Computer Engineering 5170: Real-Time Systems
Prior Number – Computer Engineering 331

Class/laboratory schedule:
3 credit hours lecture (Three 50-minute lectures per week)

Instructor
Maciej Zawodniok, Ph.D.

Text
Professor-provided supplemental notes
Readings from various IEEE journals and magazines

Catalog Information
Introduction to real-time (R-T) systems and R-T kernels, also known as R-T operating systems, with an emphasis on scheduling algorithms. The course also includes specification, analysis, design, and validation techniques for R-T systems. Course includes a team laboratory assignments and a semester project to design an appropriate R-T operating system.

Prerequisite
Computer Engineering 3150 (213) or Computer Science 3800 (284)

Required or Elective Course
Elective course

Course Goals
General Outcomes
1. Understand the basics and importance of real-time systems
2. Analyze real-time properties of various basic scheduling schemes and validate against requirements specifications
3. Generate a high-level design document based on analysis documentation and state-of-the-art literature review
4. Generate a test plan based on requirements specification
5. Generate a validation plan based on all documentation
6. Understand basic multi-task scheduling algorithms for periodic, aperiodic, and sporadic tasks as well as understand the impact of the latter two on scheduling
7. Understand capabilities of at least one commercial off-the-shelf R-T kernel
8. Participate in a team design project, utilizing varying skill sets of members.
### Relationship of Course to Program Outcomes

<table>
<thead>
<tr>
<th>ECE Outcome</th>
<th>Course Outcome</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>M M M M S</td>
<td>Students develop and document a real-time project</td>
</tr>
<tr>
<td>b</td>
<td>S S S S M</td>
<td>Students must design, test, and validate their project</td>
</tr>
<tr>
<td>c</td>
<td>S S S S M</td>
<td>Students develop and complete an engineering design based on professor-assigned requirements</td>
</tr>
<tr>
<td>d</td>
<td>S S S S M</td>
<td>Projects and laboratory assignments are performed in teams</td>
</tr>
<tr>
<td>f</td>
<td>W S S S S M</td>
<td>R-T hazards and failures are discussed in week 1</td>
</tr>
<tr>
<td>g</td>
<td>S S S S M</td>
<td>Students write mid-term progress report; students prepare documentation and present their final project.</td>
</tr>
<tr>
<td>h</td>
<td>W W S S S S M</td>
<td>Importance and ubiquitous nature of R-T systems discussed.</td>
</tr>
<tr>
<td>i</td>
<td>W W S S S S M</td>
<td></td>
</tr>
<tr>
<td>j</td>
<td>W W S S S S M</td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>S S S S W</td>
<td>Project covers entire life-cycle from analysis through testing and validation before delivery.</td>
</tr>
<tr>
<td>l</td>
<td>W W S S S S M</td>
<td></td>
</tr>
</tbody>
</table>

S – strong connection; M – medium connection; W – weak connection

### Topics Covered

1. Hard vs. Soft R-T Systems and Life-Cycle Models (1 week)
2. Specifications and Architecture Model (1 week)
3. Multi-Tasking and Real-Time Scheduling (1.5 weeks)
4. Clock-Driven Scheduling (1.5 weeks)
5. Priority-Driven Scheduling (2 weeks)
6. Resource and Access Control (1.5 weeks)
7. Multi-processor Scheduling (1 week)
8. Communications and Networking (1 week)
9. Discussion and Demonstration of real time operating systems (1 week)
10. Embedded real-time system projects (3 weeks outside of class)
11. Real-time communication project (2 weeks outside of class)
12. Mid-Term report on semester project (outside of class time 0.5 week)
13. Final semester project report and presentation (1 week)
14. Reviews and 2 Examinations (1.5 weeks)
15. Final Examination (1 week)