Federal University of Pernambuco
Graduation in Computer Engineering

Informatics Center
2009.1

JEDBG: An ARMJTAG Debugger for the ERESI Reverse Engineering Framework

Undergraduate Thesis Proposal

Author: Jesus Sanchez-Palencia Fernandez Filho (jspff@cin.ufpe.br)
Supervisor: Sérgio Cavalcante (svc@cin.ufpe.br)

Recife, February 26, 2009

Table of Contents

1. Context

Reverse Engineering is the art of understanding the principles of a system or a device by studying its structure. Software Reverse Engineering is a research area that was born decades ago, when computer systems became part of everyone's life. Back there, in the 80's, developers didn't have access to source code of almost anything they could need - drivers, software, operational systems, etc – so they had to find ways of better understanding what they were using. Even before that, there were analyses of hardware for commercial or military advantage, Hardware Reverse Engineering. The main principle was the same: discovering the whole function and operation of a device, without having access to source plans.

The methodical process of following a system's execution, instruction by instruction, is called Debugging. We use a debugger for performing such task, trying to find bugs on the software and/or to understand it better. For note, debugging is a very important step on Reverse Engineering and there are very advanced researches on this.

Nowadays we have the advent of embedded systems everywhere. They are special-purpose computer systems designed to perform a set a functions, usually embedded as a part of other major systems. For many years now, a considerable amount of effort has been put into making the automated analysis of this kind of system a powerful and practical tool in the workbench of professionals and researchers that deal with hardware and embedded software.

A great deal of work has been done among the IEEE community in order to develop a method of testing printed circuit boards, and embedded systems as well. An industry group called JTAG – Joint Test Action Group – was formed in 1985, and in 1990 the standard IEEE 1149.1 was released. The common name for this standard remained JTAG, and it can be summarized as an access port to test and to perform boundary-scan on electronic boards. Currently, JTAG is used for debugging embedded systems by being the transport mechanism for the CPU integrated on-chip debug module. It's an in-circuit emulator that enables us to access this debug module through the JTAG interface. In summary, JTAG provides a backdoor into the embedded system.

The final implementation of this work's results shall be done upon the framework provided by the ERESI project. ERESI (ELF Reverse Engineering Software Interface) is an open-source project that has been developed for the past 6 years and that aims to help its users to gather relevant information from compiled programs using various means, such as disassembling or debugging/live analysis. One of the ultimate goals of ERESI is to be able to perform automated static analysis that can efficiently spot security breaches in binary software. After this work, though, it will also reach embedded software.

The ERESI project is moving towards the complete analyses and debugging of embedded systems by the development of this work. ARM-based systems were initially chosen, and so any result of this project aims the ARMJTAG standard. The ERESI framework runs on a variety of UNIX systems, working with the ELF executable format and having most of its features implemented mainly for Intel x86 and SPARC architectures, and now the under development ARM support. Any code resulting from this work shall target Linux over Intel x86 platform, but for analyzing ARM native code.

2. Objectives

As previously stated, the main objective of this work is to enable the ERESI framework to debug and analyze ARM embedded systems, using the JTAG interface. Therefore, a library for interfacing with the ARMJTAG protocol is needed besides the debugger itself.

All initial effort will be on researching the protocol and the JTAG standard. Solid knowledge about all JTAG pins – TDI, TDO, TCK, TMS and TRST -, about its serial protocol and its instructions, is needed before starting the development of the ERESIS' *libjtag*. Once this library becomes ready, or partially ready at least, the author will focus on writing the debugger code, the so-called JEDBG – JTAG ERESI Debugger. As we can see, this work is very research-oriented, but a certain amount of proof-of-concept code has to be written.

In order to demonstrate the correctness of the proposed work, the JEDBG and so the libjtag will be tested against a "real world" embedded system. The chosen platform was an LPC-E2124 development board, running a *FreeRTOS* with *Webserver* setup. The LPC board fits perfectly for testing purposes since it's an ARM7TDMI based board with JTAG interface. A small bibliography at the end of this document can reference deeper information on all above presented subjects.

3. Work Plan

Based on the project objectives described on the previous session, the following work plan is proposed.

	Period															
Task	**March**				**April**				**May**				**June**			
ARMJtag Standard study	🟦	🟦	🟦	🟦												
ERESI development study		🟦	🟦	🟦	🟦											
Architecture Proposal					🟦											
Libjtag implementation					🟦	🟦	🟦	🟦								
JEDBG implementation						🟦	🟦	🟦	🟦	🟦						
Document writing									🟦	🟦	🟦	🟦	🟦	🟦		
Document revision											🟦	🟦	🟦	🟦		
Work presentation															🟦	🟦

4. References

[1] E. J. Chikofsky and J. H. Cross, II, "Reverse Engineering and Design Recovery: A Taxonomy," IEEE Software, vol. 7, no. 1, pp. 13-17, January 1990

[2] Vanegue J., Garnier T., Auto J., Roy S. & Lesniak R., "Next-Generation Debuggers for Reverse Engineering", 2007

[3] Bisolfati, E. "Kedbg: the ERESI kernel debugger", 2008

[4] The ERESI project. http://www.eresi-project.org

[5] Auto J., "Developing an Intermediate Representation for the Analysis of Binary Code", 2007

[6] The LPC2124 Specification. http://www.keil.com/dd/chip/3647.htm

[7] The LPC2124 Datasheet. http://www.olimex.com/dev/images/lpc-e2124-sch.gif

Signatures

Sérgio Cavalcante
(Supervisor)

Jesus Sanchez-Palencia Fernandez Filho
(Student)

Recife, February 26, 2009