Microprocessor and Computer Architecture (MPCA) Laboratory

**UE20CS252 4th Semester, Academic Year 2021-22**

**Date: 04/02/2022 Date: 28/01/2022**

|  |  |
| --- | --- |
| Name : SUNDEEP A | SRN: PES1UG20CS445 |
| SEC: H | ROLL NO : 48 |

# Week # 2 Program Number: 1

**Title of the Program**

## Write a program in ARM7TDMI-ISA to find GCD of two numbers.

* 1. Assume operands to be in the CPU registers
  2. Assume operands in the memory locations.

**Program Code**

1.a.

.TEXT

MOV R0,#6 MOV R1,#4

GCD:CMP R0,R1

BEQ RES BLT LOOP

SUB R0,R0,R1 B GCD

LOOP:SUB R1,R1,R0 B GCD

RES:MOV R2,R0

SWI 0X011

.END

1.b

.DATA

A:.WORD 6

B:.WORD 4

.TEXT

LDR R3,=A LDR R4,=B LDR R0,[R3] LDR R1,[R4]

GCD:CMP R0,R1

BEQ RES BLT LOOP

SUB R0,R0,R1 B GCD

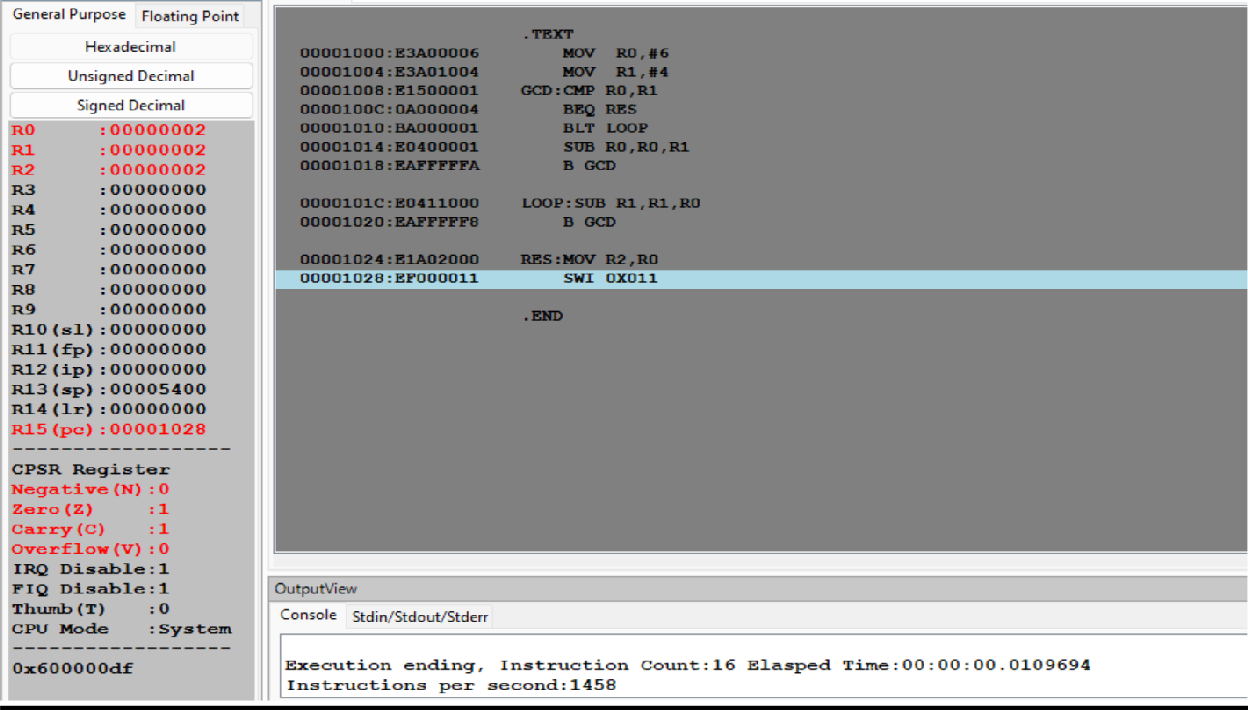
LOOP:SUB R1,R1,R0 B GCD

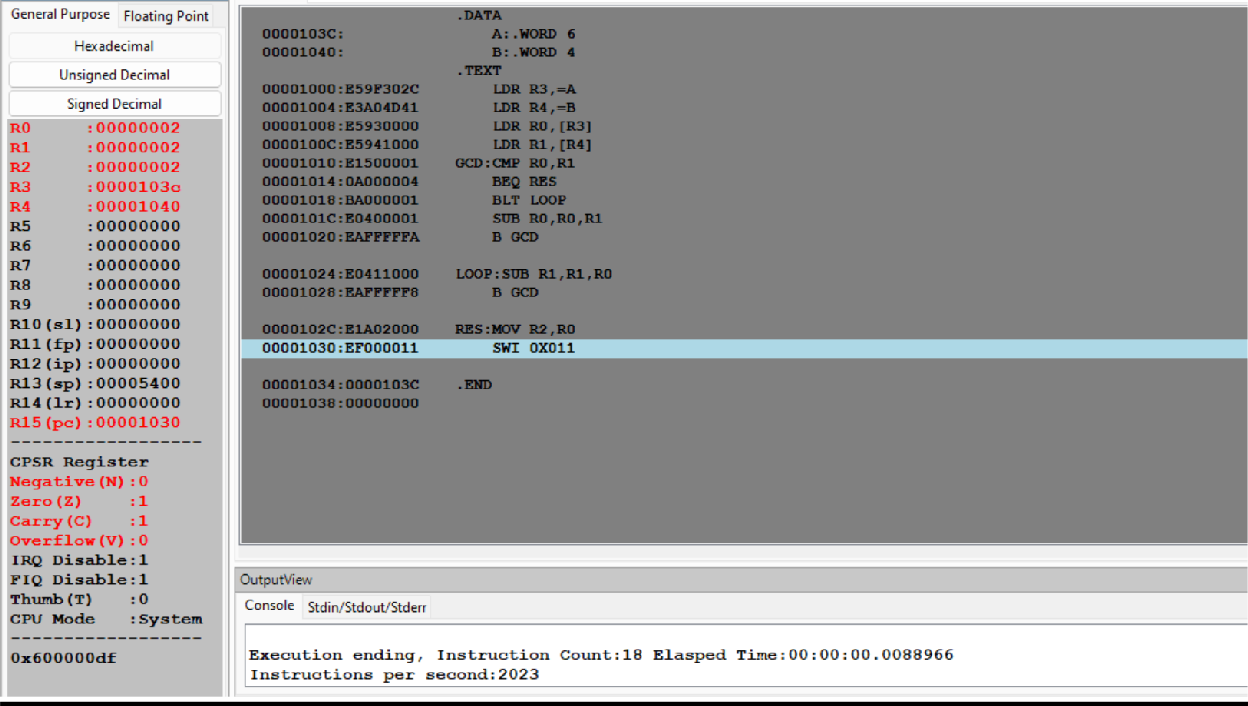
RES:MOV R2,R0

SWI 0X011

.END

**Screenshot of ArmSimulator of the Program Executed 1.a**



**1.b**

**Program Number: 2**

**Title of the Program**

1. Write a program in ARM7TDMI-ISA to find the sum of N data items in the memory. Store the result in the memory location.

**Program Code**

1. Use Pre-indexing addressing mode

.DATA

A:.WORD 10,20,30,40

SUM:.WORD 0

.TEXT

MOV R2,#0 LDR R1,=A LDR R3,=SUM MOV R6,#0 SUB R1,R1,#4

LOOP:

LDR R4,[R1,#4] ADD R1,R1,#4

ADD R2,R2,R4 ADD R6,R6,#1 CMP R6,#4 BNE LOOP

STR R2,[R3]

.END

1. Use Post- Indexing addressing mode

.DATA

A:.WORD 10,20,30,40

SUM:.WORD 0

.TEXT

MOV R2,#0 LDR R1,=A LDR R3,=SUM MOV R6,#0

LOOP:

LDR R4,[R1],#4 ADD R2,R2,R4 ADD R6,R6,#1 CMP R6,#4

BNE LOOP STR R2,[R3]

.END

1. Use Auto-indexing addressing mode

d. .DATA

e. A:.WORD 10,20,30,40

1. SUM:.WORD 0
2. .TEXT
3. MOV R2,#0
4. LDR R1,=A
5. LDR R3,=SUM
6. MOV R6,#0
7. SUB R1,R1,#4
8. LOOP:

n. LDR R4,[R1,#4]!

o. ADD R2,R2,R4

p.

q.

r.

s.

t.

u.

v.

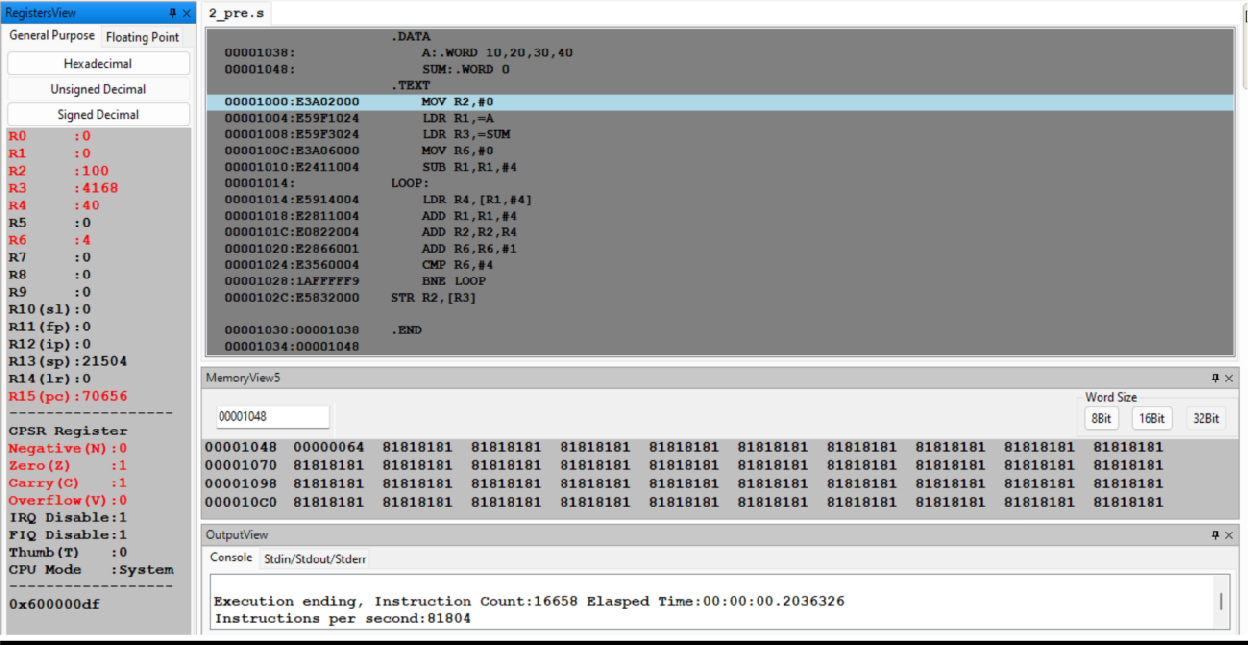
ADD R6,R6,#1

CMP R6,#4 BNE LOOP

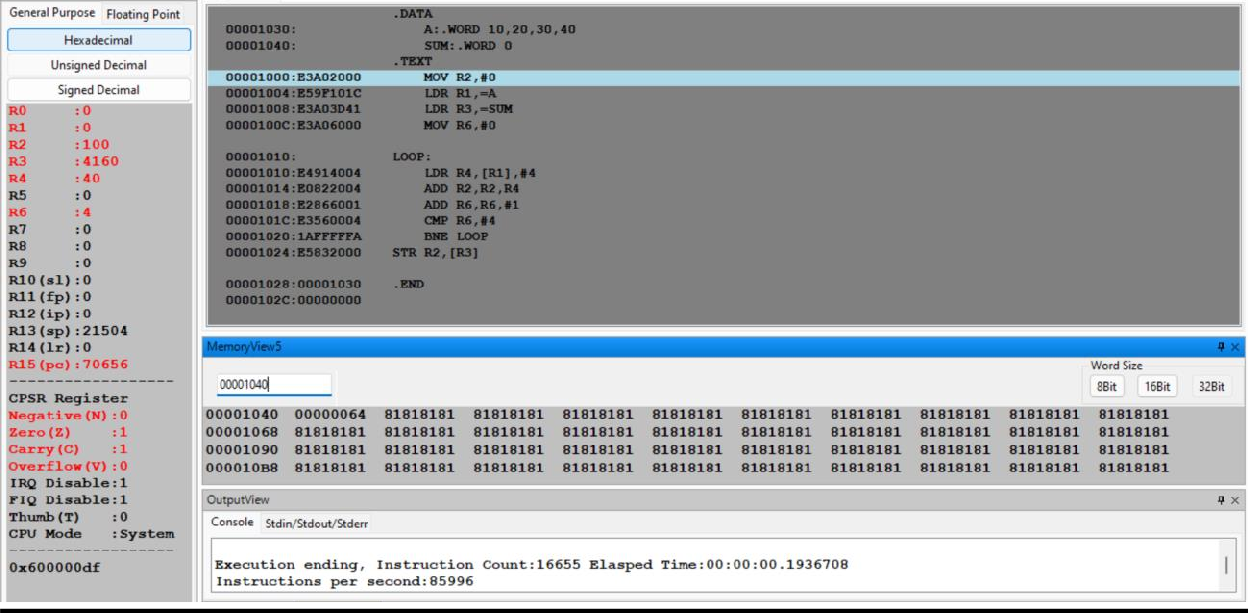
STR R2,[R3]

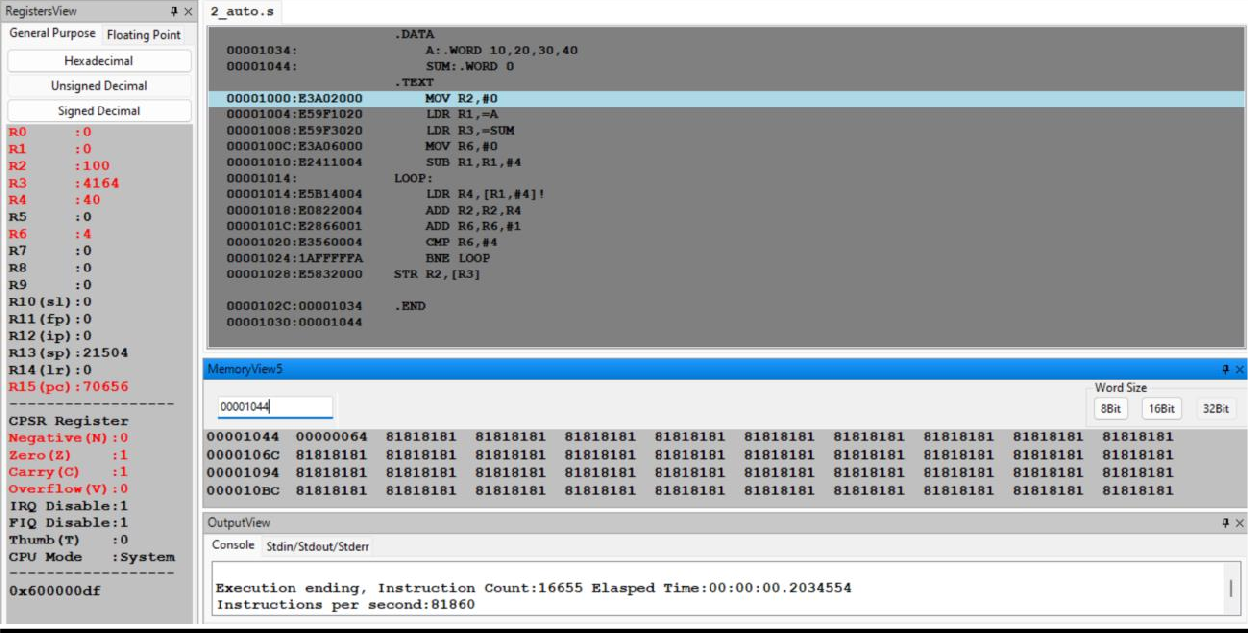
.END

**Screenshot of ArmSimulator of the Program Executed**

**2.a**

**2.b**



**2.c**

**Program Number: 3**

**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.

**Program Code**

1. Use Pre-indexing addressing mode

;here we are finding the sum of n numbers at odd locations

.DATA

A:.WORD 10,20,30,40,50,60

SUM:.WORD 0

.TEXT

MOV R2,#0 LDR R1,=A LDR R3,=SUM MOV R6,#0

SUB R1,R1,#8 ;change #8 to #4 to find the sum of numbers at even positions

LOOP:

LDR R4,[R1,#8] ADD R1,R1,#8 ADD R2,R2,R4 ADD R6,R6,#2 CMP R6,#6

BNE LOOP STR R2,[R3]

.END

1. Use Post- Indexing addressing mode

;here we are finding the sum of n numbers at odd locations

.DATA

A:.WORD 10,20,30,40,50,60

SUM:.WORD 0

.TEXT

MOV R2,#0 LDR R1,=A LDR R3,=SUM MOV R6,#0

LOOP:

LDR R4,[R1],#8 ADD R2,R2,R4 ADD R6,R6,#2 CMP R6,#6

BNE LOOP STR R2,[R3]

.END

1. Use Auto-indexing addressing mode

;here we are finding the sum of n numbers at odd locations

.DATA

A:.WORD 10,20,30,40,50,60

SUM:.WORD 0

.TEXT

MOV R2,#0 LDR R1,=A LDR R3,=SUM MOV R6,#0

SUB R1,R1,#8;change the #8 to #4 to get the sum of n numbers at even positions

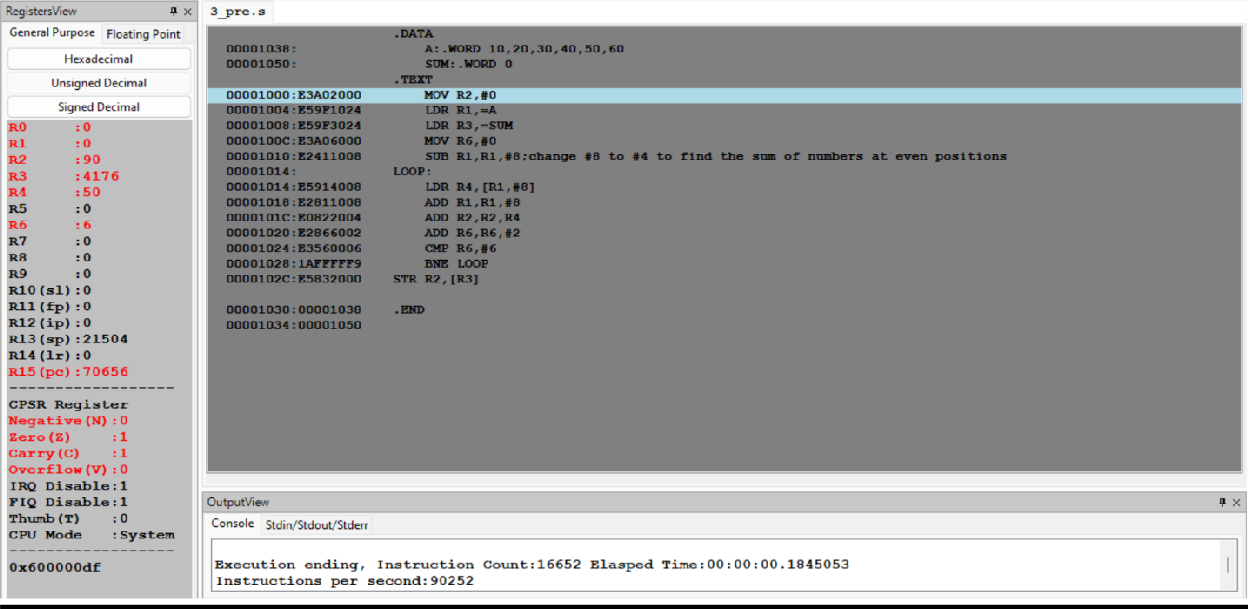
LOOP:

LDR R4,[R1,#8]! ADD R2,R2,R4 ADD R6,R6,#2 CMP R6,#6

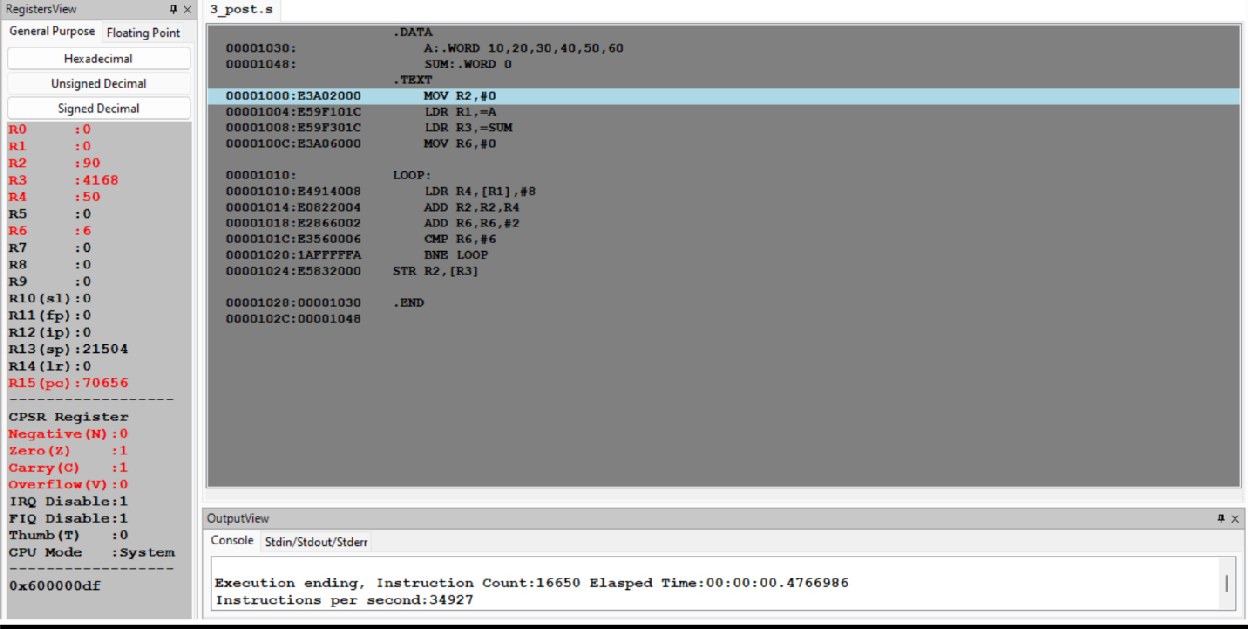
BNE LOOP STR R2,[R3]

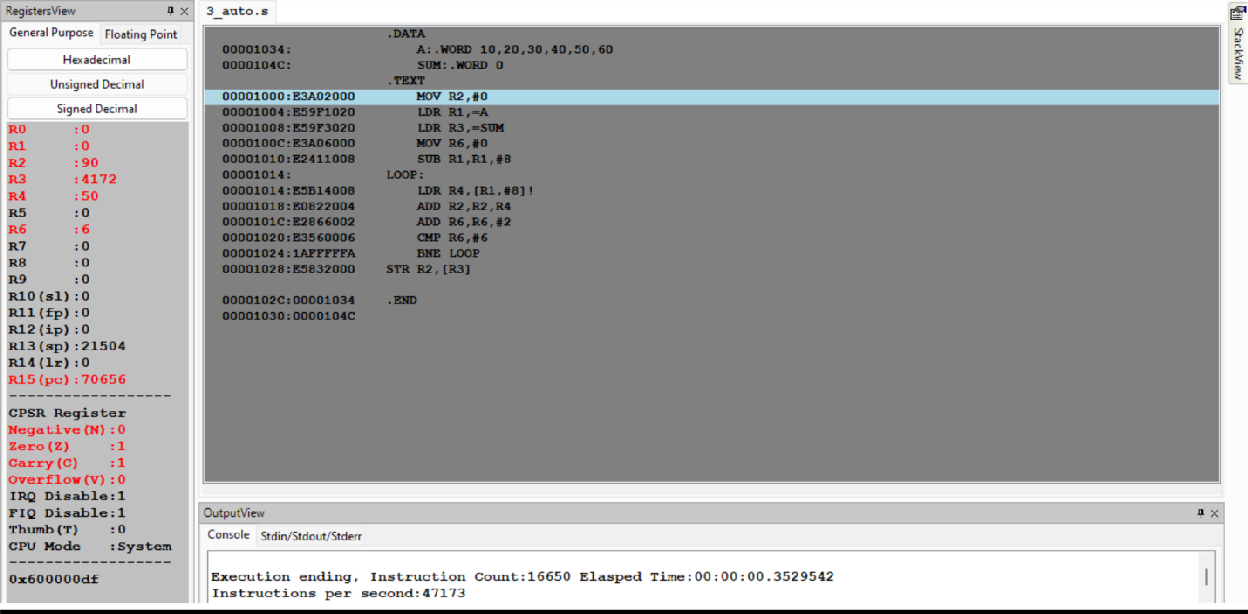
.END

**Screenshot of ArmSimulator of the Program Executed**

**3.a.**

**3.b.**



**3.c.**

**Program Number: 4**

**Title of the Program**

Write a program in ARM7TDMI-ISA to search for an element in an array.

Store 00 if the search is unsuccessful and 01 if the search is successful in the register.

* 1. Use Linear Search Technique

**Program Code**

.DATA

A:.WORD 10,20,30,40,50

KEY:.WORD 40

.TEXT

LDR R0,=A LDR R1,=KEY LDR R5,[R1] MOV R4,#1

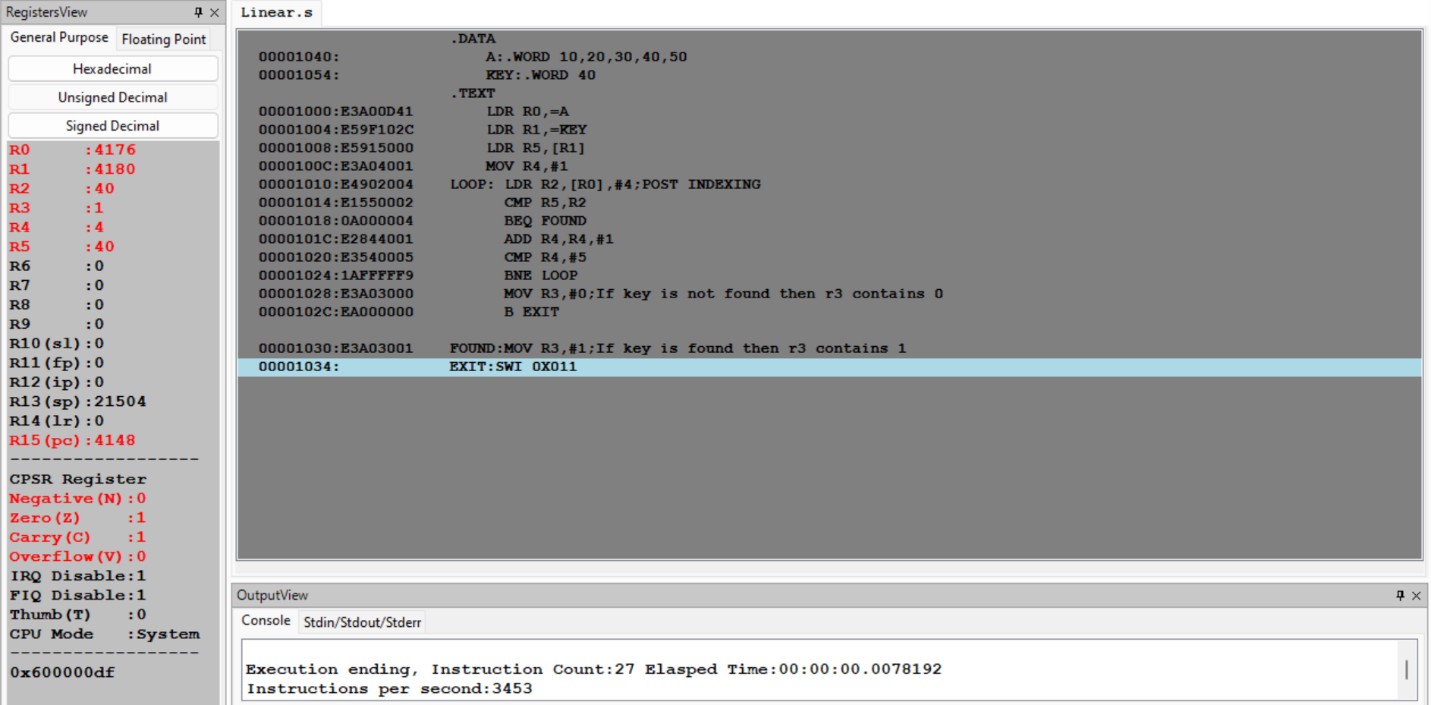
LOOP: LDR R2,[R0],#4;POST INDEXING CMP R5,R2

BEQ FOUND ADD R4,R4,#1 CMP R4,#5 BNE LOOP

MOV R3,#0;If key is not found then r3 contains 0

B EXIT

FOUND:MOV R3,#1;If key is found then r3 contains 1 EXIT:SWI 0X011

**Screenshot of ArmSimulator of the Program Executed**