

Microprocessor and Computer Architecture

UE23CS251B

4th Semester, Academic Year 2024-25

Date:01/02/2025

Name: Keerthan P.V	SRN: PES2UG23CS272	Section: 4E
-----------------------	-----------------------	----------------

Week#____3_____ Assignment Program : ____1____

Title of the Program

5a) Write an ALP using ARM7TDMI to perform Convolution using MUL instruction (Addition of multiplication of respective numbers of loc A and loc B)

5b) Write an ALP using ARM7TDMI to perform Convolution using MLA instruction (Addition of multiplication of respective numbers of loc A and loc B).

I. ARM Assembly Code(1)

Typed Code to be Included. Screenshot of Code not permitted

a)Using MUL instruction

.data

A:.word 1,3,5,7,9

B:.word 2,4,6,8,10

```
.text
LDR R0,=A
LDR R1,=B
MOV R2,#5
MOV R4,#0

LOOP:LDR R5,[R0],#4
      LDR R6,[R1],#4
      MUL R7,R5,R6
      ADD R4,R4,R7
      SUB R2,R2,#1
      CMP R2,#0
      BNE LOOP
      B EXIT

EXIT:SWI 0x11

.end
```

b)Using MLA instruction
data
A:.word 1,3,5,7,9
B:.word 2,4,6,8,10

```
.text
LDR R0,=A
LDR R1,=B
MOV R2,#5
MOV R3,#0

LOOP:LDR R4,[R0],#4
      LDR R5,[R1],#4
      MLA R3,R4,R5,R3
      SUB R2,R2,#1
      CMP R2,#0
      BNE LOOP
      B EXIT

EXIT:SWI 0x11

.end
```

II. Output Screen Shot (2)

The output should be shown for both the case a and b

The results should be clearly visible in the screenshots. The screenshot should include the code window, register window, memory window and console window

A) Using MUL instruction

The screenshot displays the ARMSim interface with the following components:

- RegistersView:** Shows the CPSR Register and General Purpose Registers (R0-R15). R0 is 00001050, R1 is 00001064, R2 is 00000000, R3 is 00000000, R4 is 000000be, R5 is 00000009, R6 is 0000000a, R7 is 0000005a, R8 is 00000000, R9 is 00000000, R10 (s1) is 00000000, R11 (fp) is 00000000, R12 (ip) is 00000000, R13 (sp) is 00005400, R14 (lr) is 00000000, and R15 (pc) is 00001030.
- Code View:** Shows the assembly code for arm19.s. The program includes data labels A and B, a loop that loads R0 and R1, multiplies them (MUL R7, R5, R6), adds R4 to R7, subtracts R2 from R7, compares R2 with R0, and branches if not less than or equal (BNE LOOP). It ends with an SWI instruction.
- MemoryView:** Shows a memory dump starting at address 00000000, with values 00000000, 00000068, and 0000009C.
- WatchView:** Shows a table with columns Label and Value.
- OutputView:** Shows the console output, including the loading of the assembly file, execution starting, and execution ending with an instruction count of 41 and an elapsed time of 00:00:00.0273574.

B) Using MLA instruction

The screenshot displays the ARMSim interface with the following components:

- RegistersView:** Shows the CPSR Register and General Purpose Registers (R0-R15). R0 is 4172, R1 is 4192, R2 is 0, R3 is 190, R4 is 9, R5 is 10, R6 is 0, R7 is 0, R8 is 0, R9 is 0, R10 (s1) is 0, R11 (fp) is 0, R12 (ip) is 0, R13 (sp) is 21504, R14 (lr) is 0, and R15 (pc) is 4140.
- Code View:** Shows the assembly code for arm19.s. The program includes data labels A and B, a loop that loads R0 and R1, multiplies them (MLA R3, R4, R5, R3), subtracts R2 from R3, compares R2 with R0, and branches if not less than or equal (BNE LOOP). It ends with an SWI instruction.
- MemoryView:** Shows a memory dump starting at address 00000000, with values 00000000, 00000068, and 0000009C.
- WatchView:** Shows a table with columns Label and Value.
- OutputView:** Shows the console output, including the loading of the assembly file, execution starting, and execution ending with an instruction count of 36 and an elapsed time of 00:00:00.0084392.

Week#____3_____ Assignment Program : ____2__

6)Write an ALP using ARM7TDMI to find the sum of all the BCD digits of a given 32-bit number.

(hint:788 =7+8+8)

I. ARM Assembly Code(1)

Typed Code to be Included. Screenshot of Code not permitted

.data

A:.word 0x12345678

.text

LDR R0,=A

MOV R1,#8

MOV R2,#0

LDR R3,[R0]

LOOP:AND R4,R3,#0xF

ADD R2,R2,R4

MOV R3,R3,LSR #4

SUB R1,R1,#1

CMP R1,#0

BNE LOOP

B EXIT

EXIT:SWI 0x11

.end

II. Output Screen Shots (1)

The screenshot displays the ARMSim - The ARM Simulator interface. The main window shows assembly code for a file named 'arm19.s'. The code includes a data section with a word 'A' at address 0x12345678, followed by a text section with various instructions: LDR R0,=A; MOV R1,#8; MOV R2,#0; LDR R3,[R0]; LOOP: AND R4,R3,#0xF; ADD R2,R2,R4; MOV R3,R3,LSR #4; SUB R1,R1,#1; CMP R1,#0; BNE LOOP; B EXIT; EXIT: SWI 0x11. The registers window on the left shows R0 through R15, with R15 (PC) at 0x000102c. The CPSR register shows Negative(N):0, Zero(Z):1, Carry(C):1, Overflow(O):0, IRQ Disable:1, FIQ Disable:1, Thumb(T):0, and CPU Mode: System. The memory window shows a table of memory addresses and values, mostly containing 0xFFFFFFFF. The output window at the bottom shows the execution log, including the file path, execution start and end times, and instructions per second.

```

.data
00001034:      A: word 0x12345678
.text
00001000: E59F0028  LDR R0,=A
00001004: E3A01008  MOV R1,#8
00001008: E3A02000  MOV R2,#0
0000100C: E5903000  LDR R3,[R0]
00001010: E203400F  LOOP: AND R4,R3,#0xF
00001014: E0822004  ADD R2,R2,R4
00001018: E1A03223  MOV R3,R3,LSR #4
0000101C: E2411001  SUB R1,R1,#1
00001020: E3510000  CMP R1,#0
00001024: 1AFFFFF9  BNE LOOP
00001028: BFFFFFFF  B EXIT
0000102C: EF000011  EXIT: SWI 0x11
00001030: 00001034  .end

```

RegistersView

General Purpose Floating Point

Hexadecimal
Unsigned Decimal
Signed Decimal

R0 : 00001034
R1 : 00000000
R2 : 00000024
R3 : 00000000
R4 : 00000001
R5 : 00000000
R6 : 00000000
R7 : 00000000
R8 : 00000000
R9 : 00000000
R10 (sl): 00000000
R11 (fp): 00000000
R12 (ip): 00000000
R13 (sp): 00005400
R14 (lr): 00000000
R15 (pc): 0000102c

CPSR Register
Negative(N):0
Zero(Z):1
Carry(C):1
Overflow(O):0
IRQ Disable:1
FIQ Disable:1
Thumb(T):0
CPU Mode : System

0x600000df

MemoryView0

Word Size
8Br 16Br 32Br

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

00000034 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

00000068 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

0000009C 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

WatchView

Label Value

OutputView

Console Stdin/Stdout/Stderr

Loading assembly language file C:\Users\keert\OneDrive\Desktop\CSE\SEM-4\MCPA\LAB\arm19.s
Execution starting ...
Execution ending, Instruction Count:54 Elapsed Time:00:00:00.0113164
Instructions per second:4771

Disclaimer:

- The programs and output submitted is duly written, verified and executed by me.
- I have not copied from any of my peers nor from the external resource such as internet.
- If found plagiarized, I will abide with the disciplinary action of the University.

Signature: Keerthan P.V

Name: Keerthan P.V

SRN: PES2UG23CS272

Section: 4E

Date: 01/02/2025