Project Title
TFTP and a Flash Memory Interface for Embedded XINU

Abstract
The XINU operating system was developed in the 1980's and has been ported to multiple architectures. Several universities use XINU as a teaching tool for undergraduate systems courses. A goal of XINU development is to maintain simplicity to allow undergraduate students to participate in its development, research, and implementation.

Embedded XINU, created by Dr. Dennis Brylow, is an ongoing research project at Marquette University. Its goal is to port XINU to the MIPS platform; specifically Linksys WRT54GL routers. A goal specific to Embedded XINU is to replace the router's native Linux distribution in flash memory without losing router functionality. A critical step is the ability to properly write a XINU boot file into flash memory and update it as newer releases occur. Initially, the routers contain a Linux distribution and Common Firmware Environment in flash memory. Currently, TFTP commands from CFE allow the router to acquire a remote XINU boot image and run it on startup. A web interface for the Linux distribution allows the overwriting the main boot system in flash memory. However, once complete no interface exists for interacting with flash memory successfully.

The proposed project is two parts; first I will add TFTP to Embedded XINU. Secondly, I will refine and fix the current rudimentary flash memory interface. TFTP gives Embedded XINU the ability to acquire and send files using the various commands. A proper and simple flash memory interface allows read and write commands on areas of flash memory from within XINU. Once completed, files will be maintained in flash memory even when the router is powered off. Combining both parts will let Embedded XINU to acquire a XINU boot file and write it properly to flash memory making the router to boot XINU on power up. This will also allow the Linux distribution to be written back into flash memory, returning the router to its original state. The project must maintain the simplicity aspect of XINU; allowing undergraduates to learn about TFTP and flash memory directly from the project implementation and build it from pieces or add to it.

After this significant step in development other areas of research become feasible. Projects include: implementation of a simple web interface, proper CORE dump for debugging, various file system projects, and flash memory based projects such as maximizing the amount writes to flash memory.

Mentor Contact Information
Dr. Dennis Brylow
(414)288-6342
Assistant Professor, Marquette University
Department of Mathematics, Statistics and Computer Science
brylow@mscs.mu.edu
http://www.mscs.mu.edu/~brylow/xinu/

Project Details

Several universities use the XINU operating system to aide in the instruction of various courses such as Operating Systems. Dr. Dennis Brylow of Marquette University has adapted significant portions of the XINU operating system to the MIPS architecture, called Embedded XINU. At SIGCSE 2008 Dr. Brylow presented his findings and paper on teaching using Embedded XINU in an undergraduate operating systems course. Marquette University and the University of Buffalo have integrated this version of XINU into courses and Oberlin College will join the group in the next academic year. Multiple other universities have older versions of XINU implemented and may be interested in converting to Embedded XINU.

Currently, a router running Embedded XINU must acquire a remote XINU boot file through an existing Common Firmware Environment (CFE) each time the router is powered on. The native on boot system is a variant of Linux. My proposal has two parts, first to add TFTP services to Embedded XINU for file acquisition and sending. Secondly, to fix, refine, and finish the currently limited flash memory interface in Embedded XINU. When combined a XINU boot file can be received and written to flash memory making it persistent through the router powering off. The parts are incremental implementations to Embedded XINU, a cut and dry version of TFPT exists in prior versions of XINU and there is a limited, partially tested flash memory interface within Embedded XINU.

Benefits

Given a simple yet complete flash memory interface universities that use Embedded XINU as a teaching tool can choose to use it as an undergraduate teaching tool for flash memory. Coursework in understanding, working with, and studying flash memory is seldom offered to undergraduate students due to the complexity. Providing a simple teaching tool for institutions gives the student exposure to a topic relevant in academia and industry and a topic seldom taught to undergraduates.

Presently, the target platform for Embedded XINU is a Linksys router that natively boots to a Linux distribution. A goal of Embedded XINU is to replace this Linux distribution without losing basic router capabilities. Completion of both parts of the project will allow students to use their own work by using the router with their operating system on it. At the moment the router only runs the student's operating system while it is powered on. TFTP commands integrated with a flash memory interface would allow the student to put the latest version of the operating system on the router as the student moves it along in development. Beyond the world of the student the user of the router might also wish to have the most up to date version of Embedded XINU or revert back to initial settings; the tools implemented with this project make this possible.

TFTP commands and a simple sufficient and working flash memory interface are part of the critical path for development of Embedded XINU. With these two things implemented a wealth of new research and development for Embedded XINU can commence. Areas in flash memory research and development include file systems and maximization of flash memory life. With TFTP commands sending and receiving files become possibilities, which allow for activities like a proper CORE dump for debugging. Thus, Embedded XINU development benefits greatly too from moving past this critical point.

Uncertainties & Risks

Constructing a flash memory interface can be a tricky problem because Embedded XINU uses systems currently in flash memory to start itself. The CFE is used when booting a router to let it know where to find the XINU boot file, if this is overwritten by a partial boot file or other file the router temporarily becomes an oversized paperweight. Fortunately a restoration tool exists; it was developed during initial attempts at developing a flash memory interface and allows an individual to manually restore the router from paperweight exile.

Incomplete projects can be viewed in both a negative and positive light. Although, the possibility exists for an incomplete project the summer provides ample amount of time to make great strides towards the parts of the project if not complete them. Finding solutions to non-trivial problems that have not been previously solved can be a very time consuming and tedious task. Any headway made on research and implementation of the project's parts will give future undergraduate researchers a huge leap forward towards the project's completion. However, given the abundant time of the summer, a significant amount of work should be done if not the completion of the project.

License

All Embedded XINU code is released under a BSD-style license.

More descriptive versions of major and working deliverables can be found along with another copy of the project schedule at:
http://mscs.mu.edu/~akoehler/gsoc_deliverables.pdf

Project Schedule (Deliverable Due Dates)

May 26 – TFTP Proof of Concept
June 18 – TFTP Client
June 30 – TFTP Daemon
June 30-July 7 – Flash Memory Code Review
July 7 – Flash Memory Interface Proof of Concept
August 11 – Flash Memory Interface Completion
August 18 – Summer of Code Summary

Major Deliverables

TFTP Proof of Concept:

Part 1: Testing the network interactions to ensure they work as intended.
Part 2: Given the network features, proof that TFTP client and daemon services are achievable.
Part 3: Specific timeline for the implementation of TFTP.

TFTP Client

Part 1: Code turn in – feature works; mark as complete
Part 2: Summary of process, tests and problems

TFTP Daemon

Part 1: Code turn in – feature works; mark as complete
Part 2: Summary of process, tests and problems

Flash Memory Interface Code Review

Step 1: Understanding the current implementation
Step 2: Verifying code works as intended, modifying if needed
Step 3: Altering code to conform to predetermined standards
Deliverable: Code review summary

Flash Memory Interface Proof of Concept

Part 1: Proof flash memory interface is possible
Part 2: Timeline for flash memory interface completion

Flash Memory Interface Completion

Part 1: Code turn in – features work; mark interface complete
Part 2: Summary of process, tests and problems

Summer of Code Summary

Part 1: Test TFTP and Flash Memory Interface together
Part 2: Document with final tests, summary of work done and foreword thinking.

Working Deliverables

SVN

- used as a revision system
- "public" view to working code base
    - access can be granted upon request

Wiki

- accessible display of on going work
- publicly viewable general timeline
- notes on discovered problems

BIO

I participated in the development of Embedded XINU within and outside the classroom. I have taken several of Dr. Brylow's courses using XINU as a teaching tool. My senior project is a frontend interface that allows Windows computers to directly participate in the research and development of XINU. Currently the tool set is available to UNIX variant systems. My full curriculum vita is available at: http://mscs.mu.edu/~akoehler/cv.pdf

Eligibility

I am enrolled as a full time undergraduate student at Marquette University. I am legally eligible to work in the United States—my country of residence. Documentation to verify the aforementioned will be provided.