

Homework 2

0. If you haven't already, download and install uVision software. Set up and debug via simulation question 5 below.

1. Regarding the Cortex M3

(a) How many bits in a word, and half-word?

(b) If BLT is not a tasty sandwich, what is it? What bits does it look at?

(c) How do we make an instruction load the flags?

(d) Under what circumstances will the BAL instruction cause a branch to be taken?

2. Consider the assembly code below. When all of these instructions have been executed, what is the hexadecimal value stored in registers R0 through R5? If you feel that any of these registers would be undefined, or any of the instructions is not legal, explain your answer.

```
1    MOVW r2, #0x5050
2    MOVT r2, #0x5050
3    MOVW    r1,    #0x a b c d
4    MOVT    r1,    # 1 2 3
5    MOV    r3,    #1
6    MOV    r5,    # b e e f
7    MOVT    r4,    # 1 2 3
8    MOVW    r4,    #0x a b c d
```

3. Consider the Thumb2 assembly code below. When all of these instructions have been executed. . .

(a) What is the hexadecimal value stored in registers R0 through R7? If you feel that any of these registers would be undefined, or any of the instructions is not legal, explain your answer.

(b) What are the states of the N and Z flags in the PSR?

```
1    MOV r0, #1
2    MOV r1, #3
3    MOV r2, #5
4    MOVW r7, #0xFFFF
5    MOVT r7, #0xFFFF
6    ADD r3, r1
7    ADD r4, r0, r2
8    ADD r5, r1, r7
9    ADDS r6, r7, r0
```

4. With the following assembly, state the values of R0-R5 after execution.

```
1    movw R0, #0xABCD
2    movt R0, #0x1234
3    movw R1, #0x0000
4    movt R1, #0x0000
5    str R0, [R1]
6    movw R2, #0x5678
7    movt R2, #0x90EF
8    str R2, [R1, #2]
9    ldr R3, [R1, #2]
10   ldrb R4, [R1, #1]
11   ldrsh R5, [R1, #6]
```

5. Write assembly code for lab1 part 3. Simulate it on your own computer.