

ELEC 2220 Computer Systems
Homework #10
Due: Friday, June 18

PROGRAM 1

Design a program to compute $A1 = B1 + C1 - D1$, where all variables are 32-bit binary numbers. You may ignore overflow conditions for this exercise. Define the values of A1, B1, C1, and D1 as 32-bit (4-byte) numbers, stored in “big-endian format”, using “dc.b” directives. (Recall that the proper storage convention for multi-precision numbers is for the most significant byte to be stored at the lowest address.)

Example: `A1 db $01,$27,$3f,$44 ;A1 = $01273f44`

Run the program two times, using the following values.

FIRST RUN:

A1 = undefined initially
B1 = \$12345678
C1 = \$4567bcde
D1 = \$2345ef01

SECOND RUN:

\$fabcd1
\$ffffff
\$eeeeeee

Print one copy of your program and the two Code Warrior debug windows, circling or highlighting the values of the four variables after each run.

Notes:

1. Multi-precision arithmetic requires the use of add-with-carry and subtract-with-carry.
2. If adding, an entire multi-precision result must be computed before moving on to another operation.
3. Consider using loop structures to do the operations needed for the 32-bit add and the 32-bit subtract.

PROGRAM 2

Given five two-element arrays, A1, B1, C1, D1, and E1, design a program to compute

$$A1[i] = (B1[i]*D1[i]) + (C1[i]/E1[i]) - E1[i]^2 \quad \text{for } i = 0 \text{ and } 1.$$

Note that $E1[i]^2$ signifies $E1[i]$ squared.

All variables are 16-bit signed binary numbers. Use the following initial data values (shown here as decimal values).

A1 = undefined
B1 = 1000, -1000
C1 = -30000, 30000
D1 = 10, -10
E1 = -100, 100
i = undefined

Print one copy of your source program and the Code Warrior debug window, showing the final values of all variables **in decimal**.