Indian Institute of Information Technology, Allahabad

Topic : Low Power Optimization for GCC Compiler

SOFTWARE REQUIREMENTS SPECIFICATION

Team Members

D. Rajeswar Rao	IIT2014055
Saurabh Tanwar	IIT2014140
Sachin Agarwal	IIT2014501
Nishit Gupta	IIT2014502

Project Guide
Dr. Bibhas Ghoshal

Table of Contents

1.0. Introduction

1.1. Purpose

The purpose of this document is to present a detailed description of the project titled "Low Power Optimization of GCC Compiler". It will explain the effects of GCC Optimizations on Execution Time and Power Consumption of the system which are the most widely focused attributes in today's scenario. This document is intended for the associates seeking research in the field and developers for the patch.

1.2. Motivation

Since, the current optimization techniques primarily aim at reducing Execution Time, the Power Optimization Techniques have remained unexplored. Hence, there is a need to research on these techniques and develop a patch aiming at Power Reduction of a process.

1.3. Scope of Project

This project is intended for developers who are associated with production of Low Power or Battery Operated Devices.

More specifically, this system is to be designed to facilitate low power functioning of a program or process which would otherwise consume lot of power thereby draining the battery. This would have an adverse effect on the outcome of the process or program.

1.4. Glossary

Term	Definition
O0	Optimization Level 0 (Default)
O1	Optimization Level 1
O2	Optimization Level 2
O3	Optimization Level 3
Ofast	Optimization Level 4
Gem5	Simulator for running the executable binary on ARM Architecture
ARM	Advanced RISC Machine (Reduced Instruction Set Computing)
McPAT	Multicore Power, Area and Time. Measures the power consumed by a process.
Stats.txt	The statistics file generated by the simulator for power analysis.
Patch	A piece of software for a specific purpose.

1.5. References

[1] IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

[2] D Branco and P.R Henriques: Impact of GCC Optimizations Levels in Energy Consumption during C or C++ program execution.
2015 IEEE 13th International Scientific Conference on Informatics.

[3] V.Tiwari, S Malik and A. Wolfe : Compilation Techniques for Low Energy, An Overview.
-Low Power Electronics, 1994. Digest of Technical Papers, IEEE Symposium. Oct 1994.

[4] M. Kandemir, N Vijaykrishnan and M. Jane Irwin: Power Aware Computing, Chapter: Compiler Optimizations for Low Power Systems.
-Pages : 191 to 210. Kluwer Academic Publishers, Norwell, MA, USA – 2002

[5] M. Valluri and Lizy K. John: Is Compiling for Performance – Compiling for Power?
Springer, USA, Boston, MA – 2001

[6] U Kremer: Low Power/Energy Compiler Optimizations
In Low-Power Electronics Design, CRC Press, 2005

[7] https://bitbucket.org/dskhudia/gem5tomcpat,
[8] http://www.hpl.hp.com/research/mcpat/

1.6. Overview of Document

The next chapter of this document describes the system environment and technical requirements which is required to conduct the survey and develop a patch as explained above.

The next chapter gives an overview of the functional requirements of the project which includes input data, operations performed by patch, the work-flow of the system and system outputs. It describes the informal requirements and in the next section.

The last part of this document discusses the goals of implementation i.e. making a part of code to optimize power.

2. System Environment and Technical Requirements

Testing Platform : Laptop HP 15R014TX, running under Linux.

Hardware/Software Specifications:

Architecture : x86

Operating System : Ubuntu GNOME 14.04.2 LTS 64-bit (Linux Kernel 3.16);

Microprocessor : 1.7 GHz Intel Core i5-4210U with Intel HD Graphics 4400

Cache : 3 MB cache

Memory : 4 GB 1600 MHz DDR3L SDRAM (1 x 4 GB)

Video Graphics : NVIDIA GeForce 820M (2 GB DDR3 dedicated)

Display : 15.6" diagonal HD BrightView LED-backlit (1366 x 768)

Hard Drive : 1 TB 5400 rpm SATA

Technical Requirements

1. Cross Compiler for ARM Architecture
2. Gem5 Simulator
3. Gem5toMcPat Converter
4. McPAT : Power Measuring Tool

Description

Cross Compiler : Cross Compilation is a process of compiling and creating executable code for a platform other than the one on which the compiler is running. Cross compiler compiles the input program for the ARM Architecture.

Gem5 Simulator : Modular platform for computer-system architecture research, encompassing system-level architecture as well as processor microarchitecture.

Gem5toMcPAT Converter : Converts statistics file to suitable input file for Power Measuring tool.

McPAT : An architectural modeling tool for chip multiprocessors (CMP).
The main focus of McPAT is accurate power and area modeling, and a target clock rate is used as a design constraint.

3. Functional Requirements

3.1 First Phase

The input to the system will be a C program file . We will generate statistics for Execution Time and Power consumption of the input file . The workflow of the system is as mentioned below:

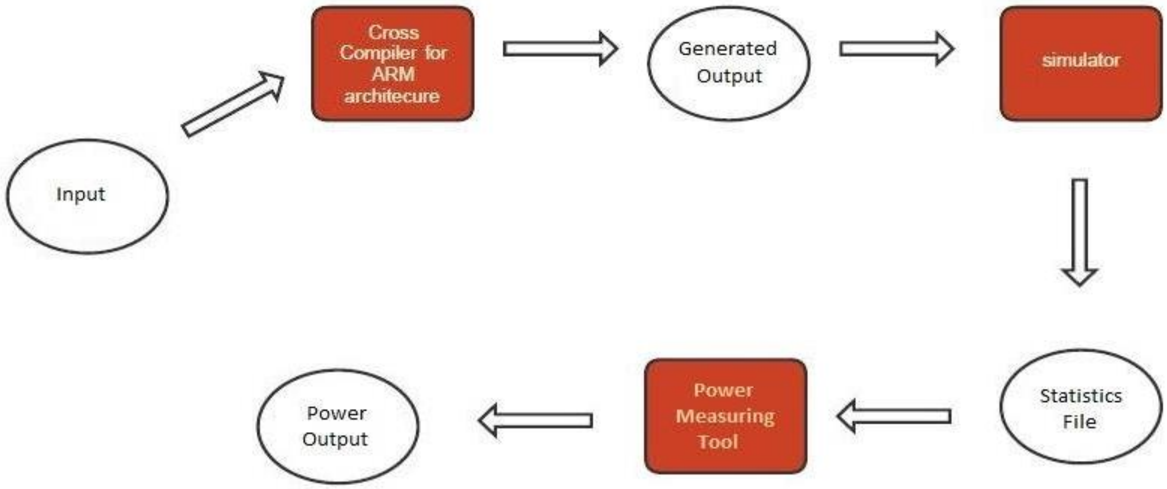

Figure 3.1 Survey Work Flow

3.2 Second Phase

The patch will perform a set of operations on the input file focusing on reducing power consumption . The workflow is as shown

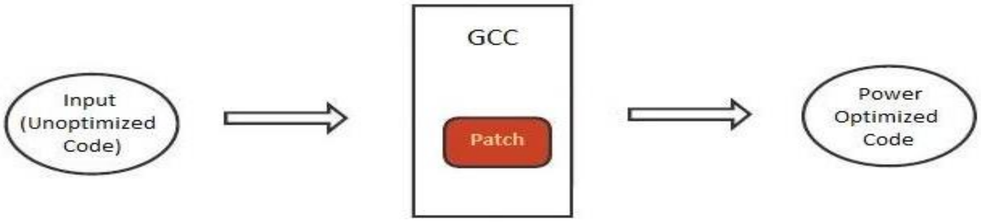

Figure 3.2 Power Optimizing Functionality

The output will be an optimized code which consumes less power.

3.3 Third Phase

The patch developed should ensure that the functionality of the original code is preserved.

Proving techniques will be used.

4. Non Functional Requirements

Performance

The system must be efficient and the delays involved must be less .So in every action-response of the system, there are no immediate delays. Also the delay in statistics generation should be less. Sufficient memory should be available to avoid page fault.

Safety

Desired output should not be modified during the course of optimization.

Reliability

System should be reliable in the sense that functionality of the program is preserved and desired output is obtained.

Usability

The patch should be easy to install for future developers associated with research in power optimization and production of low power and battery operated devices.

5. Goals of Implementation

With increase in usage of embedded devices, power optimization has become an important issue. Power optimization would contribute in advancement of technology and increase in usage of battery operated devices which were limited in use previously. Thus this patch is a simple and useful tool for power optimization.