System-on-Chip (SoC) Design

ECE382M.20, Unique: 19190, Fall 2026

Lectures: TTh 12:30-2:00pm, ECJ 1.214

Instructor: Andreas Gerstlauer <gerstl@ece.utexas.edu>**,** Office hours: TBD

Teaching Assistants: James Boyle <james.boyle@utexas.edu>, Office hours: TBD
Daniel Peralta Velazquez <dperaltav@utexas.edu>, Office hours: TBD

Class website: Canvas and http://users.ece.utexas.edu/~gerstl/ece382m_f26/

Description

With technological advances that allow us to integrate complete multi-processor systems on a single die, Systems-on-Chip (SoCs) are at the core of most embedded computing and consumer devices, such as smart phones and automotive, aerospace, robotics or medical electronics. This course will provide an understanding of the concepts, issues, and process of designing highly integrated SoCs following systematic hardware/software co-design & co-verification principles. Specifically, the class project involves taking public domain C++ code for a large language model (LLM) and mapping it to an ARM-based virtual and FPGA prototyping platform using state-of-the-art synthesis and verification tools and design flows.

Goals

This course is designed for students to learn and be able to:

- Model and specify embedded systems at high levels of abstraction.
- Analyze functional and nonfunctional requirements of the system early in the design process to support design decisions.
- Analyze hardware/software tradeoffs, algorithms, and architectures to optimize the system based on requirements and implementation constraints.
- Analyze tradeoffs and explore architecture and mico-architecture design spaces to develop and synthesize custom hardware accelerators.
- Understand hardware, software, and interface design and synthesis.
- Use co-simulation to validate system functionality.
- Describe examples of applications and systems developed using a co-design approach.
- Appreciate issues in system-on-chip design associated with co-design, such as intellectual property, reuse, and verification.

Course Description and Course Topics

Methodologies and tools for System-on-Chip (SoC) and hardware/software co-design and co-verification:

- Hardware/software co-design: partitioning, real-time scheduling, hardware acceleration;
- Virtual prototyping: electronic system-level languages and hardware/software co-simulation;
- High-level synthesis: allocation, scheduling and binding algorithms for C-to-RTL synthesis;
- SoC integration: SoC communication architectures, IP interfacing, verification and test;
- FPGA prototyping of hardware/software systems.

Prerequisites

- Working knowledge of C/C++, including software development and debugging (e.g. ECE412 Software Design and Implementation, or equivalent);
- Digital hardware design, hardware description languages (HDLs) and hardware synthesis (e.g. ECE460M Digital System Design using Verilog, or equivalent);
- Embedded real-time system design and hardware/software interfacing (e.g. ECE445L Embedded System Design Lab, ECE445M Embedded & Real-time Operating Systems, or equivalent);
- Basic computer architecture and computer system design knowledge (e.g. ECE460N Computer Architecture, or equivalent);
- It is helpful but not required to have some basic knowledge of advanced linear algebra and machine/deep learning algorithms.

Textbooks

No required textbook. Optional textbooks:

- P. Marwedel, *Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things*, Fourth Edition, Springer, 2021.
- D. C. Black, J. Donovan, B. Bunton, A. Keist, *SystemC: From the Ground Up*, Second Edition, Springer, 2010.
- S. Pasricha, N. Dutt, *On-Chip Communication Architectures (System on Chip Interconnect)*, Morgan Kaufman, 2008.
- G. De Micheli, *Synthesis and Optimization of Digital Circuits*, McGraw-Hill, 1994.

Grading

Homework:	10%
Exam:	30%
Labs:	20%
Project:	40%

Late penalty: 20% per day (24 hours).

Oral discussion of homework problems is encouraged but solutions have to be submitted individually and independently. Labs and the final project will be done in teams. Copying of any part of a homework, lab or project solution without explicit reference to its source, including use of artificial intelligence (AI), which must be disclosed in lab and homework reports, is plagiarism and considered cheating.

Tentative Course Outline and Schedule

Week	Lecture Topic
1 (Aug 25/27)	Class and Project Overview
2 (Sep 1/3)	System-Level Design Methodology
3 (Sep 8/10)	HW/SW Co-Design and Accelerated System Design
4 (Sep 15/17)	Application Partitioning and Scheduling
5 (Sep 22/24)	System Modeling in SystemC
6 (Sep 29/Oct 1)	C-to-RTL High-Level Synthesis (HLS)
7 (Oct 6/8)	HLS Operation Scheduling and Resource Binding
8 (Oct 13/15)	HLS Design Space Exploration
9 (Oct 20/22)	System Integration and Accelerator Interfacing

10 (Oct 27/29)	SoC Communication Architectures
11 (Nov 3/5)	Review, **Exam**
12 (Nov 10/12)	Emulation and FPGA Prototyping
13 (Nov 17/19)	SoC Verification and Test
14 (Nov 24/26)	*Fall Break / Thanksgiving*
15 (Dec 1/3)	**Project Design Reviews**
Finals (Dec 8)	**Project Presentations**

Sharing of Course Materials

Sharing of course materials is prohibited. No materials used in this class, including, but not limited to, lecture hand-outs, videos, assessments (quizzes, exams, papers, projects, homework assignments), in-class materials, review sheets, and additional problem sets, may be shared online or with anyone outside of the class unless you have my explicit, written permission. Unauthorized sharing of materials promotes cheating. It is a violation of the University's Student Honor Code and an act of academic dishonesty. The University is well aware of the sites used for sharing materials, and any materials found online that are associated with you, or any suspected unauthorized sharing of materials, will be reported to Student Conduct and Academic Integrity in the Office of the Dean of Students. These reports can result in initiation of the student conduct process and include charge(s) for academic misconduct, potentially resulting in sanctions, including a grade impact.

Class Recordings

HOP 2-9970 prohibits students from recording class instruction (audio or video) unless a student obtains the instructor's permission or Disability & Access has approved audio recording as an accommodation. Class recordings provided by the instructor are reserved only for students in this class for educational purposes and are protected under FERPA. The recordings should not be shared outside the class in any form. Violation of this restriction by a student could lead to Student Misconduct proceedings.

Artificial Intelligence (AI) Policy

In accordance with the University's Institutional Rules on Student Services and Activities, Chapter 11, students accept the responsibility to always uphold academic integrity and an honor code reflective of a scholarly community devoted to academic and personal success. All members of the University community are fully accountable and responsible for any output they produce as part of academic work. They are also responsible for following the guidance specified in the Texas Statement on Academic Integrity and avoiding prohibited uses of generative AI tools outlined in the Information Security Office's guidance on Acceptable Use of Generative AI Tools.

Generative AI use is permitted for academic work in this course, provided that students 1) use AI responsibly, 2) practice discernment and conduct meaningful human review of any output generated by AI, and 3) properly disclose use according to the disclosure policies in this syllabus. Using generative AI tools

without proper adherence to the disclosure policy in this course, even when AI use is permitted, may constitute academic misconduct under UT Austin's Institutional Rules and may be referred to Student Conduct and Academic Integrity in the Office of the Dean of Students for resolution.

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. By UT Austin policy, you must notify your instructors of your pending absence for a religious holy day as far in advance as possible of the date of observance. For questions regarding religious accommodations, please contact the Office of the Dean of Students.

University Policy and Resources

For a list of other important university policies and helpful resources that you may need as you engage with and navigate your courses and the university, see the University Policies and Resources Students Canvas page. The page includes the language of the University Honor Code, Title IX legal requirements for Texas employees, and information about how to receive support through the office of Disability & Access.