EE 308/MENG483 – Homework 2

1. Consider the following program:

```
prog:   equ     $2000           ; Starting address from program
data:   equ      $1000          ; Starting address for data
        org     prog            ; Set initial program counter value
        ldx     #1234           ;
        ldab    #235
        abx                     ;
        stx     result          ;
        swi
        org     data            ; Put data starting at this location
resut:  ds.w    1               ;
```

(a) Hand-assemble the program. That is, figure out what the op codes of the instructions are and where they will be located in memory, the addressing mode, number of cycles, and the status of the NVCZ bits.

(b) How many cycles will it take the MC9S12 to execute this program. (Do not include the swi instruction.)

(c) How long will it take an MC9S12 with a 24 MHz Eclock to execute this program?

2. Consider the following program:

```
; MC9S12 program to copy a table of data from one location to another
; The copied data is the negative of the original data

prog:   equ     $2000           ; Starting address from program
data:   equ     $1000           ; Starting address for data
count:  equ     8               ; 8 elements in the table
        org      prog           ; Set initial program counter value
        ldab     #count         ; ACCB keeps count of number to transfer
        ldx      #table_1       ; X points at table_1
        ldy      #table_2       ; Y points at table_2
repeat: ldaa     1,X+           ; get data from table_1, X points to next element
        nega
        staa     1,Y+           ; save into table_2, Y points to next element
        decb                    ; Decrement counter
        bne      repeat         ; If not done, continue with next element
        swi
```

```
        org     data            ; Put data starting at this location
table_1: dc.b   $44,$61,$74,$61,$20,$54,$61,$62
table_2: ds.b   count           ; Reserve count bytes of memory for results
```

(a) Hand-assemble the program. That is, figure out what the op codes of the instructions are, where they will be located in memory, and the addressing mode.

3. Write an instruction sequence to set the upper four bits of the number at address $0050 to 0, and leave the lower four bits unchanged.

4. Consider the following program fragment:

```
        ldy     #5000
loop1:  ldx     #5000
loop2:  dbne    x,loop2
        dbne    y,loop1
        swi
```

(a) Hand assemble the program. (Add an org assembler directive to put the program in memory starting at address 0x2000.)

(b) How many instruction cycles will it take the MC9S12 to execute the program? (Do not consider the swi instruction.)

(c) How many seconds will this take the MC9S12 with an 24 Mhz E-clock? (You should give the answer to the nearest millisecond.)