

ELEC 2220 Homework #5

Due Friday, June 1

PART 1

1. In the ARM program on the next page, a block of “constant data” has been defined, which will be placed by the assembler in the CODE area in memory, following the codes for the instructions. Assuming that program code begins at address 0x20000000, the first data value will be placed by the assembler at address 0x2000002C.
 - Assembly directives “dcd”, “dcb”, and “space” are described in section 3.6 of the text.
 - Program instructions are described briefly in section 3.5.
 - Instructions “ldr”, “mov”, and “movt” are described in sections 5.1 and 5.3.
2. Using the format of Table 1.3 in the text, but with a **separate row for each byte**, sketch a diagram of the memory allocated by the assembler for the constant data part of the test program (locations pp through tmp), beginning at address 0x2000002C. In each row of the table, enter the address of that byte of memory and the value that will be stored in that byte. Also, next to the appropriate rows, indicate which byte of memory corresponds to each symbolic label. Use **hexadecimal form** for each number.
3. For each of the CPU instructions in the program, determine the 32-bit hex values that you predict would be loaded into the indicated registers as the program executes, and enter those on the lines to the right of the instructions.
4. Create a new ARM project, enter the program, assemble it, load it into the simulator with the debugger.
 - a. Use the Memory window to check your answers for Step 2 above.
 - b. Single-step through the program in the debugger to check your answers from Step 3 above. Next to your answers, record the register values observed in the debugger, and comment on any values that are different from what you found in part 3,.

Submit the table from Step 2 and a printout of the program with your values from parts 3 and 4 entered on the lines provided.

(Test program on next page.)

```

; Test program
    AREA RESET, CODE
    THUMB
    ENTRY

Main                                ;Register contents after instruction:
    mov    r0,#10                    ; r0 = _____
    movt   r0,#23                    ; r0 = _____
    ldr    r1,=pp                    ; r1 = _____
    ldr    r2,[r1]                   ; r2 = _____
    add    r3,r2,r0                  ; r3 = _____
    ldr    r1,=fr                    ; r1 = _____
    ldr    r2,=ir                    ; r2 = _____
    ldr    r3,[r2]                   ; r3 = _____
    ldr    r4,=reg                   ; r4 = _____
    ldr    r5,[r2]                   ; r5 = _____
    ldr    r6,[r2,#4]                ; r6 = _____
    ldr    r7,[r2,#8]                ; r7 = _____
    mov    r8,#-5                    ; r8 = _____
    add    r9,r7,r8                  ; r9 = _____
    sub    r0,r7,r8                  ; r0 = _____
Here    b    Here                    ;Effectively "halts" the program

; Data defined in code memory, so that initial values will be stored
pp    dcd    0x12345678
ir    dcb    5,6,7,8
ts    dcd    0x0805
fr    dcb    'A'
reg   dcd    18,-20,15,-10
mem   space 32
tmp   dcd    0x1234,0x5678

; No DATA area for this program – above data is in the CODE area

    END

```