

University of Dhaka

Department of Computer Science and Engineering CSE-3113:

Microprocessor and Assembly Language Lab

Lab Report 4

Submitted By:

Syed Mumtahin Mahmud, Roll: 50

Submitted On:

February 28, 2023

Submitted To:

Dr. Upama Kabir

Dr. Md. Mustafizur Rahman

Table of Contents

February 27, 2023

Contents

1 Objectives																	2								
2	Lab	Ta	sks																						2
	2.1	Qι	estion	n 1												•									2
	2.2	Qι	estion	n 2																					4
	2.3	Qι	estion	n 3																					5
	2.4	Qı	estion	n 4																					6
	2.5	Qι	estion	n 5																					7
	2.6	Qι	estion	ı 6																					8

1 Objectives

The objectives of this lab is to understand and have familiarize with register based assembly programming for Cortex M4 processor for arithmetic operation.

2 Lab Tasks

2.1 Question 1

Write an assembly language to perform all the logical operations (AND,OR,NOR,NAND,XOR,XNOR) on two 16-bit variables. Repeat it for two 32-bit variables.

Listing 1: Adding 3 NumbersLogical operations on two 16-bit variables

```
AREA lab4, CODE, READONLY
             ENTRY
             EXPORT main
3
             EXPORT problem_1_16
4
5
6
    X EQU 0x1234
7
    Y EQU 0x4321
    P EQU 0x10000000
    Q EQU 0x10000000
10
11
    main
12
        MOV ro, \#X
13
        MOV r1, \#Y
14
        BL problem_1_16
15
    stop B stop
16
17
    problem_1_16
18
        AND r2, r0, r1 ; AND
19
        ORR r3, r0, r1 ; OR
20
        MVN r4, r3 ; NOR
21
        MVN r5,r2; NAND
22
        EOR r6, r0, r1 ; Bitwise XOR
23
        MVN r2, r2 ; Bitwise NOT
24
        MVN r3, r1 ; Bitwise NOT
25
        ORR r7, r2, r3; Bitwise XNOR
26
             BX lr;
27
28
             BX lr;
29
             END
30
```

Listing 2: Logical operations on two 32-bit variables

```
AREA lab4, CODE, READONLY
1
             ENTRY
2
             EXPORT main
3
             EXPORT problem_1_32
4
    X EQU 0x1234
    Y EQU 0x4321
    P EQU 0x10000000
    Q EQU 0x10000000
9
    main
10
        MOV ro, #X
11
        MOV r1, #Y
^{12}
13
        BL problem_1_32
14
    stop B stop
15
16
    problem_1_32
17
        AND r2, r0, r1 ; AND
18
        ORR r3, r0, r1 ; OR
19
        MVN r4, r3 ; NOR
20
        MVN r5, r2; NAND
21
        EOR r6, r0, r1; Bitwise XOR
22
        MVN r2, r2 ; Bitwise NOT
23
        MVN r3, r1 ; Bitwise NOT
24
        ORR r7, r2, r3 ; Bitwise XNOR
25
^{26}
             BX lr;
^{27}
             END
```

2.2 Question 2

Write an assembly language to perform all the shift operations (LSR, ASR, LSL) on a 32-bit variable.

Listing 3: Perform all the shift operations on 32-bit variable.

```
AREA lab4, CODE, READONLY
1
             ENTRY
2
             EXPORT main
3
             EXPORT problem_2
4
5
    X EQU 0x1234
6
    Y EQU 0x4321
    P EQU 0x10000000
    Q EQU 0x10000000
9
10
    main
11
        MOV rO, #P
12
        MOV r1, #Q
13
         BL problem_2
14
15
    stop B stop
16
17
    problem_2
18
           ASR r1,r0,#1
19
           LSL r2,r0,#1
20
              LSR r3,r0,#1
21
22
              BX lr;
23
             END
^{24}
```

2.3 Question 3

Write an assembly language to perform all the arithmetic operations (Addition, Subtraction, Division and Multiplication) on two variables. Restrict input values to avoid overflow. Repeat the same operations to handle overflow.

Listing 4: Perform all the arithmetic operations on two variables.

```
AREA lab4, CODE, READONLY
             ENTRY
2
             EXPORT main
3
             EXPORT problem_3
4
5
    X EQU 0x1234
6
    Y EQU 0x4321
    P EQU 0x10000000
8
    Q EQU 0x10000000
9
    main
10
        MOV ro, #P
11
        MOV r1, #Q
12
        BL problem_3
13
14
    stop B stop
15
    problem_3
16
              ADDS r2,r0,r1; addition
17
              BVX handle_overflow
18
              SUBS r3,r0,r1; subtraction
19
              BVX handle_overflow
20
              DIVS r4,r0,r1; signed division
21
              BVX handle_overflow
22
              MULS r5, r0, r1; multiplication
23
              BVX handle_overflow
24
              BX lr;
25
    handle_overflow
26
             B srop
27
             END
28
```

2.4 Question 4

Write an assembly language program to find the average of n numbers.

Listing 5: Find the average of n numbers

```
1
            AREA Avrg, CODE, READONLY
2
            ENTRY
3
            EXPORT main
4
    main
5
            MOV r1, #0
                                      ; Sum
6
            LDR r2, =numbers
                                  ; Load the address of the numbers array
7
            LDR r3, =0x0005
                                  ; n
8
            LDR r4, =0x0005
9
    loop
10
            LDR r5, [r2], #4
11
            ADD r1, r1, r5
12
            SUBS r3, r3, #1
                                 ; Decrement n and update the flags
13
            BNE loop
                                 ; Repeat until n = 0
14
15
            SDIV r0, r1, r4
                                 ; Divide the sum by n
16
17
    Stop B Stop
18
    numbers DCD 1, 2, 3, 0,9
                                ; Array of numbers
19
            END
20
```

2.5 Question 5

Write an assembly language program to find the largest among n different numbers.

Listing 6: Find the largest among n different numbers

```
AREA Largest, CODE, READONLY
1
                     ENTRY
2
                     EXPORT main
3
                     EXPORT fnd_max
4
    main
5
                     MOV r1, #0
                                           ; Maximum numbe
6
                     BL fnd_max
7
    Stop B Stop
8
    fnd_max
9
                                          ; Load the address of the numbers array
                     LDR r2, =numbers
10
                     LDR r3, =0x0005
                                          ; Load the value of n
11
    loop
12
                     LDR r4, [r2], #4
                                          ; Load next nmbr
13
                     CMP r4, r1
14
                     BGT update_max
15
                     SUBS r3, r3, #1
                                           ; Decrement n and update the flags
16
                     BNE loop
                                           ; while n>0
17
    update_max
18
                     MOV r1, r4
19
                     SUBS r3, r3, #1
20
                     BNE loop
^{21}
                     BX LR ; return to main function
^{22}
    numbers DCD 1,5,3,4,7 ; Array of numbers
23
            END
24
25
26
```

2.6 Question 6

Write an assembly language program to find the average of n numbers using function call

Listing 7: Find the largest among n different numbers

```
AREA Avrg, CODE, READONLY
1
             ENTRY
2
3
             EXPORT main
             EXPORT fnd_avg
4
    main
5
             MOV r1, #0
                                        ; Sum
6
             LDR r2, =numbers
                                   ; Load the address of the numbers array
7
             LDR r3, =0x0005
8
             LDR r4, =0x0005
9
             BL fnd_avg
10
    Stop B Stop
11
    fnd_avg
12
13
    loop
14
             LDR r5, [r2], #4
15
             ADD r1, r1, r5
16
             SUBS r3, r3, #1
17
             BNE loop
18
             SDIV r0, r1, r4
19
             BX LR
20
    numbers DCD 1, 2, 3, 0,9
^{21}
             END
22
```

References

- [1] Documentation arm developer. https://developer.arm.com/documentation/ddi0439/b/CHDDIGAC. [Online; accessed 2023-01-31].
- [2] Nucleo-F446RE. https://www.st.com/en/evaluation-tools/nucleo-f446re.html. [Online; accessed 2023-01-31].