

Energy Efficiency in Processors- A Survey

Rashmi K Gundakalli,
Student, M.Tech
Computer Science and Engineering,

Mohan G Kabadi,
Dean and Head
Computer Science and Engineering

Abstract-Energy being consumed in a circuit has been a major concern in the electronic industry and also in Digital System Design. Hence, to reduce the energy consumed by the underlying circuits, CMOS circuits were preferred. With the advancement in the integration technology, more transistors could be fit on the chip. This however led to increase in the energy dissipation. In a computer system, the processors memory and the Disk subsystems are the power hungry units. The numbers of computing systems are increasing drastically and will keep dissipating the energy and thus have an impact on environment too. This paper presents a review of various techniques in energy optimisation at all levels of the processor and hence suggests the ways to optimise the energy along the optimisation metrics associated with each of the technique.

Keywords- Energy, cache, hardware, Dynamic Power, Circuits, Efficiency.

I. INTRODUCTION

With continuous improvement in the CMOS technology, more and more transistors are being incorporated on a single die. Majority of these transistors are being used for caches which include trace caches, Level 1 and 2 caches, renaming registers and predictor structures. With increase in the number of transistors, the energy being dissipated increases [9]. The energy being dissipated in any component developed using CMOS Technology can be classified as: Dynamic Power Dissipation, Static Power Dissipation and the Short Circuit Power Dissipation. The dynamic power dissipated is given by [13]:

$$P_d = 0.5 * f_d * C_L * V_{dd}^2 \quad (1)$$

where f is the clock frequency, f_d is the switching activity, C_L is average capacitance loading of the circuit, and V_{dd} is the supply voltage. Reducing f_d , C_L or V_{dd} leads to reduction in overall dynamic energy consumption. Similarly, Static power essentially consists of the power used when the transistor is not in the process of switching and is determined by [40]

$$P_{static} = I_{static} V_{dd} \quad (2)$$

where, V_{dd} is the supply voltage and I_{static} is the total current flowing through the device. CMOS technology has been praised for its low static power. Whenever, gate oxide thicknesses decrease and probability of tunnelling increases, this leads to larger amount of leakage current. [cmos tech].

Collectively, the average power being dissipated is being given by [2]

$$P_{avg} = P_{switching} + P_{shortcircuit} + P_{static} = \alpha C_L V_{dd}^2 f + I_{static} V_{dd} + I_{leakage} V_{dd} \quad (3)$$

where, $P_{switching}$ is the dynamic component of power, C_L is the load capacitance, f is the clock frequency, and α is the node transition activity factor. This equation also assumes the voltage swing is equal to the supply voltage, V_{dd} [2]. $P_{shortcircuit}$ is due to the direct-path short circuit current I_{static} , which arises when both the NMOS and PMOS transistors are simultaneously active, conducting current directly from supply to ground. This conduction will be for very short period of time. Significant short-circuit power dissipation can be avoided if the output rise/fall time of a gate is much longer than the input rise/fall time. The switching power is given by [8]

$$P_{switching} = C_L \cdot V_{DD}^2 \cdot a \cdot f \quad (4)[8]$$

where a , is the activity factor.

Lowering V_{DD} increases circuit delay and consequently reduces the overall throughput of device, suggestive of need for low voltage design techniques. However, lowering of voltage means rising noise, cross-talk etc. A next level solution could be to minimizing the switching activity of the circuit. This method can be used to reduce power consumption once the supply voltage and process in the processor are chosen. Currently most of the research is being carried out at the layout and logic levels [13].

Much parallel to this, energy efficiency techniques at cache, bus, processor and memory levels are being designed. The energy dissipated, when individually evaluated, energy optimisation techniques are applied range from 18-50%.

Dynamic power can be reduced by in one of the two ways: either by reducing the number of gates or by reducing the clock frequency [8]. The system level clock gating disables the entire block along with its functionality. Whereas, combinational and sequential clock-gating selectively suspend clocking while the block continues to produce output. The method has proven to be efficient by reduce the energy so consumed by 15-64%.

Md. Qadri et al [8]. further suggest power gating to reduce sub-threshold and gate leakage current and go on to give the following figure of gating circuit. There review on power gating suggests that, by applying the following

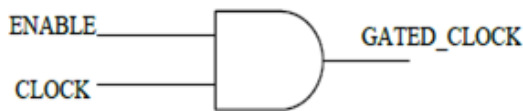


Fig.1. Gating Circuit

circuit to header or footer of a switch, while the logic block is not active, a sleep signal can be applied to turn off the logic block. Hence the logic block is disconnected which in turn reduces the leakage current.

At hardware level, processors are mostly being cooled by the fan with algorithms which make use of digital sensors and various other algorithms to control the speed of the fan that is a part of SMPS [16].

Micro-architectures such as IVY Bridge make use of Thermal Interface Material (TIM) that lies between the chip and the heat spreader. However TIM has been under speculation since reports suggest that IVY Bridge is 10 °C higher compared to Sandy Bridge when a CPU is over-clocked, even at default voltage setting due to low quality of TIM. The lower thermal conductivity has led the heat to trap on the die. Intel claims that the smaller die of Ivy Bridge and the related increase in thermal density is expected to result in higher temperatures when the CPU is over-clocked; Intel also stated that this is as expected and will likely not improve in future revisions.

Even the most recent micro-architecture, Haswell is around 15 °C hotter than Ivy Bridge, while clock frequencies of 4.6 GHz are achievable. However, the TDP[b3] values attained by Haswell range from 10 to 47 W for various versions of it.

Comparatively, BroadWell micro-architecture, gives TDP values from 3.5W to 6W for operating frequency ranging from 600MHz to 1.4GHz.

II. ENERGY OPTIMISATION TECHNIQUES IN CMOS TECHNOLOGY

The following sections give detailed review of the energy optimisation techniques adopted components such as cache, bus, processor etc. In a CMOS circuit, the power is dissipated in the following two ways: Static Power Dissipation and Dynamic Power dissipation.

A. Dynamic Power dissipation

In a CMOS circuit, switched power dissipation is caused due to continuous charging and discharging of the load capacitance. This charging and discharging is necessary for transmitting the load across the CMOS circuits [10].

Short Circuit Power is dissipated due to zero rise and fall time of the signals. If the input signals to a gate are distorted in time, there is absolute signal of present danger in a generated glitch at the output. If a glitch approaches at the input of a gate and if the input is sensitive at the moment, a propagated glitch will be created [10]. In some of the circuits this glitch power so dissipated constitutes a major part as they have peak voltages and hence cannot be ignored.

B. Static Power Dissipation

In CMOS circuits, whenever there is no activity and all the transistors are being biased to a particular voltage, the power is still dissipated. The power so dissipated is called Static Power. The major components of this static power are reverse biased leakage current and sub-threshold current. This sub-threshold leakage current is more profound and hence has attracted the attention of many researchers. The sub-threshold currents mainly depend on the threshold voltage. Even below this threshold voltage value, the transistors are not completely off. Another major cause of concern comes from the leakage currents. These currents are circulated in the circuit whenever the clock falls. These leakage currents are prominent in large circuits. With decreasing die size; the leakage current increases as the feature size is scaled down.

The following are some of the ways to overcome the aforementioned problems. To overcome the sub-threshold leakage current, the sub-threshold voltage can be decreased. However, this in turn can increase the leakage currents. Therefore, an optimal sub-threshold and supply voltages to be connected to the circuit. Yet another solution could be to use architectural scaling [10], however, the solution is not that feasible as extra hardware components would be required. The power burden of the additional circuit, if compensated, the technique is suitable otherwise it is not possible to use the technique. Apart from this, reducing clock frequency can also prove to be useful. A method called clock gating can be used to make the clock inactive over the applications that are currently not being used. On circuit level, circuits performing XOR and flip flops with fewer numbers of transistors are much sought after [10].

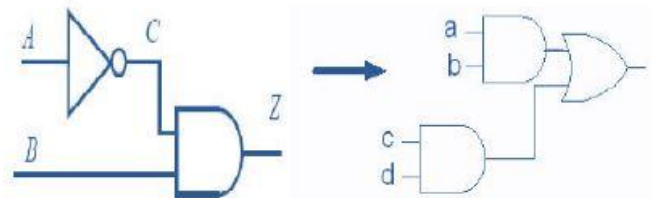


Fig. 2. Arrangement of logical gates to reduce the glitches in the circuit.

Figure 2 shows a method to reduce glitches in the circuit. At the transistor level, type of transistor called the sleep transistor can be used to reduce the static power. These sleep transistors are connected in series with low V_T . Whenever the circuit containing the transistors with low V_T gets on, the sleep transistor are also put on. Since High threshold (V_T) is connected in series with Low threshold (V_T) circuit the leakage current power loss is measured by High threshold (V_T) i.e. the sleeping transistor devices and is quite low. Therefore, the resultant static power dissipation is reduced.

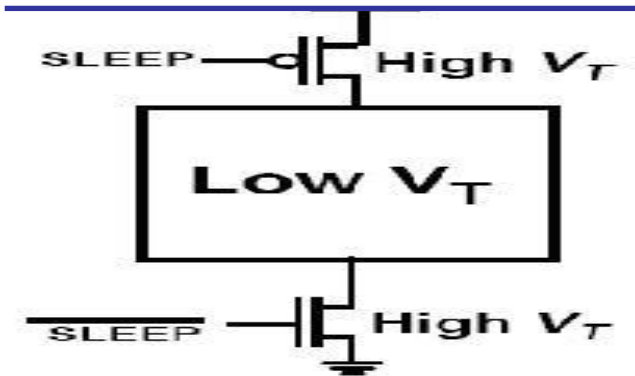


Fig.3 Sleep Transistors

In CMOS circuits, the threshold voltage can be varied dynamically. A low threshold voltage allows higher current drives in the active mode of operation. However, high threshold voltage in standby mode, gives low leakage currents. To reduce the short circuit current, when the load capacitance is very large, the drain source voltage in PMOS transistor is Zero. Hence the short circuit power is also zero [10].

III. ENERGY OPTIMISATION AT THE CACHES

Caches consume major portion of on chip transistor energy. The present day processors consists of atleast two on chip caches. Hence, the caches that are on the chip are the most preferred components for energy control. Mohan G Kabadi et al [5]. suggested the removal of dead block information. The following are the two ways to reduce the power consumed by the cache:

- At the hardware level, reduce the supply voltage to the block in the cache which are found containing dead information. The most appropriate way to do this is to, observe the inactive period of the cache and hence reduce the voltage of the cache for that period. This technique is shown as most effective for the I-cache of level 1,
- Mohan G Kabadi et al.[5] have also suggested that, the above method can be used on the block level in the cache. That is, to reduce the voltage of the block in the cache that is inactive at the moment. However this requires hardware modifications which requires addition of bits such as turn-off bit and the flag bit and work on the annotations of basic block and call instructions.

C. Heather Hanson suggestion

Heather Hanson et al. [9], have suggested the following three ways to optimise the energy consumed by the caches:

- In the first method, it is shown that, the caches should be designed with the transistors that use high voltages. Since the transistors that use higher V_T give lesser leakage currents. The amount of leakage current being dissipated is controlled at the design phase itself. However, such transistors possess slow switching speeds.

- The second method, whenever dynamically adjusts the effective size of the array by employing a circuit technique dubbed *gated- V_{DD}* . In this method, a low leakage transistor is used to control a subset of transistors. Whenever, this subset of transistors has to be deactivated, a sleep signal is applied to the low leakage transistor.
- A third technique called the *MTCMOS*, changes the threshold voltage by modulating the back-gate bias voltage. By using this technique, memory cells can be placed into a low-leakage “sleep” mode yet still retain their state. Cells in the active mode are accessed at full speed. However, accesses to cells in the sleep mode must wait until the cell has been awakened by adjusting the bias voltage. The MTCMOS technique has been implemented for an entire SRAM. This method proves to be advantageous because in this method the threshold voltage is dynamically being changed. This in turn helps to save the state of the memory cells along with any further cache misses. But the method can as well be disadvantageous as the cells have to woken up from the sleep mode before the contents of the cell can be accessed, it adds to additional overhead in terms of access time.

IV. ENERGY OPTIMISATION IN BUSES

While transmitting data on buses, major amount of energy is being dissipated due to switching activity on the buses. This type of energy can be classified as dynamic power. Literature survey suggests the use of techniques such as BITS and BI. However, these techniques are constrained by the fact that these methods require extra hardware to indicate whether the data is encoded or decoded [12].

Isha Sood et al. [12] have suggested various methods which make use of XORing of bits at the encoding side. XOR operation is particularly being selected, because, XOR operation consumes lesser power [18]. For this, gray code is being used. The MSB bit of the present greycode so encoded is sent along with the output of the XOR operation. Depending on the bit value, XNOR operation is being performed for n-1 bits of the current sequence and the previously encoded sequence.

Similarly at the decoder side, XOR operation on the previous encoded sequence and lower n-1 bits of the current sequence is performed. This operation is followed by XNOR operation based on the bit value. The method has been evaluated to be 24% power efficient [18].

They have also suggested another technique called the Modified Bits technique. Which make use of XORing the bits at 3rd and 4th positions followed by an AND operation at the encoding side. A similar operation is being carried at the decoder side. The researchers have suggested that this method produces better energy efficiency than the earlier method.

V. ENERGY OPTIMISATION IN MEMORY

D. Energy efficiency in SRAM

SRAM array or sub-array consists of set of rows and columns. The selection of number of column is an important factor while designing the array structure. The number of column, j is given by

$$j = \sqrt{D} \times \sqrt{\frac{P_r \times 0.5C_{RBL} + P_w \times C_{WBL}}{P_r \times C_{RWL} + P_w C_{WWL}}}$$

Where, j is the number of columns, C_{RWL} is the read wordline capacitance per cell, C_{RBL} is the read bitline capacitance per cell, C_{WWL} is the write wordline capacitance per cell, C_{WBL} is the write bitline capacitance per cell, and V_{DD} is the supply voltage, P_r is the probability of read and P_w is the probability of write. In the read energy equation[14], it is assumed that the probabilities of data "1" and those of data "0" are equal and are 0.5. The dynamic energy component is mainly determined by the wordline capacitance and the bitline capacitance, while the static energy component coming from the leakage current is determined by the memory density. The effects of the read and write operations on the leakage current of the accessed row and column are insignificant. Thus, they are neglected in the energy estimation.

From figure 4, it can be inferred that, the energy is minimum at $k=128$, $V_{DD}=1.2V$ and $1.0V$. Hence, $j=64$, which is indicative of minimum switching capacitance. With decrease in V_{DD} , the number of rows for energy efficient array structure also decreases. This observation is particularly true when V_{DD} is high and the dynamic energy component is an important factor. But when $V_{DD} \approx V_{TH}$, the changes in the number of rows and column becomes significant as depicted in the above figure (c) and (d). Here, the minimum energy points are found at $k = 32$ and $j = 256$ at $V_{DD} = 0.4 V$ and at $k = 16$ and $j = 512$ at $V_{DD} = 0.3 V$. When Compared to the optimal SRAM array structures at $V_{DD} = 1.0 V$, j and k have stronger impact on energy consumption. Lowering supply voltage transforms the SRAM array structure for minimum energy from results to a short and wide structure. Also, Fig. 4(d) does not show results for array structures with rows >64 due to excessive leakage and bitline discharge failure.

Simulation results demonstrate that the energy reduction up to 10% can be achieved using the optimal number of rows in the 8-kbit array structure operating at 0.4 V. The energy reduction is further enhanced when leakage energy becomes more significant. In the 64-kbit (8 kbit \times 8 banks) SRAM array, the optimal number of rows can improve the energy efficiency up to 38% at 0.4 V when compared with the array with 128 rows. It can be inferred that, in larger SRAMs where majority of the arrays are not activated, wider array structures are more beneficial in terms of energy efficiency.

Brian Zimmer et al.[39] are further suggestive of the fact that the technology under discussion is low Power CMOS technology, hence leakage current was negligible for worst case scenarios such as a SRAM consisting of 512 cells.

However, leakage can easily be taken into account by using Monte Carlo to characterize the leakage current of N worst-case cells as a lognormal and including it into IS as an additional variable described by the fit distribution.

E. Energy Optimisation in DRAM

The research conducted by ZiliShao et al.[29] indicates that the Phase Change Module is an promising alternative for DRAM due to its high density, bit alterability, and low standby power. However, some existing compiler techniques may need to be revised in order to solve problems caused by the disadvantages of PCM writes. PCM writes may be scheduled into positions so as to introduce minimum time delay or minimum power variation.

At the operating system level, Zili Shao et al [29] are further suggestive that if OS can be modelled such that it would group pages which are not write sensitive into a single bank; this would help save the energy.

Researches further lay down that PCM, unlike DRAM, need not be refreshed, resulting into energy optimisation.

Loi et al.[11] have researched that, a workload equivalent to 50% page misses, every second data access to the 3D-DRAM [3] opens and closes a new row. As result of which, the total power in application relevant mode is about 39% between bandwidth switching and disable for the 2048 Mb density. To reduce the energy being consumed, Mingson Bi et. al.[42] have given the following method. They suggest that predict ranks in the DRAMS at the system call entry level where the method to do this being Most Recently Accessed (MRA). This mechanism predicts the current I/O will access same rank as that of last I/O. The buffers are being allocated in the order they are being accessed. MRA method is said to be energy efficient since only one rank is being predicted and turned on for each I/O. When an incorrect rank is turned on, the transition of the actual accessed rank is delayed until the request arrives at the memory controller, exposing the full transition delay. They further go on to suggest that an early turn-off optimisation which can predict the rank to be accessed and turned off the un-need active ranks can save the energy. An additional variable to record the rank that the corresponding block is placed is added to the cache buffer.

VI. ENERGY OPTIMISATION AT THE COMPILER LEVEL

Compiler optimization have the potential for energy savings with less or no changes to the existing hardware or software. Compilers can include statement in object code to put the functional units not in use to lower power mode for a longer period. However, most current compiler optimizations focus on improving execution time. Compiler optimizations that are traditionally used to increase performance have shown much promise in also reducing cache energy consumption.

The following are some of the methods that can be adopted for achieving energy optimisation at the compiler level.[15]

F. Loop Unrolling

Loop unrolling removes the execution of many of the branches found in the loop iteration limit test code. It also

has the capabilities of improving software pipelining. However, loop unrolling increases the length of the code. Thus, increasing the per-access energy cost of the instruction memory. Hence, it would be important for the compiler to leverage the performance.

G. Function In-lining

It is an inter-procedural compiler optimisation method wherein the code for the procedure call is replicated at a call site. By applying Inlining eliminates the overhead related to the procedure call. Inlining further paves for code or data optimisation. The disadvantage with this function inlining is that it increases the code size. It would also lead to an unwanted instruction memory energy. Therefore it is necessary to strike a balance between the energy consumption and proportional performance gains.

H. Data Transformation[6]

The increased inter-access times would provide an opportunity for operating in a lower power mode for a longer time. The increased inter-access time could be due to two different reasons. First, improved cache behaviour can increase the time between two accesses to memory. Secondly, the memory accesses can be confined to a particular memory module for a period of time allowing the other modules to operate in a lower power mode for extended periods of time. If it is stored in row-major form and accessed in column-major form, successive references may access different modules. However, if it is stored and accessed in column-major form, successive accesses will be confined to the same module except at module boundaries.

VII. CONCLUSION

CMOS technology has proven to be power efficient, but is accompanied by static power and dynamic power and hence needs optimisations at several levels starting from circuit level to application level. However, these optimisations as well come with disadvantages such as addition of extra hardware or changes at the software level. This is only suggestive that more research is required to achieve better performance in terms of energy without trading off hardware or software level changes.

ACKNOWLEDGEMENT

The support of the guide, Lectures and friends at Acharya Institute of Technology and family members is greatly acknowledged.

REFERENCES

- [1] Po Kuan Huang and Soheil Giasi, "Efficient and Scalable Compiler Directed Energy Optimisation for Realtime Applications", EDAA, 2007.
- [2] Jayanth Srinivasan, "An Overview of Static Power Dissipation".
- [3] Shekar Borkar, "3-D Integration for Energy Efficient System Design", Symposium on VLSI Technology, 2009.
- [4] Jhon Sartori and Rakesh Kumar, "Compiling for Energy Efficiency on Time Speculative Processors", DAC, 2012.
- [5] Mohan G Kabadi and Ranjani Parthasarathi, "Using Dead Block Information to Minimise I-Cache Leakage Energy", IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.5, May 2007.
- [6] Mahmut Kandemir, N. Vijaykrishnan and Mary Jane Irwin, "Compiler Optimisation for Low Power Systems".
- [7] Ulrich Kremer, "Low Power/Energy Compiler Optimisation".
- [8] Muhammad Yasir Qadri, Hemal S Gujarathi and Klaus D McDonald Meier, "Low Processor Architecture and Contemporary Techniques for Power Optimisation- A Review", Journal Of Computers, 2009.
- [9] Heather Hanson, M.S. Hrishikesh, Vikas Agarwal, Stephen W Keckler and Doug Burger, "Static Energy Reduction Techniques for Microprocessors Caches", IEEE Transactions on Very Large Integration, 2002.
- [10] Amit Singh Gaur and Jyoti Budakoti, "Energy Efficient Advanced Low Power CMOS Design to reduce Power Consumption in Deep Submicron Technologies in CMOS circuit for VLSI Design", International Journal of Advanced Research in Computer and Engineering, 2014.
- [11] Christian Weis, Igor Loi, Luca Benini and Noebert Wehn, "An Energy Efficient DRAM Subsystem for 3-D integrated SoCs", EDAA, 2012.
- [12] Isha Sood and Candy Goel, "Power Aware Data bus Encoding and Decoding Schemes".
- [13] Ching Long Su and Alvin M Despain, "Cache Design for Energy Efficiency", Proceedings of the 28th Annual Hawaii International Conference on System Sciences, 1995.
- [14] Achiranshu Garg and Tony Tae Hyung Kim, "SRAM Array Structures for Energy Efficiency Enhancement", IEEE transactions on Circuits and Systems-II, 2013.
- [15] Joseph Zambreno, Mahmut Taylan Kandemir and Alok Chaudhary, "Enhancing Compiler Techniques for Memory Energy Optimisations", EMSOFT 2002, LNCS 2491, pp. 364-381, 2002, Springer-Verlag Berlin Heidelberg 2002.
- [16] Desktop 4th generation Intel® Core™ Processor Family, Desktop Intel®, Pentium® Processor Family, Desktop Intel® Celeron® Processor and Intel® Xeon® Processor E3-1200 v3 Product Family.
- [17] Intel® 64 and IA-32 Architecture Optimisation Reference Manual.
- [18] Subhod Wairya, Rajendra Kumar and Sudharshan Tiwari, "Comparative Performance Analysis Of XOR-XNOR Function Based High Speed CMOS Full Adder Circuits For Low Voltage VLSI Design", International Journal of VLSI design & Communication Systems (VLSICS) Vol.3, No.2, April 2012.
- [19] Olivera Jovanovic, Peter Marwede, Iuliana Bacivarov and Lothar Thiele, "MAMOT: Memory-Aware Mapping Optimization Tool for MPSoC". In 15th Euromicro Conference on Digital System Design, 2012.
- [20] Patrick P. Gelsinger, Intel Corporation, Hillsboro, OR, "Microprocessors for the New Millennium: Challenges, Opportunities, and New Frontiers".
- [21] Allen C. Cheng and Gary S. Tyson, "An Energy Efficient Instruction Set Synthesis Framework for Low Power Embedded System Designs", IEEE Transactions On Computers, VOL. 54, NO. 6, June 2005.
- [22] Saurabh Dighe, Sriram R. Vangal, Paolo Aseron, Shashi Kumar, Tiju Jacob, Keith A. Bowman, Jason Howard, James Tschanz, Vasanth Erraguntla, Nitin Borkar, Vivek K. De and Shekhar Borkar, "Within-Die Variation-Aware Dynamic-Voltage-Frequency-Scaling With Optimal Core Allocation and Thread Hopping for the 80-Core TeraFLOPS Processor", IEEE Journal Of Solid-State Circuits, Vol. 46, No. 1, January 2011.
- [23] Ozcan Ozturk, Mahmut Kandemir, and Guangyu Chen, "Compiler-Directed Energy Reduction Using Dynamic Voltage Scaling and Voltage Islands for Embedded Systems", IEEE Transactions On Computers, Vol. 62, No. 2, February 2013.
- [24] Efraim Rotem, Alon Naveh, Doron Rajwan, Avinash Ananthakrishnan, Eliezer Weissmann, Intel®, "Power-Management Architecture Of The Intel Micro-architecture Code-Named Sandy Bridge".
- [25] Vinay Hanumaiah and Sarma Vrudhula, "Energy-Efficient Operation of Multicore Processors by DVFS, Task Migration, and Active Cooling", IEEE Transactions On Computers, Vol. 63, No. 2, February 2014.
- [26] Alaa R. Alameldeen, Ilya Wagner, Zeshan Chishti, Wei Wu, Chris Wilkerson and Shih-Lien Lu, "Energy-Efficient Cache Design Using

- Variable-Strength Error-Correcting Codes", ISCA'11, June 4-8, 2011, San Jose, California, USA.
- [27] WissamChedid, Chansu Yu and Ben Lee, "Power Analysis and Optimization Techniques for Energy Efficient Computer Systems".
- [28] V. Delaluz, M. Kandemir, N. Vijaykrishnan, M. J. Irwin, A. Sivasubramaniam, and I. Kolcu, "Compiler-Directed Array Interleaving for Reducing Energy in Multi-Bank Memories".
- [29] Nadeem Firasta, Mark Buxton, Paula Jinbo, Kaveh Nasri, Shihjong Kuo, Intel Software Solutions, White Paper, "Intel® AVX: New Frontiers in Performance Improvements and Energy Efficiency".
- [30] Zili Shao, Yongpan Liu, Yiran Chen, and Tao Li, "Utilizing PCM for Energy Optimization in Embedded Systems", An invited paper in IEEE computer Society Annual Symposium on VLSI, 2012.
- [31] Efi Rotem, Alon Naveh, Micha Moffie and Avi Mendelson, "Analysis of Thermal Monitor features of the Intel® Pentium® M Processor".
- [32] Souvik Singha and G.K. Mahanti, "Optimisation of Delay and Energy in On-Chip Buses using Bus Encoding Technique", International Journal of Computer Applications (0975 – 8887) Volume 86 – No 12, January 2014.
- [33] Mamtha S., Centre for VLSI and Embedded System technologies, International Institute of technology, Hyderabad, "Energy Efficient SRAM".
- [34] Punya Prakash and Khazinobu Shin, Texas Instruments, "Power Optimisation Techniques for Energy Efficient Systems".
- [35] Wissam Chedid and Chansu Yu, Department of Electrical and Computer Engineering, Cleveland State University, "Survey on Power Management Techniques for Energy Efficient Computer Systems".
- [36] John S Seng and Dean M Tullsen, "Architecture level Power Optimisation-What are the limits?", Journal Of Instruction Level Parallelism, 2007.
- [37] Rakesh Kumar, Keith Farkas, Norman P Jouppi, Partha Ranganathan and Dean M Tullsen, "A Multi Core Approach to Addressing the Energy Complexity Problem in Microprocessors", In Proceedings of the Workshop on Complexity Effective Design", June, 2003.
- [38] Jane Christian Meyer and Lasse Natvig, "Power Instrumentation of Task Based Applications Using Model Specific Registers in the Sandy Bridge Architecture", Partnership for Advanced Computing in Europe.
- [39] Qiang Wu, V.J. Reddi, Youfeng Wu, Jin Lee, Dan Connors, David Brooks, Margaret Martonosi, Douglas W. Clark, "A Dynamic Compilation Framework for Controlling Microprocessor Energy and Performance".
- [40] Brian Zimmer, Seng Oon Toh, Huy Vo, Yunsup Lee, Olivier Thomas, Krste Asanović and Borivoje Nikolić, "SRAM Assist Techniques for Operation in a Wide Voltage Range in 28-nm CMOS", IEEE Transactions On Circuits And Systems—II: Express Briefs, Vol. 59, No. 12, December 2012.
- [41] Shrikanth Iyer, Stanford University, "CMOS Power Consumption", 2010.
- [42] Matt Bach, "Impact of Temperature on Intel® CPU Performance", 2014.
- [43] Mingson Bi, Ran Duan and Chris Gniady, "Delay Hiding Energy Management Mechanisms for DRAM".
- [44] Juan M Cebrian, Lasse Natvig and Jan Cristian Meyer, "Improving Energy Efficiency through Parallelisation and Vectorisation in Intel® Core™ i5 and i7, 2012 SC Companion: High Performance Computing, Networking Storage and Analysis.