Western New England University
College of Engineering

CPE 355 - Real Time Embedded Kernels - Spring 2012

TR 11:00 am to 12:20 pm @ Sleith-301

http://www.nunoalves.com/classes/spring_2012_cpe355

Instructor: Prof. Nuno Alves

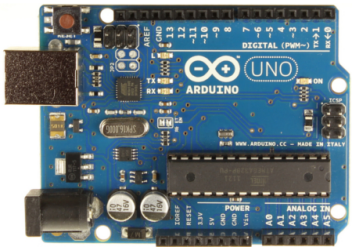
Contact Information: 401-633-4660 or nalves@wne.edu

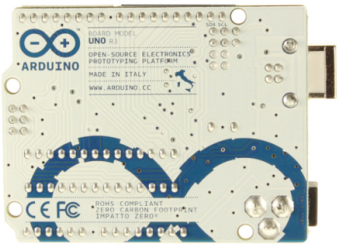
Office Hours: TR 1:00pm to 2:00pm at Sleith-313

Pre-requisite: CPE 305 and CPE 310.

Course Description: This is an introductory course in the theory, design, and use of a real-time kernel for an embedded system. A real-time kernel is the control software that manages the time resources of a microprocessor. Students learn the basic structure and services of a kernel. Topics include dispatching, hierarchical scheduling, priority-driven scheduling, real-time schedulers (including non-preemptive and preemptive), scheduling groups, and multitasking. Students also learn to utilize tasks to describe multiple threads of execution in a computation. Students study methods to manage and control task execution as well as other kernel services.

Required Material: Arduino UNO R3 which you can buy at www.sparkfun.com.





Recommended Materials:

• "An embedded software primer" by David E. Simon, 1999 Addison-Wesley Professional. ISBN: 020161569X.

• "Introduction to Embedded Systems: Using ANSI C and the Arduino Development Environment" by David Russell, 2010 Morgan and Claypool Publishers. ISBN: 1608454983.

Learning Objectives and Assessment: After taking the course, students should be able to:

Objectives/Outcomes	Assessment Techniques
1. Understand embedded systems concepts	Assignments, exams
2. Understand real time scheduling issues	Assignments, exams
3. Understand issues in task communication and synchronization	Assignments, exams
4. Understand performance issues	Assignments, exams

Relationship of Course Outcomes to ABET Student Outcomes

	ABET Student Outcomes													
	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**	**n**
1					x						x	x		x
2	x											x	x	
3	x										x		x	x
4	x				x									

Course Requirements:

1. *Assignments* (70% of final grade)*:* There will be very regular take-home graded assignments. Shorter take-home assignments (homework) are due at the start of the following class. Longer take-home assignments (projects) are due one week after are assigned, also at the start of the class. A possible solution will be posted on the course website after its due date.

2. *Examinations* (30% of final grade): There will be two hour-long exams throughout the course. These exams will consist of pen-and-paper exercises.

The final grade for each student will be linearly scaled with respect to the highest grade in class. For example, if the final grades for Joe, Richard and Peter are 78, 56 and 86. After linearly scaling them, the final grades will be 92, 70 and 100.

Grading: The range of numerical grades and the traditional letter equivalents are as follows:

A 93-100	B+ 87-89	C+ 77-79	D+ 67-69	F 0-59
A- 90-92	B 83-86	C 73-76	D 60-66	
	B- 80-82	C- 70-72		

The final grade will be rounded to the nearest integer. For Example, 89.5 will be rounded to a 90, whereas a 89.4 will be rounded to a 89.

Examinations Policy: In case of documented emergencies or medical conditions you may retake the exam at a later date.

Assignments Policy:

- Assignments are due at the beginning of the class.

- Unless there is a documented emergency late assignments will **not** be accepted, and a grade of zero will be assigned.

- If I see something very unusual about your work I will have to consider your entire solution wrong. Now, what is my definition of "unusual work"? Well, things like 95% of code that have been copied from a website or chunks of your implementation that are extremely close to one of your colleagues.

- I really encourage communication amongst yourselves but don't copy solutions from each others, as it goes against the university integrity of scholarship guidelines.

WNEU integrity of scholarship guidelines: "Honesty in all academic work is expected of every student. This means giving one's own answers in all class work, quizzes, and examination without help from any sources not approved by the instructor. Written material is to be the student's original composition. Appropriate credit must be given for outside sources from which ideas, language, or quotations are derived".

Attendance Policy: While it is university policy is that students are expected to attend all class sessions for courses in which they are enrolled, students will not be penalized if they do not show up to regular classes.

Mid-Semester Grades: Instructors are required to submit mid-semester grades. These mid-term grade will reflect the instructor's actual evaluation of student's progress.

Other Information:

1. Student grades will be posted regularly on manhattan (https://manhattan.wnec.edu). Manhattan will also be used for class related posts.

2. Classes will be a mixture of hands on laboratory work and standard presentation of material and examples. Class discussion and participation is encouraged.

3. Changes in syllabus and assignment sheet may be modified as deemed appropriate. All changes will be announced in class.

4. Students with a disability who are requesting academic accommodations should contact the SDS office in Deliso GO6, or call 782-1257/1528 for an appointment.

5. Any student who is unable, because of his religious beliefs, to attend classes or to participate in any examination, study, or work requirement on a particular day shall be excused from any such examination or study or work requirement, and shall be provided with an opportunity to make up such examination, study, or work requirement which he may have missed because of such absence on any particular day; provided, however, that such makeup examination or work shall not create an unreasonable burden upon such school.

6. Faculty evaluations will be administered at the end of the semester.

7. There will be no extra-credit offered on this course.

Topics to be covered:

1. Digital logic and information representation review (Russell Chap 1)

2. Introduction to RTOS Characteristics (Simon Chap 1)

3. Processor architectures (Simon Chap 2 and 3)

4. Interrupts (Simon Chap 4)

5. Introduction to Arduino (Russell Chap 3)

6. ANSI C (Russell Chap 2 + Russell Appendix D)

7. Embedded debugging (Russell Chap 4)

8. The ATMEL AVR Processor (Russell Chap 5)

9. Using ATmega328P I/O, timer and analog ports (Russell Chap 6, 7, 8)

10. Interrupt processing on ATmega328P (Russell Chap 9)

11. Serial communications on ATmega328P (Russell Chap 10)

12. Advanced topics on ATmega328P (Russell Chap 11,12)

13. Software architectures (Simon Chap 5)

14. Tasks and task states, shared data problem, semaphores and shared data (Simon Chap 6)

15. Message queues, mailboxes and pipes, timer functions, events (Simon Chap 7)

16. RTOS specification and design (Simon Chap 8, 9 and 10)