2026-Spr-E_E-234-PULLM-1-01-02975-Microprocessor Systems

Jump to Today

EE234: Microprocessor Systems

EE234 is a four-credit course that introduces students to the principles and practices of ARM-based microprocessor programming and system design through a hands-on, project-based learning model. The course includes three weekly lectures and a dedicated three-hour lab session, balancing theoretical instruction with practical experience.

At the core of the course are seven self-paced, self-guided design projects, each offering the opportunity to develop and implement digital systems using industry-standard tools and hardware. These projects form the foundation of the course and are designed to cultivate independent problem-solving, technical proficiency, and engineering insight.

Students will use AMD's Vitis software design tool to design, implement and debug several embedded software projects. Vitis is a powerful and widely used software development tool that runs on both Windows and Linux, and is freely available from AMD's website. For hardware implementation and verification, students will work with the Real Digital Blackboard, a versatile circuit development platform featuring an ARM processor and a wide array of I/O devices. The Blackboard brings digital designs to life, enabling real-world testing and interaction. Each student is required to purchase their own board, available in class, through the IEEE office, or directly from Real Digital's website.

Because students own their software and hardware tools, they have full flexibility to complete project work when and where it suits them. Each project includes detailed design requirements that are supported by readings and lectures that provide the foundational knowledge, design examples, and tool guidance needed for successful completion.

All course materials are hosted on Canvas and on the Real Digital website under the Microprocessor course section. While there is no required textbook, students looking for supplemental resources are encouraged to consult the "Arm Assembly Language Programming & Architecture Second Edition" book that is available for purchase from Amazon at **www.amazon.com/ARM-Assembly-Language-Programming-Architecture/dp/1970054867/ (http://www.amazon.com/ARM-Assembly-Language-Programming-Architecture/dp/1970054867/)** . An on-line version of the first edition is also available here: **https://eecs.wsu.edu/~ccole/ee234/uploads/book.pdf (https://eecs.wsu.edu/~ccole/ee234/uploads/book.pdf)** . Further, the class will use several on-line resources for additional information.

Lecture sessions track closely with the posted course content but often extend far beyond the basics. In addition to exploring core concepts, lectures frequently respond to student questions, leading to deeper

discussion, practical design strategies, and context for real-world applications. Lectures also regularly include exam preparation, design tips, and project-specific advice. Attendance is not formally recorded but is considered essential for success.

Lab sessions provide time to collaborate with TAs and peers, ask in-depth questions, troubleshoot issues, and gain guided experience with design tools and engineering methods. Although attendance is not required for all lab periods, it is mandatory when demonstrating completed project work to a TA.

Collaboration and discussion with classmates are encouraged throughout the course, but each student must independently write their own code and submit original work. By signing a project submission form, you affirm that the work is your own.

All completed projects must be demonstrated during a one-on-one, in-person lab session with a TA. Students are responsible for bringing a printed submission form, which outlines the project's design requirements and includes space for scoring. During the demo, the TA will:

- Ask you to demonstrate each required feature.
- Review your source code for consistency with your demo.
- Ask questions to evaluate your understanding of your design and code.

The TA's primary role is instructional - helping you understand the tools, concepts, and design tasks. Their secondary role is to evaluate your work. For each requirement, the TA will assign a score from 0 to 4 based on completeness, correctness, and your ability to explain the design. If features are incomplete or unclear, you're encouraged to revise and resubmit the project. If you choose not to make revisions, your current work will be scored as-is.

Project scores are calculated by multiplying each requirement score by its respective weighting factor (shown on the submission form). The resulting weighted total is entered into both the form and Canvas. Signed submission forms must be placed in the secure drop box in the lab. Early submissions receive a bonus, and late submissions are accepted up to two weeks late with a penalty.

Final grades are determined by adding the project scores (54%) and test scores (46%). Baseline grades will be assigned using percentage breaks (e.g., 100% - 90%+ = A; 80%-90% = B; etc.), with all scores normalized to the highest earned score in the class. Next, individual student scores will be determined using a grading curve, with an average grade of C+, and with the upper 20% of scores receiving an A. The assigned grade will be the higher of the baseline grade or curve grade.

The tables below provide some guidance and insight on scoring, and some reasons why a given score may have been awarded.

Project Scoring

Questions & Answers Demo	Clear knowledge of design Confidently discusses all blocks/signals Can justify design choices Knows all source files well Can discuss tradeoffs	Mostly familiar with design Knows some blocks/signals well Unsure of design approach Knows most source files Can't discuss tradeoffs	Unclear on multiple aspects Unaware of many functions Unable to justify design choices Unaware of tradeoffs Unfamiliar with source files
Exemplary All requirements met Concise, well-organized code Good comments Good partitioning (if relevant) Solid design approach Could serve as class example	**4** *Exemplary!!*	**2-3** *Borrowed too much*	**1** *Not student's original work*
Good All requirements met Code less concise/organized Few/poor comments	**3** *Nice work*	**2-3** *Decent effort, but incomplete*	**1** *Not student's original work*

Poor partitioning (if relevant) Less efficient design approach			
Room for Improvement Some/most requirements met Code poorly organized No comments Poor partitioning (if relevant) Disjointed design approach	**2-3** *Didn't put in enough time*	**1-2** *Didn't put in enough time*	**1** *Not student's original work*
Minor Effort Few/incomplete features Poor design and organization Unstructured design approach Mashup of unrelated circuits	**2** *Minor effort, short on time*	**1** *Minor effort, short on motivation*	**0**
No Effort	**0**	**0**	**0**

Course Goals

To become a successful digital design engineer, a large body of knowledge and wide array of design skills must be mastered. This course moves through the material at a brisk and steady pace, and every

new lecture and design project builds on the previous one. The lectures and design projects are tightly coupled, and students should plan on attending all lectures and completing all design projects on time. Most projects require just a few hours each week, but missing any lectures or projects can significantly increase the amount of time required, and make it difficult to catch up.

This course presents the design skills and theoretical knowledge needed to design, implement, and debug embedded software systems on an ARM processor. Biweekly lab assignments provide students with the opportunity to use state-of-the-art technologies to design and implement software systems that are relevant to industry. These design projects strengthen and support the concepts covered in lecture, and they allow students to develop highly relevant skills that are prevalent in industry today.

Course Structure

Approximately 38 lectures will be delivered during the class, and they include additional examples and explanations beyond the written materials. Lecture attendance is strongly correlated with higher scores and success rates, and is highly encouraged.

One midterm and a final exam will be given during the semester. Exams are closed note and closed book, and exam problems are based on homework problems and examples presented in class.

Lab projects are due at the close of lab during the “due” week shown. For full credit, you must complete all design activities in the requirement section of each project, and demonstrate your solutions to a TA during a lab session. You may optionally complete any or all of the design challenges for extra credit on any project. (Note: You are highly encouraged to complete the challenges for every project). You are welcome to attend any lab session to demo your project.

Lab:

Times:

Tuesday 7:45-10:35
Thursday 7:45-10:35

TAs:

Academic Integrity

You are encouraged to familiarize yourself with the student guidelines, and your rights and responsibilities, as described in the **Washington state University handbookLinks to an external site. (https://www.handbook.wsu.edu/handbook-home/)** . Within the handbook, you will find information on WSU’s general academic integrity policies.

In this class, you are encouraged to work with classmates, friends, family, the course instructor, and any other relevant resource to complete homework and design project assignments. However, any work you

submit for credit must be your own, original work – team submissions, or copies of work you did not complete, are not allowed and will not be accepted. The TAs are instructed to look for areas where design work may have been copied, and to ask more detailed questions if they suspect non-original work is being submitted for credit.

School of Electrical Engineering and Computer Science Policy

EECS faculty who observe instances of academic dishonesty, or who receive credible/verifiable reports from TAs, will have the full range of options available to them that are outlined in the Student Handbook. All instances of academic dishonesty will be reported to either the Graduate or Undergraduate Program Coordinators, whichever is appropriate. The Coordinator will maintain in-house records of academic dishonesty and will forward all information to the **Office of Student Conduct (https://communitystandards.wsu.edu/)** . This information will also be shared with the EECS Instruction Committee. For undergraduates interested in pursuing a degree in the School, the Instruction Committee will weigh the evidence and make a decision concerning the student's certification status. Students who commit acts of academic dishonesty in an EE or CptS course who have not been certified may be ineligible for certification, while certified undergraduates may be decertified. If the student chooses to appeal the decision, a panel will be appointed by the Associate Director consisting of three faculty members from the School who were not previously involved in the decision process and two students from the appropriate degree program. The panel will review the evidence and hear any additional arguments the accused student may wish to present. A simple majority vote by the panel shall uphold or overturn the Instruction Committee's decision. The Director maintains ultimate decision authority and may overrule the decision of the Instruction Committee or the appeal panel.

One instance of academic dishonesty on the part of a graduate student may result in termination of support. The decision will be made by the Instruction Committee subject to the recommendation of the Graduate Studies Committee and input from the student's faculty advisor. An appeal panel will be appointed by the Associate Director should the student wish to appeal the decision of the Committee. As with the undergraduate case, the panel will consist of three faculty members from the School who were not previously involved in the decision process and two students from the respective degree program. A simple majority vote by the panel shall uphold or overturn the Committee's decision. Again, the Director maintains ultimate decision authority.

Faculty are encouraged to explain these consequences of academic dishonesty at the beginning of each semester and explain unambiguously what constitutes academic dishonesty in each course. However, ignorance of these consequences or of the definition of academic dishonesty in a particular class does not serve as an excuse. Students who observe acts of academic dishonesty may report their observations to the course instructor or to the Associate Director of the School.

Course Summary:

Date	Details	Due
Thu Jan 29, 2026	Project 1 Challenges (https://wsu.instructure.com/courses/1845766/assignments/11063866)	due by 8pm
	Project 1 Requirements: Introduction to Vitis and the ARM assembly language (https://wsu.instructure.com/courses/1845766/assignments/11063867)	due by 8pm
Thu Feb 12, 2026	Project 2 Challenges (https://wsu.instructure.com/courses/1845766/assignments/11063874)	due by 11:59pm
	Project 2 Requitements: Controlling LED Brightness with PWM (https://wsu.instructure.com/courses/1845766/assignments/11063875)	due by 11:59pm
Thu Feb 26, 2026	Project 3 Challenges (https://wsu.instructure.com/courses/1845766/assignments/11063876)	due by 11:59pm
	Project 3 Requirements: Building a Stopwatch Using ZYNQ's Triple Timer Counter (TTC) Module (https://wsu.instructure.com/courses/1845766/assignments/11063877)	due by 11:59pm
Thu Mar 5, 2026	Proejct 4 Challenges (https://wsu.instructure.com/courses/1845766/assignments/11063865)	due by 11:59pm
	Project 4 Requirements: UARTs (https://wsu.instructure.com/courses/1845766/assignments/11063878)	due by 11:59pm
Fri Mar 6, 2026	Midterm Assessment (https://wsu.instructure.com/courses/1845766/assignments/11063864)	due by 9am
Thu Mar 26, 2026	Project 5 Challenges (https://wsu.instructure.com/courses/1845766/assignments/11063879)	due by 11:59pm
	Project 5 Requirements: Configuring Interrupts on Zynq (https://wsu.instructure.com/courses/1845766/assignments/11063881)	due by 11:59pm

Date	Details	Due
Thu Apr 9, 2026	Project 6 Challenges (https://wsu.instructure.com/courses/1845766/assignments/11063882)	due by 11:59pm
	Project 6 Requirements: Using the SPI and I2C bus (https://wsu.instructure.com/courses/1845766/assignments/11063883)	due by 11:59pm
Thu Apr 23, 2026	Project 7 Challenges (https://wsu.instructure.com/courses/1845766/assignments/11063885)	due by 11:59pm
	Project 7 Requirements: Working with Servomotors (https://wsu.instructure.com/courses/1845766/assignments/11063886)	due by 11:59pm
Tue May 5, 2026	Final Assessment (https://wsu.instructure.com/courses/1845766/assignments/11063863)	due by 7am