EE 319K: Introduction to Embedded Systems
Sections - 16275, 16280, 16285, 16290

Andreas Gerstlauer

Spring 2012

General Information

Class Time	TTh 2:00-3:30pm
Classroom	**ACA 1.104**
Lab	**ACA 1.106** and **ACA 1.108**
Contact	gerstl@ece.utexas.edu
Office	ACES 6.118
Office hours	M 3:00-4:30 pm, W 1:30-3:00pm, or after class/by appointment
Website	UT Blackboard and class site: http://www.ece.utexas.edu/~gerstl/ee319k_s12
TAs	Peter Garatoni, peter.garatoni@gmail.com Manan Kathuria, manan@mail.utexas.edu Jae Hong Min, jmin@mail.utexas.edu Pratyusha Nidamaluri, pratyusha29@gmail.com Dayo Lawal, dayo.lawal@utexas.edu Aditya Srikanth, aditya.srik@utexas.edu Kurt Fellows, kfellows85@gmail.com James Beecham, james.d.beecham@gmail.com Austin Blackstone, austinblackstone@aol.com Razik Ahmed, razik.ahmed@utexas.edu Brandon Carson, brandon.carson@utexas.edu Kin Hong Mok, kinhong.mok@utexas.edu Omar Baca, omar_bacas@hotmail.com Sam Oyetunji, djsho729@yahoo.com Zack Lalanne, zack.lalanne@gmail.com Nathan Quang Minh Thai, mnathanqthai@gmail.com Paul Fagen, pfagen@utexas.edu The TAs will hold office hours in the lab (ACA 1.106), a schedule will be posted there. About 32 students per TA.
Mailing list	sp12_ee319k@utlists.utexas.edu (all Professors and TAs)

Teaching Philosophy

I strongly encourage students to take an active role in this class. Questions are welcome before, during and after class. Please email, visit my office hours or call me if you have any type of question.

Catalog Description

Embedded systems; machine language execution; assembly and C language programming; local variables and subroutines; input/output synchronization; analog to digital conversion and digital to analog conversion; debugging; and interrupts.

Overview

EE319K will continue the bottom-up educational approach, started in EE302 and EE306. The overall educational objective is to allow students to discover how the computer interacts with its environment. It will provide hands-on experiences of how an embedded system could be used to solve EE problems. The focus will be understanding and analysis rather than design. The analog to digital converter (ADC) and digital to analog converter (DAC) are the chosen mechanism to bridge the CE and EE worlds. EE concepts include Ohms Law, LED voltage/current, resistance measurement, and stepper motor control. CE concepts include I/O device drivers, debugging, stacks, FIFO queues, local variables and interrupts. The hardware construction is performed on a breadboard and debugged using a multimeter (students learn to measure voltage and resistance). Software is developed in Freescale 9S12 assembly and C; 3 out of the 10 labs are simulated-only in TExaS and the other 7 are first simulated in TExaS and then run on the real 9S12. Software debugging occurs during the simulation stage. Verification occurs in both stages. Labs 1 through 6 are written in 9S12 assembly language. Labs 8 and 9 are written in a combination of assembly and C. Labs 7 and 10 are written in a C.

Prerequisites

EE306 or BME303 with a grade of at least C-. You should recall:

1. Bits
    a. Numbers - Unsigned and Signed Integer representation in 2's Complement
    b. True/False - Logical Operations
    c. Characters - ASCII representation
2. Gates
    a. AND, NAND, OR, NOR, NOT, XOR
    b. DeMorgan's Laws
3. Computer Components
    a. Central Processing Unit: Arithmetic Logic Unit and Control Unit
        i. Registers
    b. Memory – ROM and RAM
        i. Address Space and Addressability
    c. I/O
4. LC3 Assembly Language
    a. Instruction Set (ISA)
    b. Pseudo-ops
    c. Op-codes and Operands
        i. Memory operations
        ii. Arithmetic and Logic operations
        iii. Control operations – Branches/Jumps
    d. Addressing Modes
5. Programming
    a. Algorithms, Flow-Charts
    b. Data Types
    c. Data Structures (Arrays and Linked Lists)
    d. Interrupts

Course Objectives and Outcomes

1. Understanding how the computer stores and manipulates data (characters, integers, and fixed-point numbers), the basic arithmetic and logical operations performed by the computer.
2. The understanding of embedded systems (a system with the computer hidden inside) using modular design and abstraction.
3. Assembly and C language programming: considering both function and style.
4. Understanding how the computer executes instructions (fetch opcode, fetch operand, read data, operate, and write data).
5. The use of a microcontroller (strategic use of RAM, ROM and I/O). Microcontrollers typically have a little RAM and a lot of ROM. Globals, locals and the heap go in RAM. Constants and programs go in ROM.
6. Debugging and verification using a simulator and on the microcontroller (embedded systems typically do not have a print function). Debugging using breakpoints, scanpoints, profiles, monitors, voltmeters, oscilloscopes, logic analyzers.
7. How input/output actually happens (the students wire up analog and digital signals to the 9S12 and measure them with a voltmeter), synchronization, including switches, LEDs, LCDs, DACs, ADCs, and serial ports.
8. The implementation of an I/O driver (a set of programs that perform I/O).
9. Understanding, from an architecture standpoint, how local variables and parameters work (e.g., space on the stack is dynamically created, the local variable is accessed using stack-pointer relative addressing, then the space is deallocated).
10. Analog to digital conversion (ADC), e.g., the students interface a slide potentiometer to the ADC and write software that measures the position of the slide, creating a display like “1.23 cm”.
11. Interrupt synchronization, real-time ADC sampling (periodic timer interrupts), introduction to multithreaded programming.
12. Simple motors (e.g., open and closed-loop stepper motor control).
13. Digital to analog conversion (DAC), used to make simple sounds.
14. Design and implementation of elementary data structures, such as linked lists, stacks and queues.

After the successful conclusion of EE319K students should be able to understand the basic components of a computer, write assembly and C language programs that perform I/O functions and implement simple data structures, manipulate numbers in multiple formats, and understand how software uses global memory to store permanent information and the stack to store temporary information.

Text and Reference Materials

J. W. Valvano, *Introduction to Embedded Systems: Interfacing to the 9S12*, Cengage Publishing, 2009, ISBN-10: 049541137X / ISBN-13: 9780495411376, list of known errors at: http://users.ece.utexas.edu/~valvano/typo3.pdf

Reading assignments will be given from Patt's *Introduction to Computing Systems* (text-book for EE306 and BME303). Data sheets for most of the devices used in this class are available as PDF files on Dr. Valvano's site. Please make sure you have access to the CPU12 instruction manual, either in print (from Freescale) or PDF form. You should download these data sheets and have them available when you are developing code:

- CPU12 quick reference (32 pages): http://users.ece.utexas.edu/~valvano/Datasheets/CPU12rg.pdf
- CPU12 programming reference (458 pages): http://users.ece.utexas.edu/~valvano/Datasheets/S12CPUV2.pdf
- 9S12DP512 datasheet (1406 pages): http://users.ece.utexas.edu/~valvano/Datasheets/MC9S12DP512.pdf
- 9S12DG128 datasheet (595 pages): http://users.ece.utexas.edu/~valvano/Datasheets/MC9S12DG128.pdf
- Tech arts board information: http://users.ece.utexas.edu/~valvano/Datasheets/TechArts9S12DP512.pdf

Supplementary material:

- C programming book (C Primer) specific for the Metrowerks compiler and 9S12: http://users.ece.utexas.edu/~ryerraballi/CPrimer/
- Short Movies on TExaS: http://users.ece.utexas.edu/~valvano/Readme.htm
- Web videos of the example lessons from the book: http://users.ece.utexas.edu/~valvano/Lessons/
- I strongly advise you work through the tutorials in the book

Equipment

Board: We are no longer giving microcontroller boards to students. We can tell you that the discussions on how to maintain educational excellence in the face of repeated budget cuts have been long and careful. Every group of two students will be required to have one 9S12 board. Either the Technological Arts Adapt9S12DP512 or the Adapt9S12DG128 will suffice. The first possibility is to buy or borrow a board from a previous EE319K/EE445L student. If you obtain a kit from a previous student, you will need the board, power supply, RS232 cable, and a protoboard. If you do buy a board from another student, I recommend you run the hardware checker to make sure the board is functional. The second possibility is to purchase a new Technological Arts Adapt9S12DG128 kit, which includes a board, power supply, serial cable, and solderless breadboard. The student cost will be $79.00 per kit. The link to pay for this kit is active. The buying process will be:

1. Once you have selected your partner for Labs 3-9, you go to the http://www.technologicalarts.ca/uta/ web site and pay $79 to TechArts for the kit via PayPal. Please pay for your kit at least 7 days before the demonstration lab period. Our experience is that many credit card companies will add $1 to $3 to the transaction because TechArts is a Canadian company. Neither UT nor TechArts sees this added fee. The fee is charged by the credit card company.
2. The department will get an email from TechArts that you have paid.
3. You will go to ENS234 to pickup the kit. There will be specific scheduled hours for the pickup phase, which will be around Feb 8 to 15.
4. There will be a board demonstration lab between labs 2 and 3, the week of Feb 13-17, and you should bring your board to this demonstration. Your TA will show

you how to test your board to verify it is operational. If the board does not work during the testing procedure, we will replace it. Our experience with EE319K is if the board works during the initial test, almost all students complete EE319K labs without damaging the board. However, if you do damage your board, then you must purchase another one.

Unfortunately we will not be using the board in EE319K in Fall 2012, so you will not have many opportunities to sell it. However, it could be useful for senior lab or other projects. There is no difference to a EE319K student between the Adapt9S12DP512 and Adapt9S12DG128 board. The DG128 board has a little less memory and is a lot less expensive.

Tools: Every student should own their own voltmeter and their own wire strippers. The NI MyDAC you bought in EE302 can be used as a voltmeter. Otherwise, a voltmeter less than $20 will do, see for example:

BG Micro:
http://www.bgmicro.com/MET1014.aspx

Jameco:
http://www.jameco.com/webapp/wcs/stores/servlet/Product_10001_10001_220812_-1

Harbor Freight, has 3 locations around Austin, usually sells voltmeters for less than $10:
http://www.harborfreight.com/7-function-digital-multimeter-90899.html

Since you will be making only 5 solder joints all semester, all students will be able to share the soldering iron in lab. However, all EE319K students will need their own voltmeter and wire strippers.

Safety warning: Due to the lead in most solder, please wash your hands after soldering, before eating or drinking. If you drop the soldering iron, let it fall to the ground. Do not try and catch it. If you are pregnant or think you might be pregnant, have someone else do the soldering.

Kit handed out by TAs:

- 1 7406
- 6 LEDs (20 mA, 2 red, 2 yellow, 2 green)
- 6 220 ohm 5%, 0.25 watt resistors
- 3 push-button switches
- 3 10k ohm 5%, 0.25 watt resistors
- 3 1.5k ohm 5%, 0.25 watt resistors
- 3 12k ohm 5%, 0.25 watt resistors
- 2 feet of 4-Wire 24 gauge solid wire
- 1 0.1 uF ceramic bypass cap
- 1 headphone jack
- 1 20k ohm slide pot

Those do not need to be returned.

Long term checkout:

- 1 LCD display and serial cable

If you check out either of these two and do not return them by 4/29, we will reduce your overall grade in EE319K by one letter grade.

Computer: The following remarks are the personal opinions of the EE319K instructors, and do not reflect an official position of the department, college, or university. We feel

very strongly that you should have your own computer on which you develop your software and write your reports. Both software development and report writing should be done without paper, pencil and erasers. Having a computer at home allows you to organize your information (files, directories etc.) as well as your schedule (allocate your software development time for that the time of day during which you are most creative and energetic.) Physiologically most people are more energetic in the morning. On the other hand, there are fewer distractions late at night. Some students do all their EE319K lab work on campus, but most find it convenient to configure their home computers to work with the EE319K hardware.

To work at home you will need to install TExaS and Metrowerks Codewarrior for the 9S12. For drawing electrical circuit diagrams some students use PCBArtist. All three of these applications run on a PC computer running Windows XP, Vista or Windows 7. The software development systems will run on any PC with a COM port.

Lab computer usage: Computers in ACA 1.106 and 1.108 are available for your usage. TA's in the laboratory are checking off programs and supervising while on duty, thus you can expect to have only a brief consultation with them. You should learn to develop software while on the computer. This course involves some projects that require extended periods of time to complete and a project cannot be done just overnight. Get started on an assignment early so you can get help if you need it. There are not enough machine hours to give everyone more than enough time to complete each program. We expect most students to have a laptop have access to a machine to run at home. Unfortunately, expect the laboratory to be crowded and machine time to be scarce if you attempt to work in the lab a day or two before an assignment is due. You will be competing with students in other courses for computer time. We need your assistance in the laboratory. Please report any equipment problems to the TA if they are present. If a TA is not present use the form provided in the lab. If you do this we can rapidly get repair service. Please help to keep the lab clean.

Software

TExaS: The simulator application, called Test EXecute And Simulate (TExaS), is not freeware, so please don't post it on the net or otherwise send it to others. On the other hand, Prof. Valvano has granted EE319K students indefinite usage of the software, including installing the application on each of your personal computes. If you know of someone interested in the application have them contact Prof. Valvano directly. It is the 14th semester we will be using the application in EE319K, but Prof. Valvano continues to make updates to fix bugs or add features. Please work through the tutorials and examples to bring you up to speed on the various aspects of the system. Instructions for installing/upgrading TexaS will be provided on Blackboard. The TExaS application itself runs on Vista/Windows7, but the help system does not run without a patch. Ask your TA how to get the TExaS help system to run under Vista/Windows7.

Metrowerks Codewarrior: The instructions to download the free (special) 9S12 version can be found at http://users.ece.utexas.edu/~valvano/S12C32.htm#Metrowerks. The most recent version will run on Windows 7.

PCBArtist: You will be drawing a 5 or 6 circuit diagrams as part of your lab assignments. You are free to draw these diagrams in any manner you wish as long as the information is presented in a complete and professional manner. One option for drawing circuits is PCBArtist, which is available at http://www.4pcb.com/free-pcb-layout-software. The use of PCBArtist is optional for EE319K students.

Grading Criteria

Task	Date	Percentage
Homework/Programming Assignments	Usually due on Mondays	10%
Laboratory Assignments	Due at lab times (Tuesdays/Wednesdays)	30%
Test 1 (in class)	Thursday, February 23, 2:00-3:30pm	15%
Test 2 (in lab, ACA 1.106 & 1.108)	Tuesday/Wednesday, March 20/21	20%
Final Exam	Saturday, May 12, 2:00-5:00pm	25%

There will be no re-tests, make-ups, or incompletes.

Attendance: Students are expected to attend lectures. The book covers more information than the class, and we will use lectures to map our way through the book. If you miss class you may find it difficult to catch up.

Laboratory Policies and Lab Partners: See the EE319K Laboratory Manual, available at http://users.ece.utexas.edu/~valvano/EE319K/EE319K_LabManualSp12.pdf

There will be a signup for lab checkout times. Lab partners need to be enrolled in the same lab section. Each student or team of students will have a specific checkout time during their lab section, which will be when their lab will be demonstrated to their TA. You can only switch lab sections using official add/drop procedures.

Homework Assignments: There will be approximately one homework assignment each week involving programming in assembly or C. Homeworks will be submitted online through Blackboard. You may work in groups and discuss homework problems with others, but every student has to independently develop his own, separate homework solution. Late homework submissions will not be accepted under any circumstances.

Tentative Lecture Schedule

Week	Reading	Topic
1	Ch. 1 Syllabus	Introduction: Course administration, Embedded systems, Development cycle, Flow charts, Data flow and call graphs
	Ch. 2, 3 App. 1.1-1.8.2	Numbers: Binary, Decimal, Hexadecimal Architecture: 9S12 assembly programming, TExaS simulator
2	Ch. 2, 4 Patt Ch. 11	Execution: 9S12 architecture, Subroutines, Simple addressing modes C Programming: Introduction to C, Structure of a C program (C Primer)
	Ch. 2, 3 Patt Ch. 12	I/O: Parallel Ports, Direction registers, Logical and shift operations C assignments and boolean expressions
3	Sec. 4.7, 5.7 Patt Ch. 14, 15	Debugging in TExaS C functions
	Ch. 3, Sec. 5.2 Patt Ch. 12, 13	Arithmetic: Addition/subtraction operations, Condition code bits, Conditionals

Week	Reading	Topic
4	App. 2, Sec. 2.9	Board: Demonstration of the board (bring your board to class) Switch Input and LED Output
	Sec. 5.1, 5.2 Patt Ch. 13, 14	Control structures: If-then, Loops Modular programming: Subroutines and the Stack
5	Sec. 6.1-6.3 Patt Ch. 16	Pointers: Indexed Addressing, Arrays, Strings C pointers and arrays
	Sec. 4.5 Sec. 6.11	Timers: Timer programming Functional debugging dump
6	Ch. 1 - 5	Review for Test 1
	Ch. 1 - 5	**Test 1 (closed book), in class**
7	Sec. 6.5, 6.8 Patt Ch. 19	Finite State Machines (FSMs): Structures, Linked lists C structs
	C Primer Patt Ch. 19	C data structures, Linked lists FSMs in C
8	Sec. 7.1-7.5 Sec. 5.4	Local variables, Stack frames and Parameter passing Recursion
	C Primer Patt Ch. 14, 17	C scoping (local/global) C function calling conventions, parameter passing, recursion
9	-	*Spring Break*
10	Ch. 5, 6	**Test 2 (closed book), in lab (ACA 1.106 & 1.108)**
	Sec. 8.5, 9.1 Sec. 10.1	LCD interface: LCD programming, I/O Synchronization, Fixed-point numbers, Number conversions
11	Sec. 9.2, 9.4 Sec. 9.6, 9.10	Interrupts: 9S12 Interrupts and interrupt processing Timer and periodic (Output Compare) interrupts
	C Primer	C/assembly interface Metrowerks
12	Sec. 11.2, 11.3	Digital-to-Analog Conversion (DAC): Sound Generation
	Sec. 11.1, 11.4	Analog-to-Digital Conversion (ADC): Lab 8 design methods
13	Sec. 10.2	Numerical calculations: Multiply/divide Table Lookup, Interpolation
	Sec. 8.1	Serial I/O: Serial Communications Interface (SCI), Serial interrupts
14	Sec. 12.1-12.4	Thread communication: Producer-consumer problems, FIFO queues
	Ch. 8.8	Stepper Motors: Motor control, Controller state machines
15	-	Advanced Embedded Systems Design, Systems Engineering
	-	Wrapup, Outlook, TRobots

Week	Reading	Topic
16	Ch. 1 - 13	Review for Final Exam
	Ch. 1 - 13	**TRobots Competition**
Finals	Ch. 1 – 13	**Final Exam, regularly scheduled time and place**

Tentative Lab Schedule

Week	Task
1	None
2	Go to ACA 1.106 for demonstration
3	**Lab1**: Digital lock I/O, parallel port, direction register and logical function, written in assembly (simulated, groups of two)
4	**Lab2**: LED and switch interface, written in assembly (simulated, groups of two)
5	Real board demonstration, bring your board to lab
6	**Lab3**: LED and switch interface, written in assembly (board, groups of two)
7	**Lab4**: Debugging techniques, one switch, one LED, written in assembly (simulated and board, groups of two)
8	**Lab 5**: Traffic Light Controller using the real 9S12, written in assembly (simulated and board, groups of two)
9	*Spring Break*
10	**Test 2 (closed book), during regularly scheduled lab time, in lab (ACA 1.106/108)**
11	**Lab 6**: LCD device driver, decimal fixed-point output, local variables, written in assembly (simulated and board, groups of two)
12	Go to ACA 1.106 for Metrowerks C demonstration
13	**Lab 7a or 7b**: Digital piano or pacemaker using a 4-bit DAC, written in C (simulated and board, groups of two)
14	**Lab 8**: Real-time position monitor, ADC, interrupts, LCD, mixture of assembly and C (simulated and board, groups of two)
15	**Lab 9**: Distributed DAS, serial port interrupts, FIFO queue, mixture of assembly and C (simulated and board, groups of four)
16	**Lab 10:** TRobot programming competition, written in C

Tentative Homework Schedule

Date	Reading	Task
01/30		**Homework 1:** Hand Assemble

Date	Reading	Task
02/6		**Homework 2:** Assembly concepts
02/13	Patt Ch. 11, 12	**Homework 3:** Signed/unsigned numbers, Arithmetic/logic operations
02/27	Patt Ch. 13, 14	**Homework 4:** If-then-else, Loops, and Functions
03/5	Patt Ch. 14, 16	**Homework 5:** Functions and arrays
03/19		**Homework 6:** Practice Exam 2s
03/26	Patt Ch. 15, 16	**Homework 7:** Solve Exam Problem 1 in C, Pointers and testing
04/2	Patt Ch. 15, 16	**Homework 8:** Solve Exam Problem 2 in C, Pointers and testing
04/9	Patt Ch. 17	**Homework 9:** Recursion problem
04/23	Patt Ch. 19	**Homework 10:** A FSM tester problem, structures

Legal Notices

Add/Drop: The 12th class day is Wednesday, February 1, 2012. The drop policy has recently changed. See your academic advisor or the Dean of Students for more information.

Course Evaluations and Final Exam: Course evaluation is conducted on the last class day in accordance with the Measurement and Evaluation Center form. The final exam is at the time and place stated in the course schedule.

Academic Dishonesty: "*Faculty in the ECE Department are committed to detecting and responding to all instances of scholastic dishonesty and will pursue cases of scholastic dishonesty in accordance with university policy. Scholastic dishonesty, in all its forms, is a blight on our entire academic community. All parties in our community -- faculty, staff, and students -- are responsible for creating an environment that educates outstanding engineers, and this goal entails excellence in technical skills, self-giving citizenry, an ethical integrity. Industry wants engineers who are competent and fully trustworthy, and both qualities must be developed day by day throughout an entire lifetime. Scholastic dishonesty includes, but is not limited to, cheating, plagiarism, collusion, falsifying academic records, or any act designed to give an unfair academic advantage to the student. The fact that you are in this class as an engineering student is testament to your abilities. Penalties for scholastic dishonesty are severe and can include, but are not limited to, a written reprimand, a zero on the assignment/exam, re-taking the exam in question, an F in the course, or expulsion from the University. Don't jeopardize your career by an act of scholastic dishonesty. Details about academic integrity and what constitutes scholastic dishonesty can be found at the website for the UT Dean of Students Office and the General Information Catalog, Section 11-802.*"

You are encouraged to study together and to discuss information and concepts with other students. You can give "consulting" help to or receive "consulting" help from such students in oral form. However, this permissible cooperation should never involve one student having possession of a copy of all or part of work done by someone else, in the form of an email, an email attachment file, a portable storage device, or a hard copy. Copying of any part of a program is cheating without explicit reference to its source. We do enter lab and homework assignments turned in by EE319K students through a pla-

giarism checker, comparing them to assignments of this and previous semesters. If we find two programs that are copied, there will be a substantial penalty to both students, e.g., failure in the course. Students who cheat on tests, in homeworks or in lab will fail. Prosecution of cases is very traumatic to both the student and instructor. It is appropriate to use software out of the book, class website as long as all copy-pasted software is explicitly referenced. Copy-pasting software from current or past EE319K students is scholastic dishonesty.

Policies concerning the use of other people's software in this class:

- I strongly encourage you to study existing software.
- All applications and libraries must be legally obtained. E.g.,
  - You may use libraries that came when you bought a compiler.
  - You may use software obtained from the web.
  - You may copy and paste from the existing source code.
- You may use any existing source code that is clearly referenced and categorized:

  original: completely written by you,
  derived: fundamental approach is copied but it is your implementation,
  modified: source code significantly edited to serve your purpose,
  copied: source code includes minor modifications.

University Honor Code: "The core values of the University of Texas at Austin are learning, discovery, freedom, leadership, individual opportunity, and responsibility. Each member of the University is expected to uphold these values through integrity, honesty, trust, fairness, and respect toward peers and community." (see the university catalog)

Electronic Mail Notification Policy: In this course e-mail will be used as a means of communication with students. You will be responsible for checking your e-mail regularly for class work and announcements. The complete text of the University electronic mail notification policy and instructions for updating your e-mail address are available at http://www.utexas.edu/its/policies/emailnotify.html.

Use of Blackboard and Class Web Site:This course uses the class web page and Blackboard to distribute course materials, to communicate and collaborate online, to submit assignments and to post solutions and grades. You will be responsible for checking the class web page and the Blackboard course site regularly for class work and announcements. As with all computer systems, there are occasional scheduled downtimes as well as unanticipated disruptions. Notification of disruptions will be posted on the Blackboard login page. Scheduled downtimes are not an excuse for late work. However, if there is an unscheduled downtime for a significant period of time, I will make an adjustment if it occurs close to the due date.

Religious Holidays: Religious holy days sometimes conflict with class and examination schedules. If you miss an examination, work assignment, or other project due to the observance of a religious holy day you will be given an opportunity to complete the work missed within a reasonable time after the absence. It is the policy of The University of Texas at Austin that you must notify each of your instructors at least fourteen days prior to the classes scheduled on dates you will be absent to observe a religious holy day.

Students with Disabilities:The University of Texas at Austin provides upon request appropriate academic accommodations for qualified students with disabilities. For more information, contact the Office of the Dean of Students at 471-6259, 471-4641 TTY or the College of Engineering Director of Students with Disabilities at 471-4382.

ABET Accreditation Material

Three lecture hours and one laboratory hour a week for one semester.
Design Assignments: Labs 4, 7, 8 (1 week each)
Laboratory Projects: Labs 1, 2, 3, 5, and 6
SCH Engineering Topics 3 (Including: 1 SCH of Engineering Design)

Relationship of the Course to ABET EC2000 Program Outcomes*:*

√	**1. ABET EC2000 Program Outcomes**
√	a. An ability to apply knowledge of mathematics, science, and engineering
√	b. An ability to design and conduct experiments, as well as to analyze and interpret data
√	c. an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
	d. An ability to function on multi-disciplinary teams
√	e. An ability to identify, formulate, and solve engineering problems
	f. An understanding of professional and ethical responsibility
√	g. An ability to communicate effectively
	h. The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context
√	i. A recognition of the need for, and an ability to engage in life-long learning
√	j. A knowledge of contemporary issues
√	k. An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice

ABET Criterion 9, program criteria for electrical engineering curriculum achieved:

√	**Programs must demonstrate that graduates have a knowledge of:**
	1. Probability and statistics, including applications appropriate to the program name and objectives;
√	2. Mathematics through differential and integral calculus, basic sciences, computer science, and engineering sciences necessary to analyze and design complex electrical and electronic devices, software, and systems containing hardware and software components, as appropriate to program objectives.
√	3. (Electrical) Advanced mathematics, typically including differential equations, linear algebra, complex variables, and discrete mathematics.