

# Reducing Power Consumption at Computer Architectures to Improve the Performance

**Rajesh Mallela<sup>1</sup>, K. Suresh<sup>2</sup>, Kadiyala Ramana<sup>3</sup> and M. Subba Rao<sup>4</sup>**

1, 2, 3 Assistant Professor, Department of IT, AITS, Autonomous, Rajampet, AP.

4 Professor, Department of IT, AITS, Autonomous, Rajampet, AP.

(razeshmallela, sureshkallam, ramana.it01, msraoswap@gmail.com)

## ABSTRACT

Power and Energy Consumption is a Key constraint and crucial Technology for High Performance computing era. Energy optimization is an enabling power Management. The Consumption of Energy should be ascertainable not only to Gate Level or Register Transfer (RT) Level but also to the System Level. Reducing the Energy Consumption without degrading performance of the system. The compiler-time power and Energy optimization that will be complementary to current hardware and Operating System Techniques. The Computing science developers focus on the compiler-time power and energy optimization that will be complementary to current hardware and Operating System techniques, compiler have the advantage that they can analyse whole program behaviour, and reshape the behaviour if considered profitable for a given optimization objective. Power and Energy management strategy will be investigating optimization criteria to minimization of overall energy consumption. The focuses on various and tools for minimization of energy without increasing runtime. The Energy consumption and run time computed for various compiler techniques on XScale Architecture using XEEMU tool. The optimized code picked out and code is tuned dynamically by varying voltage-frequency. The optimized codes are tuned dynamically.

**Key Words:** Compiler Optimization, Performance Evaluation, Voltage-Frequency Scaling, XscaleArchitecture.

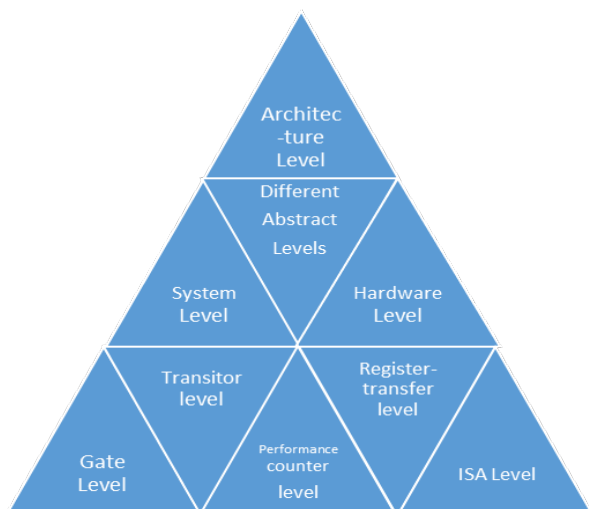
## 1. INTRODUCTION

In present day world every joule of energy is valuable because all aspects of our system are related to energy consumption. Energy has become an important aspect of life as the factors that generate power are on the edge of extinction.

So it has become very important for us to conserve energy for future in any form like computing systems, which can be either by battery driven or driven by ac power supply. By using effective operating system, the consumption of energy can be reduced. This can be applicable in compiling programmes on system and by using compatible machine codes. Power aware compilation is technique by which we make every developer or user to know the amount of energy used by their codes. If it is reasonable our system reduces the consumption of energy. Performance is always plays major role in Computer Science. Most power reduction techniques focused on minimizing the static power consumption rather than system level dynamic power consumption.

Energy is an essential asset because the factors that generate it are mainly depleting resources. The effect of reconfiguration granularity particularly on energy savings is also analysed with help of compiler approach to optimize energy results is presented. Hence it becomes an implicit requirement to conserve energy, be it in any form i.e. Computing systems, which may be either battery driven or driven by AC power supply. Power Consumption can be reduced by having efficient operating systems that consume lesser power. The same can be applied while compiling programs on systems where we can produce energy efficient machine codes. We propose a technique called power aware compilation. Using this technique, each and every developer or user could know the amount of energy consumed by their code; further, if feasible our system optimizes the energy consumption. The energy efficiency of them is becoming an important issue. The processor is one of the most important power consumers in any computing system. Considering that state-of-the-art real-time systems[17] are evolving in complexity and scale, the demand for high-performance processors will continue to increase. A processor's performance, however, is directly related to its power

consumption. As a result, the processor power consumption is becoming more important issue as their required performance standards increase. With power having become a critical issue in the operation of data centers today, there has been an increased push towards the vision of "energy-proportional computing"[18], in which no power is used by idle systems, very low power is used by lightly loaded systems, and proportionately higher power at higher loads. Unfortunately, given the state of the art of today's hardware, designing individual servers that exhibit this property remains an open challenge. However, even in the absence of redesigned hardware, we demonstrate how optimization-based techniques can be used to build systems with off-the-shelf hardware that, when viewed at the aggregate level, approximate the behavior of energy-proportional systems. The increasing importance of Energy consumption and power reduction are the major problems for computer systems. From computer to smart phones, in order to run these devices all we need is power. Low power design is a critical design consideration even in high-end computer systems where expensive cooling and packaging costs and lower reliability often associated with high levels of on-chip power dissipation are the important concerns. we are trying to reduce the consumption of power on Chip-Level[19], Gate-Level[20], Operating System Level[21], Processors and Compiler Level[22], but we are reducing the power at compiler level. When it comes to computer scientists a steady progress has been achieved basically in the form of Dynamic power management (DPM) and Dynamic voltage scaling (DVS)[23].



**Figure 1 Classification of model**

## 2. RELATED WORK

The most effective power reduction technique is Dynamic voltage scaling. This result reducing the power supply voltage that can notably reduce power dissipation. It could be appropriate for eliminating idle times at low workload hours. So power is not wasted by an idle processor. CPU consumes much power in convex fashion with frequency that can be reduced by using dynamic voltage scaling which makes CPU lower dynamic energy consumption.

Power-reduction can be done in two ways static and dynamic. Static techniques are applied at the time of design, such as compilation. Dynamic techniques are applied at the time of run time based on the workloads. Dynamic power management (DPM). When high performance is required, DPM allows hardware to consume more power; otherwise, the hardware enters a lower-power state. DPM techniques include dynamic voltage/frequency scaling (DVS/DFS) and clock gating. DVS/DFS finds the program section where voltage and frequency can be tuned on CPU with minimum loss in performance. To maintain the both energy and performance is vital role in DVS was introduced, this will help to apply different voltages for different executions of frequencies. (DVS) will allows the devices with change in voltage, increasing energy levels and efficiency of their operation in progress. DVS is used to reduce power by varying the voltages according to the load on the processor.

Basically processors obtaining a power in two ways. One is through a compiler, second is an Assembly code manipulation or by another non-compiler method. Dynamic voltage scaling is a non-compiler method. On-compiler method checks the load on the processors and dynamically increases or decreases the processor frequency. DVS is one of the feasible and effective solutions to power reduction techniques. As a result, lowering the supply voltage can reduce significantly lowering the power dissipation. It is suitable for eliminating idle times during low workload periods it leads no power wasted by an idle processor usually.

Since the System processor power consumption increases in convex fashion, but DVS will help to considerably reduce the system energy consumption. (DVS) is a mechanism dynamically adjust CPU voltage and frequency. DVS in embedded devices variation in processor utilization, lowering the frequency when the processor in less load, and running at maximum

frequency when the processor is very largely loaded. DVFS will reduce energy systems. Because the frequencies are proportional to voltages.

A major challenge in DVS are utilizing the application are need to reduce the power. Voltage scaling is a common technique to reduce power by simply adjusting the supply voltage either at design time or at run time to maximize energy efficiency. The developer can implement different optimization techniques and can choose the one which gives the best result in terms of energy (Joule) and run-time (Sec). The code can be tuned dynamically by varying frequency and voltage across the blocks or the regions in the code. In such a way that minimization in the energy consumption can also be obtained dynamically.

### 3. ANALYSIS AND PERFORMANCE

The less power consumes by the CMOS Technology. A Power Consumption of CMOS Formula:

$$p = c v^2 f \quad (1)$$

Where  $p$  = power in watts,  $c$  = switch capacitance,  $v$  = supply voltage, and  $f$  is the clock frequency in hertz [15] this Suggests that there are essentially three ways to reduce power:

DVFS technique proposed to achieve low power consumption for the CPU. Describe the relationship between CPU clock frequency, power and energy using the equations provided in the Intel optimization documentation. We let  $V_{dd}$  represent the supply voltage and  $f$ .

$$\text{Power} \propto f V_{dd}^2 \quad (2)$$

$$\text{Delay} = 1/f \alpha 1/V_{dd}$$

$$\text{Energy} \propto V^2 dd \quad (3)$$

Traditional(DVS) will not fit address scaling on system power consumption as the leakage power increases.

The various power analysis tools are JouleTrack [16], WATTCH [17], SimpleScalar [18], XTREM [19], U [20], Simics, Cache Access and Cycle

Time Information: CACTI, Simple Power, General Execution-driven Multiprocessor Simulator (GEMS), and WARTS - Wisconsin Architectural Research Tool Set. Joule Track is MIT research lab product and a very efficient web based tool for software profiling. WATTCH is CPU power estimation tool. It analyses and optimizes power dissipation at micro architectural level, where as Simple Scalar is the complete tool set. XTREM and XEEMU is Xscale architecture specific tool. SIMICS is full system simulator. CACTI is the tool for measuring performance based on cache sizes and organization. GEMS simulator based on SIMICS. WARTS performs profiling and tracing of the programs. Among all XTREM and XEEMU is Intel(c) Xscale(c) architecture specific tool. XEEMU developed to simulate the runtime and power consumption of the Intel(c) Xscale(c) core. With the experimental results it showed XEEMU is faster and efficient than XTREM.

Ideal vitality utilization of  $k$  number, undertaking can be characterized as discovering the best mix of accessible voltages and frequencies to perform a predefined task with  $k$  clock ticks inside a predefined time  $T$ .

$$E^{(k)} = \sum_{i=1}^N t_i^{(k)} P_d(f_i, v_i) + P_T(T^{(k)} - \sum_{i=1}^N t_i^{(k)}) \quad (4)$$

s.t

$$\sum_{i=1}^N t_i^{(k)} f_i = K^{(k)}$$

$$\sum_{i=1}^N t_i(K) \leq I^{(k)}$$

$$t_i^{(k)} \geq 0; \text{ for } i = 1, 2, \dots, N$$

$$\text{Minimize } E = t_1 p_d(f_1, v_1) + t_2 p_2(f_2, v_2) + t_3 p_3(f_3, v_3) \quad (5)$$

s.t

$$1. t_1 f_1 + t_2 f_2 + t_3 f_3 = K$$

$$2. t_1 + t_2 + t_3 = T$$

$$3. t_1 \geq 0; \text{ for } i = 1, 2, 3$$

$$t_2 = T - t_1 - t_3 \quad (5)$$

$$t_1 f_1 + [T - t_1 - t_3] f_2 + t_3 f_3 = K$$

$$t_1 f_1 + T f_2 - f_1 f_2 - t_3 f_2 + t_3 f_3 = K$$

$$[f_3 - f_2] t_3 = K - t_1 f_1 - T f_2 + t_1 f_2$$

$$= K - t_1 f_1 + (t_1 - T) f_2$$

$$T_3 = \frac{(K - T f_2) - t_1 (f_1 - f_2)}{f_3 - f_2} \quad (6)$$

$$f_2 = T - t_1 - t_3$$

$$= T - t_1 - \frac{(K - T f_2) - t_1 (f_1 - f_2)}{f_3 - f_2}$$

$$= \frac{T(f_3 - f_2) - t_1(f_3 - f_2) - (K - T f_2) + t_1(f_1 - f_2)}{f_3 - f_2}$$

Optimize the solution

$$t_2 = T f_3 - T f_2 - t_1 f_3 + t_1 f_2 - K + T f_2 + t_1 f_1$$

$$- t_1 f_2$$

$$t_2 = \frac{(t f_3 - K) + t_1 (f_1 - f_3)}{f_3 - f_2}$$

$$t_1 \geq 0,$$

$$t_2 \geq 0$$

$$(T f_3 - K) + t_1 (f_1 - f_3) \geq 0$$

$$(T f_3 - K) \geq -t_1 (f_1 - f_3)$$

$$(T f_3 - K) \geq t_1 (f_3 - f_1)$$

$$t_1 \leq \frac{T f_3 - K}{f_3 - f_1}$$

$$t_1 \leq \frac{K - T f_3}{f_3 - f_1}$$

$$t_3 \geq 0$$

$$(K - T f_3) - t_1 (f_1 - f_2) \geq 0$$

$$(K - T f_3) \geq t_1 (f_1 - f_2)$$

$$\frac{K - T f_2}{f_1 - f_2} \geq t_1$$

$$t_1 \leq \frac{K - T f_2}{f_1 - f_2}$$

$$t_1 \geq 0$$

Result

$$0 \leq t_1 \leq \frac{K - T f_2}{f_1 - f_2}$$

$$E = t_1 P_1 + t_2 P_2 + t_3 P_3 \quad (7)$$

$$E = t_1 P_1 + \left\{ \frac{(T f_3 - K) + t_1 (f_1 - f_3)}{f_3 - f_2} \right\} P_2 + \left\{ \frac{(K - T f_3) - t_1 (f_1 - f_2)}{f_3 - f_2} \right\} P_3$$

$$= \frac{t_1 P_1 (f_3 - f_2) + (T f_3 - K) P_2 + t_1 (f_1 - f_3) P_2 + (K - T f_3) P_3 - t_1 (f_1 - f_2) P_3}{f_3 - f_2}$$

$$= t_1 \left[ \frac{P_1 (f_3 - f_2) + (f_1 - f_3) P_2 - (f_1 - f_2) P_3}{f_3 - f_2} + \frac{P_2 (t_3 f_3 - K) + (K - T f_2) P_3}{f_3 - f_2} \right]$$

$$= t_1 \alpha + \beta$$

$$\text{if } \alpha \leq 0$$

$$t_1 = \frac{K - T f_3}{f_1 - f_2}$$

$$t_3 = 0$$

$$t_2 = T - t_1 - t_3$$

$$t_2 = T - \frac{K - T f_2}{f_1 - f_2}$$

$$= \frac{T f_1 - T f_2 - K + T f_2}{f_1 - f_2}$$

$$= \frac{f_1 - f_2}{T f_2 - K}$$

• The objective is to apply compiler optimization transformations which help in maximum energy reduction.

• Besides reducing energy consumption of the program, it should not degrade its quality of execution, so minimization of Runtime in Sec and Number of instructions executed is also considered.

• The optimized code is taken and tuned dynamically by varying voltage-frequency and the best combination of voltage-frequency taken out.

There are various types of power models available followed by various tools available for power performance evaluation. The power models are classified on the basis of the level of abstraction of the description of the system and are reviewed. These are Transistor Level Power Estimation, Gate Level Power Estimation, RT Level Power Estimation, and System Level Power Estimation. The power model gives the measurement of power dissipated or power consumed by any system. According to the abstraction level the effect on power estimation accuracy, simulation time and power saving Opportunity is explained. Here Power Estimation Accuracy and Simulation time reducing from bottom to top, whereas Power Saving Opportunity is reducing from top to bottom. Basically two methodologies exist for estimating the power dissipation at different levels of abstraction. These are simulation based methods and probabilistic methods. Along with the power models there are various power analysis tools available now days. It measures the power performance evaluation of laptop to smart phones, from tablet PCs to hand held devices. Scientists are putting great effort in the development of power analysis tools for various devices and its implementation in various domains. The various power analysis tools are Joule Track[31], WATTCH[32], Simple Scalar[33], XTREM[34], XEEMU[35], Simics, Cache Access and Cycle Time Information:

CACTI, Simple Power, General Execution-driven Multiprocessor Simulator (GEMS), WARTS - Wisconsin Architectural Research Tool Set. WATTCH is CPU power estimation tool. It analyses and optimizes power dissipation at micro architectural level, where as Simple Scalar is the complete tool set. XTREM and XEEMU is XScale architecture specific tool. SIMICS is full system simulator. CACTI is the tool for measuring performance based on cache sizes and organization. GEMS simulator based on SIMICS. A WART performs profiling and tracing of the programs. Among all XTREM and XEEMU is Intel(c) XScale(c) architecture specific tool. XEEMU developed to simulate the runtime and power consumption of the Intel(c) XScale(c) core. With the experimental results it showed XEEMU is faster and efficient than XTREM.

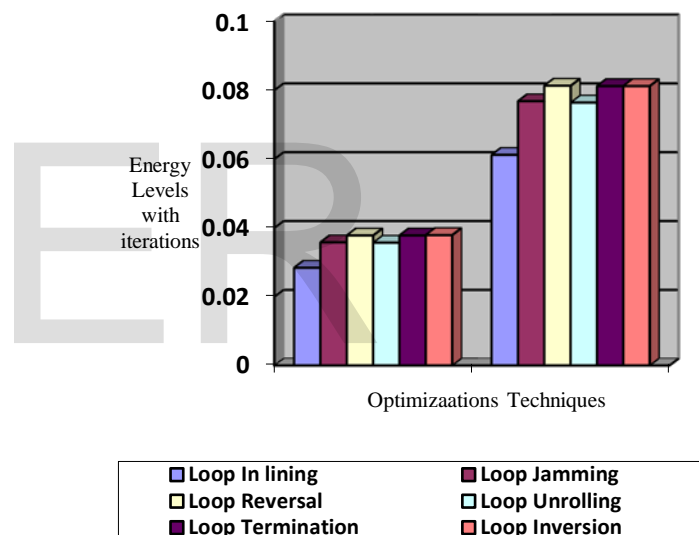
**ENERGY**

The energy E, measured in Joules (J), consumed by a computer over T seconds is equal to the integral of the instantaneous power, measured in Watts (W). The instantaneous power consumed by components implemented in CMOS, such as microprocessors and DRAM, is proportional to  $V^2 \times F$ , where V is the voltage supplying the component, and F is the frequency of the clock driving the component. Thus, the power consumed by a computer to, say, search an electronic phone book, may be reduced by reducing V, F, or both. However, for tasks that require a fixed amount of work, reducing the frequency may result in the system taking more time to complete the work. Thus, little or no energy will be saved. There are techniques that can result in energy savings when the processor is idle, typically through clock gating, which avoids powering unused devices. In normal usage pocket computers run on batteries, which contain a limited supply of energy. When the system is idle, the integrated power manager disables the processor core but the devices remain active. If the system clock is 206 MHz, a typical pair of alkaline batteries will power the system for about 2 hours; if the system clock is set to 59 MHz, those same batteries will last for about 18 hours. Although the battery lifetime increased by a factor of 9, the processor speed was only decreased by a factor of 3.5. Based on proposed optimal energy consumption computing for kth task of different voltage frequencies DVFS based the closest energy optimized calculated from equation 1 to 7.

Table 1 Possible frequency and voltage Combinations(XScale)

Optimization Techniques	Eavg(Average Energy Performance Percentage)	Rtavg(Average Runtime Performance Percentage)
Loop In lining	0.0284	0.0612
Loop Jamming	0.0358	0.0768
Loop Reversal	0.0378	0.0813
Loop Unrolling	0.0357	0.0764
Loop Termination	0.0378	0.0812
Loop Inversion	0.0379	0.0812

**Energy levels of different at Run time**



**REFERENCES**

- [1] W. Kim, D. Shin, H. Y. Un, J. Kim, and S. Min . Performance comparison of dynamic voltage scaling algorithms for real-time systems. In proceedings of the symposium on Real-time and Embedded Technology and Applications, 2002
- [2] L. Barroso and U. Holzle, "The case for energy-proportional computing," Computer, vol. 40, pp. 33-37, December 2007
- [3] J. Tsao, Interpolation artifacts in multimodality image registration based on maximization of mutual information, IEEE Trans. Med. Imaging 22 (7) (2003) 854-864, doi:10.1109/TMI.2003.815077.
- [4] Chih-Shun Ding, Chi-Ying Tsui, Member, IEEE, and Massoud Pedram, Member, IEEE "Gate-Level Power Estimation Using Tagged Probabilistic Simulation", IEEE Transactions On Computer-Aided Design Of Integrated Circuits And Systems, Vol.17, No.11. (November 1998), Page No.54-66.
- [5] K. Flautner, S. Reinhardt, and T. Mudge. Automatic performance setting for dynamic voltage scaling .In proceedings of the 5th symposium on Operating systems Design and Implementation, December 2002. IS Department , Ghent

University SinitPietersnieuwstraat 41,B-9000 Gent,Belgium  
2010.

multiprocessor systems with time constraints," IEEE  
Transactions on Emerging Topics in Computing, 2014.

- [6] kennethHoste, LievenEeckhout at.al "COLE: Compiler Optimization Level Exploration" Kenneth HosteLievenEeckhout E.
- [7] D.Marculescu.On the use of microarchitecture-driven dynamic voltage scaling. In Workshop on Complexity-Effective Design, June 2000.
- [8] Advanced microdevices, Inc. Mobile AMD athlon 4 processor model 6 CPGA data sheet. Publication 24319, November 2001.
- [9] Intel corporation Intel 80200 Processor based on Intel Xscale Microarchitecture: Developer's Manual. Order Number: 273411-003 (March 2003)..
- [10] D. Shin, J. Kim, and S.Lee. Low-Energy intra-task Voltage scaling using static timing analysis. In proceedings of Design Automatic Conference, pages 13-23,1994.
- [11] Zilishao,mengwang, yingchen, chunXue, Meikang Qui, Laurence T. Yang, and Edwin H. -M.Sha, "Real-Time Dyanamic Voltage Loop scheduling for multi-core Embedded systems"IEEE Transactions On circuits and systems-li:Express Briefs, vol. 54, No. 5,May 2007,page No.445.
- [12] R.J. Rost, OpenGL Shading Language, 2nd edition,Addison-Wesley Professional, 2006.
- [13] T. Burd and R.Brodersen, "Energy Efficient CMOS Microprocessor Design," proc. 28<sup>th</sup> Hawaii Int'l Conf.on system sciences, 1995.
- [14] D. Blythe, The Direct3D 10 system, ACM Trans. Graph. 25 (3)(2006) 724–734,doi:http://doi.acm.org/10.1145/1141911.1141947.
- [15] W.R. Mark, R.S. Glanville, K. Akeley, M.J. Kilgard, Cg: a systemfor programming graphics hardware in a C-like language, in:SIGGRAPH'03: ACM SIGGRAPH, ACM Press, New York, NY,USA, 2003, pp. 896–907,doi:http://doi.acm.org/10.1145/1201775.882362.
- [16] I. Buck, T. Foley, D. Horn, J. Sugerma, K. Fatahalian, M.Houston, P. Hanrahan, Brook for GPUs: stream computing ongraphics hardware, ACM Trans. Graph. 23 (3) (2004) 777–786,doi:http://doi.acm.org/10.1145/1015706.1015800.
- [17] D. Brokks, V. Tiwari, and M. Martonosi, "Wattch: A framework for Architectural-Level power analysis and optimizations", in proc. ISCA, Jun.2000,pp.83-94. Contreras, G., Martonosi, M., Peng, J., Ju, R., Lueh, G.Y.: XTREM: a Power simulator for the Intel Xscale core. SIGPLAN Not. 39(7), 115-125(2004).
- [18] R. Strzodka, M. Droske, M. Rumpf, Fast image registration inDX9 graphics hardware, J. Med. Inform. Technol. 6 (2003)43–49.
- [19] zolt' an Herezegl, AkosKissl, Daniel Schmidt2, NorbertWehn2, and tabor Gyim' Othyl "XEEMU: An improved Xscalepowewr simulator", PATMOS conference held in Gothenburg,Sweden in September 2007.
- [20] N. Courty, P. Hellier, Accelerating 3D non-rigid registrationusing graphics hardware, Int. J. Image Graph. 8 (1) (2008)1–18.
- [21] P. Muyan-Özc,elik, J.D. Owens, J. Xia, S.S. Samant, Fastdeformable registration on the GPU: a CUDAimplementation of demons, in: The 2008 InternationalConference on Computational Science and its Applications,ICCSA 2008, IEEE Computer Society, 2008, pp. 223–233.
- [22] ÁronCsendes"Survey of Dynamic Voltage Scaling Methods for Energy Efficient Embedded" 8th International Conference on Applied Informatics Eger, Hungary, January 27–30, 2010. Vol. 1. pp. 413–420.
- [23] Y. Wang, K. Li, H. Chen, L. He, and K. Li, "Energy-aware data allocation and task scheduling on heterogeneous



**Rajesh Mallela**, Completed his bachelor degree in Information Technology and Masters from Computer Science and Engineering from Jawaharlal Nehru Technological University, Anantapuramu. He is currently working as Assistant Professor, Department of I.T, AITS, Autonomous InstituteRajampet, A.P, India. He has Six years of teaching experience.



**K.Suresh** received his Bachelor and Master degree from Jawaharlal Nehru Technological University, Hyderabad and Ph.D. Pursuing from VIT University, Vellore, TN. He is currently working as Assistant Professor, Department of I.T, AITS, Autonomous Institute Rajampet, A.P, and India. Previously he worked as Foreign Faculty in East China University of Technology, ECIT Nanchang Campus, Jiangxi-330013, P.R.China and Visiting Faculty for Jiangxi Normal University, P.R.China. He has published more than 20 national and International conference papers and 9 international journals. He received best paper award in 2005 and Got First prize in National Paper Presentation in 2008. His field of interest is Internet of Things, Energy-aware Computing, Data Mining and High Performance Computing.



**RamanaKadiyalareceived** M.Tech Degree from Sathyabama University and B.Tech degree from JNTUniversity, Hyderabad, India in Information Technology. Presently he is with Annamacharya Institute of Technology & Sciences, Rajampet, A.P., India working as an Assistant Professor in Dept. of IT. His research interests include Distributed Computing, Cloud Computing and Wireless and Sensor Networks. He is a member of Professional societies like ISTE (India), IACSIT (Singapore), IAENG (Hongkong), UACEE (India), CSTA(USA),APCBEEES(Asia). He is reviewer of International Journal of Advanced Computer Science (IJACS), International Journal of Wireless and Mobile Computing (IAWMC), International Journalof Computer Science and Information Security (IJCSIS), Journal of Parallel and Distributed Computing (JPDC).



**M.Subba Rao, received his bachelor degree from NBKR, SV University and M.tech from JNTU University, Ananthapur.He is currently working as Professor and Head, Department of I.T, AITS, Autonomous InstituteRajampet, A.P, India. He has Sixteen years of teaching experience.**

IJSER