**Microprocessor and Computer Architecture (MPCA) Laboratory**

**UE20CS252 4th Semester,**

**Academic Year 2021-22**

**Date: 04/02/2022**

**Sriram R Section : H SRN : PES1UG20CS435**

**Week # 2 Program Number: 1**

**Title of the Program**

Write a program in ARM7TDMI-ISA to find the sum of N data items at alternate [odd or **even** positions] locations in the memory. Store the result in the memory location.

a. Use Pre-indexing addressing mode

b. Use Post- Indexing addressing mode

c. Use Auto-indexing addressing mode

**Program Code**

a.

; odd alternate pre indexing

.data

a: .word 10, 20, 30, 40

sum: .word 0

.text

mov r2, #0

ldr r1, =a

ldr r3, =sum

mov r5, #0

sub r1, r1, #4; initializing r1 for pre indexing

loop: ldr r4, [r1, #4]

add r1, r1, #8; increment to skip over even locations

add r2, r2, r4

add r5, r5, #2; loop count double increment

cmp r5, #4

bne loop

str r2, [r3]

swi 0x011

.end

b.

; even alternate auto indexing

.data

a: .word 10, 20, 30, 40

sum: .word 0

.text

mov r2, #0

ldr r1, =a

ldr r3, =sum

mov r5, #0

sub r1, r1, #4; initializing r1 for pre indexing, starts at 1 and skips first element

loop: ldr r4, [r1, #8]!; writeback included

add r2, r2, r4

add r5, r5, #2

cmp r5, #4

bne loop

str r2, [r3]

swi 0x011

.end

c.

; odd alternate post indexing

.data

a: .word 10, 20, 30, 40

sum: .word 0

.text

mov r2, #0

ldr r1, =a

ldr r3, =sum

mov r5, #0

loop: ldr r4, [r1], #8

add r2, r2, r4

add r5, r5, #2

cmp r5, #4

bne loop

str r2, [r3]

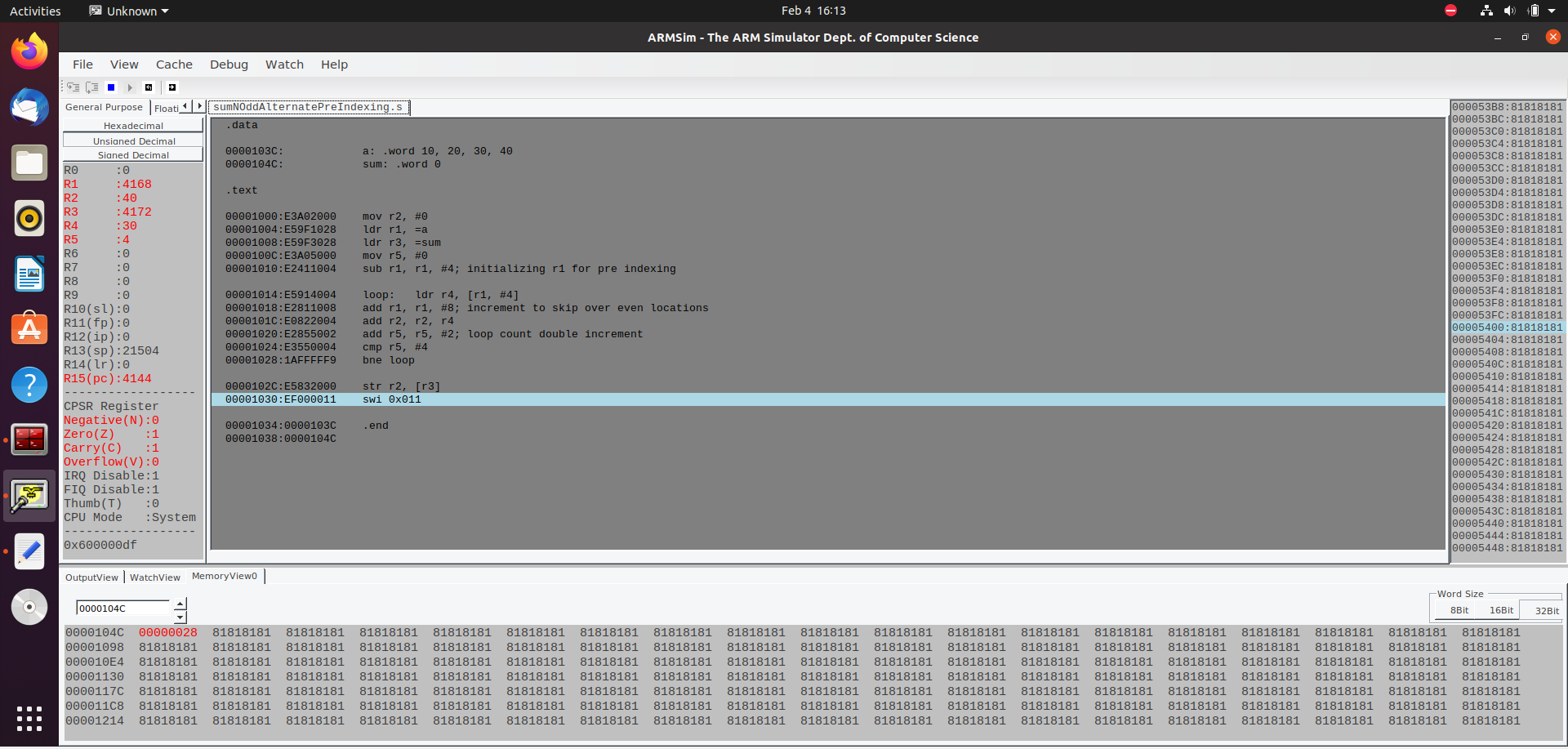
swi 0x011

.end

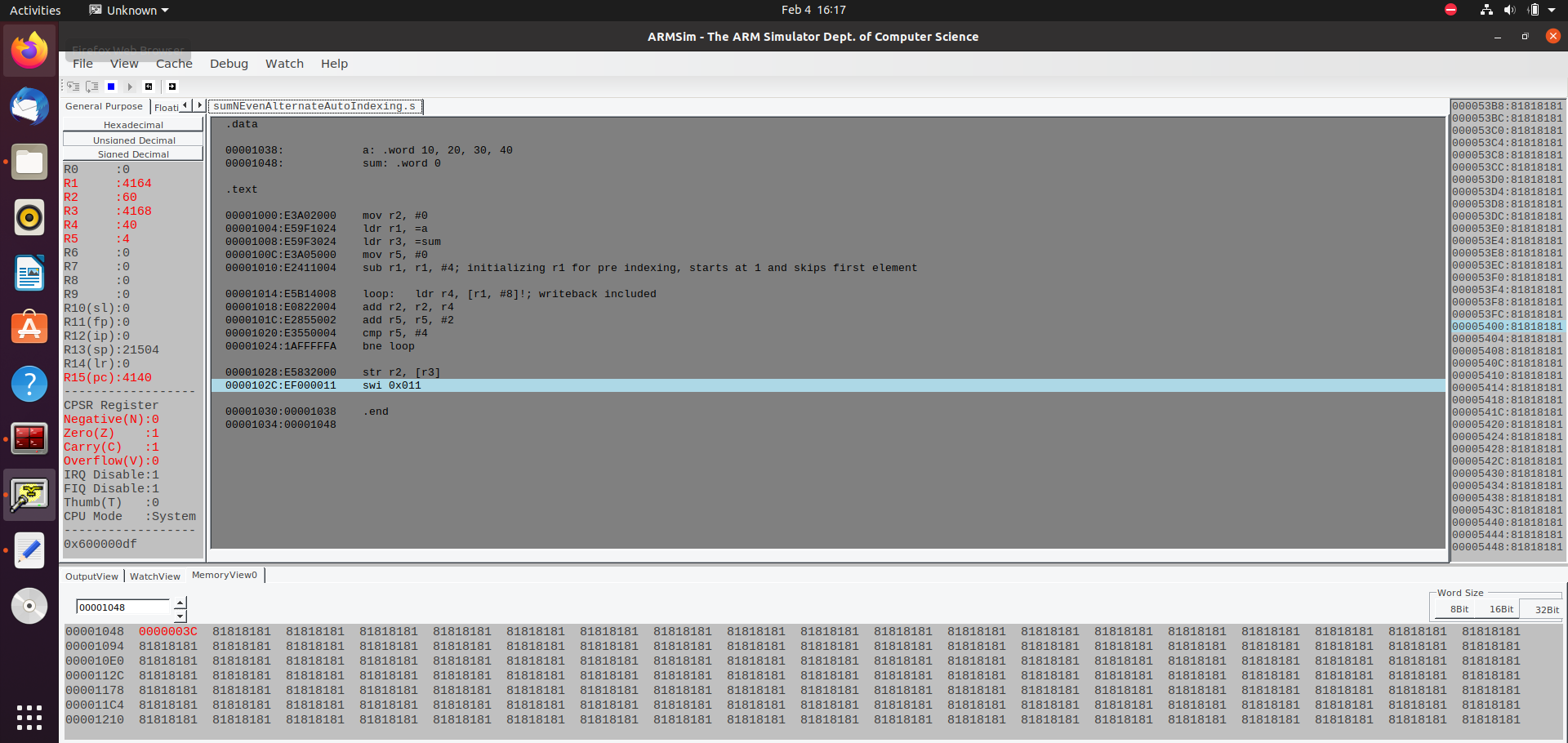
**<Screenshots on the next page>**

**Screenshot of ArmSimulator of the Program Executed**

a.



b.



c.

