ECE 373 – EMBEDDED OPERATING SYSTEMS AND DEVICE DRIVERS
Spring 2019

Instructor:
Peter (PJ) Waskiewicz – ppw@pdx.edu
Charles Stoll – stoll@pdx.edu

Office Hours:
After class, or as-needed
Lab Time: Tues 6:40 – 9:30pm
Fri 3:00 – 5:50pm

TA's:

Peng Gao –

Ram Bhattarai – ram9@pdx.edu

Texts:
Linux Device Driver Development, Madieu, Packt, ©2017

Course Overview and Objectives:
ECE 373 extends the microprocessor interfacing skills gained in ECE 372 to the design of hardware and device drivers for a microprocessor system using an embedded operating system. Course includes discussions of core OS concepts of scheduling, mutual exclusion, and virtual memory as implemented in embedded systems; a study of basic Linux OS structure and operations; and practice developing simple Linux device drivers.

At the conclusion of this class, should be able to:

1. Describe the operation and use of operating system features such as mutual exclusion, pre-emptive and non-pre-emptive scheduling, real-time operation, and virtual memory.
2. Describe the basic structure and operation of the Linux operating system with real-time capability.
3. Analyze and write Linux application programs in the C programming language using the GNU tool chain.
4. Compile, install, run, and remove a live Linux source module.
5. Design, write, build, install, and run a specified custom module.
6. Understand and correctly use the Linux kernel internal services of mutual exclusion, memory mapping, DMA operations, interrupt handling, timekeeping and scheduling to write a variety of custom modules.
7. Design and build a custom device driver.
8. Modify or design a custom kernel module.
9. Describe the interfaces for character and block devices, video displays, and network links.

10. Describe the security and power management techniques commonly used in embedded systems.

Course Procedures and Grading:

Assignments will be given in class as homework and labwork. Lab time with a teaching assistant is available and expected to be used for most assignments, as some assignments could require the use of little project computers in the lab in the absence of personal virtual machine access.

All assignments should be done individually.

Test questions will closely parallel homework problems and design/programming projects, so to do well on tests it is highly suggested that you complete all these assignments individually rather than as part of a group.

Approximate ratios for final grade: Midterm Exam – 25%, Homework – 40%, Final Exam – 35%. A = 94-100, A- = 90-93, B+ = 87-89, B = 84-86, B- = 80-83, C+ = 77-79, C = 74-76, C- = 70-73, D+ = 67-69, D = 64-66, D- = 60-63, F = 0-59. No curve is used in determining grades. Anyone caught copying or cheating in any way will receive a 0 for that test, homework, of design project. Make up exams will be given only if arranged well before the exam time. Late design or programming projects will start at half-credit prior to grading.

The projects must be completed satisfactorily to receive credit for class. This includes clear and correct writing and spelling along with technically correct work.

Computer Based Design/Programming Development Tools

The lab design/programming projects in this class will use virtual machines (x86). Labs and projects will start with overviews of the C programming language, the GNU tool chain, moving into the innards of the Linux kernel and the device drivers.