EEE3002 Microprocessors Tutorial 4 (2014 Sem 1)

1) Describe the contents of register r13 after the following instructions complete, assuming that memory contains the values shown below. Register r0 contains 0x24, and the memory system is little-endian.

Address	Contents
0x24	0x06
0x25	0xFC
0x26	0x03
0x27	0xFF

   a) LDRSB r13, [r0]
   b) LDRSH r13, [r0]
   c) LDR r13, [r0]
   d) LDRH r13, [r0]

2) Calculate the effective address of the following instructions if register r3 = 0x4000 and register r4 = 0x20:
   a) STRH r9, [r3, r4]
   b) LDRB r8, [r3, r4, LSL #3]
   c) LDR r7, [r3], r4
   d) STRB r6, [r3], r4, ASR #2

3) Write a program that sums word-length values in memory, storing the result in register r2. Include the following table of values to sum in your code:

```
TABLE   DCD   0xFEBBA, 0x1234, 0x8888
        DCD   0x13, 0x8080808, 0xFFFF
```

4) Assuming you have a little-endian memory system, what would register r4 contain after executing the following instructions? Register r6 hold the value 0xBEEFFACE and register r3 holds 0x8000.

```
    STR   r6, [r3]
    LDRB  r4, [r3]
```

   What if you had a big-endian memory system?

EE3002 Microprocessors Tutorial 4 Solutions

1) [r0] refers to a memory address starting from 0x24 since r0 contains 0x24
   a) LDRSB r13, [r0]
      This instruction loads a Sign Byte (SB) from address 0x24 into r13
      Hence r13 contains 0x06

   b) LDRSH r13, [r0]
      This instruction loads a Sign Halfword (SH) from address 0x24 into r13
      Note that the Halfword is 0xFC06 which is a negative number. Sign extension has to be applied to the 16-bit result to form a 32-bit result.
      Hence r13 contains 0xFFFFFC06

   c) LDR r13, [r0]
      This instruction loads a 32-bit number from address 0x24 into r13
      Hence r13 contains 0xFF03FC06

   d) LDRH r13, [r0]
      This instruction loads a Halfword from address 0x24 into r13
      Hence r13 contains 0xFC06

2) The effective address can be computed from whatever within [..].
   a) STRH r9, [r3, r4]
      The instruction store a halfword in r9 into the effective address given by the sum of the base register r3 (0x4000) and offset register r4 (0x20). Shorthand notation : ea<r3+r4>
      Effective address = 0x4020

   b) LDRB r8, [r3, r4, LSL #3]
      This instruction loads a byte from ea<r3+r4*8> into r8
      Effective address = 0x4100

   c) LDR r7, [r3], r4
      This is a post-index instruction. The instruction will load ea<r3> into r7 and add r4 to r3 after the load instruction.
      Effective address = 0x4000 but r3 becomes 0x4020

   d) STRB r6, [r3], r4, ASR #2
      This is another post-index instruction. The instruction will store content of r6 into ea<r3> and add r4/4 to r3 after the load instruction.
      Effective address = 0x4000 but r3 becomes 0x4008

3)

```
        AREA Exercise3, CODE
        ENTRY
                ADR     r0,     TABLE       ; r0 is pointer to TABLE
                MOV     r1,     #6          ; r1 is length of TABLE
                MOV     r2,     #0          ; r2 is sum
loop            LDR     r3,     [r0],   #4  ; load TABLE value and
                                            ; post-increment pointer
                ADD     r2,     r2,     r3  ; sum = sum + r3
                SUBS    r1,     r1,     #1  ; decrement counter
                BNE     loop

stop            B       stop

TABLE           DCD   0xFEBBA, 0x1234, 0x8888
                DCD   0x13, 0x8080808, 0xFFFF

                END
```

4) r6 contains 0xBEEFFACE and r3 contains 0x8000
After the instruction, STR r6, [r3], a little-endian memory system will be as follows:

Address	Contents
0x8000	0xCE
0x8001	0xFA
0x8002	0xEF
0x8003	0xBE

The instruction, LDRB r4, [r3] will load **0xCE** into r4

After the instruction, STR r6, [r3], a big-endian memory system will be as follows:

Address	Contents
0x8000	0xBE
0x8001	0xEF
0x8002	0xFA
0x8003	0xCE

The instruction, LDRB r4, [r3] will load **0xBE** into r4