“This Assignment is not a group project. I am the alone contributor. Please do not include it’s grading as group.”

FECS ASSIGNMENT

BY

SHUBHAM KUMAR

18IE10021

PROBLEM STATEMENT:

Assume that the data memory locations $244-$247 contain the following data:

                                $244= $ 4D

                                $245=$CB

                                $246=$B6

                                $247=$F1

Write program in Assembly and C (assuming signed numbers) to:

(i) Add all the (above) four numbers together and place the result in locations $210 and $211

(ii) Subtract content of $245 location from content of $244 followed by subtracting content $247 from $246. Thereafter, multiply the results of subtraction and store in the locations $212 to $215

(iii) Also, divide the results of subtraction with each other (first result of subtraction as numerator and second result of subtraction as denominator) and place the results in location starting $80.

My solution:

1. Total Execution time of .asm Code : 454 cycles = 454 \* 0.125µs = 56.75µs
2. Total Execution time of .c Code : 670 cycles = 670 \* 0.125µs = 83.75µs.
3. Separate execution time of instruction is not provided as it can be seen from Atmega32 datasheet, and a matter of copy paste.
4. Important aspects of my solution and problems encountered:

* No instructions were already present in AVR assembly for signed operation, except that of MULS.
* I had taken several cases and manipulate registers according to S and N flags for signed implementation, Branching was readily used on these flags.
* Also, we have operated on 16-bit using 8-bit multipliers, 16 bit addition and subtraction.

That too in signed assumption.

* I have to use 16bit X 16bit = 32bit signed multiplication standard algorithm in .asm
* Also, 16bit signed division was implemented using several 8-bit instructions.
* C program was pretty simple due to presence of 8bit(char), 16bit(int) and 32bit(long) datatype to handle and operate on multiple bytes.

Program is presented below: