

# Pre-lab Assignment 1

Due: Wed, 02 Sep 2020 23:59:59 (approximately 99 days ago)

[Score: 10 / 10 points possible]

Do a little bit of reading about ARM Cortex-M0 microprocessors so that you will be ready to do the lab experiment.

---

## Academic Honesty Statement [0 ... -100 points]

By typing my name, below, I hereby certify that the work on this prelab is my own and that I have not copied the work of any other student (past or present) while completing it. I understand that if I fail to honor this agreement, I will receive a score of zero for the lab, a one letter drop in my final course grade, and be subject to possible disciplinary action.



---

### (1) [1 point]

Write the instruction to add the contents of the R3 register to the constant value 0x4d and write the result to the R3 register. **Specify the instruction using syntax you would use for an assembler.** Try it in the simulator to make sure it assembles.



---

### (2) [1 point]

Write the instruction to add the contents of the R2 register to the constant value 0x3 and write the result to the R1 register. **Specify the instruction using syntax you would use for an assembler.** Try it in the simulator to make sure it assembles.



---

### (3) [1 point]

For the instruction you specified in (2), what is the largest constant you could add to R2?



### (4) [1 point]

Look at the ARMv6-M Architecture Reference Manual, section A6.7.2 and write the encoding for the instruction with maximum-sized operand that you listed in question (3). Express your answer as a four-digit hexadecimal number.



### (5) [1 point]

There is no instruction to multiply the value of a register by a constant. Write a two-instruction sequence that multiplies the value of register R2 by 0x3f and writes the result to register R1. Make sure that the only register modified by these instructions is R1. **Specify the instructions using syntax you would use for an assembler.** Try them in the simulator to make sure they assemble.

```
movs r1, #0x3f
muls r1, r1, r2
```



### (6) [1 point]

Write a multi-instruction sequence that adds the constant 0x4e to R0 and then multiplies it by the contents of R1. In other words,  $R0 = (R0 + 0x4e) * R1$ . Do not modify any register other than R0 with these instructions. **Specify the instructions using syntax you would use for an assembler.** Try them in the simulator to make sure they assemble.

```
adds r0, #0x4e
muls r0, r0, r1
```



### (7) [1 point]

What is the maximum immediate value that can be "mov"ed into a register?  
E.g., for the instruction "MOVS R0,#n" what is the largest number that you can write for n?



**(8) [1 point]**

Look at the ARMv6-M Architecture Reference Manual, section A6.7.39 and write the encoding for the instruction with maximum-sized operand that you listed in question (7). Express your answer as a four-digit hexadecimal number.

**(9) [1 point]**

What is the instruction encoding of an instruction that adds the value 8 to register R1 and deposits the result in register R1? Express your answer as a four-digit hexadecimal value. (You can and should use the simulator or System Workbench to verify your result, but you need to know how to look this up in the appropriate documentation and determine the value manually for this and any other instruction.)

**(10) [1 point]**

Write the assembly language specification for the instruction represented by the 16-bit hexadecimal code 3809. In other words, write the assembly language phrase that will cause the assembler to create the instruction with that hexadecimal code. (You can and should use the simulator or System Workbench to verify your result, but you need to know how to look this up in the appropriate documentation and determine the value manually for this and any other instruction.)

