



ECE 255 Architecture Laboratories



You are expected to read each lab experiment manual carefully and prepare in advance of your lab session.

Pay particular attention to the parts that are **bolded and underlined**. You are required to address these parts in your lab report.

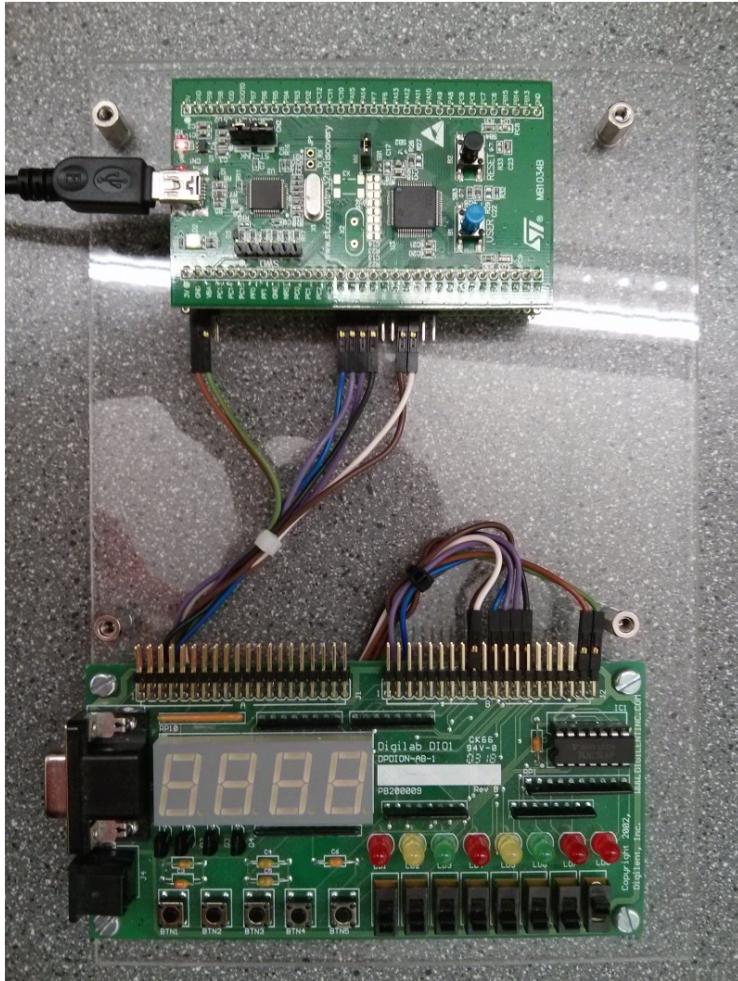
In particular, all items in the **Prelab** section must be prepared in a **written form** before your lab.

You are required to submit your written preparation during the lab, which will be graded by the lab instructor.

The instructor may ask you questions for grading purpose.



Hardware: STM32 Microcontroller



ECE 255 Lab Web Page:
<https://www.ece.uvic.ca/~ece255/lab/>



LAB 0: Introduction to Eclipse



Eclipse for STM32F0

an integrated development environment (IDE) that
can be used to develop applications with various
programming languages



Using Eclipse



1. Create a Blinky C Project
2. Build the Project
 - Convert source code to executable binaries
3. Configure Debugging
4. Create an ARM Assembly Project
5. Tips for using Eclipse
6. Examine Memory



LAB 1: Using Eclipse



Objectives:

1. Assemble and link ARM architecture assembly programs.
2. Load program onto an ARM development board.
3. Run and debug programs.
4. Examine / Modify memory locations (including program and data) and registers.
5. Set breakpoints and single step execution.



LAB 1: PreLab



Prelab (In addition to your written submission, your Lab Instructor may ask you these questions during the lab and your answers will be graded individually.)

- What is a cross-assembler?
- What is an exception?
- Comment on the program in 1.4.2 Part 2 (Explain what it does).
- Comment on the program in 1.4.3 Part 3 (Explain what it does).



Part 1



This part introduces how to import an existing project to Eclipse, and then how to view and read the memory after downloading the executable binary file to the ARM board.

Exercise: Change the content of the memory with address “0x20000146” to a “SUBTRACT” instruction and run the program. What are the content of the registers and why?



Part 2



Figure. 1.6

Exercise: Describe the function of this program



Part 3



Deliverable for Part 3:

- Description of the program
- Program Listing
- Snap shots of the data section before and after execution of the program
- The contents of the memory and r3, r4, r5, and r0 registers at the stop breakpoint
- Change the program in such a way that it performs its operation for 32 elements. Is there any limitation in the provided code?



In-Class Exercise



Given an array with Max number of elements.

Write a program to add all the elements from 1 to Max.