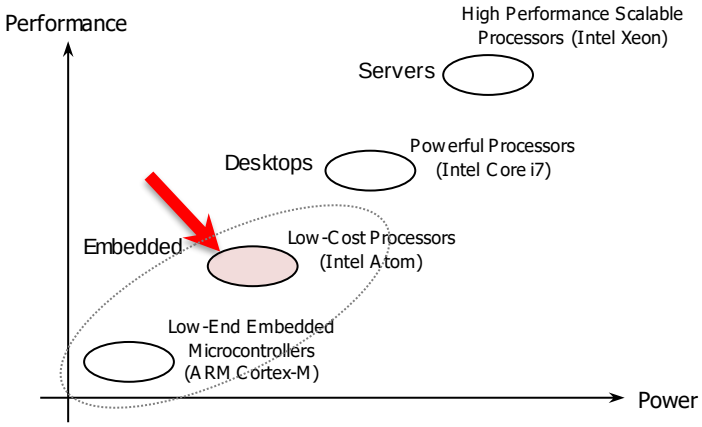
Course Information

INSTRUCTOR	Daniel Llamocca
CONTACT INFO	email: llamocca@oakland.edu
OFFICE HOURS	Tuesday 3:00 to 5:00 pm @ Room EC-438, or by appointment. Virtual Office hours also available (on Moodle → Virtual Office hours via Zoom)
LECTURES	CRN 43672, 44796: Tuesday/Thursday 7:30 – 9:17 pm @ Room MSC-185 (Mathematics and Science Center)
LABORATORY	See schedule

COURSE CATALOG DESCRIPTION

ECE 4772/5772 – High-performance Embedded Programming (4 credits)
Real-time embedded system programming, analysis, and optimization using the Intel Atom® processor. Topics covered include real-time programming, multi-threaded systems, multi-core software development, as well as optimization of processor utilization, speed and memory requirements. Offered: Fall. Prerequisite (for ECE4772 only): (ECE 3720) and major standing.



COURSE MATERIAL

- The course material will be hosted on Moodle (moodle.oakland.edu). Grades will be periodically posted via this system.
- As a backup resource, the material will also be posted at: www.secs.oakland.edu/~llamocca/Fall2024_ece4772.html
- High-performance Embedded Programming Tutorial: Available at: www.secs.oakland.edu/~llamocca/emb_intel.html

TEXTBOOK

- There is no required textbook. Students are encouraged to use the extra references.

EXTRA REFERENCES:

- Lori Matassa, Max Domeika, *Break Away with Intel Atom Processors: A Guide to Architecture Migration*, Intel Press, 2012.
- Max Domeika, *Software Development for Embedded Multi-core Systems*, Newnes, 2008.
- David R. Butenhof, *Programming with POSIX Threads*, Addison-Wesley Professional, 1997.
- M. McCool, A. Robison, J. Reinders, *Structured Parallel Programming: Patterns for Efficient Computation*, M. Kaufmann, 2012
- M. Voss, R. Asenjo, J. Reinders, *Pro TBB: C++ Parallel Programming with Threading Building Blocks*, Apress, 2019 (free)
- Intel® Threading Building Blocks Documentation.

COURSE OBJECTIVES

1. Describe the generalized architecture of the Intel Atom® microprocessor. (7)
2. Implement software applications with C/C++ on Ubuntu Linux. (1)
3. Implement real-time embedded applications on Ubuntu Linux. (1, 6)
4. Design and implement multi-threaded software applications (1, 6)
5. Design and implement multi-core applications to enable parallelism and pipelining. (1, 6)
6. Design applications that utilize the computer resources in a scalable fashion (1, 4, 6)
7. Work in a team environment to design a real-time multi-threaded embedded application and communicate the results in a written report and an oral presentation (1, 2, 3, 5, 6)

ABET Course Outcomes:

1	2	3	4	5	6	7

LABORATORY MATERIALS

- **Hardware (students can borrow it)**:
  - ✓ Terasic DE2i-150-FPGA Development Kit. It includes an Intel Atom® N2600.
    https://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=11&No=529

- **Software**:
  - ✓ Ubuntu Linux 12.04.4 (or Ubuntu 14) distribution. This is pre-installed in the boards.
  - ✓ MATLAB®. OU students have access to it.

GRADING SCHEME

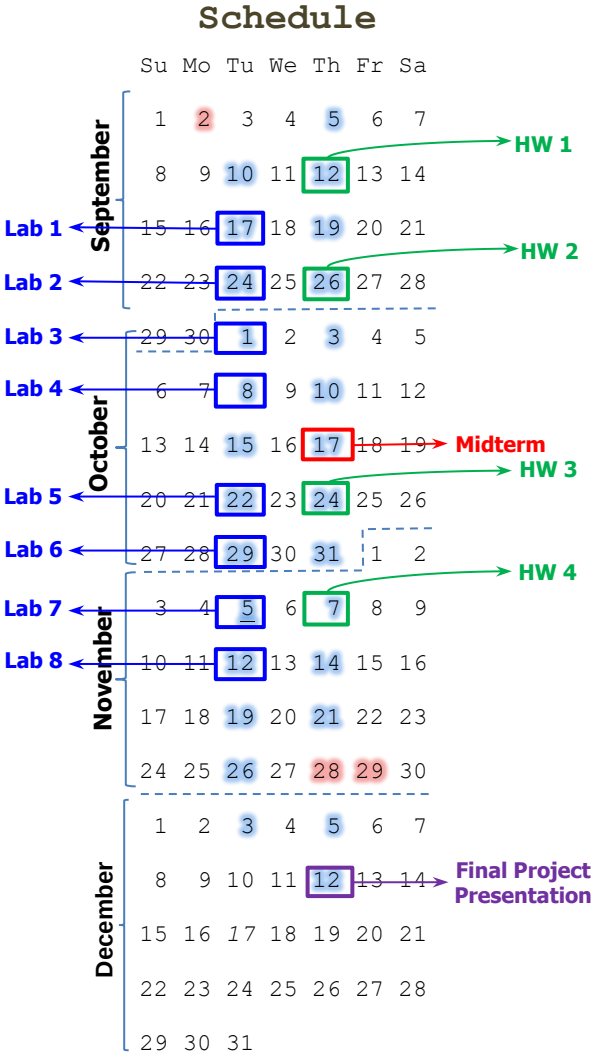
Homeworks:	20%	**Midterm Exam**: 15% (October 16th, 5:30 pm – October 17th, 11:59 pm)	
Laboratory:	35%	**Final Project:**	30% (December 12th, 7:00 – 10:00 pm).

- **Homeworks (4):** Homework assignments are meant to strengthen your conceptual understanding of the topics. Completing homework assignments is a key component of this course as it will help students master the course material and prepare them for the examinations.
  Homeworks will be posted according to the schedule (green rectangles). Depending on the assignment, students have 1 or 2 weeks to turn in the completed assignments in class. Late submissions are NOT accepted.

- **Midterm Exam:** Open-book, online exam. Students are not allowed to take the exams either before or after the exam date. Make-up exams are given *only* under extreme circumstances (e.g.: medical emergency, jury duty).
  ** You will have 24 hours to complete the exam.*

- **Laboratory (8):** This important component of the class will reinforce your understanding of the topics. There will be eight (8) lab experiments throughout the semester. Students have 1 week to complete them and have them checked off by the instructor.
  ** There is a late policy on laboratory assignments*

- **Final Project:** Students will work in groups on a Final Project. Each group will prepare an oral presentation and submit a final report. Presentations will take place on Dec. 12th.



GRADE ASSIGNMENT:

96-100	A	4.0
90-95	A-	3.7
85-89	B+	3.3
80-84	B	3.0
72-79	B-	2.7
66-71	C+	2.3
60-65	C	2.0
56-59	C-	1.7
53-55	D+	1.3
50-52	D	1.0
49 and below	F	0.0

OUTLINE OF TOPICS

Embedded Multi-Core Systems	▪ Multi-core Processors and Embedded Systems ▪ Intel® Atom™ processor. Terasic DE2i-150 FPGA Dev. Kit
C/C++ Language Programming Fundamentals	▪ C Programming: Basics, pointers, functions, function pointers, structures ▪ C++ Programming: Objects, Function and Operator Overloading, functors
Multi-threaded applications	▪ POSIX Threads: Declaration. ▪ Thread synchronization: mutexes, condition variables
Multi-core Applications	▪ Threading Building Blocks: *parallel_for, parallel_invoke, parallel_reduce, parallel_scan* ▪ Pipelining: *parallel_pipeline*
Real-Time Programming	▪ Signal Handling: Alarm (software interrupt) and Keyboard (User Interrupt) ▪ Real-Time Clock
Design & Optimization of Embedded Real-Time Systems	▪ Power and Performance Analysis Tools ▪ Power Optimization
Applications	▪ Convolutional Neural Network ▪ Median Filter ▪ Adaptive Beamforming ▪ Cylinder Pressure Estimation

TUTORIAL: HIGH-PERFORMANCE EMBEDDED PROGRAMMING WITH THE INTEL® ATOM™ PLATFORM

Topics		#
Getting Started with the Hardware and Software Platform		1
C/C++ Programming		2
Pthreads	*2D Convolution*	3
	Matrix Multiplication	4
Threading Building Blocks (TBB)	*Parallel_for*	5
	Parallel_reduce	6
	Pipelining	7
Real-Time Programming		8
Optimizing Real-Time Embedded Applications		9
Applications: CNN		10

TECHNICAL ASSISTANCE

- If you have general questions about the course (such as due dates, content, etc.) or trouble accessing any of the content in this course, please contact the instructor.
- For Moodle technical issues that you cannot resolve on your own, please contact the e-LIS (e-Learning and Instructional Support) office:
  - ✓ e-LIS Helpdesk Phone: (248) 805-1625
  - ✓ Submit a Moodle help ticket

REQUIRED TECHNOLOGY AND BACKUP PLAN

- To fully participate in this class, you will need an internet connected computer with the most updated version of your favorite web browser installed.
  - ✓ In the event that your computer crashes or internet goes down, it is essential to have a "backup plan" in place where you are able to log in using a different computer or travel another location that has working internet.
- In order to present your laboratory work remotely, you need to have a device with a camera than can stream video via Zoom or Google Video.
- **Fall 2024**: The hardware kit will be leased to students. Students need to acquire a mouse, keyboard, and a VGA or HDMI screen. The mouse and keyboard should be preferably non-wireless (to avoid having to deal with driver issues).
- Any files you intend to use for your course should be saved to a cloud solution (Google Drive, Dropbox, etc.) and not to a local hard drive, USB stick or external disk. Saving files this way guarantees your files are not dependent on computer hardware that can fail.
- **Homeworks and exams**: They are posted as pdf files, and students need to post their work as pdfs. In order to do this, students need to be proficient in editing pdfs or generating pdfs out of scanned pages or pictures. It is the student's responsibility to:
  - ✓ Ensure that the submitted file is correct. Corrupted, unrelated, or invalid files will be assigned 0 (no exceptions).
  - ✓ Submit the assignment on time (by 11:59 pm on the due date). Late submissions will NOT be accepted.

CLASS POLICIES

- **The instructor is expected to**:
  - ✓ Grade assignments within a week of the assignment deadline.
  - ✓ The instructor will login into the course every day, at least 5 days a week.
  - ✓ Respond to emails and to Q&A forums replies within 1-2 days.

- **Students are expected to**:
  - ✓ Ensure that their computer is compatible with Moodle.
  - ✓ Follow the calendar of events and complete all assignments by their deadline. Students are responsible for ensuring the timely and correct submission of their assignments (should an issue arise with Moodle, students can email the assignment to the instructor as a last resort).
  - ✓ Respond to emails within 2 days
  - ✓ Participate in a thoughtful manner
  - ✓ Respect rules of etiquette
    - Respect your peers and their privacy
    - Use constructive criticism
    - Refrain from engaging in inflammatory comments.

- **E-mail communication**: The instructor will only respond to emails from students that use their Oakland.edu account. Answering student emails from an email other than an Oakland.edu email is in violation of FERPA because the identity of the sender or receiver cannot be verified.

- **Course Questions & Answer Forum**: Students are encouraged to use this forum to post questions (associated with the course content) that they deem of interest to their classmates. The instructor will intervene periodically.

- **Assignments (Homeworks, Laboratory)**: Note that the homework and laboratory work is individual, and students are not allowed to submit their work in groups.

- **Academic conduct policy**: All members of the academic community at Oakland University are expected to practice and uphold standards of academic integrity and honesty. Academic integrity means representing oneself and one's work honestly. Misrepresentation is cheating since it means students are claiming credit for ideas or work not actually theirs and are thereby seeking a grade that is not actually learned. Academic dishonesty will be dealt with seriously and appropriately. Academic dishonesty includes, but it is not limited to cheating on examinations, plagiarizing the works of others, cheating on lab reports, unauthorized collaboration in assignments, hindering the academic work of other students.

- **Special Considerations**: Students with disabilities who may require special consideration should make an appointment with campus Disability Support Services, 106 North Foundation Hall, phone 248 370-3266. Students should also bring their needs to the attention of the instructor as soon as possible. For academic help, such as study and reading skills, contact the Academic Skills/Tutoring Center, 103 North Foundation Hall, phone 248 370-4215.

- **Add/Drops**: The university policy will be explicitly followed. It is the student's responsibility to be aware of deadline dates for dropping courses.

- **Attendance**: It is assumed that the students are aware of and understand the university attendance policy. Attendance is mandatory and maybe monitored. Students are responsible for all material covered in classes that they miss. There will be no excuses for being late to quizzes/exams.

- **Athlete Excused Absences**: Students shall inform the instructor of dates they will miss class due to an excused absence prior to the date of that anticipated absence. For activities such as athletic competitions whose schedules are known prior to the start of a term, students must provide their instructors during the first week of each term a written schedule showing days they expect to miss classes. For other university excused absences, students must provide the instructor at the earliest possible the dates that they will miss.

- **Special Circumstances:** The instructor should be notified as early as possible regarding any special conditions or circumstances which may affect a student's performance during the course timeframe (e.g., medical emergencies, family circumstances).

- **Mental Health Resources**: Oakland University is committed to advancing the mental health and well-being of its students. If you or someone you know is feeling overwhelmed, depressed, and/or in need of support, services are available. For help, contact the OU Counseling Center in the Human Health Building at (248) 370-3465 or the SEHS Counseling Center at 250A Pawley Hall, (248) 370-2633, https://oakland.edu/counseling/sehs-cc/. Student resources can also be found at

https://www.oakland.edu/deanofstudents/student-health-safety-resources/. For immediate 24/7 services contact Common Ground at https://commongroundhelps.org/#/ via chat or call or text the word "hello" to 1-800-231-1127.

- **Cellphones**: A ringing cellphone going off during a lecture is disruptive to other students as well as the instructor. Students are strongly advised to set their cellphones to vibrate (not ringing) and leave the classroom discretely to answer the phone.

OUTLINE OF COURSE TOPICS, ASSOCIATED ASSIGNMENTS AND REFERENCE MATERIAL.

TOPICS SHADED IN GRAY: SYNCHRONOUS LECTURES

TOPICS SHADED IN RED: ASYNCHRONOUS LECTURES AVAILABLE (PANOPTO)

Week		Unit	Topics	Associated Material	Assignments
1	09/07	1	Class policies. Class structure. Multi-core Processors and Embedded Systems	Syllabus	
2	09/12	1	Intel Atom microprocessor, chipset	Lecture Notes – Unit 1	
			Getting Started with the Hardware and Software Platform	Emb. Prog. Tutorial # 1	
		2	C Programming basics.	Lecture Notes – Unit 2	
	09/14	2	C Programming. Memory management. Functions, function pointers	Lecture Notes – Unit 2	Homework 1
3	09/19	2	Embedded Programming Tutorial #2. C examples	Emb. Prog. Tutorial # 2	*Laboratory 1*
			C++ Programming: introduction	Lecture Notes – Unit 2	
	09/21	2	C++ Programming: constructors, destructors, function/operator overloading	Lecture Notes – Unit 2	
			C++ Programming: Objects. Embedded Programming Tutorial # 2: C/C++ examples	Emb. Prog. Tutorial # 2	
4	09/26	2	C++ Programming: Functors Embedded Programming Tutorial #2: C++ examples	Lecture Notes – Unit 2	*Laboratory 2*
	09/28	3	Multi-threading: pthreads	Lecture Notes – Unit 3	Homework 2
			· Multi-threading: pthreads – create/use threads · Multi-threading: pthreads – examples	Emb. Prog. Tutorial # 3	
5	10/03	3	Multi-threading: mutexes	Lecture Notes – Unit 3	*Laboratory 3*
	10/05	3	Multi-threading: condition variables		
			Multi-threading: pthreads – mutexes examples	Emb. Prog. Tutorial # 4	
6	10/10	4	TBB (Threading Building Blocks): Overview and *parallel_for*.	Lecture Notes – Unit 4 Emb. Prog. Tutorial # 5	*Laboratory 4*
			TBB: *parallel_for* examples	Emb. Prog. Tutorial # 5	
	10/12	4	TBB: parallel_for examples	Lecture Notes – Unit 4 Emb. Prog. Tutorial # 6	
7	10/17	4	TBB: *parallel_invoke* examples	Lecture Notes – Unit 4	
	10/19		Midterm Exam		
8	10/24	4	TBB: *parallel_reduce* and examples	Lecture Notes – Unit 4 Emb. Prog. Tutorial # 7	*Laboratory 5*
	10/26	4	TBB: *parallel_reduce*: summarize reduce+map patterns in Tut #6, Lab 6 TBB: *parallel_scan* and examples	Emb. Prog. Tutorial #7	Homework 3
9	10/31	4	Pipelining TBB: *parallel_pipeline:* intro	Lecture Notes – Unit 4	*Laboratory 6*
			TBB: *parallel_pipeline* – examples	Emb. Prog. Tutorial #7	
	11/02	4	TBB: *parallel_pipeline* – examples	Lecture Notes – Unit 4	
10	11/07	5	TBB: *parallel_for_each* TBB: *parallel_for_each* – examples Real-Time Programming	Lecture Notes – Unit 5	*Laboratory 7*
	11/09	5	Real-Time Programming – examples	Emb. Prog. Tutorial #8	Homework 4
			Real-Time Programming - examples		
11	11/14	6	Design and Optimization of Real-Time Systems	Lecture Notes – Unit 6	*Laboratory 8*
	11/16	6	Design and Optimization of Real-Time Systems		
			Single-Core Performance Tools: sysprof, GNU prof		
12	11/19	7	Applications: CNN	Lecture Notes – Unit 7	
	11/21	7	Applications: CNN		
13	11/26	7	Applications: Beamforming.		
14	12/03	7	Applications: Cylinder Pressure Estimation		
	12/05		Final Project: Discussion about presentation		
15	12/12		Final Project – Presentation		