ECE 341 Microcontrollers Lab

Credits and Contact Hours: 1 credit, 2 laboratory hours per week

Course Instructor or Coordinator: James F. Frenzel, PhD

Textbook: "Programming 32-bit Microcontrollers in C - Exploring the PIC32," Lucio Di Jasio, 2008.
Supplemental Materials: Laboratory handouts (course web page)

Course Catalog Description:
ECE 341 Microcontrollers Lab (1 cr) Lab to accompany ECE 340.

Coreq: ECE 340

Course Type: Required

Course Goals:

- •To instruct student how to solve engineering problems using microcontrollers
- To instruct students how to partition programs for sustainable software development
- To instruct students how to use modern embedded system development tools
- To instruct students how to use hardware instrumentation

Student Outcomes:
Data collected in this course are used to assess achievement of Student Outcome (6): An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions

Course Topics:

1. Using the Microchip MPLAB IDE
2. Microprocessor I/O and Lab Planning
3. Timers and software delays
4. Software finite state machines and stepper motor control
5. Multi rate process control
6. Interrupts
7. Parallel Communication and Handshaking
8. Serial Communications using RS232
9. I2C Communications
10. Pulse Width Modulation and dc motor control
11. Timing input signals using the Input Capture module

ECE 440 Digital Systems Engineering

Credits and Contact Hours: 3 credits, 3 lecture hours per week

Course Instructor or Coordinator: James F. Frenzel, PhD

Textbook: "Logic Design and Verification using SystemVerilog (Revised)", D. Thomas, 2016.
Supplemental Materials: Course Handouts (course web page)

Course Catalog Description:
Design of digital systems using a hardware description language and field-programmable gate arrays; projects emphasize a top-down design process using software tools; topics include data path optimization, pipelining, static and dynamic memory, technology issues, intra-system communication, and design for testability.

Prerequisites: ECE 240, 241, or permission

Course Type: Selected Elective

Course Goals:
- Learn fundamental concepts of digital system engineering
- Learn the use of hardware description languages for synthesis and verification
- Learn different design principles for implementing complex digital systems
- Learn the principles and basic functionality of programmable logic devices

Student Outcomes:
Data collected in this course are used to assess achievement of Student Outcome (2): An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.

Course Topics:
- Introduction to digital systems engineering
- Review of fundamental digital logic topics
- Use of a Hardware Description Language
- Functional, data-path and control-path description
- Implementation of complex digital systems
- Timing analysis and crossing clock domains
- Testability of digital designs

ECE 443 Distributed Processing and Control Networks

Credits and Contact Hours: 3 credits, 3 lecture hours per week

Course Instructor or Coordinator: James F. Frenzel, PhD

Textbook: "Programming 32-bit Microcontrollers in C - Exploring the PIC32," Lucio Di Jasio, 2008.
Supplemental Materials: Course handouts. (course web page)

Course Catalog Description:
This course has three major parts: real-time computing, distributed processing, and control networks. Analysis of hardware and software performance with respect to speed, accuracy, and reliability. Investigation of ways of maximizing the three essential processor resources: memory, CPU time, and input/output. Methods for writing error free programs and designing fault tolerant computing systems.

Prereqs: ECE 340, 341, 350, and 351

Course Type: Selected Elective

Course Goals:

- Teach students to verify embedded system functionality and analyze performance
- How to implement multitasking embedded systems utilizing a real-time operating system
- How to manage multi-processor control using network communications

Student Outcomes:
Data collected in this course are used to assess achievement of Student Outcome (2): An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.

Course Topics:

- Designing embedded systems using a real-time operating system
- Use of visualization tools for characterizing performance
- Various networking technologies
- TCP/IP and web-based control
- Software best practices