CS326: Embedded Systems and the Internet of Things  
Calvin University  
Spring 2021

Course Instructor: Prof. Derek Schuurman


Other Tools: A customized Raspberry Pi IoT kit will be provided by the CS department at a reasonable cost. The kit contains parts that will be used for assignments and labs.

Course organization: There will be two lectures and a hands-on lab session each week. Lectures will be held on Mondays and Wednesdays and a lab session will be held in SB354 on Fridays.

Catalog Description
An introduction to topics in embedded systems and the Internet of Things (IoT) including hardware and software considerations for special-purpose computing applications that interact with the physical world. Hardware topics include embedded processors, I/O interfacing, sensors, and actuators. Software topics include scheduling considerations, IoT network protocols, the Web of things, state machines, and embedded programming. Additional topics include discussions of related social and ethical issues such as security, privacy, reliability, and the impact of automation. Lectures will be combined with hands-on lab exercises and a final project.

Prerequisites: Computer Science 112 (which may be taken concurrently) and Engineering 204 or 220, or permission of the instructor.

Student Learning Objectives: Upon successful completion of this course, a student will be able to:
- describe theoretical concepts in embedded systems and the Internet of Things (IoT)
- analyze a problem in embedded systems and IoT and design an appropriate solution
- implement software that applies concepts in embedded systems and IoT
- discuss social and ethical implications related to embedded systems and IoT from a Reformed Christian perspective

Lab Assignments: There will be weekly lab assignments posted in Moodle, most of which will be completed in pairs using kits that are provided. Weekly lab reports must include the names of the lab partners and must be submitted to Moodle by the following Thursday at midnight. Late lab submissions will receive at most 75% full credit. No labs more than one week late will be accepted. Students may discuss aspects of the assignments with each other, but every group must hand in their own work.

*Before attempting any labs, students must read and understand the safety guidelines posted on Moodle.*

Quizzes: There will be regular quizzes scattered throughout the semester comprised of multiple choice questions, short answers, true/false questions, and definitions.

Laptop policy: Laptops used in lectures must be used strictly for note-taking. Regular “hands-on” practice will be provided in the weekly lab sessions. Furthermore, the use of cell phones is not permitted during classes or labs.
**Grading:** The grading for the course will be weighted as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quizzes</td>
<td>10%</td>
</tr>
<tr>
<td>Lab assignments</td>
<td>40%</td>
</tr>
<tr>
<td>Final Project Presentation</td>
<td>5%</td>
</tr>
<tr>
<td>Final Project Report</td>
<td>20%</td>
</tr>
<tr>
<td>Final exam</td>
<td>25%</td>
</tr>
</tbody>
</table>

Grading queries must be raised within one week of the grades being posted. No work will be accepted after the last day of classes.

**Course Outline:** A tentative schedule for the course is shown below. The weekly reading assignments should be read since not all of the material in each chapter can be covered in detail during lectures.

<table>
<thead>
<tr>
<th>Week</th>
<th>Text and Readings</th>
<th>Topics</th>
<th>Labs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1</td>
<td>Feb. 3,5</td>
<td>Chapter 1 Chapter 8.1, 9.1 Introduction to embedded systems and IoT</td>
<td>Lab 1 Introduction</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Introduction to the Raspberry Pi hardware and software</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded processors, microcontrollers, DSPs, ASICS, FPGAs</td>
<td></td>
</tr>
<tr>
<td>Week 2</td>
<td>Feb. 8,10,12</td>
<td>Getting Started With Bash On The Raspberry Pi</td>
<td>Lab 2 Programming the Pi</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Memory architectures; flash memory considerations Operating systems and software considerations Programming the Raspberry Pi • Bash shell, assembly language, GNU C/C++ compiler; cross compiling, Python and Cython</td>
<td></td>
</tr>
<tr>
<td>Week 3</td>
<td>Feb. 15,17,19</td>
<td>Chapter 7 General Purpose I/O (GPIO) ports Sensors and actuators: models and examples Sensor issues: calibration, non-linearity, noise, and failures Sensor fusion example: Marzullo’s Algorithm</td>
<td>Lab 3 Using GPIO ports</td>
</tr>
<tr>
<td>Week 4</td>
<td>Feb. 22,24,26</td>
<td>Section 7.1 Section 10.1 A/D conversion: Quantization, noise, sampling, resolution Pulse Width Modulation (PWM) Microservos and actuators SPI, I’C, TIA232 and serial data communications</td>
<td>Lab 4 A/D Conversion</td>
</tr>
<tr>
<td>Week 5</td>
<td>Mar. 1,3,5</td>
<td>Chapter 3 State machines, state diagrams Reachability analysis, traces</td>
<td>Lab 5 PWMs and Servos</td>
</tr>
<tr>
<td>Week 6</td>
<td>Mar. 8,10,12</td>
<td>Sections 16.3, 16.4 Scheduling and timing analysis Interrupts and latency Real time operating systems (RTOS)</td>
<td>Lab 6 Scheduling and Latency</td>
</tr>
<tr>
<td>Week 7</td>
<td>Mar. 15,17,19</td>
<td>Sections 12.1, 12.2 Rate Monotonic Scheduling (RMS) Priority inversion and the Mars Pathfinder incident Watchdog timers Introduction and evolution of IoT</td>
<td>Lab 7 M2M Communications</td>
</tr>
</tbody>
</table>
| Week 9  
Mar. 29,31  
Apr. 2 | The Web of Things (WoT)  
HTTP and webservers; HTTP request/response & polling  
MQTT and websockets  
WebRTC protocol | No lab |
|---|---|---|
| Week 10  
Apr. 5,7,9  
| Intro to Edge AI  
(video) | SQL databases and IoT data  
IoT and Blockchain  
Big data, data analytics,  
Cloud vs. Fog computing  
Machine learning for embedded systems and Edge AI | Lab 9  
Camera sensors and the Web of Things |
| Week 11  
Apr. 12,16  
(April 14 Advising day) | Chapter 17  
6 Reasons Why  
IoT Security Is Terrible (IEEE Spectrum) | IoT and security issues:  
• the problem of default passwords  
• authentication and encryption  
• software updates  
• Exploits and DDoS attacks  
• IoT data security and privacy  
MQTT security: authentication and SSL encryption  
Embedded certificates  
Testing considerations | Lab 10  
Security and M2M communications |
| Week 12  
Apr. 19,21,23 | Are you sure your software will not kill anyone?  
(CACM)  
How the Boeing 737 Max Disaster Looks to a Software Developer (IEEE Spectrum)  
Engineers, Ethics, and the VW Scandal (IEEE Spectrum) | Social and ethical issues: reliability, safety, sustainability  
Privacy: cybersecurity laws and regulations  
Normative design principles: IoT for flourishing  
Case studies: Ariane 5 rocket explosion, Therac-25 incident, VW emissions scandal, Boeing 737 Max 8 incident | Work on final project |
| Week 13/14  
Apr. 26,28,30  
May 3,5 | Final Project Presentations  
Review | No labs |

**Academic Honesty:** Students are expected to display honesty and responsibility in completing assignments. Students are responsible for understanding the information on plagiarism contained in the Student Conduct Code (Article IV. B). For more information, refer to the [Calvin plagiarism policy](#).

**Accommodations:** Calvin University will make reasonable accommodations for persons with documented disabilities. Students should notify a disability coordinator in the [Center for Student Success](#) in order to arrange accommodations. Then, come and talk to me within the first two weeks of class so we can put your accommodations in place.

**Communication outside of Class Times:** Important announcements will be sent via Moodle to Calvin email, so students should check their Calvin email on a regular basis. Generally, the instructor will be happy to help you during lab times, during office hours (which are posted on my office door) or whenever I am in my office. The preferred way of communication outside the classroom is through Teams. Tutors are available by contacting the Center for Student Success.