Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.1188 – 1192

RESEARCH ARTICLE

Client-Server Version of Energy Management through the Computational Outsourcing

Sridevi.K[1], George Christober.J[2]

[1]Asst. Prof, Department of Electronics and Communication Engineering, Sri Eshwar College Of Engineering, Tamilnadu, India

[2]PG Scholar, Department of Computer Science Engineering, Sri Eshwar College Of Engineering, TamilNadu, India

[1] sece.sri@gmail.com; [2] paulistheking@gmail.com

Abstract— In This work we optimize the Energy resources done by the Computational Outsourcing Process. The Cost of energy is higher than the Cost of Communication as per the analysis of Experimentation so we using the Computational outsourcing. The functions of Computational outsourcing having the nature to act like Deputy Server on the mobile phone like devices with in the storage space. In the System should decide the outsourcing process on the dynamic allocation that means run time while decision making process and it having the client server version of device programs that can be run on the device which makes the utilization of energy during computational process. In this Approach having that the independent application and require the minimal energy awareness of the programmer and It can be implemented in the real time system and it can have decision logic and computation process that can be run time support within the dynamic allocation process It can be saving the energy resources of mobile like devices.

Keywords— Energy Conservation; Mobile outsourcing; Computation; Client- Server

I. INTRODUCTION

The energy management Researches have so many challenges. The main issue was energy management of the pervasive computing. The power sources are limited to the mobile devices handled by the mobility management and even hardware also include these resources. These devices usability's are growing a part of the human life so the growth rate is high compared to the past decade and they need the power sources to consume the power and the operation makes their energy consumption become vital role in this area. The issue of Energy management has gained lots of attention by the researchers. The mobile device usability is growing day by day so increasing the wide range of spectrum users so the mobile devices are stemmed because of the increasing capability of users. The battery technologies are grown slowly and the wearable components like

PDA and laptop increased simultaneous growth on that time. But the Capacity of the battery improvements is in slower advance. The Capability of the mobile devices and increase reliance represents the energy consumption problems. The mobile device utilization is ranging from the teenagers to the elders on the wide range of backgrounds. Mostly the mobile devices having the features set like multimedia applications. The teenagers using the devices to play multimedia games, taking videos, edit pictures, record and play music on them. The graphical designs, Interfaces and Voice recognition features are more attractive to the users. Controlling the energy consumption is the main role of this research includes both pervasive computing and mobile device. The computer system having various levels presented n the solutions and the energy consumption problem is involved in certain type of the tradeoffs. The compiler based optimization is one the most effective high level solution for the energy management. The system power can be mostly optimized by the optimization techniques and the optimizations increased in the energy consumption process by the processor core.

II. OVERVIEW OF THE COMPILE-TIME AND RUN TIME SOLUTION

To solve the previous issue stated that has been composed in the two parts .Compilation part consist the optimization technique in the high level source code and the outcome of the compilation part will be the support of run time part

A. Compilation part

The source code can be tested and Verified in its original form. The validation process done in the syntax checking of the source code and assure that having no errors and it is reassembled and it introduced the mnemonics representation of the resulting process and the target achieved in the every instruction of the assembly representation. The contribution of High level source code and low level source code is in the part of optimization techniques. After that they recognizes the program instruction blocks that includes source code and assembly code .Then the computation of the loop can be recognized in the high level representation of the source code and it can yield the energy conservation cost for the communication to sending data and receiving only the data that changes when the loop iteration will be calculated. The instructions are involved in the each loop. And the execution of a single iteration of the each loop will be yielding the entire energy cost in the system. In addition to testing machine instructions and verifying their cost, we tested pre-existing library function and verified their energy cost in a similar manner to the individual machine instructions.

B. Runtime Part

The outsource code will be supported in the run time part when using the dynamic allocation in the system. The modes of operations includes when the application process runs on the devices. The modes are normal mode and energy-saving mode. The battery is down even if in the energy-saving mode so only we are going to consider the computation outsourcing. The application can make to monitoring battery at the start up process. If the battery will be monitored surely the outsourcing will be performed by the computational process through the network and if the run in the energy saving mode listing the application that runs on the deputy servers. The intelligence to execute code remotely is done at runtime as opposed to compile-time, and that is why their approach is a coarse-grained approach to energy management. The operating system advance power management has been straight forward to their monitoring battery in the solution part. The battery monitoring and the network monitoring are the two important areas in this system.

III. OUTSOURCING FRAMEWORK OF COMPUTATION

Outsourcing computation is not a new terminology here. However, the motivation behind outsourcing the computation to a remote server, and the approach under which we are outsourcing the computation is the contribution here. Our goal from this research is to show that an intelligent runtime decision can be made to

decide if it is better to execute a section of code locally on the mobile device, or would it be more energy-beneficial to send its data to a remote server The idea is that a server machine accessible via a wireless network can serve as a surrogate server for a host of mobile devices such as handhelds, PDAs and laptop computers. This server at runtime will receive requests from client programs running on any of these devices for outsourcing code to the server. The code that is in charge of making this decision is completely transparent to the programmer. All the programmer is required to do is to compile the code to optimize for energy. This will result in two version of the program being generated which the programmer will eventually have to compile and install. We believe that this is not a burden on the programmer in any way, and it is not a requirement for the programmer to have any knowledge of energy requirements/constraints.

Once an application is compiled, and two versions have been generated (a server version and a client version), and they are installed on their respective machines, the user can then execute a client application on the mobile device. This client application executes normally until it reaches a section of code that has been designated as outsource-able (having the potential for outsourcing), this is what we call the outsourcing candidate. Once this section is reached, then the intelligent code that was inserted at compile-time is executed to make the outsourcing decision. As a matter of fact, the candidate code will not be until the decision making code is executed.

Figure 1 Runtime Computational Outsourcing Framework

In the figure 1, the mechanism of computational framework at run time has explained. The mobile device can run the client application, when it reached the outsource able part it can determine the benefits of energy and that will run the process on locally. The outsourcing processing can be held it will send the data to the server and it will wait the results otherwise the local executions are made in the system. At all times, the

server running on the surrogate machine is waiting for requests from client programs. Once it services the client's request it goes back to waiting for client requests again, which occur once a candidate section of code decides to outsource its computation.

IV. COMPILATION AND OPTIMIZATION PROCESS

In the optimization techniques are implemented in this part and it introduce the three levels of the system. They are high, intermediate and low. The each loop contains the information about the data that can be collected at the higher level, in the intermediate level analysis that can analyse the source code representation and the register transfer list. It will also find the Count of loop iterations, In the low level calculating the machine instruction generated by the assembler and it can be determined the each loop execution in that process of levels.

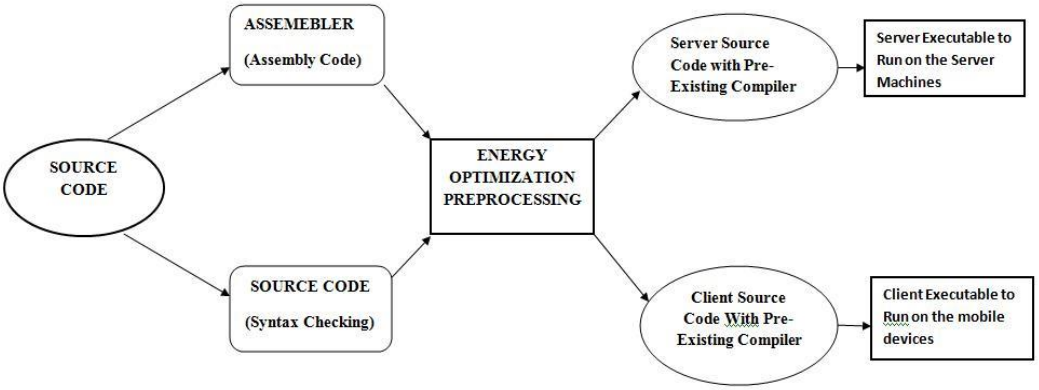

Figure 2 Energy Optimization process

In the Figure 2 shows that the energy optimization process. The source code can be the input of the process and the source code should be validated through syntax checking and the file containing the assembly code in the energy optimization process then it having the server source code with pre existing compiler and the client source code with pre existing compiler, then the client and server compiler introduce the executables of the client and server of the system to reduce the conservation of energy

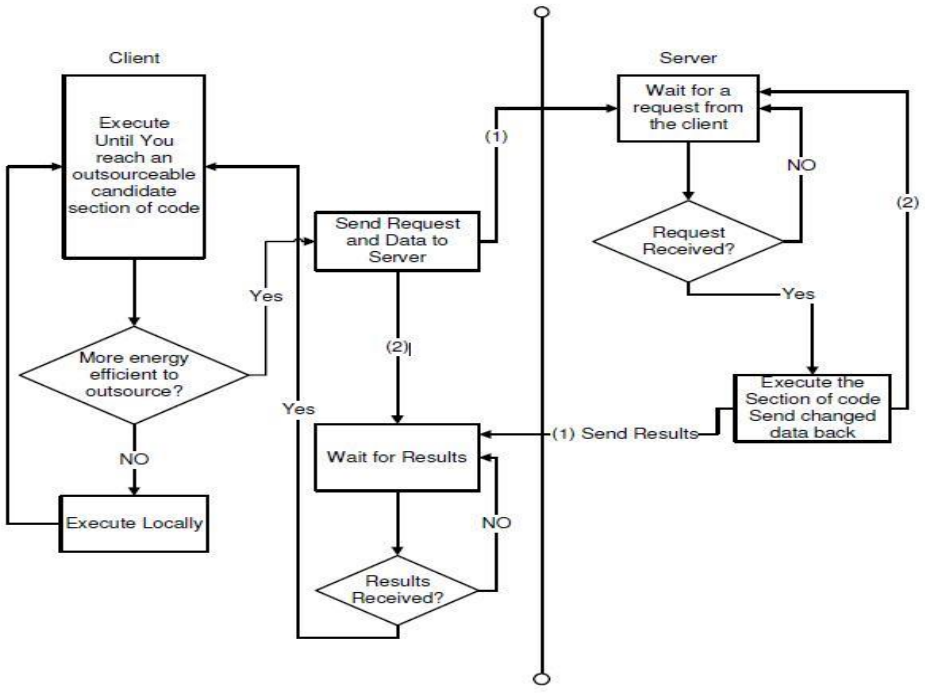

Figure 3 the steps for executing the programs in the outsourcing framework

The outsourcing takes place by the client communicating its data to the server, let the server process the data, and then the client will get the results back.
In the Energy optimization process outsourcing the basic program code and execute it locally in the system and it having various process they are

1. Calculating the Number of Loop Iterations
2. Loop Data and Iterations Acquisition
3. Calculating the Size of Loop Data
4. Identifying Loop Instructions and Total Loop Execution Cost
5. Insert Outsourcing Code

V. CONCLUSIONS

In the evaluation of experiment that computation outsourcing in the pervasive computing and it has great energy efficient technique. The energy consumption was less in the mobile device functionality and it having more time utilize the battery. Therefore the battery will not drain by the communication data. The benefits of the every outsourcing mechanism having that the basic block that can saves more energy and it will having the hybrid process of the outsourcing mechanisms.

REFERENCES

[1] Yuan W, Nahrstedt K, Gu X., "*Coordinating Energy-Aware Adaptation of Multimedia Applications and Hardware Resources*", October 2003.
[2] Starner T," *Batteries and Possible Alternatives for the Mobile Market*", IEEE Pervasive Computing 2003.
[3] Kremer U, Hicks J, Rehg J, "*Compiler-Directed Remote Task Execution for Power Management, in: Workshop on Compilers and Operating Systems for Low Power*", Philadelphia, PA, October 2000.
[4] Hsu C-H, Kremer U, Hsiao M, "*Compiler-Directed Dynamic Frequency and Voltage Scheduling*",1st International Workshop on Power-Aware Computer Systems, Cambridge, MA, November 2000.
[5] Gu X, Messer A, Greenberg I, Milojicic D, Nahrstedt K, "*Adaptive Offloading for Pervasive Computing*", IEEE Pervasive Computing 2004.
[6] Ellis C, *"The Case for Higher Level Power Management*",7th Workshop on Hot Topics in Operating Systems, Rio Rico, AZ, March 1999.
[7] Abukmail A, Helal A, *"A Pervasive Internet Approach to Fine-Grain Power-Aware Computing",* IEEE International Symposium on Applications and the Internet, Phoenix, Arizona, January 2006.
[8] Yeh T, Long D, Brandt S. *"Conserving Battery Energy through Making Fewer Incorrect File Predictions",* IEEE Workshop on Power Management for Real-Time and Embedded Systems, Taipei, Taiwan, May 2001.

Authors Bibliography

K.Sridevi has completed Bachelor of Engineering in Electronics and Communication Engineering under Anna University, Coimbatore, India. Completed Master of Engineering in VLSI DESIGN from Sri Eshwar College of Engineering, Anna University, Chennai, India. She has published one international journal. Her research areas include Low power VLSI Design, Testing of VLSI circuits and Network Security.

J. George Christober born in Tirunelveli, Tamilnadu, India in 1991. He received B.Tech Degree in Information Technology from Anna University, Coimbatore, India. He is pursuing M.E Degree in Computer science and Engineering in Sri Eshwar College of Engineering, Anna University, Coimbatore, Tamilnadu, India. He is a member of an IEEE Association. His research interests include, Image Processing, Network Security and Web Technology.

