EE 319K: Introduction to Embedded Systems
Sections - 16305, 16310, 16315, 16353

Andreas Gerstlauer

Spring 2011

General Information

Class Time	TTh 2:00-3:30 pm
Classroom	**ACA 1.104**
Office	ACES 6.118
Contact	gerstl@ece.utexas.edu
Pre-requisites	EE306 or BME303 with a grade of at least C- EE302 or BME102L with a grade of at least C-
Office hours	W 2:00 - 4:00 pm, or after class/by appointment
Website	UT Blackboard and class site: http://www.ece.utexas.edu/~gerstl/ee319k_s11
TAs	TBD

Catalog Description

Embedded systems; machine language execution; assembly and C language programming; local variables; input/output synchronization; analog to digital conversion, 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 7 and 8 are written in a combination of assembly and C. Labs 9 and 10 are written in a C.

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 Microcomputer Systems: Interfacing to the 9S12*, Cengage Publishing, 2009, ISBN-10: 049541137X / ISBN-13: 9780495411376

Reading assignments will be given from Patt's *Introduction to Computing Systems* (textbook 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:

- 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/

Equipment

Board: 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 $69.00 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 $69 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 or 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 ENS220 (checkout window) to pickup the kit. There will be specific scheduled hours for the pickup phase, which will be around Feb 10 to 17.

There will be a board demonstration lab between labs 2 and 3, the week of Feb 14-18, 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. We will be using the board in EE319K in Fall 2011, so if your board still works, you will have the opportunity to sell it. 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: You will need a voltmeter (one less than $20 will do), soldering iron (with solder), and a wire stripper. Since you will be making only 5 solder joints all semester, it is acceptable to borrow a soldering iron. However, all EE319K students will need their own voltmeter and wire strippers. The NI box you bought in EE302 can be used for as the voltmeter.

Kit handed out by TAs:

- 1 7406
- 6 LEDs (20 mA, red yellow, 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 one letter grade.

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 11th 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, but the help system does not run on Vista without a patch. Ask your TA how to get the TExaS help system to run under Vista.

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 can be downloaded at http://www.4pcb.com. 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)	30%
Test 1 (in class)	Thursday, February 24, 2:00-3:30pm	15%
Test 2 (in lab, ACA 1.102 & 1.106)	Thursday, March 24, 2:00-3:30pm	20%
Final Exam	Wednesday, May 11, 2:00-5:00pm	25%

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

Laboratory Policies: See the EE319K Laboratory Manual, available at http://users.ece.utexas.edu/~valvano/EE319K/EE319K_LabManualSp11.pdf

Homework Assignments: There will be approximately one homework or programming practice assignment each week. Homeworks will be administered over the web. You may work in groups of any size, but everyone enters a separate online solution. Other than breaking into the system and changing grades, you may get as much help from anyone on any way you please to complete the online homework:

1. Log into Quest Homework Service at: https://quest.cns.utexas.edu/student Click on the "Get Started" link and login with your EID. You should see EE319K as one of the choices (unique #16305).
2. Download student's instructions and homework.
3. Work on the homework offline.
4. Log in again any time (possibly multiple times) to submit or continue submitting answers until the due time (usually Monday 11:59pm). You can resubmit answers until correct, but you will get reduced or even negative scores for repeated mistakes.
5. Download solutions after due time.

Late homework submissions will not be accepted under any circumstances. The lowest homework score will be dropped.

Tentative Lecture Schedule

Week	Reading	Topic
1	Ch. 1, App. 1.1-1.8.2	Course Description, Embedded System Development, Flow Charts, Data Flow and Call Graphs, 9S12 Programming, Memory Map, Registers
	Ch. 2, 3	TExaS Simulator, Numbers, Hexadecimal
2	Ch. 2, 4	9S12 Architecture, Memory Allocation, Simple Addressing Modes, Execution, Subroutines
	Ch. 2, 3	Parallel Ports, Direction Registers, Logical and Shift Operations
3	Ch. 4.7, 5.7	Debugging in TExaS
	Ch. 3, 5	Arithmetic Operations, Condition Code Bits

Week	Reading	Topic
4	App. 2, Ch. 2.6	Demonstration of the board (bring your board to class): Switch Input and LED Output
	Ch. 5.1, 5.2, 6.11	If-then, Loops, Subroutines and the Stack, Debugging Dump
5	Ch. 6.1-6.3	Pointers and Indexed Addressing, Arrays, Strings
	Ch. 4.5, 5.7	Timers, Functional Debugging
6	Ch. 1 - 5	Review for Test 1
	Ch. 1 - 5	**Test 1 (closed book) in class, during regularly scheduled class time**
7	Ch. 6.8, 10.1	Finite State Machines (FSMs), Fixed-point numbers
	Ch. 7.1-7.5, 9.1	Local Variables, Stack Frames and Parameter Passing, I/O Synchronization
8	C Primer	C Programming, Structure of a C Program, Functions, Expressions
	C Primer	Variables in C, Structures and Data Types, FSMs in C
9	-	*Spring Break*
10	Ch. 8.4, 10.1, 10.5	LCD interface and Programming, Number Conversions Review for Test 2
	Ch. 5, 6	**Test 2 (closed book) in lab, during regularly scheduled class time (ACA 1.102 & 1.106)**
11	Ch. 9.2, 9.4, 9.6, 9.10	9S12 Interrupts and Interrupt Processing, Timer and Periodic (Output Compare) Interrupts
	C Primer	C/assembly interface, Metrowerks
12	Ch. 11.1, 11.4	Analog-to-Digital Conversion (ADC), Lab 7 Design Methods
	Ch. 10.2	Numerical Calculations, Mul/Div, Table Lookup, Interpolation
13	Ch. 8.1	Serial Communications Interface (SCI), Serial Interrupts, Lab 8
	Ch. 12.1-12.4	Producer-Consumer Problems, FIFO Queues
14	Ch. 11.2, 11.3	Digital-to-Analog Conversion (DAC), Sound Generation
	Ch. 8.7	Stepper Motors, TRobots Discussion
15	-	Advanced Embedded Systems Design, Systems Engineering
	-	Wrapup, Outlook
16	Ch. 1 - 13	**TRobots Competition**
	Ch. 1 - 13	Review for Final Exam
Finals	Ch. 1 – 13	**Final Exam, regularly scheduled time and place**

Tentative Lab Schedule

Date	Task
1/18	None
1/25	Go to ACA 1.106 for demonstration
2/1	**Lab1**: Digital lock I/O, parallel port, direction register and logical function, written in assembly (simulated, groups of two)
2/8	**Lab2**: LED and switch interface, written in assembly (simulated, groups of two)
2/15	Real board demonstration, bring your board to lab
2/22	**Lab3**: LED and switch interface, written in assembly (board, groups of two)
3/1	**Lab4**: Debugging techniques, one switch, one LED, written in assembly (simulated and board, groups of two)
3/8	**Lab 5**: DNA sequence detector using the real 9S12, written in assembly (simulated and board, groups of two)
3/15	*Spring Break*
3/24	**Test 2 (closed book), during regularly scheduled class time, in lab (ACA 1.102/106)**
3/29	**Lab 6**: LCD device driver, decimal fixed-point output, local variables, written in assembly (simulated and board, groups of two)
4/5	Go to ACA 1.106 for Metrowerks C demonstration
4/12	**Lab 7**: Real-time position monitor, ADC, interrupts, LCD, mixture of assembly and C (simulated and board, groups of two)
4/19	**Lab 8**: Distributed DAS, serial port interrupts, FIFO queue, mixture of assembly and C (simulated and board, groups of four)
4/26	**Lab 9a or 9b**: Digital piano or pacemaker using a 4-bit DAC, written in C (simulated and board, groups of two)
5/3	**Lab 10:** TRobot programming competition, written in C

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 assignments turned in by EE319K students through a plagiarism 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 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.

Add/Drop

The 12th class day is Wednesday, February 2, 2011. The deadline for dropping without possible academic penalty but requiring the Dean's approval is Monday, March 28, 2011.