COMPE 571 – Embedded Operating Systems Term Project Description

Requirements

The term project will be done in teams of two/three. The project has to have both HW and SW components and needs to show integration of the two, especially how the SW part is related to the embedded operating systems topic. It will be graded on a combination of the proposal, progress demo/presentation/report, and the final demo/presentation/report. More challenging/novel projects are more likely to get a good score. The suggested number of team members is two but three is also allowed if the chosen project has fair workload (a bigger/more challenging project as compared to a project for two). There is a list of potential project ideas at the end of this document along with supporting links. It is acceptable to do a previously done project as long as the students can demonstrate their own implementation and analysis efforts.

The project requires three written deliverables and two presentations/demos:

1. **Project proposal:** This document is the first required item. It should clearly show the team members, the chosen project, why this project is relevant, what is going to be delivered by the progress demo and the end of the project (a clear timeline) and the anticipated workload share among the team members. This document should be 3 pages maximum with a minimum 10-pt font. Please make sure to talk to and agree with the instructor about your project proposal before the proposal due date. The students should also include any applicable references to support the proposal. The proposal, in particular, should answer the following questions:
    a. Title of the project
    b. List team members names, email addresses and student IDs
    c. What you plan to do, why & what is hard about it (i.e. why should it be a grad level class project)
    d. List HW components will you interface with and, if applicable, where you will get them from
    e. Describe the SW you will implement
    f. Metrics of success (e.g. power consumption decrease, response time improvement)
    g. Experiments you will run and the results you expect
    h. Deliverables by project progress report
    i. Describe the final project demo & what you will report in the final project report
    j. A clear timeline of milestones and deliverables (This is really important for your project grade at the end). You should not overpromise or have too few milestones.
2. **Project progress report:** This document should discuss the status of the project at the corresponding date. It should show the completed milestones (and how they have been achieved), any results to the date and the remaining goals. The report should clearly identify the workload among the group members in detail. This document should be 5 pages maximum with a minimum 10-pt font. The students should also include any applicable references to support the report. . The progress report, in particular, should answer the following questions:
    a. Title of your project
    b. List team members names, email addresses and student IDs
    c. Related work
        i. Discuss what others have already accomplished in this area (this can include published academic papers or commercial products).
        ii. Highlight what is interesting in your particular implementation relative to other's previous work
    d. Accomplishments so far
        i. List HW components are working with and the status of your HW development. Discuss any changes and issues that may have arisen in the HW area since the proposal.
        ii. List SW frameworks that you use, the SW you plan to design and the status of that implementation. Discuss any changes and issues that have arisen in SW area.
        iii. Experimental design
            1. Success metrics
            2. Outline experiments you already did, and discuss results to date
    e. Future deliverables/milestones
3. **Project progress demo/presentation:** The students should give a presentation and a demo (if applicable) in class to demonstrate the current status of their project. The students will be given a fixed time interval to

finish their presentation and demo. The instructor will announce the duration of this time interval ahead of time. All of the team members should participate during the presentation.

4. **Final project report:** This document should present the final status of the project. It should clearly demonstrate that all project milestones are achieved and how the success of the project is measured. The students should include any possible results of their implementation and quantitative analysis. If any of the milestones is not completed, the students should clearly explain the reason (but in general, they should complete all the milestones). This document should be 8 pages maximum with a minimum 10-pt font. The project report should also be accompanied by any code that is implemented by the students in an archive file. The final report, in particular, should answer the following questions:
    a. Title of your project
    b. List team members names, email addresses and student IDs
    c. Accomplishments
        i. List HW components are working with and the status of your HW development. Discuss any changes and issues that may have arisen in the HW area since the proposal.
        ii. List SW frameworks that you use, the SW you plan to design and the status of that implementation. Discuss any changes and issues that have arisen in SW area.
        iii. Experimental design
            1. Success metrics
            2. Results
            3. Possible extensions to the project
    d. Missing milestones
        i. What have you not delivered from your milestone list
        ii. Negative results, what they mean and how they can be fixed later
5. **Final project demo/presentation:** The students should give a final presentation and a demo outlining the results shown in the final project report. The students will be given a fixed time interval to finish their presentation and demo. The instructor will announce the duration of this time interval ahead of time. All of the team members should participate during the presentation. The presentation should clearly demonstrate the implementation efforts, the crucial results and the importance of the project.

Grading

- Proposal report: 5%
- Progress report: 20%
- Progress demo and presentation: 15%
- Final report: 30%
- Final demo and presentation: 30%

The grading of the reports will be based on how well the report addresses the questions listed above for each different report.
The grading of the presentations/demons will be based on how well the students can communicate their findings through the presentation, how all team members participate in the process and how well they can answer questions from the audience.

Timeline

1. Project proposal – Due: Sep 25th, Mon (Week 5)
2. Project progress report – Due: Oct 30th, Mon (Week 10)
3. Final project report – Due: Dec 11th, Mon (Week 16)

Sample Project Ideas and Links

Here are some project ideas that can be used as a starting point or to explore further.

- Real-time scheduling evaluation
    - Timing
    - API study

    - Scheduling latency measurement
    - Worst case execution time measurement
    - Real-time scheduling theory
    - Energy efficient scheduling
    - Dynamic voltage frequency scaling (DVFS) – based scheduling
- Kernel hacking
    - EDF scheduler on an embedded device (e.g. Raspberry Pi)
- Virtualization
    - Virtualization on an embedded device (e.g. Raspberry Pi)
    - Real-time Xen
    - Light-weight virtualization
- Real-time embedded applications
    - Internet of Things
    - Cyber-physical systems

Other example projects that can be found online:

- UCSD Embedded Systems course:
    - https://cseweb.ucsd.edu/classes/wi16/cse237A-a/finalproject/submissions/
- Embedded systems project ideas (note that the projects chosen for the class should also be related to embedded operating systems):
    - http://www.electronicshub.org/embedded-systems-projects-ideas/
- University of Trento, RTOS course:
    - http://disi.unitn.it/~abeni/RTOS/projects.html

Sample Hardware/Software Resources

The project requires implementation on an embedded device/hardware. The students can use a real device (preferred), an embedded emulation platform that mimics the behavior of an embedded device or a simulation platform that evaluates the behavior of an embedded device. Some examples are provided below. Note that you may choose your own hardware (other than the list below) provided with a clear explanation and description of the chosen hardware.

- Hardware
    - Raspberry Pi: https://www.raspberrypi.org/
    - Intel Edison: http://www.intel.com/content/www/us/en/do-it-yourself/edison.html
    - Arduino: https://www.arduino.cc/

- Emulation platform
    - Proemulator: http://proemulator.sourceforge.net/
    - QEMU: http://wiki.qemu.org/Index.html
    - Emul8: http://www.emul8.org/
    - Open source ARM emulators: http://www.thefreecountry.com/emulators/arm.shtml

- Simulation platform
    - Using simulation tools for embedded systems software development: http://www.embedded.com/design/real-time-and-performance/4007090/Using-simulation-tools-for-embedded-systems-software-development-Part-1
    - Open Virtual Platforms: http://www.ovpworld.org/
    - Embedded System Simulator: https://sourceforge.net/projects/ess-sim/

If your project tests/evaluates one or more systems, you may need to have standardized code (software) for uniform testing, i.e. benchmarks. Some open source benchmarks are listed below. You may choose your own benchmark (other than the list below) or implement a testing code, provided with a clear explanation and description.

- Linux benchmark suite: http://lbs.sourceforge.net/
- RocksDB: http://rocksdb.org/
- ParMiBench: https://github.com/cota/parmibench
- Phoronix test suite: http://www.phoronix-test-suite.com/