ECE 3411: MICROPROCESSOR APPLICATIONS LABORATORY – FALL 2017

Lecture: MoWe 2:30-3:20, ITE 125, Lab: MoWe 3:30-5:30 PM, ITE C30

Instructor: Prof. Marten van Dijk **Office:** ITEB 431 **Email**: marten.van_dijk@uconn.edu

TA: Hamza Omar **Email**: hamza.omar@uconn.edu **Office Hours:** By appointment biweekly on Fr TBD

Course Description

The basis of sophisticated designs of modern digital systems that appear in products such as automobiles, appliances and industrial tools is the Micro Controller Unit (MCU); a microcomputer optimized for single-chip system design for controlling peripheral devices geared to real-time applications. We will use the Atmega328P microcontroller (from the Atmel MEGA series RISC microcontrollers) in a series of labs to design microcontroller based systems. In particular, we will explain how to read and use the Atmega328P datasheet and how to write self-explanatory C code for interfacing with the MCU and controlling peripheral devices. This includes initializing register values, writing Interrupt Service Routines (ISRs), constructing underlying Finite State Machines (FSMs), and using a task based programming approach based on hardware timers.

[I] Learning Objectives (EAC outcomes (a) an ability to apply knowledge of engineering, and (k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice):

To get familiarized with basic C-language concepts:

- To be able to program in basic C (without pointer constructs) by
  - **[O1]** emulation of read examples (how to apply C primitives and how to layout your C code and well-comment your C-code such that it can be interpreted and understood by colleagues), by
  - **[O2]** learning how to set up an Interrupt Service Routine (ISR) and how to communicate over microcontroller pins, by
  - **[O3]** writing Finite State Machines (FSMs) that specify state transitions based on interrupt events and explain how program variables change, and by
  - **[O4]** adopting a task based programming approach (without blocking delay functionality) and
  - **[O5]** writing non-blocking procedures (to communicate with e.g. UART and LCD).
- **[O6]** To be able to debug programming errors and use the debugging tool to observe how the microcontroller steps through assembly instructions that represent procedures and ISRs.
- **[O7]** To understand the importance of a Real Time Operating System (RTOS) and to be able to implement basic schedulers.

To read schematics and datasheets:

- **[O8]** To be able to understand how the pins of the microcontroller are connected to peripheral devices.
- **[O9]** To be able to read and understand the corresponding Atmega datasheet, in particular, how to enable interrupts and program their properties.

[II] Reading:

Optional Textbook: Elliot Williams, *Make: AVR Programming*, 2014. This textbook can be used to get somewhat familiarized with each topic before attending lecture – this will allow you to take in the material presented during lecture. We will also post lecture slides before each lecture – and these (even though they are at a more abstract level) can also be used to become familiarized with the material before it is presented in lecture.

Great website: Bruce Land's course at Cornell: http://people.ece.cornell.edu/land/courses/ece4760/

Distribution of slide decks, lab problems, other reading material, etc.: If you have not already received an invitation, you can sign up for Piazza yourself by following the link piazza.com/uconn/fall2017/ece3411. We will use Piazza as the method of preference to distribute general announcements, solutions to lab problems and problem sets, as well as additional reading material.
Website http://scl.uconn.edu/courses/ece3411/index.php will post lecture slides, lab problems, and problem sets.

[III] Organization Lectures and Labs:

- The lectures are organized in 7 blocks of two weeks each:
    - Block 0 (B0) revisits and tests your basic programming skills as a prerequisite to the remaining course, i.e., we will use "if statements", "while loops", "procedures", "arrays", etc. (an understanding of 'complex' pointer based data structures is not needed for a successful completion of this course).
      Also we will ask you to write an essay titled "How I plan to get an A for the Microprocessor Applications Laboratory" with a minimum length of 300 words.
    - Blocks 1, 2, and 3 (B1, B2, and B3) will guide students step-by-step towards solutions: That is, *you need to take the initiative to ask questions during lab and office hours.* If you can clearly explain where you get stuck and what you have been thinking of, we will provide hints to help you move forward with your problem set and/or labs. You can also ask us whether your partial solution is correct and how we did it. Final solutions to problem sets and labs will be posted once all students have submitted their work.

  - In blocks 4 and 5 (B4 and B5) we will mostly let you work on lab problems independently: you can still and should ask us questions – but our hints may be sparse because we want you to think more on your own.
  - Block 6 (B6) is completely independent – we will offer no (or very little) help: In this block you are able to demonstrate what you have learned and how to apply what you have learned.
- Each block (of two weeks each) is structured as follows:
  - There are 3 lectures and a final review and Q&A lecture.
  - The 3 lectures are each followed by an in-class lab which you need to complete yourself in your own words/code, however, you are allowed to ask us questions and you can also work together with your colleague students.
  - The review lecture is followed by a lab which can be used by students to catch up where needed and/or can be used to start an independent lab assignment testing what you have learned during the block. *This independent lab assignment should be completed on your own without help from us or your colleague students*.
  - In each block you will be asked to complete and submit a problem set. You are allowed to ask us questions but you will need to do the questions on your own without help from colleague students. You should make sure to explain solutions in your own words.
- For a maximum grade, you will need to demonstrate the completion of each of the 3 labs as well as the independent lab to us. Also, each block has a problem set which you need to complete and submit. We will give plenty of time for you to finish your assignments.

[IV] Teaching Philosophy: This course implements adult-based learning theory or "andragogy." Andragogy maintains that adults learn best when they have a flexible but challenging learning environment. We will create assignments from which each student can choose how much to do in a positive/safe but challenging learning environment. This will allow students to direct their learning in the manner that best suits their learning objectives and to get the grade she or he wishes. The open-ended assignments allow students to pursue tasks in a manner customized to individual needs and interests.

The grading is "safe" in that each assignment (a lab or a question in the problem set) is graded either pass or fail, and the requirements for a pass are clearly specified. Incomplete work justifies a "fail." It is easy for students to tell whether their work is complete, done in good faith, and consonant with the intended learning outcome. We also reinforce the safety of the learning environment by giving a number of tokens representing opportunities to revise work and allowing 24-hour extensions.

Lab assignments will be pass/fail and each question in a problem set will be pass/fail. The latter means that each correct answer in a problem set is awarded the full number of points corresponding to the answered question. If an answer is only slightly incorrect, zero points (i.e., no credit is given). This will enforce you to ask whether you are on the right track during lab hours and biweekly (at the end of each block) office hours. Also this reflects an industry environment where you need to submit fully working code by a certain preset deadline – if it does not fully work, it cannot be shipped out in a product.

Pass/fail grading without partial grades holds students to high expectations – this brings academic rigor to what it means to receive a satisfactory B or good A grade. Arum and Roksa, "Academically adrift: Limited learning on college campuses," 2011, indicates that 36% of students showed no significant progress in their learning over four years, and those who did improve made only modest gains, according to their Collegiate Learning Assessment scores. Spec grading, see Nilson, "Specifications grading: Restoring rigor, motivating students, and saving faculty time," 2014, as used in this course offers a method to counter this pessimistic trend.

[V] Grading: The following explains in words the required tasks you will need to pass in order to obtain your letter grade of choice. The grading scheme is also depicted in a grading table (after the grade enumeration below).

- Minimum requirements for a D:
    - Pass the 2 dependent and 1 independent lab in B0.
    - Pass the 9 dependent labs in B1, B2, and B3.
    - You can use 5 dependent lab revision tokens.
    - Pass questions in problem sets P0, P1, P2, and P3 with corresponding points aggregating to at least 60% of the total number of points allocated to the questions in P1, P2, and P3.
    - You can use 4 24-hour extension tokens for labs or problem sets.
- Minimum requirements for a C:
    - Pass the 2 dependent and 1 independent lab in B0.
    - Pass the **15 dependent labs in B1, B2, B3, B4, and B5**.
    - You can use 5 dependent lab revision tokens.
    - Pass questions in problem **sets P1, P2, P3, P4, and P5** with corresponding points aggregating to **at least 70%** of the total number of points allocated to the questions in P1, P2, P3, P4, and P5.
    - You can use 4 24-hour extension tokens for labs or problem sets.
- Minimum requirements for a B:
    - Pass the 2 dependent and 1 independent lab in B0.
    - Pass the 15 dependent labs in B1, B2, B3, B4, and B5.
    - You can use **4 dependent lab revision tokens**.
    - **Pass the 10 independent lab problems (2 each block) in B1, B2, B3, B4, and B5.**
    - **You can use 2 independent lab problem revision tokens.**
    - Pass questions in problem sets P1, P2, P3, P4, and P5 with corresponding points aggregating to **at least 80%** of the total number of points allocated to the questions in P1, P2, P3, P4, and P5.
    - **Pass questions in the advanced problem sets A1, A2, A3, A4, and A5 with corresponding points aggregating to at least 70% of the total number of points allocated to the questions in A1, A2, A3, A4, and A5.**
    - You can use **6 24-hour extension tokens** for labs or problem sets.

- Minimum requirements for an A:
  - Pass the 2 dependent and 1 independent lab in B0.
  - Pass the 15 dependent labs in B1, B2, B3, B4, and B5.
  - You can use **3 dependent lab revision tokens**.
  - Pass the 10 independent lab problems (2 each block) in B1, B2, B3, B4, and B5.
  - You can use **1 independent lab problem revision token**.
  - **Pass the independent RedBot lab of B6 (no revision token allowed). The last lab is reserved for demonstrating each student's RedBot project.**
  - Pass questions in problem sets P1, P2, P3, P4, and P5 with corresponding points aggregating to **at least 85%** of the total number of points allocated to the questions in P1, P2, P3, P4, and P5.
  - Pass questions in the advanced problem sets A1, A2, A3, A4, A5, **and A6** with corresponding points aggregating to **at least 80%** of the total number of points allocated to the questions in A1, A2, A3, A4, A5, **and A6.**
  - You can use 6 24-hour extension tokens for labs or problem sets.
- If you fail to meet the requirements for a D- (see below), you will receive an F.
- It is not recommended to not do P0 because you expect to be able to meet the requirements for C- (see below) or higher: Because if you would fail to meet C- requirements without P0, then you would automatically also fail D- requirements and receive an F.
- If you write an essay titled "How I plan to get an A for the Microprocessor Applications Laboratory" with a minimum length of 300 words in B0, then you will receive **1 additional 24-hour extension token**.

Letter grade or Pass/Fail requirements	Dependent Labs (B0 has 2; B1-5 each 3)	Independent Lab Questions (B0-5 each 2)	PSets	Adv. PSets	24hr Ext. Tok.	RedBot Project (No tokens)
A/Pass	17 (B0-B5) 3 Rev. Tokens	12 (B0-B5) 1 Rev. Token	P1-P5 85%	A1-A6 80%	6 plus 1 for essay	Pass
B/Pass	17 (B0-B5) 4 Rev. Tokens	12 (B0-B5) 2 Rev. Token	P1-P5 80%	A1-A5 70%	6 plus 1 for essay	Not Req.
C/Pass	17 (B0-B5) 5 Rev. Tokens	2 (B0)	P1-P5 70%	Not Req.	4 plus 1 for essay	Not Req.
D/Fail	11 (B0-B3) 5 Rev. Tokens	2 (B0)	P0-P3 60%	Not Req.	4 plus 1 for essay	Not Req.

Fine tuning to letter grades with plus/minus signs:

- Passing A requirements by using one extra token (for revision of a dependent lab, revision of an independent lab question, or 24hr extension) will drop you to an A-.

- Passing B requirements and also passing the **RedBot project** as well as **A1-A6 with 70%** will increase your grade to a B+
- Passing B requirements by using one extra token (for revision of a dependent lab, revision of an independent lab question, or 24hr extension) will drop you to a B-.
- Passing C requirements by **only using 4 dependent lab revision tokens** and obtaining an **80% average on P1-P5** will increase your grade to a C+.
- Passing C requirements by using one extra token (for revision of a dependent lab or 24hr extension) will drop you to a C-.
- Passing D requirements by **only using 4 dependent revision lab tokens** and obtaining a **70% average on P0-P3** will increase your grade to a D+.
- Passing D requirements by using one extra token (for revision of a dependent lab or 24hr extension) will drop you to a D-.
- If you chose to receive a pass/fail or satisfactory/non-satisfactory grade instead of a letter grade, then **PASS IS EQUIVALENT TO C- OR HIGHER** and **FAIL IS EQUIVALENT TO D+ OR LOWER**.
- Plagiarism and other anti-intellectual behavior cannot be tolerated in any academic environment that prides itself on individual accomplishment. Such behavior will be reported and will lead to academic scrutiny (including at most an F for the course).

Collaboration policy: **You can ask for help during labs and office hours from the instructors for the red colored columns in the table, i.e., dependent lab assignments and problem sets. You are allowed to collaborate with colleague students only on dependent labs – but not any of the independent labs, and problem sets. You may not discuss independent labs and problem sets in any way, shape, or form with anyone other than your instructor. You are not allowed to collaborate or receive help from instructors for the independent labs and RedBot project.**

DURING THIS COURSE YOU ARE NOT ALLOWED TO RETRIEVE OR LOOKUP SOLUTIONS FROM OTHER SOURCES (e.g., colleague students posting solutions, other websites posting solutions from this/previous offerings, etc.) – YOU NEED TO DO PROBLEM SETS AND LABS ACCORDING TO THE ABOVE COLLABORATION POLICY.

For the dependent labs you are encouraged to collaborate and study together. In fact, students who form study groups generally do better than do students who work alone. If you do work in a study group, however, you owe it to yourself and your group to be prepared for your study group meeting. Specifically, you should spend at least 30-45 minutes trying to understand lecture material and solve any of the open questions beforehand. If your group is unable to solve a problem, talk to other groups or ask the instructor.

Passing a dependent lab problem / independent lab question:

- To pass a lab problem (a lab assignment has more than one problem) you will need to satisfy the following specifications:
    - Working code:
        - Your code should compile.
        - Your code should solve the lab problem statement – i.e., the code should meet the lab problem's specification.
        - You need to demo your code on your own MCU during lab.
    - Coding style:
        - Your code should have at least one meaningful comment before each variable declaration, procedure or function definition, if-then-else statement, and loop/while/for statement. This will help you to make your code readable (to yourself as well as colleagues).
        - Unless otherwise stated, you are not allowed to use delay_ms and delay_um functionality.
        - You need to follow the coding template as taught in class (in later blocks you will do task based programming).
        - Since the first parts with hints in a lab problem lead to the final specification, you will only submit the code corresponding to this final specification in pdf form to the TA, who will verify the coding style and add questions, see below.
    - Understanding:
        - If you are using a revision token, then you also need to explain in at least 5 sentences in own words how your previous code needed to be improved (with what kind of modified design choices) in order to pass the above requirements. If you had not submitted previous code, then you need to explain in at least 5 sentences in own words what you learned from the posted solutions – what design choices are used in the solutions? You should add this explanation in the body of your email when submitting the pdf version of your code.
        - If you are not using a revision token, we may put several comments "E" (of "Explain your design choice") in the pdf version of your code. The "E" comments each represent the following question:
            - Explain in at least 3 sentences and in own words the thought process that led you to your design choice for the code referred at by the comment. If you cannot do this, **your letter grade will be reduced by 1/3 letter grade**. So, if you cannot properly explain your thought process to say 3 *different* spots in your code, then your letter grade drops by a full letter! This is a harsh measure in order to make sure you will understand why you chose to code something in a certain way (especially if you received help from instructors or colleague students). In other words you need to make your code your own.

- If we indeed decide to ask you to do the above, then (1) you will get time at the start of a lab section to formulate your answers by adding responses to the questions in the comments of the pdf file – this personalized test is closed book/laptop, and/or (2) you will get time at the start of a lab section to orally explain to us your design choices and coded solutions.

- Dependent lab assignments are handed out in class/lab:
  - 6 days later your assignment is due at noon (e.g., the Monday lab assignment is due @ noon the Sunday after; the Wednesday lab assignment is due @ noon the Tuesday after). I.e. the pdf version of your final code needs to be emailed to the TA by noon, and you need to demonstrate your code that day in lab to the TA.
  - You can make use of exactly *one* 24-hour extension token (e.g., noon Sunday becomes noon Monday; noon Tuesday becomes noon Wednesday) – if this is not sufficient to pass your lab assignment, then instead of a 24-hour extension token:
  - You can make use of exactly *one* dependent lab revision token. Solutions will be posted 7 days after the dependent lab assignment was handed out in class/lab – this means that a revision token allows you to see and study our solution, and you can use this to complete your lab assignment. This will create a safe learning environment. Your revised solution needs to be submitted after another 3 days, i.e., 10 days after the dependent lab assignment was handed out in class/lab (again at noon in pdf form and later demonstrated in lab).
  - You need to clearly state in your email to the TA in the subject line the lab assignment for which you are submitting the pdf version of your code as follows: "ECE-3411 LAB1b". **In the body of your email you write with whom you collaborated on the solution**.
- Independent lab assignments are handed out in class/lab every two weeks on Wednesdays:
  - The assignment contains two lab questions about the material presented so far (this also includes preceding blocks) and studied during the three labs of the block. Wednesday at noon the solutions of the second lab will be posted. This means that all related dependent lab solutions, except for the third and final dependent lab of the block, are available to you for solving the two independent lab questions. Therefore, it is important for students aiming at a B or A grade to use the Wednesday lab to complete the third lab (of the preceding Monday) by asking instructors for help so that its solution can be used to solve the independent lab assignment.
  - You are not allowed to collaborate on figuring out your solution. The idea is to do this completely on your own with no help (like when doing a take-home exam).
  - 4 days after the independent lab assignment was handed out in class/lab, your assignment is due at noon (Sunday at noon). (This coincides with submitting the third dependent lab and problem set, see below.) I.e. the pdf version of your final code needs to be emailed to the TA by noon, and you need to demonstrate your code later in lab to the TA.
  - You can make use of exactly *one* 24-hour extension token.
  - You can make use of exactly *one* independent lab question revision token *per lab question* (notice that each independent lab assignment has 2 lab questions). Solutions will be posted

5 days after the independent lab assignment was handed out in class/lab (i.e., Monday at noon) – this means that a revision token allows you to see and study our solution, and you can use this to complete your lab assignment. This will create a safe learning environment. Your revised solution needs to be submitted after another 3 days at noon (i.e., Thursday at noon), i.e., 8 days after the dependent lab assignment was handed out in class/lab (again at noon in pdf form and later demonstrated in lab).

  - You need to clearly state in your email to the TA in the subject line the lab assignment for which you are submitting the pdf version of your code as follows: "ECE-3411 INDEPENDENT LAB 1".

- Independent RedBot project:
  - You are not allowed to collaborate on figuring out your solution. The idea is to do this completely on your own with no help (like when doing a take-home exam).
  - Your assignment needs to be demoed on the last Wednesday of class during lab time.
  - The pdf version of your code needs to be submitted on the Tuesday before the last Wednesday class (this coincides with problem set A6.
  - No revision or 24-hour extension tokens can be used.
  - You need to clearly state in your email to the TA in the subject line the project assignment for which you are submitting files as follows: "ECE-3411 FINAL PROJECT".

Passing problem sets:

- Passing a problem set:
  - Each (sub-)question in the problem set must be answered using **at least 3 sentences**. **If you do not satisfy this requirement, then the points collected by your good answers will be halved.** So, we are checking whether you put in effort.
  - Each question part is graded pass/fail. If a question part is correct/passed, then it receives the full number of points allocated to that part. If failed, zero points are given.
  - We may put several comments "E" (of "Explain your answer") in the pdf version of your problem set. The "E" comments each represent the following question:
    - Explain in at least 3 sentences and in own words the thought process that led you to your answer for the question part referred at by the comment. If you cannot do this, **your letter grade will be reduced by 1/3 letter grade**. So, if you cannot properly explain your thought process to say 3 *different* question parts in your problem set, then your letter grade drops by a full letter! This is a harsh measure in order to make sure you understand what you wrote yourself (especially if you received help from instructors). In other words you need to make your answers your own. We note that you need to be able to explain your thought process for all your answers, the correct ones and those incorrect or incomplete. So, better is to honestly answer a question part incomplete rather than writing a correct answer obtained from another source which you do not understand.

We may put several comments "S" (of "Explain how the posted solution is different form yours") in the pdf version of your problem set. The "S" comments each represent the following question:

- Explain in at least 3 sentences and in own words how the posted solution for the question part referred at by the comment is different from yours and why (in what sense is your answer incorrect or incomplete). **This requires you to read and understand the posted solutions.** If you cannot do this, **we reduce your problem set score by 20% of the points allocated to the question part referred at by the comment**. So, if you cannot properly explain the posted solution to say 3 *different* question parts in your problem set which were allocated 4, 8, and 8 points out of a 100, then your problem set score drops by 20% of 4+8+8, which equals 4 points; i.e. 4 points are subtracted from the points accumulated by your correctly answered problem parts. This is a harsh measure in order to make sure you study posted solutions and ask help (form the instructors or colleague students) if you are unable to understand.

If we indeed decide to ask you to do the above, then (1) you will get time at the start of a lab section to formulate your answers by adding responses to the questions in the comments of the pdf file – this personalized test is closed book/laptop, and/or (2) you will get time at the start of a lab section to orally explain to us your thought process behind your answers to question parts (comments "E") and how and why your answers to question parts are different from the posted solutions (comments "S").

A personalized test will likely be split in two parts: "E" comments need to be answered before solutions are posted and "S" comments need to be answered after solutions are posted.

- Problem sets are handed out in class at the beginning of each block:
  - At the start of each block we hand out a problem set. The set contains problems which all student need to solve as well as problems specific for students who pursue a B or A grade.
  - There are no revision tokens for problem sets. You can use exactly *one* 24-hour extension token per problem set. This may seem overly strict – but you can submit what you have finished and write out your thoughts regarding the questions you have not been able to solve yet. Notice that in the end we are computing the *average* percentage of points collected by good answers over multiple problem sets – and this means you are able to compensate low scoring by better scoring problem sets. This will create a safe learning environment.
  - Your problem set is due at the same time when the independent lab assignment is due (except for A6, which is due the day before the RedBot project demo). This will give you more than 2 weeks to finish each problem set.
  - You need to clearly state in your email to the TA in the subject line the problem set for which you are submitting your answers as follows: "ECE-3411 P1" or "ECE-3411 P1 AND A1".

- **Your submission must be type written and submitted in pdf form**. You must make sure your answers are readable – if not, you will need a 24-hour extension token to rewrite/edit your answers. We do not accept unreadable answers – the answers should have complete sentences.
- Solutions to problem sets will ideally be posted together with the solution of the independent lab assignment. Whether this is possible depends on when related personalized student tests are taken and completed.

Personalized tests: We aim at making sure each student gets as many personalized tests (as explained above – see "E" and "S" comments for labs and problem sets) as possible (given the instructor's time). I.e., those students doing work for a B or A grade should expect to be mainly tested on their solutions for the independent lab questions and advanced problem sets. Students doing work for a D or C grade should expect to be tested on their solutions for the dependent labs and problem sets. For now, but this may change, we will likely have every student do personalized tests on all submitted work.

A diagram of assignments and (personalized) tests for a typical block is given below:

Mo	We	Mo	We	Mo	We
Laba out		Demo Laba "E" Test Laba Sol Laba out			
	Labb out		Demo Labb "E" Test Labb Sol Labb out		
		Labc out		Demo Labc "E" Test Labc Sol Labc out	
			LAB out	Demo LAB "E" Test LAB Sol LAB out	
Problem sets P and A out				"E" Test P and A If no test is postponed, then Sol P and A out	"S" Test P and A

Since you may need to do a personalized test, **you need to be present at the start of each lab**. In case of absence, see [IX] below and communicate with the instructor beforehand – we will want to give you your personalized test another time on the same day or the day after.

[VI] I want you to be successful in class: I look forward to helping you learn and develop a deep understanding of the course material. I would like our relationship to be such that we work together to achieve this end goal.

There is plenty of time to ask questions during lab and biweekly office hours. When asking questions you need to be prepared. General questions such as "I do not understand the whole lecture" are not concrete enough. By thinking the material through you will find the concrete spots for which you can ask guidance. **I expect you to be actively engaged in your learning.** In order to learn how to program an MCU with peripherals you will need to practice.

Come prepared for class: Look at the lecture slides beforehand, do optional reading, already look ahead at dependent labs whenever possible etc. **Prepare diligently outside of class and come to class ready to work. Don't procrastinate and ask for help when stuck. Be engaged and active in your learning – make effective use of lab time!**

Also be engaged and active in your learning after class: An effective study strategy is to immediately after lecture and lab summarize what was taught in own words, bullet points, and diagrams. This allows you to spot parts that are unclear to you and for which you can ask questions. This also helps you to learn by heart the essentials needed for successfully completing labs etc.

Adopt a "growth mindset" for your intellectual development: According to Carol Dweck, the psychologist who coined this term, those with a "fixed mindset" believe their basic qualities such as intelligence and skills are fixed quantities. On the other hand those with a "growth mindset" believe that these basic qualities can be improved through dedication and hard work, and when they fail at something, they take it as a learning opportunity and get better by learning from their mistakes. (See http://edglossary.org/growth-mindset/.)

Practice self-regulated learning: This means that you are paying not only to what you are learning but how you are learning it, and you are not dependent upon the instructor or anyone else to learn things. It means in particular:

- Be aware at all times of what you are supposed to be learning.
- When you work, work purposefully on activities that show where your learning is in relationship to the learning objectives for the course. Some of the activities are ones that you take on yourself, or you make them up yourself, even if no grade is involved.
- If you know there is a gap between your abilities and the learning objectives, take initiative to find things that will help you close it.

You are responsible for your own time management: Dependent lab assignments need to be submitted after 6 days (if no tokens are used), independent lab assignments need to be submitted after 4 days (if no tokens are used), and each problem set can be solved during the time window of its block an another 4 days (if no

tokens are used). If you push work to the last moment and something unexpected happens (like not a good night's rest, some emergency, etc. etc.), then this is your responsibility – **you should have made sure to anticipate unexpected distractions and finish your work early**.

This course uses a points-free, competency-based grading system. You choose the letter grade you want to earn in the course and then work through a variety of assignments to demonstrate the level of competency the grade requires. There are no points! **All work (questions in problem sets, lab problems, project) is graded on a pass/fail basis, so careful attention to the specifications for acceptable work is a must.**

[VII] Course Schedule and Student Activities:

Before coming to first lecture/lab:

- Check Order Kit
- Make sure to have set up Eclipse/GCC & Atmel Studio; see document Tools_Setup.pdf which explains how to do this for your laptop, you do not have time for installing software during lab!
- Sign up for Piazza

wks	Date	Lecture	Lab	Optional Reading / Notes
1	Mo 28-Aug	**Lec0a:** Course Outline + Grading policy	**Lab0a:** Examples basic C-Programming	Read syllabus **Lab0a due Su 3-Sept** *P0 and Essay out*
	We 30-Aug	**Lec0b:** Introduction to C-Programming	**Lab0b:** Examples basic C-Programming Continued	**Lab0b due Tu 5-Sept**
2	4-Sept	Labor Day – no classes	Labor Day – no classes	
	We 6-Sept	**REVIEW, Q&A** C-Programming, Grading policy	**Independent LAB0:** Basic C-Programming using the Eclipse compiler	**LAB0, P0, and Essay due Su 10-Sept** *(No Office Hours)*
3	11-Sept	**Lec1a:** Microcontroller introduction + General Purpose Digital Output (GPDO)	**Lab1a:** AVR Board Setup (soldering) + LEDs (GPDO)	Ch. 1 Ch. 2 till page 21 Ch. 3 Ch. 4 **Lab1a due Su 17-Sept** *P1 and A1 out* **DROP DATE**
	13-Sept	**Lec1b:** Universal Asynchronous Receiver & Transmitter (UART)	**Lab1b:** UART (recognizing strings)	Ch. 5 till page 97 **Lab1b due Tu 19-Sept**

4	18-Sept	**Lec1c:** General Purpose Digital Input (GPDI) + LCD Display	**Lab1c:** LCD (GPDI)	Ch. 6 **Lab1c due Su 24-Sept**
	20-Sept	**REVIEW, Q&A** GPDO, GPDI, LEDs, UART, LCD	**Independent LAB1** GPDO, GPDI, LEDs, UART, LCD	**LAB1, P1, A1 due Su 24 Sept** *Office Hours*
5	25-Sept	**Lec2a:** Interrupt Service Routine (ISR)	**Lab2a:** Non-blocking UART + debounced switches (uses ISR)	**Lab2a due Su 1-Oct** *P2 and A2 out*
	27-Sept	**Lec2b:** ISRs + Timer 0 + Task based programming	**Lab2b:** Non-blocking LCD	Ch. 9 **Lab2b due Tu 3-Oct**
6	2-Oct	**Lec2c:** Timers 0, 1, and 2	**Lab2c:** Non-blocking LCD continued	**Lab2b due Su 8-Oct**
	4-Oct	**REVIEW, Q&A** Main focus: ISRs, Timers, non-blocking UART and LCD	**Independent LAB2** Main focus: ISRs, Timers, non-blocking UART and LCD	**LAB2, P2, A2 due Su 8-Oct** *Office Hours*
7	9-Oct	**Lec3a:** Debugging	**Lab3a:** Debugging	**Lab3a due Su 15-Oct** *P3 and A3 out*
	11-Oct	**Lec3b:** External Interrupt + Pin Interrupt	**Lab3b:** Human reaction time + Capture Interrupt Timer 1	Ch. 8 **Lab3b due Tu 17-Oct**
8	16-Oct	**Lec3c:** External Interrupt + Task based programming	**Lab3c:** Stopwatch	**Lab3c due Su 22-Oct**
	18-Oct	**REVIEW, Q&A** Main focus: Debugging, External Interrupt, Timers	**Independent LAB3** Main focus: Debugging, External Interrupt, Timers	**LAB3, P3, A3 due Su 22-Oct** *Office Hours*
9	23-Oct	**Lec4a:** Pulse Width Modulation (PWM)	**Lab4a:** PWM	Ch. 10 **Lab4a due Su 29-Oct** *P4 and A4 out*
	25-Oct	**Lec4b:** Analog-to-Digital Conversion (ADC)	**Lab4b:** ADC	Ch. 7 Ch. 12 **Lab4b due Tu 31-Oct**
10	30-Oct	**Lec4c:** Eeprom + Watchdog	**Lab4c:** Eeprom + Watchdog + Assembly	Ch. 18: p. 387-396 Ch. 19 Ch. 20 **Lab4c due Su 5-Nov**

				DROP DATE / CONVERSION TO Pass/Fail (D+, D, D-, F students will have completed the course)
	1-Nov	**REVIEW, Q&A** Main focus: PWM, ADC, Eeprom, Watchdog, Assembly	**Independent LAB4** Main focus: PWM, ADC, Eeprom, Watchdog, Assembly	**LAB4, P4, A4 due Su 5-Nov** *Office Hours*
11	6-Nov	**Lec5a:** Task based programming revisited + Real Time Operating System (RTOS) global understanding	**Lab5a:** RTOS Scheduling	**Lab5a due Su 12-Nov** *P5 and A5 out*
	8-Nov	**Lec5b:** RTOS Cont'd + SPI	**Lab5b:** SPI + DAC	Ch. 16 **Lab5b due Tu 14-Nov**
12	13-Nov	**Lec5c:** I2C + RedBot (PID control) + Servo Control	**Lab5c:** I2C	Ch. 11 Ch. 15 Ch. 17 **Lab5c due Su 19-Nov**
	15-Nov	**REVIEW, Q&A** Main focus: RTOS, DAC, SPI, I2C, Servo Control	**Independent LAB5** Main focus: RTOS, DAC, SPI, I2C, Servo Control	**LAB5, P5, A5 due Su 19-Nov** *Office Hours*
	20-Nov	Thanksgiving Recess – no classes	Thanksgiving Recess – no classes	
	22-Nov	Thanksgiving Recess – no classes	Thanksgiving Recess – no classes	
13	27-Nov	**Lec6a:** Advanced Topics	**Independent LAB6/Project:** RedBot	**LAB6/RedBot:** **Code due Tu 5-Dec** **Demo We 6-Dec** *A6 out*
	29-Nov	**Lec6b:** Topics advanced MCU Applications Laboratory (Spring 2018)	**Independent LAB6/Project:** RedBot	
14	4-Dec	**REVIEW, Q&A** Anything	**Independent LAB6/Project:** RedBot	**A6 due 5-Dec**
	6-Dec	**TBD**	**RedBot Demo**	
15	11-15 Dec	**Finals Week – No final**	**Finals Week – No Final**	

[VIII] Course objectives: The grade you earn in the course will be based on how well you eventually demonstrate mastery of course learning objectives. Blocks and objectives are related as follows:

		O1	O2	O3	O4	O5	O6	O7	O8	O9
B0	Testing prerequisite coding skills before the first drop date.	Y								
B1	GPDO, GPDI, LEDs, UART, LCD	Y							Y	Y
B2	ISRs, Timers, non-blocking UART and LCD	Y	Y	Y	Y	Y				Y
B3	Debugging, External Interrupt, Timers	Y	Y		Y	Y	Y			Y
B4	PWM, ADC, Eeprom, Watchdog, Assembly		Y		Y	Y				Y
B5	RTOS, DAC, SPI, I2C, Servo Control		Y		Y	Y		Y		Y
B6	Overview advanced topics, RedBot project		Y		Y	Y	Y		Y	Y

[IX] University wide policies:

Sec. 10a-50. (Formerly Sec. 10-334g). Absence of students: Students should inform their instructor about any potential conflicts with scheduled exams or other assignments and a religious holiday that they observe. For conflicts with final examinations, students should, as usual, contact the Office of Student Services and Advocacy (formerly the Dean of Students Office). Faculty and instructors are strongly encouraged to make reasonable accommodations in response to student requests to complete work missed by absence resulting from religious observances or participation in extra-curricular activities that enrich their experience, support their scholarly development, and benefit the university community. Examples include participation in scholarly presentations, performing arts, and intercollegiate sports, when the participation is at the request of, or coordinated by, a University official. Such accommodations should be made in ways that do not dilute or preclude the requirements or learning outcomes for the course. Students anticipating such a conflict should inform their instructor in writing within the first three weeks of the semester, and prior to the anticipated absence, and should take the initiative to work out with the instructor a schedule for making up missed work. For conflicts with final examinations, students should contact the Office of the Dean of Students.

Final exam week for Fall 2017 takes place from Monday, December 11th through Sunday, December 17, 2017. Students are required to be available for their exam during the stated time. If you have a conflict with this time, you must visit the Dean of Students Office to discuss the possibility of rescheduling this exam.

Please note that vacations, previously purchased tickets or reservations, social events, misreading the exam schedule and over-sleeping are not viable excuses for missing a final exam. If you think that your situation warrants permission to reschedule, please contact the Dean of Students Office with any questions. Thank you in advance for your cooperation.

Faculty and instructors are also encouraged to respond when the Counseling Program for Intercollegiate Athletes (CPIA) requests student progress reports. This will enable the counselors to give our students appropriate advice.

Disabilities: The Center for Students with Disabilities (CSD) at UConn provides accommodations and services for qualified students with disabilities. If you have a documented disability for which you wish to request academic accommodations and have not contacted the CSD, please do so as soon as possible. The CSD is located in Wilbur Cross, Room 204 and can be reached at (860) 486-2020 or at csd@uconn.edu. Detailed information regarding the accommodations process is also available on their website at www.csd.uconn.edu.

Let the instructor know as soon as possible if you need adaptations or accommodations because of a disability (e.g. learning disability, attention deficit disorder, psychological, physical), or if you have emergency medical information which you should share with the instructor, or if you need special arrangements in case the building must be evacuated.

Policy Against Discrimination, Harassment and Related Interpersonal Violence: The University is committed to maintaining an environment free of discrimination or discriminatory harassment directed toward any person or group within its community – students, employees, or visitors. Academic and professional excellence can flourish only when each member of our community is assured an atmosphere of mutual respect. All members of the University community are responsible for the maintenance of an academic and work environment in which people are free to learn and work without fear of discrimination or discriminatory harassment. In addition, inappropriate amorous relationships can undermine the University's mission when those in positions of authority abuse or appear to abuse their authority. To that end, and in accordance with federal and state law, the University prohibits discrimination and discriminatory harassment, as well as inappropriate amorous relationships, and such behavior will be met with appropriate disciplinary action, up to and including dismissal from the University. Additionally, to protect the campus community, all non-confidential University employees (including faculty) are required to report sexual assaults, intimate partner violence, and/or stalking involving a student that they witness or are told about to the Office of Institutional Equity. The University takes all reports with the utmost seriousness. Please be aware that while the information you provide will remain private, it will not be confidential and will be shared with University officials who can help. More information is available at http://equity.uconn.edu and http://titleix.uconn.edu.

Sexual Assault Reporting Policy: To protect the campus community, all non-confidential University employees (including faculty) are required to report assaults they witness or are told about to the Office of Diversity & Equity under the Sexual Assault Response Policy. The University takes all reports with the utmost seriousness. Please be aware that while the information you provide will remain private, it will not be confidential and will be shared with University officials who can help. More information is available at http://sexualviolence.uconn.edu/.