CS1021 Tutorial 6

Reading and Writing to Memory

Q1 if a, b and c are 32bit signed integers stored at memory addresses 0x40000000, 0x40000004 and 0x40000008 respectively, write ARM assembly language instructions to compute:

(i) a = a + b
(ii) c = a – b

```
LDR     R0, =0x40000000         ; R0 -> a
LDR     R0, [R0]                ; R0 = a
LDR     R1, =0x40000004         ; R1 -> b
LDR     R1, [R1]                ; R1 = b
SUB     R0, R0, R1              ; R0 = a - b
LDR     R1, =0x40000008         ; R1 -> c
STR     R0, [R1]                ; c = R0 = a - b
```

(iii) b = a * a

Q2 If a, b and c are 256 bit integers stored at memory addresses 0x400000000, 0x40000020 and 0x40000040 respectively, write ARM assembly language instructions to compute:

(i) a = a | b
(ii) c = a & b


```
;
; a 256-bit integer occupies 256/32 = 8 x 32-bit words
;
; R1 -> a
; R2 -> b
; R3 -> c
;
        LDR     R1, =0x40000000 ; R1 -> a
        LDR     R2, =0x40000020 ; R2 -> b
        LDR     R3, =0x40000040 ; R3 -> c
        LDR     R4, =8          ; cnt = 8 (8 words)
L       LDR     R0, [R1]        ; R0 = a
        LDR     R5, [R2]        ; R5 = b
        AND     R0, R0, R5      ; R0 = a & b
        STR     R0, [R3]        ; c = R0 = a & b
        ADD     R1, R1, #4      ; R1 -> next word of a
        ADD     R2, R2, #4      ; R2 -> next word of b
        ADD     R3, R3, #4      ; R3 -> next word of c
        SUBS    R4, R4, #1      ; cnt -= 1 and set flags
        BNE     L               ; next word if cnt !=  0
```

(iii) c = a + b

Q3 If a zero terminated string of ASCII characters is stored at memory address 0x40000000, write ARM assembly language instructions to count the number of characters in the string (not including the zero). For example, if the string of ASCII characters is 0x31 0x32 0x33 0x00, its length is 3.

```
; R0 = length
; R1 -> s

        LDR     R0, =0              ; length = 0
        LDR     R1, =0x40000000     ; R1 -> s
L       LDRB    R2, [R1]            ; load next byte of string
        CMP     R2, #0              ; 0 ?
        BEQ     L1                  ; finished if 0
        ADD     R0, R0, #1          ; length += 1
        ADD     R1, R1, #1          ; R1 -> next byte in s
        B       L                   ; next byte
L1
```

Q4 If a zero terminated string of ASCII characters is stored at memory address 0x40000000, write ARM assembly language instructions to store the string in reverse order at memory address 0x40001000. For example, if the string of ASCII characters is 0x31 0x32 0x33 0x00 the reverse string stored at memory location 0x40001000 is 0x33 0x32 0x31 0x00.

Q5 If a zero terminated string of ASCII characters is stored at memory address 0x40000000, write ARM assembly language instructions to reverse the string in situ (without using any other memory locations).


```
; R1 -> first byte of string
; R2 -> last byte of string (excluding terminating zero)
;
; find last byte in string
;
```



```
        LDR     R1, =0x40000000     ; R1 -> s
        MOV     R2, R1              ; R2 -> s
L       LDRB    R0, [R2]            ; load next byte of string
        CMP     R0, #0              ; 0 ?
        BEQ     L1                  ;
        ADD     R2, R2, #1          ; R1 -> next byte in s
        B       L                   ; next byte
L1      SUB     R2, R2, #1          ; R2 = R2 - 1
;
; swap first and last bytes and work towards middle
;
L2      CMP     R1, R2              ;
        BHS     L3                  ; if R1 >= R2 then finished
        LDRB    R0, [R1]            ; swap bytes
        LDRB    R3, R2]             ;
        STRB    R0, [R2]            ;
        STRB    R3, [R1]            ;
        ADD     R1, R1, #1          ; R1 = R1 + 1
        SUB     R2, R2, #1          ; R2 = R2 - 1
        B       L2                  ;
L3                                  ;
```