

Diploma Project Proposal

2010-2011, 4th year of study, 2nd semester

POSIX multithreaded Real Time Operating System

1 Context

There is a class of microkernels, called L4[1], providing basic services (Thread Management, Memory Space and I/O Management, Interrupt Control) that were derived from research done at Karlsruhe and Dresden Universities. The most advanced version, freely available, is L4 “Pistachio”. It was used by OK Labs as a basis for their product called OKL4[2].

Part of our collaboration with VirtualMetrix, Inc. (VMX)[3], we are developing an L4-based microkernel, called VMXL4. This is an effort to develop a microkernel that will offer virtualization support on embedded systems (i.e. smartphones) and allow the running of the VMX Linux – an updated version of Linux that uses “Performance Management” (PerfMan)[4]. PerfMan is a new class of control developed by VirtualMetrix. It manages all performance-related resources such as CPU runtime (Task Scheduling), Clock/Idle Management, cache space and I/O bandwidth allocation to software.

2 General Description

The project aims to create a POSIX Real-Time multithreaded environment running on VMXL4 (POSIX – POSIX - Portable Operating System Interface, IEEE Family of Standard). You will need to implement POSIX “style” interface as the “RTOS” API (Multi-process, Threads, No File System) and design a small real-time operating system to support the API running on VMXL4.

Project delivery requires test and demonstrating an application meeting real-time requirements.

2.1 Objectives

1. create a POSIX real-time multithreaded environment running on VMXL4
2. choose a real-time multithreaded application to demonstrate the POSIX RTOS
3. get acquainted with
 - (a) POSIX multithreaded environment
 - (b) real-time operating system design and implementation
 - (c) how to implement an operating system personality on VMXL4
 - (d) using the VMXL4 microkernel

2.2 Technologies Employed

- GNU toolchain
- C, Python
- shell scripting
- VMXL4 microkernel

2.3 Resources

- **Software management:** wiki, Trac, Git repository
- **Documentation:** wiki (component description etc.), LaTeX (Bachelor Thesis)

3 Coordination

The project coordinators are Răzvan Deaconescu (razvan.deaconescu@cs.pub.ro), part of the Automatic Control and Computers Faculty, and Gary Gibson (gary@VirtualMetrix.com), part of VirtualMetrix.

Project coordination will employ:

- review on the architectural description and development of a project roadmap
- weekly meetings for progress reporting
- review and feedback sessions after each milestone
- periodic Skype calls with Gary Gibson for progress reporting and general decisions

4 Prerequisites

Students must have good knowledge on the following topics:

- Linux development environment (CLI, editors, GNU toolchain)
- Basic operating systems concepts
- Programming languages: C, Python, shell scripting

5 Planning

The project requires a 2 students team. It will be developed during the 2nd semester for 5 months.

6 Bibliography

- An Embedded Software Primer, David Simon
- The Linux Programming Interface: A Linux and UNIX System Programming Handbook, Michael Kerrisk
- Operating System Concepts, 7th Edition, Avi Silberschatz, Peter Galvin, Greg Gagne

7 Other Information

[1] http://en.wikipedia.org/wiki/L4_microkernel_family

[2] <http://www.ok-labs.com/products/okl4-microvisor>

[3] <http://www.virtualmetrix.com/>

[4] <http://embedded-computing.com/performance-management-new-dimension-operating-systems>