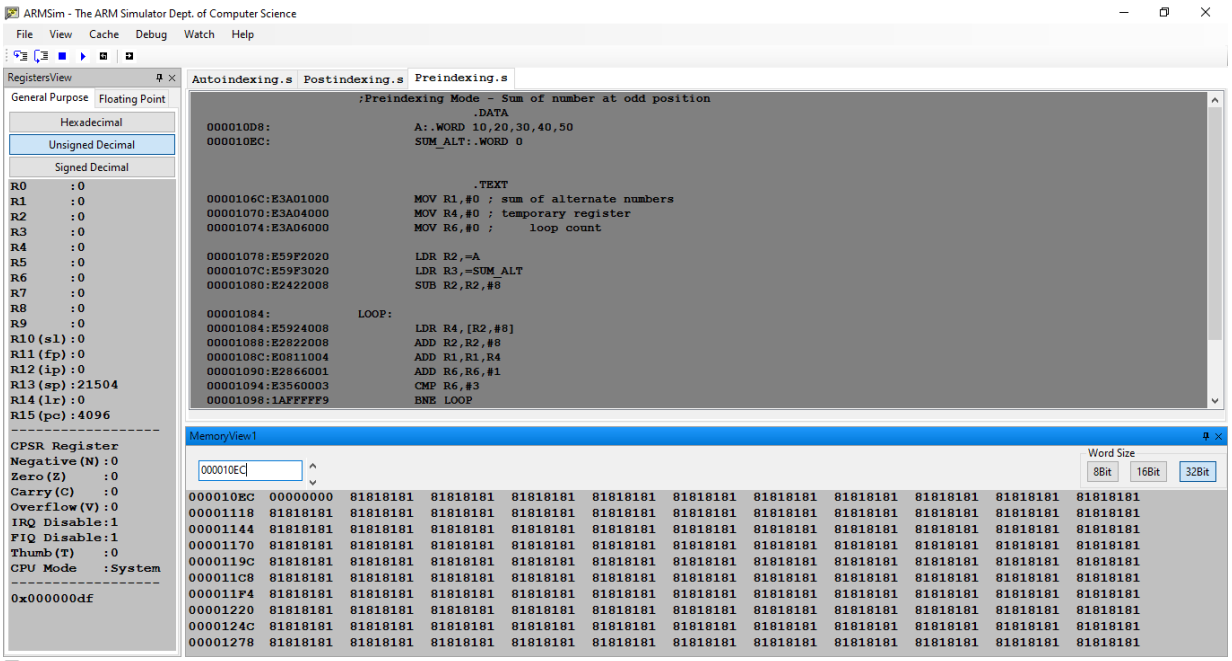
Name : Ghanashyam Mahesh Bhat
SRN : PES1UG20CS153
SECTION : C

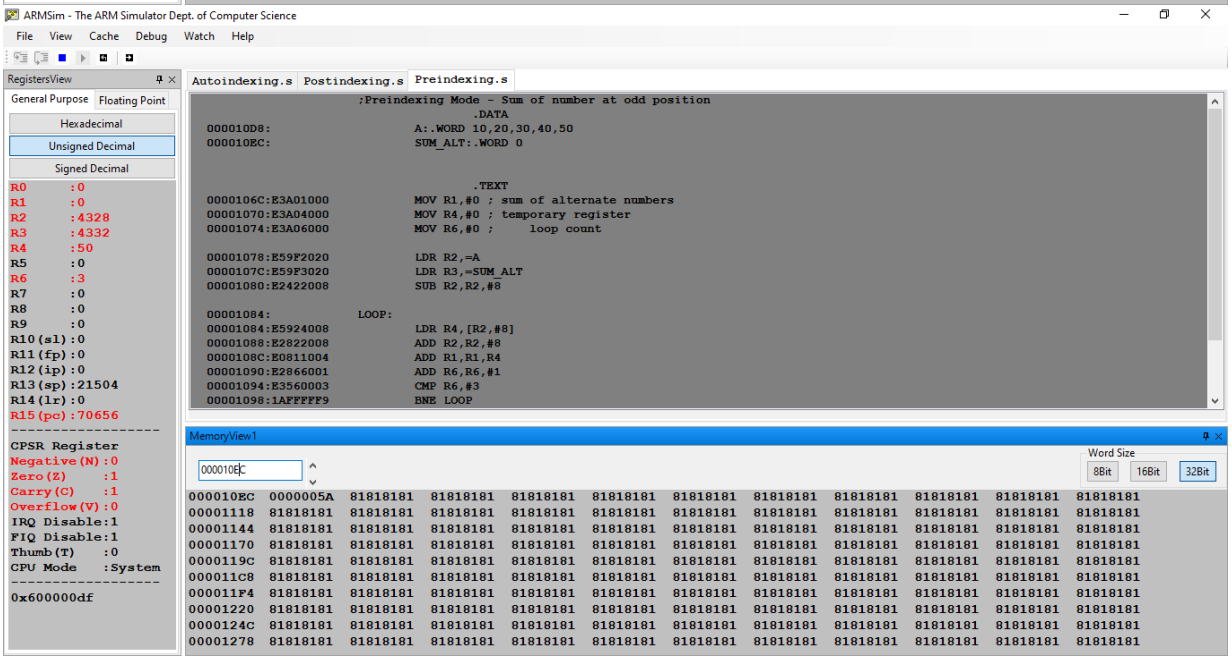
PES UNIVERSITY

Department of Computer Science & Engineering
Microprocessor & Computer Architecture
MPCA-Laboratory/Assignment/Hands-on/Project
UE20CS252

Sl. No	Programs
Week cNo.3	3. 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 **;Preindexing Mode - Sum of number at odd position** **.DATA** **A:.WORD 10,20,30,40,50** **SUM_ALT:.WORD 0** **.TEXT** **MOV R1,#0 ; sum of alternate numbers** **MOV R4,#0 ; temporary register** **MOV R6,#0 ; loop count** **LDR R2,=A** **LDR R3,=SUM_ALT** **SUB R2,R2,#8** **LOOP:** **LDR R4,[R2,#8]** **ADD R2,R2,#8** **ADD R1,R1,R4** **ADD R6,R6,#1** **CMP R6,#3** **BNE LOOP** **STR R1,[R3]**

.END





b. Use Post- Indexing addressing mode

;Postindexing Mode - Sum of number at odd position

.DATA

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

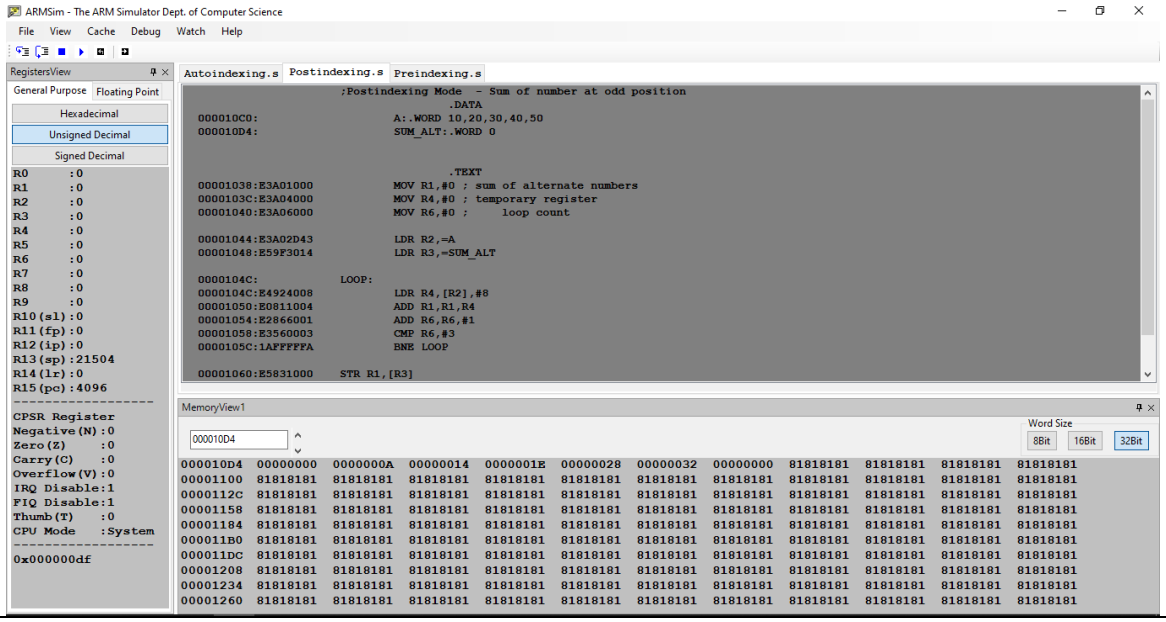
SUM_ALT:.WORD 0

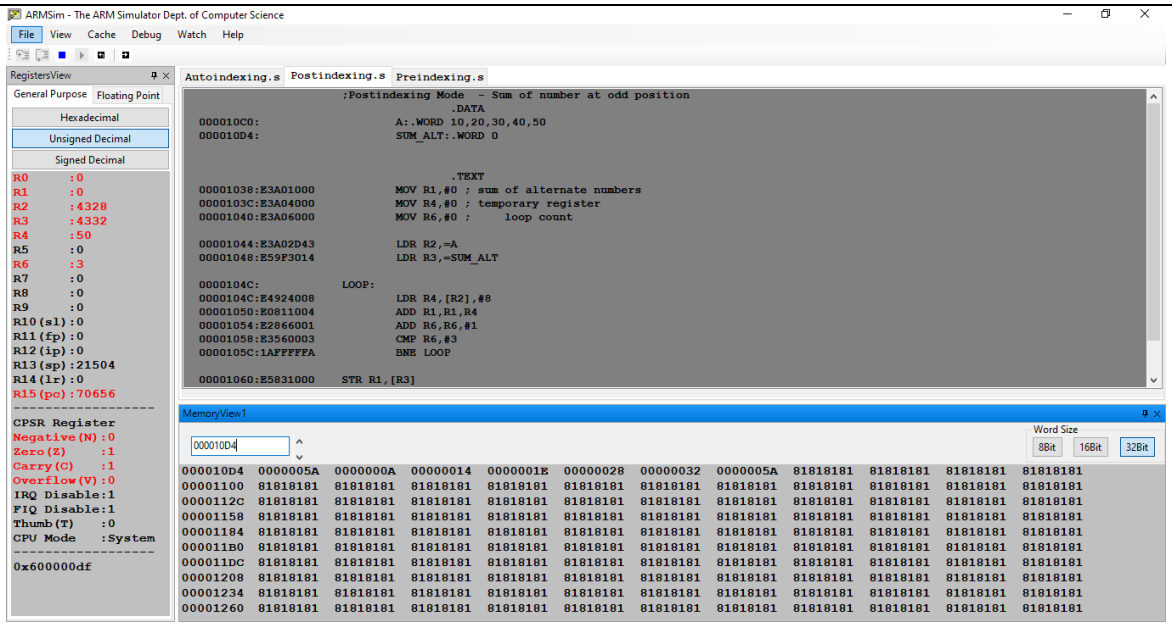
.TEXT

MOV R1,#0 ; sum of alternate numbers

MOV R4,#0 ; temporary register

```
    MOV R6,#0 ;         loop count

    LDR R2,=A
    LDR R3,=SUM_ALT

LOOP:
    LDR R4,[R2],#8
    ADD R1,R1,R4
    ADD R6,R6,#1
    CMP R6,#3
    BNE LOOP

STR R1,[R3]

            .END
```

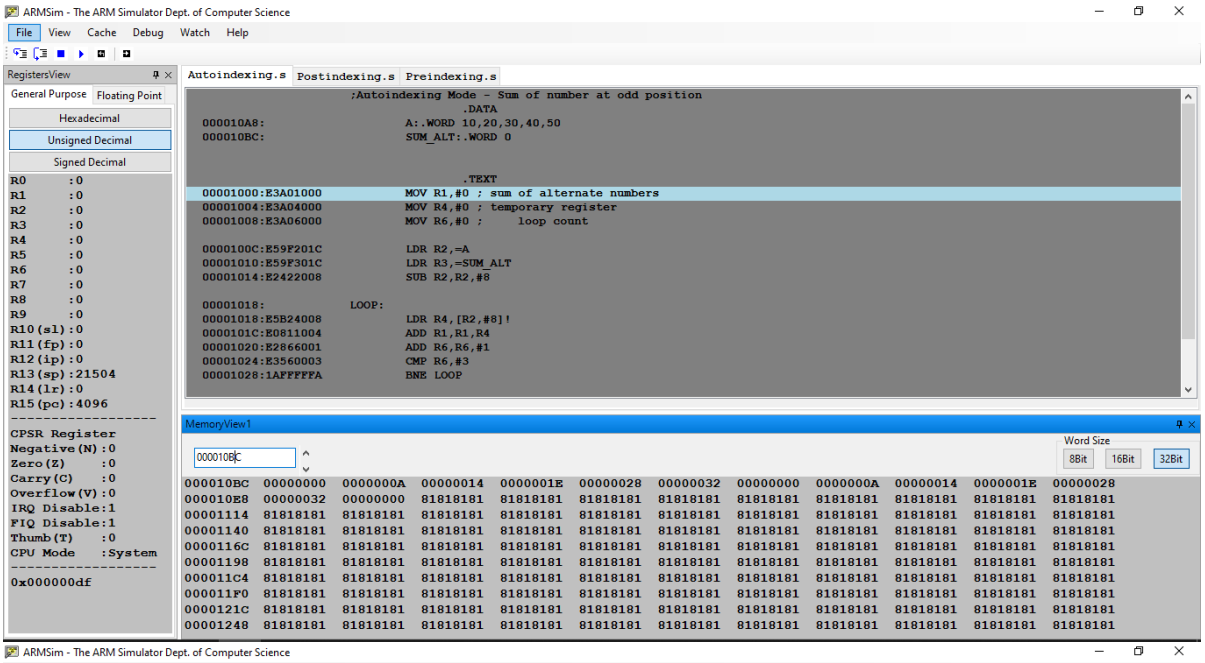


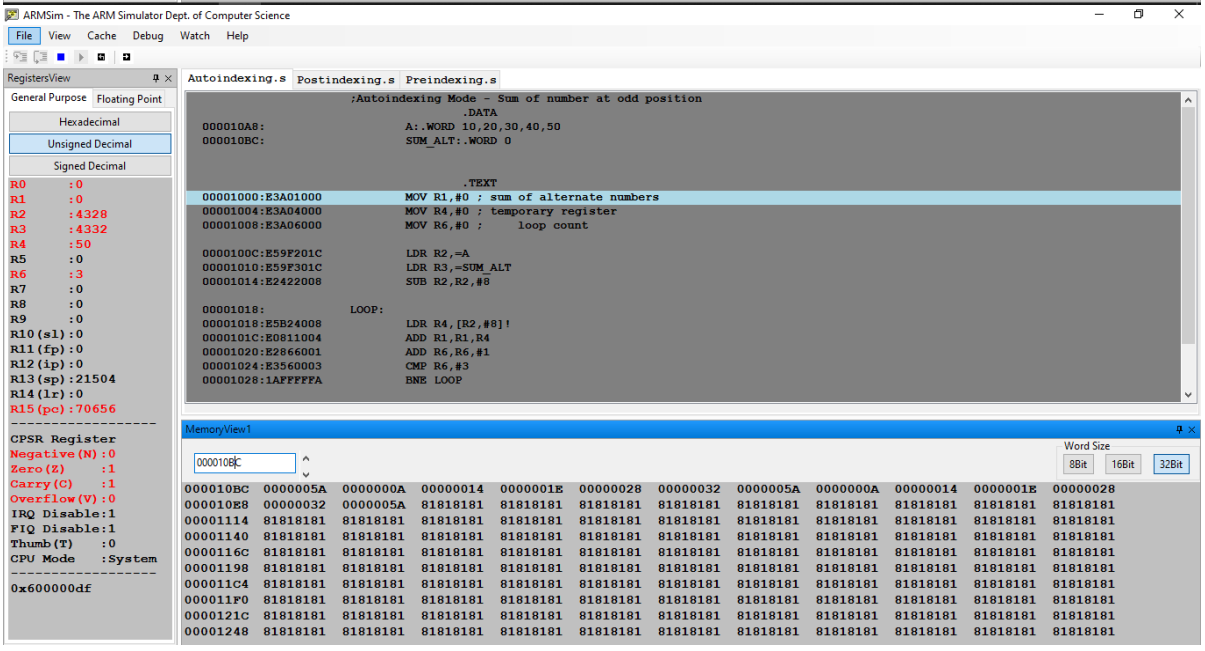


c. Use Auto-indexing addressing mode

```
;Autoindexing Mode - Sum of number at odd position
        .DATA
    A:.WORD 10,20,30,40,50
    SUM_ALT:.WORD 0


        .TEXT
    MOV R1,#0 ; sum of alternate numbers
    MOV R4,#0 ; temporary register
    MOV R6,#0 ;         loop count

    LDR R2,=A
    LDR R3,=SUM_ALT
    SUB R2,R2,#8

LOOP:
    LDR R4,[R2,#8]!
    ADD R1,R1,R4
    ADD R6,R6,#1
    CMP R6,#3
    BNE LOOP

STR R1,[R3]
```

.END


ARMSim - The ARM Simulator Dept. of Computer Science
File View Cache Debug Watch Help
RegistersView
General Purpose Floating Point
Hexadecimal
Unsigned Decimal
Signed Decimal
R0 :0
R1 :0
R2 :0
R3 :0
R4 :0
R5 :0
R6 :0
R7 :0
R8 :0
R9 :0
R10(sl):0
R11(fp):0
R12(ip):0
R13(sp):21504
R14(lr):0
R15(pc):4096
CPSR Register
Negative(N):0
Zero(Z) :0
Carry(C) :0
Overflow(V):0
IRQ Disable:1
FIQ Disable:1
Thumb(T) :0
CPU Mode :System
0x000000df
Autoindexing.s Postindexing.s Preindexing.s
;Autoindexing Mode - Sum of number at odd position
.DATA
000010A8: A:.WORD 10,20,30,40,50
000010BC: SUM_ALT:.WORD 0
.TEXT
00001000:E3A01000 MOV R1,#0 ; sum of alternate numbers
00001004:E3A04000 MOV R4,#0 ; temporary register
00001008:E3A06000 MOV R6,#0 ; loop count
0000100C:E59F201C LDR R2,=A
00001010:E59F301C LDR R3,=SUM_ALT
00001014:E2422008 SUB R2,R2,#8
00001018: LOOP:
00001018:E5B24008 LDR R4,[R2,#8]!
0000101C:E0811004 ADD R1,R1,R4
00001020:E2866001 ADD R6,R6,#1
00001024:E3560003 CMP R6,#3
00001028:1AFFFFFA BNE LOOP
MemoryView1
000010BC
Word Size
8Bit 16Bit 32Bit
000010BC 00000000 0000000A 00000014 0000001E 00000028 00000032 00000000 0000000A 00000014 0000001E 00000028
000010E8 00000032 00000000 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001114 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001140 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
0000116C 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001198 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000011C4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000011F0 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
0000121C 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001248 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181


ARMSim - The ARM Simulator Dept. of Computer Science
File View Cache Debug Watch Help
RegistersView
General Purpose Floating Point
Hexadecimal
Unsigned Decimal
Signed Decimal
R0 :0
R1 :0
R2 :4328
R3 :4332
R4 :50
R5 :0
R6 :3
R7 :0
R8 :0
R9 :0
R10(sl):0
R11(fp):0
R12(ip):0
R13(sp):21504
R14(lr):0
R15(pc):70656
CPSR Register
Negative(N):0
Zero(Z) :1
Carry(C) :1
Overflow(V):0
IRQ Disable:1
FIQ Disable:1
Thumb(T) :0
CPU Mode :System
0x600000df
Autoindexing.s Postindexing.s Preindexing.s
;Autoindexing Mode - Sum of number at odd position
.DATA
000010A8: A:.WORD 10,20,30,40,50
000010BC: SUM_ALT:.WORD 0
.TEXT
00001000:E3A01000 MOV R1,#0 ; sum of alternate numbers
00001004:E3A04000 MOV R4,#0 ; temporary register
00001008:E3A06000 MOV R6,#0 ; loop count
0000100C:E59F201C LDR R2,=A
00001010:E59F301C LDR R3,=SUM_ALT
00001014:E2422008 SUB R2,R2,#8
00001018: LOOP:
00001018:E5B24008 LDR R4,[R2,#8]!
0000101C:E0811004 ADD R1,R1,R4
00001020:E2866001 ADD R6,R6,#1
00001024:E3560003 CMP R6,#3
00001028:1AFFFFFA BNE LOOP
MemoryView1
000010BC
Word Size
8Bit 16Bit 32Bit
000010BC 0000005A 0000000A 00000014 0000001E 00000028 00000032 0000005A 0000000A 00000014 0000001E 00000028
000010E8 00000032 0000005A 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001114 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001140 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
0000116C 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001198 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000011C4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000011F0 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
0000121C 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001248 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181