## Quiz 2 <br> ECE-3120: Fall ‘08

1) Write an instruction sequence to add two 16 bit numbers stored at $\$ 2000-\$ 2001$ and \$2002~\$2003 and subtract the 16 bit number stored at $\$ 1000$ from the sum. Store the result in \$\$2100

| org | $\$ 1000$ | $;$ starting address of the program |
| :--- | :--- | :--- |
| ldd | $\$ 2000$ | $; \mathrm{D} \leftarrow \mathrm{m}[\$ 2000]$ |
| addd | $\$ 2002$ | $; \mathrm{D} \leftarrow \mathrm{m}[\$ 2002]$ |
| subd | $\$ 1000$ | $; \mathrm{D} \leftarrow \mathrm{m}[\$ 1000]$ |
| std | $\$ 2100$ | $; \mathrm{D} \leftarrow \mathrm{m}[\$ 2100]$ |

2) Write a sequence of instructions to compute the sum of N signed 8 -bit numbers stored at memory locations $\$ 1000$ and store the result in $\$ 2000$

| N | equ | 5 | ;array count |
| :---: | :---: | :---: | :---: |
|  | org | \$1000 | ;starting address of the array |
| array: | db | 1,2,3,4,5 |  |
|  | org | \$2000 | ;starting address of the array |
| $\begin{aligned} & \text { sum: } \\ & \text { i: } \end{aligned}$ | rmb | 1 | ;array sum assuming we need just 1 byte |
|  | rmb | 1 | ;array index |
|  | org | \$1500 | ;starting address of the program |
|  | ldaa | \#0 |  |
|  | staa | i | ;initialize loop (array) index to 0 |
|  | staa | sum | ;initialize sum to 0 |
| loop: | ldab | i |  |
|  | cmpb | \#N | ;is $\mathrm{I}=\mathrm{N}$ ? |
|  | beq | done | ;if done, then branch |
|  | ldx | \#array | ;use index register X as a pointer to the array |
|  | ldab | 0,x | ;place array[i] in B |
|  | ldy | sum | ;place sum in Y |
|  | aby |  | ;sum¢sum+array[i] |
|  | sty | sum | ;update sum |
|  | inc | i | ;increment loop counter by 1 |
|  | bra | loop |  |
| done: | swi |  | ;return to D-Bug12 monitor |
|  | end |  |  |

3) Write a sequence of instruction to toggle the upper four bits of an 8-bit number stored at memory location $\$ 1000$. Also write instructions to clear the lower four bits of the same number [e.g:- $10101010 \rightarrow 01010000$ ]

| Idaa | $\$ 1000$ | ;D $\leftarrow m[\$ 1000]$, loading the number to be <br>  <br> eora |
| :--- | :--- | :--- |
| $\# \$ F 0$ | ;Tonipulated in D |  |
| anda | $\# \$ F 0$ | ;clears the lower 4 bits in A |

## Logic

Example: $A \leftarrow 10111100$

| ldaa | $\$ 1000$ | ;A $\leftarrow 10111100$ |
| :--- | :--- | :--- |
| eora | $\# \$ F 0$ | ;A XOR \$F0 $\rightarrow 10111100$ |
|  |  | $; \quad$ XOR $11110000 \rightarrow 01001100$ |

So after eora A $\leftarrow 01001100$ (finished toggle part)
anda \#\$F0 ;A AND \$F) $\rightarrow 01001100$
; AND $\rightarrow 11110000 \rightarrow 01000000$
So after anda $\mathrm{A} \leftarrow 01000000$ (finished clearing lower 4 bits)

