

National Sun Yat-Sen University
ASSEMBLY LANGUAGE AND MICROCOMPUTER
Homework #3
Tested in 12/25/2014

1. Find out the 32-bit instruction coding for the following ARM instructions:
 - (a) SUBLSS r3, r2, r1, LSR #7
 - (b) STRB r5, [r1, r0, LSL #2]
 - (c) STREQ r4, [r2, #-3]!
 - (d) LDMED r13!, {r0-r2, r14} (Refer to page 130 of the textbook.)
 - (e) MULNE r1, r2, r3 (Refer to page 122 of the textbook.)
 - (f) LDRH r3, [r2], #6

Note: Coding for the “shift type”

Coding	Shift type
00	LSL
01	LSR
10	ASR
11	ROR

2. Please show the contents of the little-endian based memory for the following ARM assembly code. Assume the code will be placed in the memory starting from 0x00010000.

```
        AREA    HelloW, CODE, READONLY ; declare code area
SWI_WriteC EQU    &0                ; output character in r0
SWI_Exit   EQU    &11               ; finish program
        ENTRY
        ; code entry point
START    ADR     r1, TEXT            ; r1 -> "Hello World"
LOOP     LDRB    r0, [r1], #1        ; get the next byte
        CMP     r0, #0              ; check for text end
        SWINE   SWI_WriteC          ; if not end print ..
        BNE     LOOP                ; .. and loop back
        SWI     SWI_Exit             ; end of execution
TEXT     =       "Hello World", &0a, &0d, 0
        END
        ; end of program source
```

3. Exercise 5.1.2 in page 150 of the textbook. (Write a subprogram which copies a string of bytes from one memory location to another using the 'loop unrolling' technique. The start of the source string will be passed in r1, the length (in bytes) in r2, and the start of the destination string in r3. Assume both source and destination strings are word-aligned, and the string is a multiple of 16 bytes long. Your program has to follow the APCS standard.)
4. Do Example 6.3 and Exercise 6.3.1 of the textbook.
5. Explain why the return from the IRQ exception has to use the instruction **SUBS pc, r14, #4** instead of **MOVS pc, r14**.