ARM PROCESSOR EMULATOR

MOHAMAD HASRUZAIRIN B MOHD HASHIM

A project report submitted in partial fulfilment of the
requirements for the award of the degree of
Master of Engineering (Electrical – Electronic and Telecommunications)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JUNE 2015

Specially dedicated to my wife Aishah, my daughter, lecturers, fellow friends and those who have guided and inspired me through my journey of education

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my thesis supervisor and mentor, Dr. Usman Ullah Sheikh, for his willing encouragement, unstinting guidance, and friendship.

I am also indebted to Majlis Amanah Rakyat (MARA) for funding my MEE study and my fellow postgraduate students who have selflessly supported and enabled the pursuit of this dream. My sincere appreciation also extends to all my colleagues and others who have provided assistance at various occasions. Finally, the bedrock of my success, I am grateful to all my family members.

ABSTRACT

Technology has always enhanced learning as well as the overall teaching experience. With proper tools and resources in hand, we can easily integrate educational and information technologies into the academic environment. These are many ARM emulators in the market, however most are not freely available and complicated to use. Besides that, they are not suitable for beginners to use for learning. The objective of this project is to deliver on ARM emulator for x86 Windows platform. This project focuses on developing on ARM emulator using the GNU C/C++ compiler and associated tools for embedded systems development and ARM-based projects. The ARM emulator use GNU toolchain to convert high level language to assembly language and binary object file. GDB ARM is used to do simulation for register set and assembly code. C# was used to develop the interface for the emulator. This software can help instructors in the teaching and learning process in class and students can see the ARM process emulation in real time.

ABSTRAK

Peningkatan teknologi di dalam pembelajaran dapat meningkatkan teknik mengajar keseluruhan. Dengan penggunaan alat yang betul dan sumber yang benar, kita boleh dengan mudah mengintegrasikan teknologi pendidikan dan maklumat digunakan di dalam persekitaran akademik. Terdapat banyak ARM emulator yang berada di pasaran, namun kebanyakan tidak dapat diperolehi secara percuma dan amat rumit untuk digunakan. Selain itu, ia tidak sesuai untuk pengguna baru untuk digunakan di dalam suasana pembelajaran.Objektif projek ini adalah untuk memperkenalkan ARM emulator untuk digunakan di dalam platform Windows x86. Projek ini memberi tumpuan kepada pembangunan ARM emulator menggunakan GNU C / C ++ sebagai pengkompil dan alat-alat yang berkaitan untuk pembangunan sistem dan projek-projek berasaskan ARM.ARM emulator menggunakan GNU toolchain untuk menukar bahasa peringkat tinggi kepada bahasa himpunan dan fail objek binari. GDB ARM digunakan untuk melakukan simulasi untuk daftar set dan kod pemasangan. C# digunakan untuk membangunkan antara muka dengan emulator. Perisian ini boleh membantu pengajar dalam proses pengajaran dan pembelajaran di dalam kelas dan pelajar boleh melihat simulasi proses ARM dalam keadaan sebenar.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Background

The ARM processor is one of the most widely used processor in the market today. Many of the embedded system are based on ARM processor. The objective of this project is to help the users for use the ARM processor technology to have a platform to learn the processor architecture and instruction set. As processor ARM processor has very high demand, many ARM emulators were developed but most of them in the market are paid and complicated to use. This project will focus on considerations of using the GNU C/C++ compiler and associated tools for embedded systems development, for ARM-based projects and together with the integrated development environment for visualization.

The purpose of ARM processor emulator use GNU toolchain to convert high level language to assembly language and binary object file. GDB for ARM is used to perform emulation and to view processor registers and assembly code. C # to develop an integrated development for Windows (x86) platform. This software can help instructors in the teaching of ARM architecture in class and students can see the processor emulation in real time.

1.2 Motivation

Most of emulators developed in pass few year are outdated in the market and are designed by the previous generation of ARM processor. The available emulators are not user friendly and inconvenience to the beginner. They are also difficult to use as a tool during learning process and not many tools are specifically designed for teaching purposes. The motivation of this work is to develop on ARM processor emulator suitable for in class usage.

1.3 Objectives

The objective of this project is to develop ARM processor emulator to emulate ARM and Cortex processor. It can be used for performing simulation in class so that student can see the real-time ARM process from high level language to assembly code. This emulation will use GNU toolchain to convert high level language to assembly language.

1.4 Problem Statement

ARM emulators use on an x86 Microsoft Windows platforms are very limited. Most processors on the market are using Linux platform. Sometimes difficult to users because most users are using windows base software.

ARM emulators for x86 Microsoft Windows platform are very limited. Most ARM emulator on the market are for the Linux platform. ARM emulators that have been designed to run on an x86 Microsoft Windows platform are generally not freely available. Besides, is very hard to find ARM emulators which are free ans also has support on documentation.

1.5 Scope of Work

I. The ARM emulator only caves the ARM 7 architecture and Cortex M0 architecture,

II. The emulator is based on GNU tool-chain backend as compiler.

REFERENCES

1. David Sharp and Graham Martin , "*A dynamically Recompiling ARM Emulator*", University of Warwick, Project Report 2000-2001
2. "ARM Cortex-M0 Technical Reference Manual" Manual ARM 2012.
3. Lukas Pustina1, Simon Schwarzer1, Prof. Dr. Peter Martini1, Jari Muurinen2 and Ari Salomaki2," *A Methodology for Performance Predictions of Future ARM Systems Modelled in UML*", IEEE Conference 2008.
4. Get Better Code Density than 8/16 bit MCU'sNXP LPC1100 Cortex M0, 26 march 2014 product datasheet.
5. Bobby Clarke, ARM, "*Introduction to ARM",* Eclipse Members Meeting – Sept '06
6. Fabrice Bellard,"*Qemu,a Fast and Portable Dynamic Translator*", 2005 USENIX Annual Technical Conference.
7. ARMware http*://code.google.com/p/ARMware/*
8. Geetika Malhotra,Namita Atri,Smruti R Sarangi,"emuARM : *A Tool For Teaching the ARM Assembly Language*", IEEE Conference 25 Sept 2013.
9. http://whatis.techtarget.com/definition/ARM-processor. Internet
10. Skazikis (Nov 2012), "The ARM Processor Architecture"
11. Dillon Tellier and Dr. Christopher Lupo (June 2014) ," *Design of CPU Simulation Software for ARMv7 Instruction Set Architecture*"