Basha garu** who contributed the technical guidance and encouragement, valuable suggestions & inspirations offered me his knowledge to help me overcome many of the difficulties in all aspects and support throughout the process.

7. BIBILOGRAPHY

1. Avesta Sasan, Kiarash Amiri, Variation Trained Drowsy Cache(VTD-Cache): A history Trained Variation Aware Droswsy Cache for Fine Grain Voltage Scaling IEEE transactions on VLSI systems,vol20,N0.4,April 2012.

2. M. Powell, S.H. Yang, B. Falsafi, K. Roy, T.N. Vijaykumar, "Gated-Vdd: A circuit technique to reduce leakage in deep-submicron cache memories", In Proceedings of International Symposium on Low Power Electronics and Design, pp.90-95, 2000.

3. S. Kaxiras, Z. Hu, M. Martonosi, "Cache decay: Exploiting generational behavior to reduce leakage power", In Proceedings of International Symposium on Computer Architecture, pp.240-251, 2001.

4. K. Flautner, N.S. Kim, S. Martin, D. Blaauw, T. Mudge, "Drowsy caches: Simpletechniques for reducing leakage power", In Proceedings of International Symposium on Computer Architecture, pp.148-157, 2002.

5. R. Giorgi, P. Bennati, "Filtering drowsy cache to achieve better efficiency", In Proceedings of ACM Symposium on Applied Computing, pp. 1554-1555, 2008.

6. S. Petit, J. Sahuquillo, J. M. Such, D. Kaeli, "Exploiting temporal locality in drowsy cache policies", In Proceedings of the 2nd Conference on Computing Frontiers, pp. 371-377, 2005.

7. S.W. Chung, K. Skadron, "On-demand solution to minimize I-cache leakage energy with maintaining performance", IEEE Transactions on Computers, Vol. 57, pp. 7-24,2008.

8. M.B.C. Alioto, P. Bennati, R. Giorgi, Exploiting locality to improve leakage reduction in embedded drowsy I-caches at same Area/speed", In Proceedings of International Symposium on Circuits and Systems, pp. 37-40, 2010.

9. K. Flautner, Nam sung kim, S. Martin, D. Blaauw and T. Mudge, "Drowsy Caches: simple techniques for reducing leakage power," proc. Of IEEE/ACM Intl.Symp. on computer Architecture, PP. 148~157, 2002.

10. Nvid Aziz, student member, IEEE, Farid N. Najm, fellow, IEEE, and A.Moshovos, Associate member,IEEE, " Low-Leakage Asymmetric-Cell SRAM", IEEE Trans. On Very Large Scale Integration System, vol.11, no.4. August 2003.

11. K. Flautner, et al. Drowsy Caches: Simple Techniques for Reducing Leakage Power. To appear in Proc. of Int. Symp. Computer Architecture, 2002.

12. S. Wolf. Silicon processing for the VLSI era Volume 3 - The submicron MOSFET. Lattice Press, 1995, pp. 213-222.

13. M. Powell, et al. Gated-Vdd: A circuit technique to reduce leakage in deep-submicron cache memories. Proc. of Int. Symp. Low Power Electronics and Design, 2000, pp. 90-95.

14. S. Yang, et al. An integrated circuit/architecture approach to reducing leakage in deep-submicron high-performance I-caches. Proc. of Int. Symp.High-Performance Computer Architecture, 2001, pp. 147 - 157.

15. K. Ghose, and M. Kamble. Reducing power in superscalar processor caches using subbanking, multiple line buffers and bit-line segmentation. Low Power Electronics and Design, Proc. of Int. Symp. on Low Power Electronics and Design, 1999. pp. 70 -75.

16. Urvashi Chaudhari, Rachna Jani A Study of Circuit Level Leakage Reduction Techniques in Cache Memories (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 3, Issue 2, March -April 2013, pp.457-460.