Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)
Department of CSE

Tutorial -III

Programme: B.E
Course: Computer Organization Course Code: CS45

Term: Jan to May 2020

Name: MANAS. P.S	**Marks:** 10/10	**Date:** 4/2/2020
USN: 1MS18CS065	**Signature of the Faculty:**	10/2/2020

Objective: To simulate ARM Instruction set using ARMsim simulator.

Simulator Used: ARMSim 1.91 is a desktop application running in a Windows environment. It allows users to simulate the execution of ARM assembly language programs on a system based on the ARM7TDMI processor.

ARM enables the users both to debug ARM assembly programs and to monitor the state of the system while a program executes.

Activity to be performed by students:

1) Write an ARM program to perform basic arithmetic operations.
2) Write an ARM program to demonstrate the working of load and store instructions.
3) Write an ARM program to evaluate expression f=(g+h)-(i+j)
4) Write an ARM program to find the sum of all elements of an array.
5) Write an ARM program to find the factorial of a number.

Programs and the snapshots:

RAMAIAH
Institute of Technology

Data Sheet

MARKS :

Name :	MANAS.P.S	Branch:	CSE
USN/Roll No. :	1MS18CS065	Sem/Sec:	IV 'B'
Subject :	Computer Organization and Architecture	Subject Code:	CS45

(1) Write an ARM program to perform basic arithmetic operations

```
MOV R3, #24
MOV R4, #25
LDR R1, = 0x00000064
LDR R2, = 0x00000068

STR R3, [R1]
STR R4, [R2]

LDR R5, [R1]
LDR R6, [R2]

ADD R0, R5, R6      ; Basic arithmetic operations
MUL R0, R5, R6
SUB R0, R5, R6

SWI 0x11
```

(2) Write an ARM program to perform and demonstrate working of load and store instructions.

```
MOV R3, #36      ; Values to store
MOV R4, #37
```

```
LDR  R1,=0x00000074      ; load address into
LDR  R2,=0x00000078      ; register

STR  R3,[R1]             ; store R3 and R4 at address in
STR  R4,[R2]             ; R1 and R2

LDR  R5,[R1]             ; load from R1,R2 to R5,R6
LDR  R6,[R2]

ADD  R0,R5,R6
SUB  R0,R5,R6
MUL  R0,R5,R6

SWI  0x11
```

(3) Write an ARM program to evaluate expression
$f = (g+h)-(i+j)$

```
MOV R3,#41               ; value of g
MOV R4,#45               ; value of h
MOV R7,#35               ; value of i
MOV R8,#38               ; value of j
LDR R1,=0x00000064
LDR R2,=0x00000068
LDR R9,=0x00000074
LDR R10,=0x00000078

STR R3,[R1]
STR R4,[R2]
STR R7,[R9]
STR R8,[R10]
```

```
LDR R5,[R1]
LDR R6,[R2]
LDR R11,[R9]
LDR R12,[R10]

ADD R5,R5,R6     ; g+h
ADD R11,R11,R12  ; i+j

SUB R0,R5,R11    ; (g+h)-(i+j), store result in R0

SWI 0x11
```

(4) Write an ARM program to find sum of all elements in an array

```
MOV R0,#5      ; initialize counter register
LDR R1,=array  ; load base address
loop: LDR R2,[R1],#4  ; load array val and increment
      ADD R3,R3,R2    ; address. Sum stored in R3.
      SUB R0,R0,#1    ; Decrement counter value
      CMP R0,#0       ; checking counter value
      BNE loop
      SWI 0x11
.data
array: .word 0x00000001,0x0000000A,0x000000AB,
             0x0000000B,0x00000008
```

(5) Write an ARM program to find the factorial of a number.

```
LDR R1,=fact   ; load base address of variables
LDR R2,[R1]    ; load value at [R1] into R2 and R3
LDR R3,[R1]
loop : SUB R2,R2,#1    ; decrement R2 value
       CMP R2,#0       ; break to end if R2==0
       BEQ end
       MUL R0,R2,R3    ; calculate product in current
       MOV R3,R0       ; iteration, store into R3 for
       CMP R2,#0       ; updation. check R2 value.
       BNE loop        ; break if equal to zero
end :  SWI 0x11
.data
fact: .word 0x00000005    ; variable with value 5
```