Fall 2019

EE 382N-4: Advanced Micro-Controller Systems

Classroom: EER 1.518

Time: Mon/Wed 15:00 - 16:30

Course Overview:

This course focuses on the HW/SW architectures of “System-on-a-Chip (SoC) implementations. These SoC’s are composed of hardware and software components which must be seamlessly integrated together to produce working SOCs. These systems are becoming increasingly complex utilizing micro-architectural features from high performance computing platforms and from operating systems such as Linux and Android.

Topics covered include:

Hardware and software co-design of an SOC on a Dual-ARM core based FPGA
Linux drivers/handlers, kernel modules and interrupt handlers
RTOS, Middleware, SW Library development,
Flash based file systems
Embedded Linux debugging
Hardware accelerators, dataflow processing & fuzzy logic acceleration
Intelligent & cognitive sensor systems, sensor fusing
I/O subsystems
Networking-on-chip (NOC).

There will be 3 lab assignments, 2 exams and a class project.

The exams will be closed book, closed notes.

The class project focuses building a hardware accelerator that is coupled to a dual ARM core system via the AXI bus using the ZYNQ-7020 programmable SOC on the ZEDBOARD.

The Lab assignments focus on learning how to design, synthesize, debug and test various components of the ZYNQ SOC.

Course prerequisites:

- EE 360N undergraduate Computer Architecture, or an equivalent undergraduate computer architecture class.
- EE445L or EE445M undergraduate Embedded Systems Labs, or similar courses.
- Basic assembly language programming skills (ARM)
- VERILOG programming skills
- Basic high level language programming skills such as C or C++
- Some familiarity with Linux programming

Instructor:

Mark McDermott
Office: EER 5.826, Phone: (512) 471-3253
Office hours: TBD

E-Mail: *mcdermot at ece.utexas.edu*

TA:

TBD
Office: EER 1.810
Office Hours: TDB
E-Mail: *m at utexas.edu*

Course outline and schedule:

Week	Date	Lecture Topic	Lecture Notes	LAB Assignments
1	Aug 28	Course Overview Xilinx Development Environment Lab #1 Overview	Lecture 1 Lecture 2	LAB #1 Due Sep 25th
2	Sep 4			
3	Sep 9	Xilinx Zynq-7000 Architecture	Lecture 3	
	Sep 11			
4	Sep 16	ARM SW Programming	Lecture 4	
	Sep 18	Embedded Linux Lab # 2 Overview	Lecture 5	
5	Sep 23			
	Sep 25	Linux Device Drivers	Lecture 6	LAB #2 Due: Oct 18th
6	Sep 30	Interrupts, Interrupt Handlers & Signals	Lecture 7	
	Oct 2			
7	Oct 7	Boot Loaders & Device Tree Blobs	Lecture 8	
	Oct 9	Debugging Embedded Linux	Lecture 9	
8	Oct 14	ARM Instruction Set Architecture	Lecture 10	
	Oct 16	**Exam #1 (Closed Book & Notes)**		
9	Oct 21	ARM Processor Micro-Architecture Lab #3 Overview	Lecture 11	LAB #3 Due: Nov 15th
	Oct 23			
10	Oct 28	Accelerators & Co-Processors	Lecture 12	
	Oct 30	Linux File Systems	Lecture 13	
11	Nov 4	Sensor Systems	Lecture 14	
	Nov 6			
12	Nov 11	Dataflow Processing	Lecture 15	
	Nov 13			
13	Nov 18	I/O Subsystems	Lecture 16	
	Nov 20	Software Optimization for Power Reduction	Lecture 17	

14	Nov 25	TBD	Lecture 18	
	Nov 28	Thanksgiving		
15	Dec 2	Exam #2 (Closed Book & Notes)		
	Dec 4	**PRELIMINARY PROJECT DESIGN REVIEWS**		
16	Dec 9	**NOTE: Final Project Reviews will be during Exam Period -- No exceptions**		

Grading:	
Labs	30%
Exam #1	15%
Exam #2	15%
Project	40%

Late Submission Penalties:

Penalty for late submission of Labs and Class Project: 25% per working day (Maximum: 100%)

Lab Facilities:

EER 1.810

Useful Tutorials:

DTB & BIT File Generation Flow
Processor Micro-Architecture
Using the Zedboard

Digilent Zedboard Tutorials and Documentation

Embedded Linux Development Guide
Getting Started with Embedded Linux - ZedBoard
Embedded Linux Hands-on Tutorial - ZedBoard
Creating a Custom IP core using the IP Integrator
Getting Started with Zynq
Using Pmod IPs
Zedboard DMA Audio Demo
Zedboard LED Demo
Zedboard OLED Demo
Zedboard Programming Guide in SDK

Xilinx Zynq-7000 Tutorials and Documentation

ZYNQ Video Tutorials
ZYNQ Documentation
Zynq-7000 All Programmable SoC Technical Reference Manual
AXI Infrastructure Intellectual Property
Debugging U-Boot with SDK

Creating an AXI Peripheral
Using Xilinx SDK
Repository of useful Zynq Info

Vivado Tutorials and Documentation

Vivado Video Tutorials
Vivado Design Suite User Guide: Getting Started (UG910)
Vivado Design Suite Tutorial (UG940)
Vivado Design User Guide: Design Flows Overview (UG892)
Vivado Design Suite Tutorial: Design Flows Overview (UG888)
Vivado Design Suite Tutorial: Programming and Debugging (UG936)
Vivado Design Suite Tutorial: High-Level Synthesis (UG871)
Vivado Design Suite User Guide: High-Level Synthesis (UG902)
Vivado Design Suite User Guide: Synthesis (UG901)
Vivado Design Suite User Guide: Implementation (UG904)
Introduction to FPGA Design with Vivado High-Level Synthesis (UG998)
Vivado Design Suite User Guide: Using Tcl Scripting (UG894)
Vivado Design Suite Tcl Command Reference Guide (UG835)
Vivado Design Suite User Guide: Designing with IP (UG896)
Vivado Design Suite User Guide: Designing IP Subsystems Using IP Integrator (UG994)
Vivado Design Suite User Guide: Logic Simulation (UG900)
Vivado Design Suite User Guide: Using Constraints (UG903)
Vivado Design Suite User Guide: Design Analysis and Closure Techniques (UG906)
Vivado Design Suite User Guide: Programming and Debugging (UG908)
Vivado Design Suite User Guide: System-Level Design Entry (UG895)
Vivado Design Suite Properties Reference Guide (UG912)
Vivado Design Suite User Guide: I/O and Clock Planning (UG899)
ISE to Vivado Design Suite Migration Guide (UG911)
Vivado Design Suite User Guide: Model-Based DSP Design Using System Generator (UG897)
Vivado Design Suite User Guide: Power Analysis and Optimization (UG907)

Xilinx Wiki

Main page
Linux
U-Boot
Technical Articles
Installing Ubuntu
Zynq Base Targeted Reference Design (TRD)

Forums

Xilinx
Avnet
Digilent

Useful Websites

Zynq Design from Scratch
Zynq Development
Free Electrons
Zynq Training
An FPGA Tutorial using the ZedBoard

Reference Books

Sreekrishnan Venkateswaran Essential Linux Device Drivers (Prentice Hall Open Source Software Development Series)
Karim Yaghmour Building Embedded Linux Systems
Jonathon Corbet Linux Device Drivers
Richard Zurawski Embedded Systems Handbook: Networked Embedded Systems
Richard Zurawski Embedded Systems Handbook: Embedded Systems Design and Verification
L.H. Crockett, R.A. Elliot, M.A. Enderwitz, and R.W. Stewart, The Zynq Book: Embedded Processing with the ARM Cortex-A9 on the Xilinx Zynq-7000 All Programmable SoC, PDF copy available for free at http://www.zynqbook.com
L.H. Crockett, R.A. Elliot, M.A. Enderwitz, and R.W. Stewart, The Zynq Book Tutorials, available for free at http://www.zynqbook.com/download-tuts.html

Academic dishonesty:

Oral discussion of homework problems is encouraged. However, be sure to submit your own individual and independent solution. Labs and final projects can be done in teams. Collaboration on projects is encouraged. Copying of any part of a homework/lab solution or project report without explicit reference to its source is plagiarism and considered cheating.

"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 EE445L 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 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.

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 Canvas and Class Web Site

This course uses the class web page and Canvas 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 Canvas 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 Canvas 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.

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 Services for Students with Disabilities (SSD) at 471-6259, http://ddce.utexas.edu/disability/.

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.

Classroom Evacuation and Emergency Preparedness

Every member of the university community must take appropriate and deliberate action when an emergency strikes a building, a portion of the campus, or entire campus community. Emergency preparedness means we are all ready to act for our own safety and the safety of others during a crisis. It takes an effort by all of us to create and sustain an effective emergency preparedness system. Your support is important to achieving the best possible outcomes during a crisis event. As a University faculty member, you are responsible for pointing out your classrooms building emergency evacuation routes and for reviewing emergency procedures with students at the beginning of each semester. This review should include a mention of the monthly emergency communications test (every first Wednesday at 11:50 a.m.) and the list of communications channels the university uses during emergencies. It should also include a review of the attached document outlining emergency terms (e.g., the difference between

“shelter-inplace” and “lockdown”) and instructions for faculty and students to follow during emergencies. As a matter of convenience, we recommend including this information in your syllabus along with the phone number for the Behavior Concerns Advice Line (BCAL: **512-232-5050**). This is the number to call if you have concerns regarding the attitude or actions of students, staff, or other faculty. Finally, at the end of your emergency preparedness review, request that students requiring assistance in evacuation inform you in writing of their needs during the first week of class. This information must then be provided to the Fire Prevention Services office by fax (**512-232-2759)**, with "Attn. Mr. Roosevelt Easley" written in the subject line. Thank you in advance for taking the time to ensure the safety of your classroom. I assure you this small effort can yield much greater rewards should the unthinkable happen. If you would like more information regarding emergency preparedness, visit http://www.utexas.edu/safety/preparedness/.

Emergency Communications

Emergencies may range from inclement weather, to building evacuations, to campus closures, and the university has a variety of tools to communicate with the public in the event of these and other possible emergencies. Depending on the type of emergency, we may use some or all of the following tools to communicate with faculty, staff and students:

Siren System

This system is tested around noon on the first Wednesday of every month, and delivers a siren warning and public address in the event of certain outdoor emergencies. Read more about the siren system.

Emergency Web Site

You may want to bookmark the emergency Web site because it is updated with information during actual emergencies or campus closures.

Local Press and Social Media

University Communications staff send emergency information to the press and update social media with public safety messages. Because of the transient nature of our population, the university depends a great deal on the press and social media to keep students, faculty, and staff informed during campus emergencies.

Pager System

Our campus first responders, resident advisors, and some building managers are part of the AWACS paging system. The pagers send text messages about emergencies on campus and alert city responders (APD, AFD, EMS, Office of Emergency Management, etc.) to campus crisis situations.

Text Alerts

The university collects cell phone numbers from members of the campus community for emergency text messages. Sign up for campus text alerts online.

University Group E-mail

During emergencies, UT Safety Alert sends an “urgent” group e-mail to every student, faculty and staff member. The e-mail directs individuals to the emergency Web site for additional information and instruction.

Public Safety Patrol Car Announcements

UTPD patrol cars are equipped with PA systems, which officers can use to provide instructions to pedestrians during emergencies.

University Emergency Information Line: 512-232-9999

Students, faculty, and staff can call this main number for information about campus closures. The implementation of each tool described above is assigned to an individual who has at least two backups who can also carry out the communications task. Individuals with electronic communication tools assigned to them have remote access (from their homes, etc.) to those tools. The police department and the associate vice president for Campus Safety and Security are typically the ones who deliver emergency information to university administration. Upon considering this information, administration develops the messages and activates

campus-wide communications. The only exceptions to this are the sirens and pager system, which are activated directly by UTPD in extremely urgent situations where immediate action is required.

Last updated: Saturday, August 03, 2019