

EECS 373 Fall 2014 Homework #2

Due September 23rd at the start of class.

Name: _____ unique name: _____

You are to turn in this sheet as a cover page for your assignment. The rest of the assignment should be stapled to this page. *Assignments that are unstapled, lack a cover sheet, or are difficult to read will lose at least 50% of the possible points and we may not grade them at all.* This is an individual assignment; all work should be your own.

1. Using the ARMv7-M Architecture Reference Manual.

- In straightforward English, explain what the LSL (immediate) instruction does. **[4 points]**
- Write the hexadecimal for the machine code you would expect to get for the following instructions.

[6 points, 2 each]

- LSL R2, R3, #3
- LSL R2, R3, #23
- LSL R11, R3, #3

2. For each of the following program segments, assume you start with all memory locations in question equal to zero. Indicate the values found in these memory locations when the programs end. Write all answers in hex. **[10 points, 5 for each part]**

a. `BASE_EMC = 0x74000000;`
`uint32_t *x = (uint32_t*)BASE_EMC;`
`*x = 0x76543210;`
`*(x-1) = 0x89abcdef;`

Base Address	00	01	02	03
0x74000004				
0x74000000				
0x73FFFFFFC				
0x73FFFFFF8				

```

b. mov  r2, #100
   movw r1, #255
   movt r1, #15
   strb r1, [r2, 2]!
   str  r1, [r2], #1
   strh r2, [r2, #-3]

```

Address	Value
100	
101	
102	
103	
104	
105	
106	
107	

Hint: The ARMv7-M Architecture Reference Manual may be useful here.

3. Write a program in UAL assembly that does the same thing as the following C code. You should assume “print” is an ABI compliant function that takes a single integer argument and does something with it. Have the main return to whatever called it just as any function might. Do not use IT statements or conditional instructions (e.g. ADDNE). **[10 points]**

```

void main() {
    int i, a=1;
    for(i=0; i<5; i++) {
        a = a + i;
        print(a);
    }
}

```

4. Write a program in C that does the same thing as the following ARM assembly language code. Your C code must not exceed three lines. **[10 Points]**

```

movw r0, #0030
movt r0, #2008
ldr  r1, [r0]
add  r1, r1
str  r1, [r0]

```

5. Write an ABI compliant function named “mean” which takes four integer arguments and returns the average of the four. Do not worry about fractions or overflow. **[10 points]**