Optimal Page Allocation of Hybrid Main Memory using Page Caching Algorithm

Lordwin Cecil Prabhaker.M, Saravana Ram.R

Abstract: ***In memory management system, DRAM has long write endurance and PRAM has shorter write latency. To get the advantages of both memories, a hybrid memory is constructed using DRAM and PRAM. In this article, the task allocation strategies on hybrid memory were carried out to achieve multiple targets such as, extending the lifetime, reducing power consumption and reducing the memory size. Different types of algorithm such as, genetic algorithm, scalable algorithm and distributed algorithms are proposed by many researchers. Here, page caching algorithm is proposed to achieve an optimal performance. The Page caching algorithm includes the following steps: 1.Getting instructions from CPU and finding address in DRAM, 2. Extracting page in PRAM, 3.Reallocating the page in DRAM and 4. Updating in PRAM. The work is designed and implemented using an evaluation framework and the hybrid memories optimal performance is calculated. The user provides workload, PRAM and DRAM parameters and environmental characteristics of hybrid memory as inputs to the system. It also determines the overall performance by considering several limitations. This architecture first optimizes the page allocation and then the task write in the memory was allocated. Comparing to the existing system, the page caching algorithm reduces upto 32.8% of total power.***

Index Terms: ***Optimal Page allocation, Hybrid memory, Page caching algorithm, Memory management system, PRAM and DRAM.***

I INTRODUCTION

In recent days, the memories used in computers are fully managed by memory management mechanisms. The main task of memory management is to allocate portion of memory to programs at their request dynamically and releasing it for reuse when it is no longer required. A number of methods were developed to increase the efficiency of memory management [2, 3, 8]. The virtual memory systems split the memory addresses of process from actual physical addresses, allowing processes to be separated and effectively increase available amount of RAM through paging or swapping to secondary storage.

Power and cost budget of a computing system increases while using DRAM based main memory. This drawback can be overcome by implementing new memory technologies such as, magnetic RAM, ferro-electric RAM, and phase charge RAM (PRAM). Among these, for large scale main memory applications PRAM is the most optimum techniques, since, it it has high density and very less power consumption [1,5,7,14]. The virtual memory managing system is also an impact while determining the overall system performance.

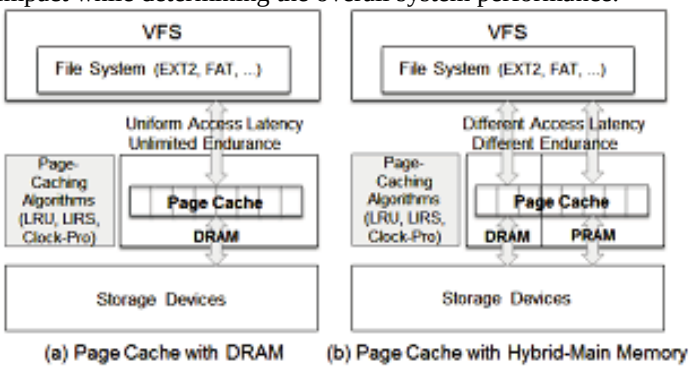


Figure 1: Page caching mechanism with different memory approaches

In the earlier studies, to compensate latency and tolerance limits of PRAM a hybrid main memory was used (a combination of PRAM and DRAM). To reduce the latency of the disk access a page caches are used and it occupies large portion of main memory. Page cache algorithm such as, clock-Pro, LRU and LRS algorithm was proposed earlier. These algorithms shows, an improvement in performance. But it only considers main memory with the following characteristics: 1. Uniform latency of access, 2. Infinite durability and 3. The hybrid memory architecture are not adoptable directly to PRAM's main memory structure [6,18,22,24]. The objective of this work is 1) To reduce energy consumption, 2) To reduce number of NVM records and 3) To reduce the NVM size of the main memory.

II LITERATURE SURVEY

Soyoon Leo et al proposed a hybrid phase changing memory mechanisms and DRAM architecture. This memory approach is used to reduce the energy dissipation. There are two drawbacks of PCM such as limited write operation for each cell and write access time which is 6-10 times lesser than DRAM [10, 16, 21]. So, to overcome the drawbacks, hybrid memory mechanisms which consist of PCM memory and minimum amount of DRAM have been suggested.

The disadvantages of PCM memory was overcome by this approach. Yibo guo et al, proposed a data placement management algorithm (DPM) with polynomial time consideration for multicore systems. This system equipped with SPM's to minimize total cost of memory accesses. It also integrates data duplication mechanism (DPM-DUP). The former technique further minimizes the cost of memory access even though several data copies are allowed. Weijia Che et al., discussed about issues in multicore processor scratchpad memories.

Revised Manuscript Received on September 15, 2019

Lordwin Cecil Prabhaker M, Department of Electronics and Communication Engineering, Vignan's Lara Institute of Technology and Science, Guntur, AP, India, Email: cecillord@gmail.com.

Saravana Ram R, Department of Electronics and Communication Engineering, Anna University, University College of Engineering, Dindigul, Tamilnadu, India, Email:saravanaramkrishnan@gmail.com.

They also, addressed some issues arises when compilation and optimization for streamed programs [11,15,19,20, 25]. The scratch pad memories are used in multicore processor due to its optimum power consumption characteristics. Performance optimization of memories used in multicore processor is a complete task, thus implementation cost also increases. Hyunsun Patric et al., developed power management technique called as runtime adaptive management technique for hybrid main memory. To reduce DRAM's refresh energy a DRAM decay was introduced. It occupies some amount of energy in the total memory energy [4,12,13]. Some other methods also proposed to further reduction in the refresh energy and latency arises due to memory access. They are DRAM bypass and dirty data keeping technique.

Jong-Hun Choi et al., implemented an evolutionary framework called as OPAMP. It provides an optimal performance for hybrid memory environment even though the calculation of hybrid memory architecture's performance is high. Also, it derives the optimal value for NP-complete functions [13, 17].The maximum performance of hybrid memory was calculated under several critical conditions for the given workload, DRAM and PRAM specification and environmental parameters.

The DRAM utilizes 30%-40%of power consumption but it is placed in main memory of many computer architectures. PRAM has access latency and low writes endurance. To have the advantages of both memories, a hybrid memory is formed combining the advantages of both memories. A new framework OPAMP is designed which calculates which calculates the optimal performance. For specific target application, OPAMP take out memory access trace and optimal value calculator derive the optimization value [17, 21]. They allocate 25% of pages in DRAM from the total pages of workload. They get the dynamic page allocation that reduces the hybrid memory system's energy consumption. Optimal approach rate is derived by dividing the energy consumption of previous work by optimal energy consumption. In comparison with DRAM only system this frame work reduces energy delay product over 20% on average. The memory- intensive process can hurt all other processes.

III PROPOSED PAGE CACHING ALGORITHM

Page Caching algorithm includes fetch instruction from CPU, search address in DRAM, extract page in PRAM ,reallocate page in DRAM and update page in PRAM. To meet different goals such as, low energy dissipation, and NVM's lifetime extension a most appropriate memory location assigning technique is proposed. Energy consumption, NVM write and NVM size are some other important parameters to be considered to attain the target [3,5,18,23]. NVM based hybrid main memory is optimized by an integer linear programming model (ILP). Different targets are considered, such as the reduction of energy consumption, NVM writing and NVM size.

An evaluation framework that calculates the optimum performance of the hybrid memory environment is developed and implemented. This system collects workload, PRAM and DRAM specifications and hybrid main memory environmental parameters from the user input and determines the uttermost performance for the given aspect. [16, 22].

Algorithm 1 : Page Caching Algorithm

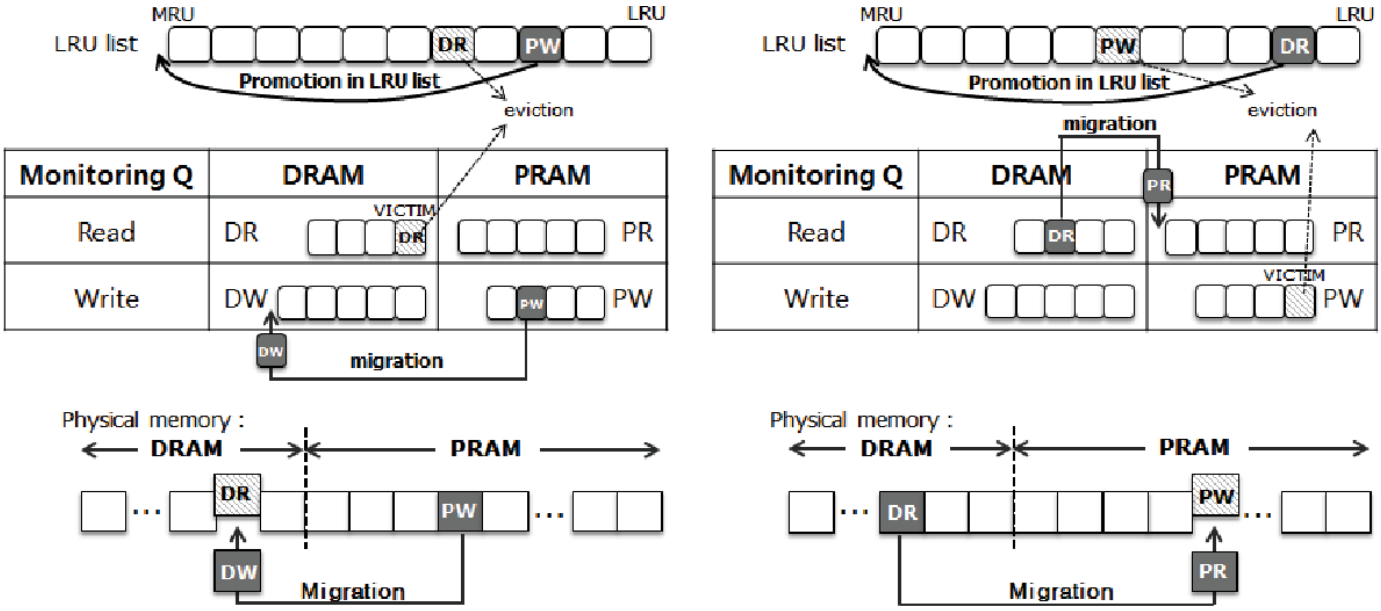
Input: page address and request type

W_{Ct}: Current weight value
Th_m: Threshold value for determining migration
Th_M: Threshold value for determining movement between read and write queues
Calculate W_{Ct}

if page in PRAM then
if $W_{Ct} < Th_m$
    then migrate the page to DRAM
else if $W_{Ct} <= Th_M$
    and page in read queue then the page moves to write queue
else if $W_{Ct} > Th_M$
    and page in write queue then the page moves to read queue
end if
else if
    page in DRAM then
if $W_{Ct} < Th_M$:
    then migrate the page to PRAM
else if $W_{Ct} > Th_M$:
    and page in write queue then the page moves to read queue
else if $W_{Ct} == Th_M$:
    and page in read queue then the page moves to write queue
end if
end if

The PRAM to DRAM migrations are explained as depicted I figure 2(a). This system selects the DRAM victim's page when there is no availability of space I DRAM. And also, it chooses the victim page at the mark down portion of DRAM reading queue and remove it. The PRAM written page is migrated to the DRAM location where the victim's page is located. This page is placed at the top of the queue in the DRAM write queue.

When the element is not available in the DRAM read queue while finding the wandering in the victim's page, choose a victim page at the mark down portion of the DRAM write queue. It means that, the victim's page is rarely used at recent time. The read bound page migrations are illustrated in figure 2(b) and its characteristics are equivalent to write bound pages. Finally, to choose victim page select the lower portion of the PRAM write queue. Suppose if PRAM's write queue does not have any pages, then allocate the lower portion PRAM read queue. While deleting the migration information from the victim page, also remove the information stored in the main memory, the LRU list and the queue [14,17,24]. The reason for not changing the pages between the migration victim page and the migration page is that, this action would lead to an additional PRAM writing.



(a) Migration of the write-bound page (b) Migration of the read-bound page

Figure 2: Migration of write and read bound page.

IV EXPERIMENTATION AND EVALUATION

The simulation is performed on ISE (Integrated Synthesis Environment) based graphical user interface simulator. A top level HDL behavioral module is implemented to evaluate the performance of the model. For the 'counter' model CLOCK,I-O DIRECTION and COUNT_OUT are assigned in which COUNT_OUT is represented in 4 bit bus. The simulation module is designed based on the target device XC2VP30_6ff1152.

The test bench runs about 1000ns with the GSR signal and the following clock settings are assigned: clk_time_max: 20 ns; clk_time_min: 2 ns; input_time: 10 ns; output_delay: 10 ns; initial_offset: 100 ns; Global_Signals: GSR (FPGA). And also we assumed that the GSR value of 100 is added to the initial_offset value automatically, test bench initial length: 1000 ns and other clock fields are assigned with their predefined values. The behavioral model simulation of counter module is exhibited the figure 3. The figure 4 describes the comparison of power with different functions like leakage, clock, logic, BRAM, DSP, PLL, phase and input – output port. The output is maximum for the given input and output port function.

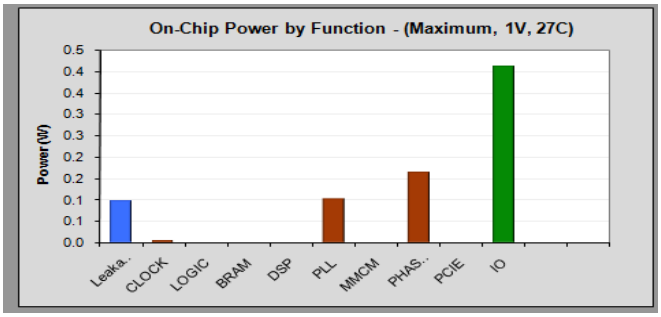


Figure 4: The active power (in watts) for different functional components.

Table 1. Total power (in watts) of different active components

Active Components	V_{cc} (in volts)	Power (in Watts)
Clock	1.00	0.002
Input and Output	1.00	0.042
PRAM	1.00	0.098
DRAM	1.00	0.087

The Table.1 represents the total power of the active components for the applied voltage (V_{cc} = 1volts). The clock consumes very less power than other active components and the PRAM and DRAM consumes almost equal power. The input and output port consumes half of the logical components. The proposed page allocation method gives 776mw power consumption which is less when compared with the existing system.

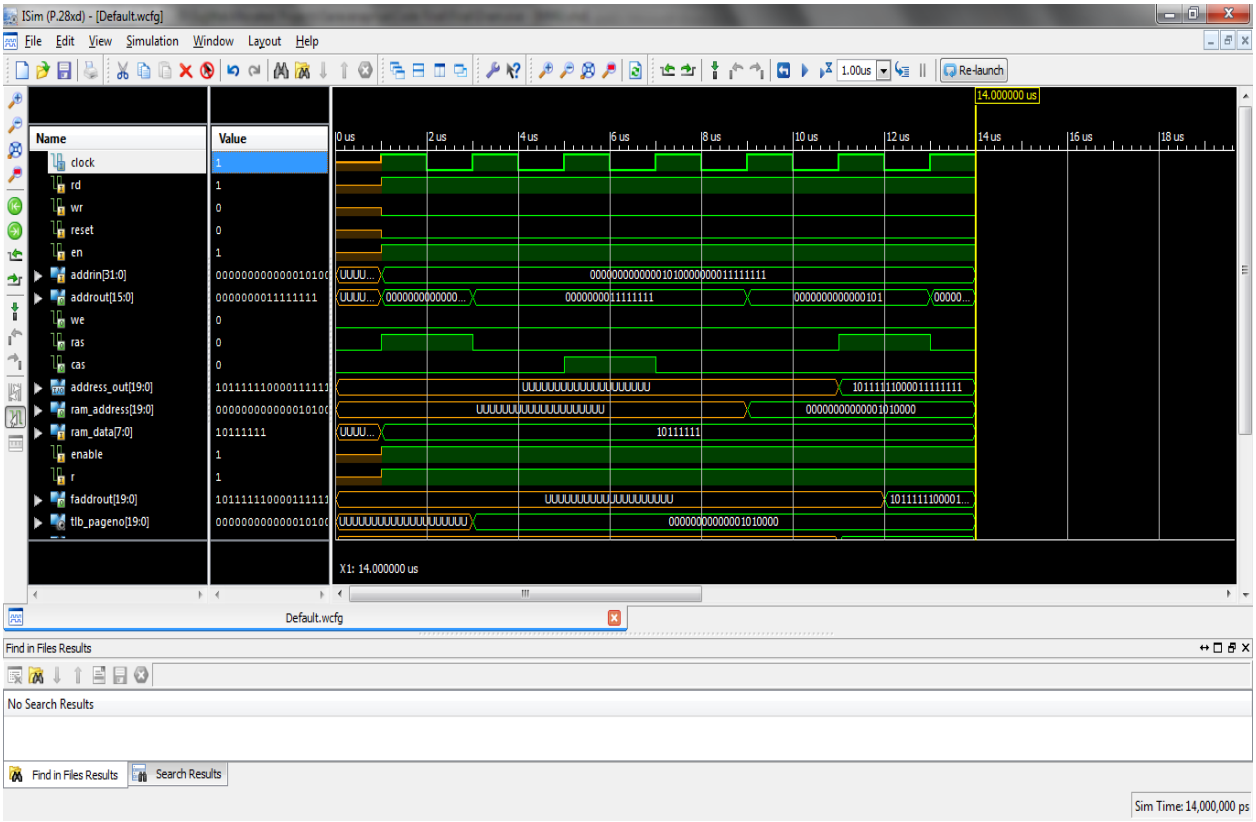


Figure 3: Behavioral model simulation result for 1000ns simulation time

V CONCLUSION

This paper addresses the issues arises while assigning the task in the main hybrid memory. This memory is a combination of PRAM and DRAM. It adopts the efficient energy structure of PRAM and the elongated writing resilience of the DRAM. The proposed technique achieves the objectives through reduce the energy consumption, the number of PRAM writings and size of PRAM without performing page migration. The proposed hybrid main memory system shows a near optimum performance comparing with other memory systems. The proposed well designed memory system lessen the energy delay product comparing with DRAM only system.

REFERENCES

1. LiangyuanWang et al, 2018 Data Scheduling Based on Data Label in Hybrid Storage Architecture IEEE 15th International Conference on obile Ad Hoc and Sensor Systems (MASS), Pages 367-372
2. Guan Wang et al 2017 “Shared Last-Level Cache Management and Memory Scheduling for GPGPUs with Hybrid Main Memory”,ACM Transactions on Embedded Computing Systems (TECS) Volume 17 Issue 4, August 2018 .
3. Jen-Wei Hsieh,Yuan-Hung Kuan 2015 “DCCS: Double Circular Caching Scheme for DRAM/PRAM Hybrid Cache” IEEE Transactions on Computers (Volume: 64 , Issue: 11 , Nov. 1 2015), Page(s): 3115 – 3127
4. M. Lordwin Cecil Prabhaker, Dr. K. Manivannan, 2018 Performance Based Investigation of Scheduling Algorithm on Multicore Processor.Advances in Natural and Applied Sciences. 11(7); Pages: 507-519
5. Jinbao Zhang,et al 2017, “An Optimal Page-Level Power Management Strategy in PCM–DRAM Hybrid Memory,” International Journal of Parallel Programming”, Volume 45, Issue 1, pp 4–16
6. Xiaojun Cai, Lei Ju, Xin Li,Zhiyong Zhang. Zhiping Jia, 2017 “Energy efficient task allocation for hybrid main memory architecture, Journal of Systems Architecture”,Volume 71, , Pages 12-22.
7. Su-Kyung Yoon et al 2019 ‘Self-learnable Cluster-based Prefetching Method for DRAM-Flash Hybrid Main Memory Architecture”ACM Journal on Emerging Technologies in Computing Systems (JETC) Volume 15 Issue 1.
8. E. Abinaya, K. Aishwarva, C. P. M. Lordwin, G. Kamatchi and I. Malarvizhi, 2018 "A Performance Aware Security Framework to Avoid Software Attacks on Internet of Things (IoT) Based Patient Monitoring System," 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), Coimbatore , pp. 1-6. DOI: 10.1109/ICCTCT.2018.8550955
9. Su-Kyung Yoon,et al, 2018 “Design of DRAM-NAND flash hybrid main memory and Q-learning-based prefetching method, The Journal of Supercomputing, Volume 74, Issue 10, pp 5293–5313
10. JUNPENG NIU et al, 2018 “Hybrid Storage Systems: A Survey of Architectures and Algorithms” IEEE Access (Volume: 6) Page(s): 13385 – 13406
11. Vinson Young ; Chiachen Chou et al, 2018 “ACCORD: Enabling Associativity for Gigascale DRAM Caches by Coordinating Way-Install and Way-Prediction”, ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA)
12. Shuo-Han Chen et al., 2016 “A Partial Page Cache Strategy for NVRAM-Based Storage Devices”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,2018, DOI: 10.1109/TCAD.2018.2887045,PP.1-14.
13. Soyoon Lee et al 2010 “Characterizing Memory Write References For Efficient Management Of Hybrid PCM And DRAM Memory,” in Proceedings of the Conference on Design.
14. Yibo Guo et al 2011 “Data Placement And Duplication For Embedded Multicore Systems With Scratch Pad Memory,” EEE Transactions on Computer-Aided Design of Integrated Circuits and Systems Volume: 32 , Issue: 6, Page(s): 809 – 817, DOI: 10.1109/TCAD.2013.2238990
15. Weijia Che, et al., 2011 “Compilation Of Stream Programs For Multicore Processors That Incorporate Scratchpad Memories” Design, Automation & Test in Europe Conference & Exhibition, DOI: 10.1109/DATE.2010.5456976.
16. Hyunsun Park, Sungjoo Yoo, Sunggu Lee, 2011, “Power Management Of Hybrid DRAM/PRAM-Based Main Memory,” IEEE 19th International Symposium on Design Automation Conference (DAC).

17. Jong-Hun Choi et al, 2011 , "OPAMP: Evaluation Framework For Optimal Page Allocation Of Hybrid Main Memory Architecture," International Conference on Parallel and Distributed Systems, DOI: 10.1109/ICPADS.2012.89.
18. G. W. Burr, et al, 2010 "Phase Change Memory Technology" Journal of vacuum science & technology B.
19. J. Li et al, 2011 "Stt-Ram Based Energy Efficiency Hybrid Cache For Cmps IEEE/IFIP 19th International Conference on VLSI and System-on-Chip, DOI: 10.1109/VLSISoC.2011.6081626.
20. J. Hu et al 2011 "Toward Energy Efficient Hybrid On-Chip Scratch Pad Memory with Non-Volatile Memory" Design, Automation & Test in Europe, DOI: 10.1109/DATE.2011.5763127
21. Y. Huang, T. Liu, and C. Xue, 2011 "Register Allocation For Write Activity Minimization On Non-Volatile Main Memory" 16th Asia and South Pacific Design Automation Conference (ASP-DAC 2011), DOI: 10.1109/ASPDAC.2011.5722171.
22. L. Shi, C. Xue, and X. Zhou, 2011 "Cooperating Write Buffer Cache And Virtual Memory Management For Flash Memory Based Systems," IEEE IEEE Real-Time and Embedded Technology and Applications Symposium, DOI: 10.1109/RTAS.2011.22
23. M. K. Qureshi et al, 2009 "Scalable High Performance Main Memory System Using Phase-Change Memory Technology" ACM/IEEE-Proceedings of the 36th annual international symposium on Computer architecture, Pages 24-33.
24. P. Zhou et al, 2009 "A Durable and Energy Efficient Main Memory Using Phase Change Memory Technology," ACM/IEEE-Proceedings of the 36th annual international symposium on Computer architecture, Pages 14-23.
25. [25] Lavanya, R., Sivarani, S., Lordwin, C.P., Jeyalakshmi, T., & Muthulakshmi, M.M. (2018). Evaluating the Performance of Various MOEA's to Optimize Scheduling Overhead in Homogeneous Multicore Architecture. 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), 1-9. DOI:10.1109/icctct.2018.8550921

AUTHORS PROFILE

Dr.M.Lordwin Cecil Prabhaker an Electronics and Communication Engineering Professor, received his Ph.D degree in Information and Communication Engineering from Anna University Chennai, India. He is an active researcher in cutting edge technologies such as, real time systems, embedded systems and Multicore Architecture. He has published more than 10 research papers in various National and International journal.

Dr.R. Saravana Ram received his B.E. degree from Anna University, Chennai, Tamilnadu, India in 2008 and M.E. degree from Anna University, Coimbatore, Tamilnadu, India in 2010. He has completed PhD degree from Anna University Chennai. He has more than 7 years of teaching experience. He has published nearly three research articles in various international conferences/journals. His research area includes Reconfigurable Architecture, Processors, and Very Large Scale Integration.