SE 5402: Architecture of Internet of Things

Course Instructor: Parasara Sridhar Duggirala, Ph.D.

Catalog Description. 3 credits. This course is designed to provide students with a thorough understanding of the design, verification, and validation of embedded/network systems and software-intensive systems. The student will develop skills in specifying requirements for embedded software systems, model based architecture and design, and verification and validation of embedded systems. Special emphasis will be placed on distributed embedded systems and real-time systems. The platform-based design (PBD) flow will be used as the common thread through the course. Examples are driven by cyber-physical systems.

Course Objective. Embedded systems are deployed in highly constrained physical scenarios. To learn a development process of an embedded system, a designer should understand all the associated constraints. This course will help the students gain a fundamental understanding of the constraints and costs involved in three primary aspects of embedded systems, namely, hardware, software, and network.

Anticipated Student Outcomes. By the end of the course, a student will be able to:

1. Develop several hardware, software, and network architectures for a given embedded system.
2. Evaluate the cost, power, and performance tradeoffs associated with each architecture.

Course Outline

---

**Module 1: Hardware**

- Foundations of microarchitectures
- x86 assembly language
- Cost and power constraints
- FPGAs and ASICs

---

**Module 2: Software**

- Hardware and software co-design
- Programmable logic controllers
- Foundations of Real-Time Operating Systems
- Worst Case Execution Time (WCET)
Module 3: Networking

- ISO stack for networking
- Network protocols like TCP/IP, UDP, ATM
- Protocols for embedded systems like ZigBee, ZWave, CAN, TTP
- Distributed computing protocols like Chord, Pastry

Texts

- Computer Architecture – a Quantitative Approach by John Hennessey and David Patterson
- Introduction to Arduino – a piece of cake! By Alan G Smith.
- FPGA prototyping by VHDL examples by Pong Chu.
- Computer Networks by Andrew Tanenbaum.

Copyright. Copyrighted materials within the course are only for the use of students enrolled in the course for purposes associated with this course and may not be retained or further disseminated.

Due Dates and Late Policy. All course due dates are identified in the Course Schedule. Deadlines are based on Eastern Standard Time; if you are in a different time zone, please adjust your submittal times accordingly. The instructor reserves the right to change dates accordingly as the semester progresses. All changes will be communicated in an appropriate manner.

Student Conduct: [http://www.dosa.uconn.edu/student_code.html](http://www.dosa.uconn.edu/student_code.html). Students are responsible for adherence to the University of Connecticut student code of conduct. Pay attention to the section on Student Academic Misconduct, “Academic misconduct is dishonest or unethical academic behavior that includes, but is not limited, to misrepresenting mastery in an academic area (e.g., cheating), intentionally or knowingly failing to properly credit information, research or ideas to their rightful originators or representing such information, research or ideas as your own (e.g., plagiarism).” Examples of academic misconduct in this class include, but are not limited to: copying solutions from the solutions manual, using solutions from students who have taken this course in previous years, copying your friend’s homework, looking at another student’s paper during an exam, lying to the professor or TA and incorrectly filling out the student workbook.
Absences. Make-up of missed exams requires permission from the Dean of Students, see “Academic Regulations.” Midterm-exams are treated the same as Final Examinations. Students involved in official University activities that conflict with class time must inform the instructor in writing prior to the anticipated absence and take the initiative to make up missed work in a timely fashion. In addition, students who will miss class for a religious observance must “inform their instructor in writing within the first three weeks of the semester, and prior to the anticipated absence, and should take the initiative to work out with the instructor a schedule for making up missed work.”

Course Schedule*

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Module No</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aug 28-Sept 1</td>
<td>Week 1: Introduction to Design Flows for Embedded Systems and System Constraints</td>
<td>1</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Sept 4-8</td>
<td>Week 2: Foundations of Computer Architecture and Embedded Systems Hardware</td>
<td>1</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Sept 11-15</td>
<td>Week 3: Power and Cost Constraints on Embedded Systems Hardware</td>
<td>1</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Sept 18-22</td>
<td>Week 4: Introduction to Arduino and FPGAs</td>
<td>1</td>
<td>Project Proposal Discussion</td>
</tr>
<tr>
<td>Oct 2-6</td>
<td>Week 6: Worst Case Execution Time and Estimation Techniques</td>
<td>2</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Oct 9-13</td>
<td>Week 7: Real-Time Scheduling and Mixed Criticality</td>
<td>2</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Oct 16-20</td>
<td>Week 8: Introduction to Networking Protocols for Embedded Systems</td>
<td>3</td>
<td>Individual/Group project commences</td>
</tr>
<tr>
<td>Oct 23-27</td>
<td>Week 9: TCP-IP and UDP Networking Protocols</td>
<td>3</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Oct 30 – Nov 3</td>
<td>Week 10: Bluetooth, ZigBee, and ZWave Protocols</td>
<td>3</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Nov 6-10</td>
<td>Week 11: Low Range Communication RFID and NFC networking.</td>
<td>3</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Nov 13-17</td>
<td>Week 12: Internet-of-Things: an introduction for integration of Hardware, Software, and Networking</td>
<td>4</td>
<td>Lecture videos on Tuesday and Thursday</td>
</tr>
<tr>
<td>Nov 20-24</td>
<td>Thanksgiving Recess</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov 27- Dec 1</td>
<td>Week 13 Open topics in Internet-of-Things</td>
<td>4</td>
<td>Project Presentations</td>
</tr>
</tbody>
</table>
Adding or Dropping a Course. If you should decide to add or drop a course, there are official procedures to follow:

- Matriculated students should add or drop a course through the Student Administration System.
- Non-degree students should refer to Non-Degree Add/Drop Information located on the registrar’s website.

You must officially drop a course to avoid receiving an "F" on your permanent transcript. Simply discontinuing class or informing the instructor you want to drop does not constitute an official drop of the course. For more information, refer to the online Graduate Catalog.

Grading. As this is a graduate course, we will adopt relative grading for this course. The distribution of points will be fitted to a normal distribution curve and the grades will be awarded according to the performance of the student.

Academic Calendar. The University’s Academic Calendar contains important semester dates.

Students with Disabilities. Students needing special accommodations should work with the University’s Center for Students with Disabilities (CSD). You may contact CSD by calling (860) 486-2020 or by emailing csd@uconn.edu. If your request for accommodation is approved, CSD will send an accommodation letter directly to your instructor(s) so that special arrangements can be made. (Note: Student requests for accommodation must be filed each semester.)

Instructors’ Contact Information. Sridhar Duggirala: psd@uconn.edu

Helpful Links:
- Virtual Computer Lab at UConn: http://skybox.uconn.edu/
- Course Material: https://lms.uconn.edu
- Institute for Advanced Systems Engineering: http://www.utc-iase.uconn.edu/