

Computer Science 230
Computer Architecture and Assembly Language
Spring 2023

Assignment 1

Due: Thursday, February 9, 11:55 pm by Brightspace submission
(Late submissions **not** accepted)

Programming environment

For this assignment you must ensure your work executes correctly on the simulator within Microchip Studio 7 as installed on workstations in ECS 249. If you have installed AVR Studio on your own then you are welcome to do much of the programming work on your machine. (The IDE is available at no charge from <https://bit.ly/311ENk7> but comes only in the Microsoft Windows flavor.) **You must allow enough time** to ensure your solutions work on the lab machines.

Individual work

This assignment is to be completed by each individual student (i.e., no group work). Naturally you will want to discuss aspects of the problem with fellow students, and such discussion is encouraged. **However, sharing of code fragments is strictly forbidden without the express written permission of the course instructor.** If you are still unsure regarding what is permitted or have other questions about what constitutes appropriate collaboration, please contact me as soon as possible. (Code-similarity analysis tools will be used to examine submitted work.) The URLs of significant code fragments you have found and used in your solution must be cited in comments just before where such code has been used.

Objectives of this assignment

- Understand short problem descriptions for which an assembly-language solution is required.
- Use AVR assembly language to write solutions to three such small problems.
- Use AVR Studio to implement, simulate and test your solution. (We will not need to use the Arduino mega2560 boards for this assignment.)
- Hand draw a flowchart corresponding to your solution to the second problem.
- **You are not to use AVR functions in this assignment.**

Problem #1: Adding 16-bit values

$$Z \leftarrow X + Y$$

$R_0 \leftarrow 1$ indicates an overflow, 0 otherwise

In this exercise, two 16-bit values stored in X ($R_{27}:R_{26}$) and Y ($R_{29}:R_{28}$) to be added and the result to be stored at Z ($R_{31}:R_{30}$). When overflow occurs, R_0 should contain a value of one, zero otherwise.

Problem #2: Even Parity

In this exercise, a seven-bit value given in XL (R_{26} , $b_6..b_0$ with b_7 value is ignored). Your task is to write an assembly code that sets b_7 when the number of ones in XL ($b_6..b_0$) is odd and clears b_7 otherwise. The result to be stored in ZL (R_{30}). For example,

ZL \leftarrow 1001 1111 when XL = ?001 1111; here XL has 5 ones

ZL \leftarrow 0101 1001 when XL = ?101 1001; here XL has 4 ones

You must also draw by hand the flowchart of your solution and submit a scan or smartphone photo of the diagram as part of your assignment submission; please ensure this is a JPEG or PDF named "assignment1-flowchart.jpg" or "assignment1-flowchart.pdf" depending on the file format you have chosen. (Please: No BMPs or Word files! We beg of you!)

What you must submit

- A scanned flowchart of your solutions (or smartphone photo) showing your name and V-number.
- Your solution for problem #1 and problem #2 copied in the provided assignment1.asm file. The copied code should not have an infinite loop and to be inserted between START and STOP marks for each problem. Your name and V-number should be written in the submitted file.
- Do not change the name of the file!
- **Please do not submit any other AVR 7 Studio project files.**
- **Please do not submit ZIP files.**
- Your work must use the provided .asm file. Any other kinds of solutions will not be accepted. (Again: Do not submit ZIP files containing the Microchip projects!)
- The scan / smartphone photo of your hand-drawn flowchart with the name of "assignment1-flowchart.jpg" or "assignment1-flowchart.pdf" depending on the format you have chosen.

Evaluation

- 3 marks: Problem #1 solution code.
- 4 marks: Problem #2 solution code.
- 2 marks: Hand-drawn flowchart for Problem 2.
- 1 mark: All submitted code is properly formatted (i.e., indenting and comments are suitably used), and files correctly named.

Total marks: 10