# CIS*2030: Structure and Application of Microcomputers
# Computer Science, Fall 2020

**Instructor**: G. Grewal
**Office**: Reynolds 2208
**Phone**: x52630
**Email**: ggrewal@uoguelph.ca
**Course Web Page:** Available through CourseLink: https://courselink.uoguelph.ca

## General Description

This course introduces the topic of *Computer Architecture*. Computer architecture refers to those attributes of a computer system that are visible to a programmer, or put another way, those attributes that have a direct impact on the logical execution of a program. As a result of this course, students will gain insight into the operation of the major components of a computer system, assembly-language programming, and computer interfacing. The Motorola 68000 has been selected as the example architecture to study, due to the simplicity and clarity of its architecture, and the availability of a simulation tool that supports many facets of the 68000 architecture, including its data types, memory organization, registers, instruction set, address modes, processor states, exemptions and exception processing facility, memory-mapped I/O, and assembler language.

## Course Outline

Each bullet corresponds to *roughly* 1-3 weeks of the semester.

- Binary encoding and data representation: number systems, binary arithmetic, complement arithmetic, ranges, arithmetic overflow, other code systems, and fixed-point and floating-point representations
- Basic computer components, importance of Instruction-Set Architecture (ISA) role of memory versus registers, and simplified instruction cycle
- An assembler language view of the Motorola 68000's ISA
- Hardware support for high-level language (i.e., C): data types, data structures, loops, conditional statements, hardware stack, functions, parameter passing, return values, stack frames, local variables, recursion and re-entrant code
- Hardware support for Operating Systems: privileged states; exception processing
- Interaction among components: device operation, device control, bus signals, input/output protocols: polling, interrupts, direct-memory access
- Memory Hierarchy and Caching (if time permits)

## Textbook

- MacKenzie S., (1995). *The 68000 Microprocessor*, Prentice-Hall. (Required)
- Lab Exercises are provided online
- Additional readings (PDFs) are provided online

## Lectures

There are two course sections, each with *three* lectures per week: MWF 1:30pm to 2:20pm and 3:30pm to 4:20pm, respectively. To provide students with maximum flexibility, lecture material is pre-recorded and available on CourseLink. Discussions with the instructor will be held online every Monday during regular class hours, and will be managed through the Zoom interface available in CourseLink. These synchronous meetings will be used to make important announcements, answer questions, and review assignment solutions from the previous week.

## Recording Lectures and Labs in an Online World

Presentations which are made in relation to course work including lectures *cannot* be recorded, downloaded or copied without the permission of the presenter, whether the instructor, a classmate or guest lecturer. Material recorded with permission is restricted to use for that course unless further permission is granted.

## Course Evaluation

Your final grade will be determined as follows:

| Weight | Description | Notes |
|---|---|---|
| 24% | Weekly Homework Assignments<br>Uniform weighting<br>See semester schedule at end of this document for dates | Assignments may be handed in up to two days after the due date listed on the assignment without penalty. **No other extension will be granted**. |
| 40% | 8 Lab Exercises<br>Uniform weighting<br>See semester schedule at end of this document for dates | Labs are due at the end of your scheduled 3-hour lab section. However, labs can be handed in up to two days after this due date without penalty. **No other extension will be granted.** |
| 36% | Final Exam** | **Date:** December 10, 8:30am to 10:30am.<br>Online using Respondus. |
| Total Grade = 24% (Homework) + 40% (Labs) + 36% (Exam) | | |

**You must achieve a passing grade on the final exam (i.e., 18 out of 36). Failure to satisfy the previous case will result in the following calculation for your final grade: Final Grade = MIN(45, Total Grade). Otherwise, Final Grade = Total Grade.

## Labs

All labs will be held in a virtual room through the Zoom interface available in CourseLink. Prior to your scheduled lab, you will receive a Zoom invitation to enter the virtual lab room at the scheduled time. This invitation should not be shared with others. Students can only attend their own scheduled lab section, not other lab sections. Therefore, do not ask the teaching assistants to allow you to move between lab sections – even temporarily. The answer will be no.

Prior to attending your scheduled lab section, you should have completed most, if not all, of the lab exercise on your own. Any outstanding questions can then be targeted to the lab instructor for clarification. Keep in mind that you are expected to submit your completed lab electronically at the end of the scheduled lab, so please come prepared.

Lab exercises will be completed using Easy68K – a Windows based 68000 assembler and simulator. This software is accessible through the School of Computer Science Windows' servers. Please visit the School's wiki (wiki.socs.uoguelph.ca) if you require technical help.

## Learning Outcomes

Successfully completing the homework and laboratory assignments in this course will contribute to the following learning outcomes:

1. Explain the digital representation of numeric and non-numeric data.
2. Identify the main abstractions that exist between programs and the actual hardware they run on, explain why these abstractions exist, and describe how they build upon each other.
3. Create assembly-language programs to implement some of the main abstractions and evaluate your program through testing and debugging.
4. Implement interfaces with external devices using common input-output strategies.
5. Describe the memory hierarchy and list tradeoffs

## Teaching Assistants

Contact information and advising times for the Teaching Assistants (TA) is posted on CourseLink. Discussions with the TAs will take place either through the Zoom facility that is part of CourseLink or via email. All requests for re-grades must be made by email to the TA who marked your assignment/lab within one week of the grade being posted on CourseLink. It is your responsibility to be vigilant and check your grades regularly.

## Electronic Mail

As per university regulations, all students are required to check their mail.uoguelph.ca e-mail account regularly: e-mail is the official route of communication between the University and its students.

## Considerations and Constraints for Online Learning

- Do not redistribute recorded interactive discussions that involve your instructor, teaching assistant or classmates. This includes advising times, lab times, and question and answer sessions with the instructor.

- Online activities such as advising times, question and answer sessions, and interactive lectures may be recorded by the instructor or TAs and posted to Courselink. By taking this

course you are agreeing that your participation in these activities can be used in this manner. If you do not wish to have your image or voice recorded as part of these activities then either do not take this course or do not ask verbal questions during these activities.

- A reliable internet connection that is sufficient for online learning is necessary for this course. If you do not have a sufficiently fast and reliable internet connection, you may not be able to view lectures or other course material. It may also not be possible to attend labs or online advising with teaching assistants or the instructor.

- This course is offered in the Eastern Standard Time (EST) zone. While taking this course you will be required to attend online activities such as advising times or labs between 8:30am and 8:20pm EST.

- Keep copies of assignments which you have submitted. You may be asked to resubmit assignments at a later time.

- Do not upload any course material (Intellectual Property) to other sites on the internet, including Chegg and CourseHero, or share course material in any other way. To do so will be treated as a violation.

## A Word of Caution

Needless to say, plagiarism in any form must be dealt with severely. *There is no group work in this course. Therefore, when answering questions do it yourself. Be original.* All submitted items will be checked for plagiarism, as well as for uploads to websites, like Chegg, in search of answers. All cases of academic misconduct are handled by the Dean, in conjunction with the Department Chair. Successive infractions of misconduct affirmed by this process could have consequences as serious as expulsion from the University. *(It is your responsibility to acquaint yourself with the definitions and ramifications of academic misconduct as described in the university's undergraduate Calendar.)* The risks are sufficiently great that they are not worth taking. If you are having trouble, please see the teaching assistant or the instructor for help. Moreover, if you are not sure whether a potential action is appropriate, check either with the instructor or Greg Klotz – the undergraduate faculty advisor for the School of Computer Science.

## Semester Schedule

There are several important dates this semester:

- Friday, September 11, 2020 class commences.
- Monday, October 12, 2020 no classes are scheduled due to Thanksgiving holiday 😊
- Tuesday, October 13, 2020 no classes are scheduled due to Fall study break 😊
- Friday, December 4, 2020 classes end.

Over the course of the semester, there are 12 assignments and 6 labs. Table 1 and Table 2 below show the due date and the topic of each assignment and lab, respectively.

**Table 1: Assignments.**

| Week | Due | Assignment Topic |
|---|---|---|
| 1 | September 18, 2020 | Binary Representations |
| 2 | September 25, 2020 | Computer Architecture |
| 3 | October 2, 2020 | Assembly Language |
| 4 | October 9, 2020 | Addressing modes |
| 5 | October 16, 2020 | Control Flow |
| 6 | October 23, 2020 | Logical and Shift Operations |
| 7 | October 30, 2020 | Runtime Stacks |
| 8 | November 6, 2020 | Subroutines and Stack Frames |
| 9 | November 13, 2020 | Exceptions and Exception Processing |
| 10 | November 20, 2020 | Computer Buses |
| 11 | November 27, 2020 | Input-Output |
| 12 | December 4, 2020 | Memory Hierarchy |

**Table 2: Labs.**

| Week | Due | Lab Topic |
|---|---|---|
| 1 | - | - |
| 2 | Week 2 | Introduction to Easy68K |
| 3 | Week 3 | Data Types and Arithmetic Operations |
| 4 | Week 4 | Assembler Directives and Address modes |
| 5 | - | - |
| 6 | Week 6 | Program Control Flow |
| 7 | Week 7 | Runtime Stacks and Functions |
| 8 | Week 8 | Stack Frames and Recursion |
| 9 | Week 9 | Operating Modes and Exceptions |
| 10 | Week 11 | Memory-Mapped I/O |
| 11 | - | - |
| 12 | - | - |