PES UNIVERSITY

Department of Computer Science & Engineering

Microprocessor & Computer Architecture Lab

UE23CS251B

WEEK 4 submission

Name of the Student	**Amit Prakash**
SRN	**PES1UG23AM042**
Section	**A**
Department	**CSE - AIML**
Campus	**RR**

UE23CS251B

1 Write an ALP using ARM7TDMI to add n numbers bytewise.

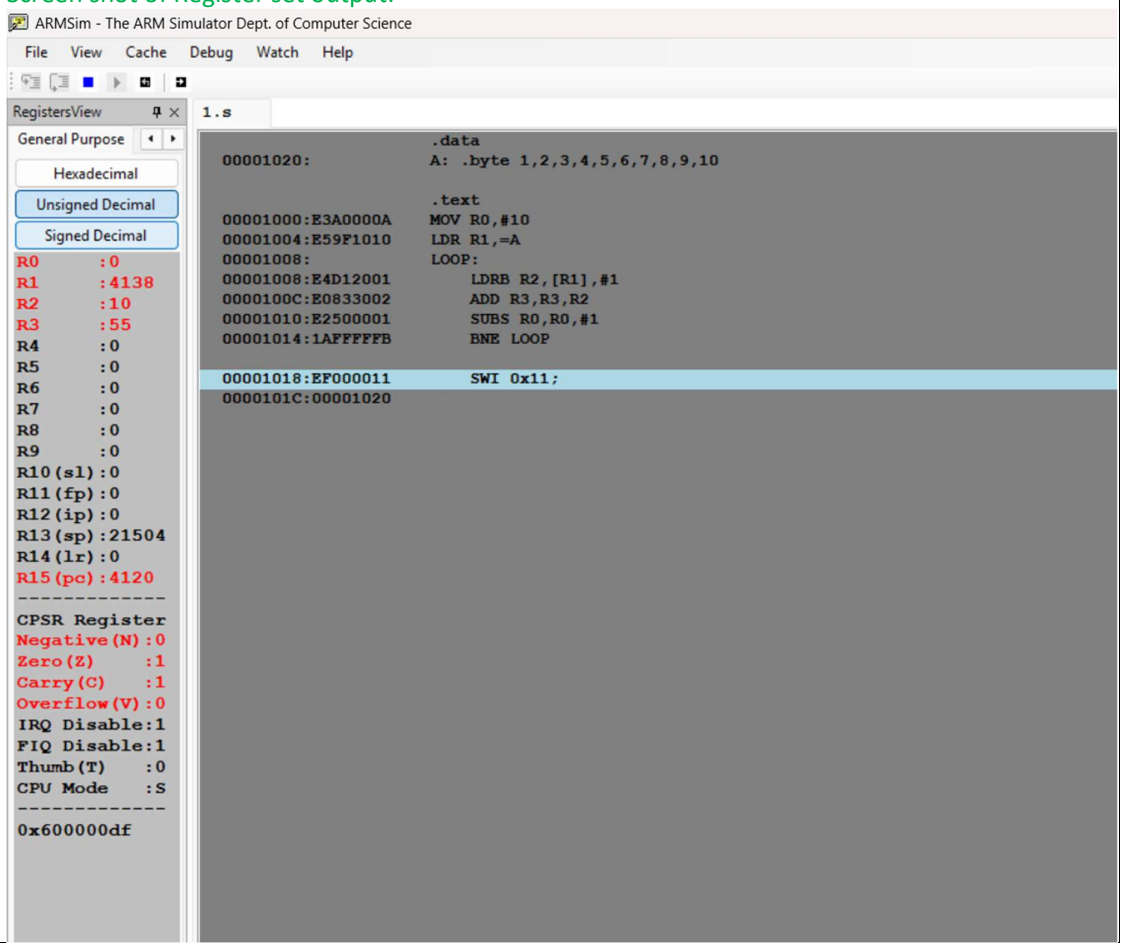
. .DATA

A: .byte 1,2,3,4,5,6,7,8,9,10

Program screen shot:

```
week-4 > ≡ 1.s
1   .data
2   A: .byte 1,2,3,4,5,6,7,8,9,10
3
4   .text
5   MOV R0,#10
6   LDR R1,=A
7   LOOP:
8       LDRB R2,[R1],#1
9       ADD R3,R3,R2
10      SUBS R0,R0,#1
11      BNE LOOP
12
13      SWI 0x11;
14
```

Screen shot of Register set output:



2 Write an ALP using ARM7TDMI to generate a square given matrix with A

If (i==j) then A[i][j]=5
Otherwise A[i][j]=0
(Note:Any size of matrix can be given as input)

Considering 4X4 matrix

Example :

5	0	0	0
0	5	0	0
0	0	5	0
0	0	0	0

Before:
A:.word 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16

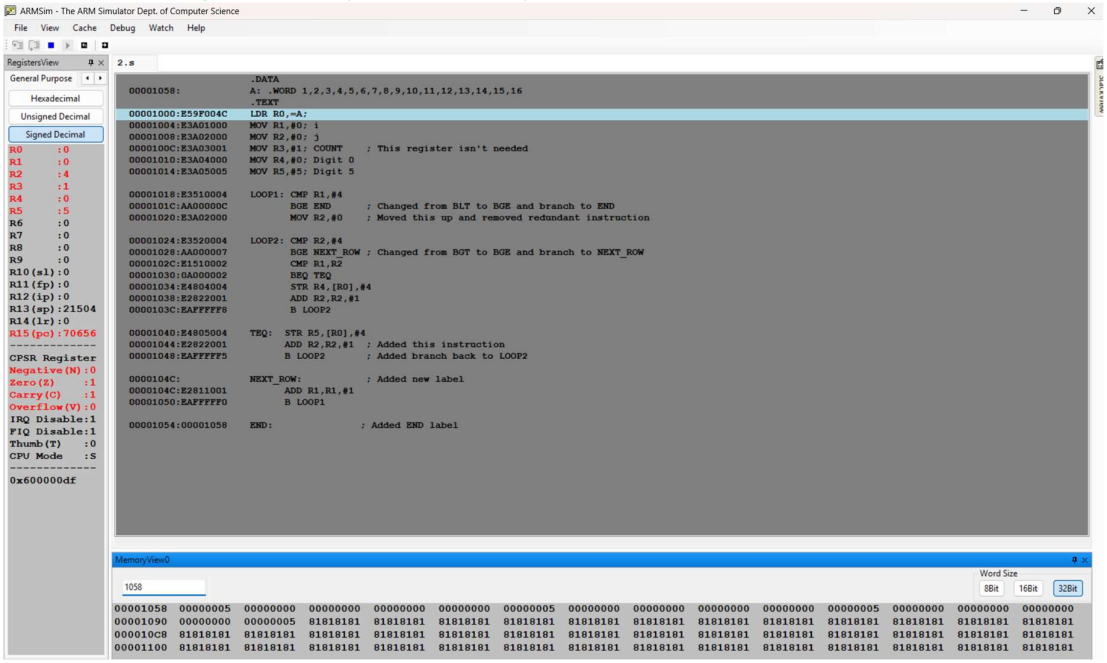
After:
A:.word 5,0,0,0,0,5,0,0,0,0,5,0,0,0,0,5

Program screen shot:

week-4 > 2.s

```
3   .TEXT
4   LDR R0,=A;
5   MOV R1,#0; i
6   MOV R2,#0; j
7   MOV R3,#1; COUNT    ; This register isn't needed
8   MOV R4,#0; Digit 0
9   MOV R5,#5; Digit 5
10
11  LOOP1: CMP R1,#4
12         BGE END      ; Changed from BLT to BGE and branch to END
13         MOV R2,#0    ; Moved this up and removed redundant instruction
14
15  LOOP2: CMP R2,#4
16         BGE NEXT_ROW ; Changed from BGT to BGE and branch to NEXT_ROW
17         CMP R1,R2
18         BEQ TEQ
19         STR R4,[R0],#4
20         ADD R2,R2,#1
21         B LOOP2
22
23  TEQ:  STR R5,[R0],#4
24        ADD R2,R2,#1  ; Added this instruction
25        B LOOP2       ; Added branch back to LOOP2
26
27  NEXT_ROW:           ; Added new label
28        ADD R1,R1,#1
29        B LOOP1
30
31  END:              ; Added END label
```

Screen shot of Register set output and memory location:



3 Write an ALP using ARM7TDMI to convert hexadecimal to decimal.

Input: **0x0124**(Hex)

Conversion Process:

- 0x124→ Extract **4**→ 0 * 16 + 4 = 4
- 0x124 → Extract **2** → 2 * 16^1 + 4 = 36
- 0x124 → Extract **1** → 1* 16^2 + 36 =292

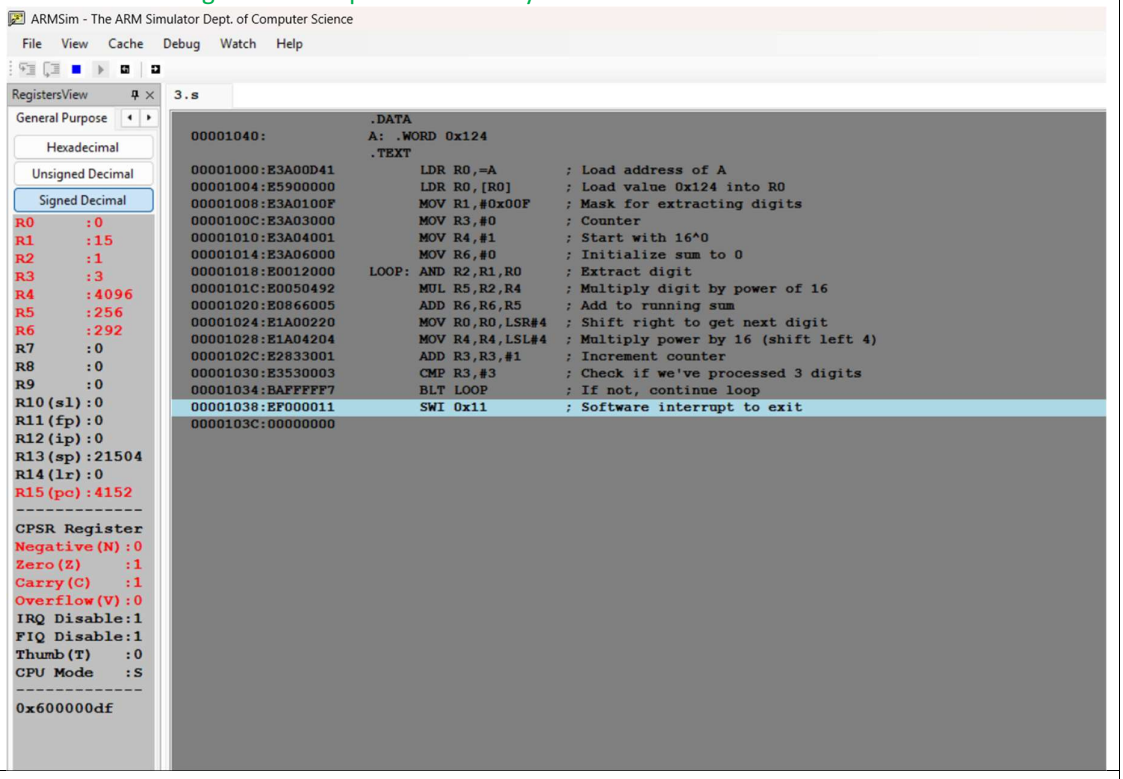
Output: **292(**Decimal)

Program screen shot:

week-4 > 3.s

```
1   .DATA
2   A: .WORD 0x124
3   .TEXT
4       LDR R0,=A       ; Load address of A
5       LDR R0,[R0]     ; Load value 0x124 into R0
6       MOV R1,#0x00F   ; Mask for extracting digits
7       MOV R3,#0       ; Counter
8       MOV R4,#1       ; Start with 16^0
9       MOV R6,#0       ; Initialize sum to 0
10  LOOP: AND R2,R1,R0  ; Extract digit
11      MUL R5,R2,R4    ; Multiply digit by power of 16
12      ADD R6,R6,R5    ; Add to running sum
13      MOV R0,R0,LSR#4 ; Shift right to get next digit
14      MOV R4,R4,LSL#4 ; Multiply power by 16 (shift left 4)
15      ADD R3,R3,#1    ; Increment counter
16      CMP R3,#3       ; Check if we've processed 3 digits
17      BLT LOOP        ; If not, continue loop
18      SWI 0x11        ; Software interrupt to exit
```

Screen shot of Register set output and memory location:



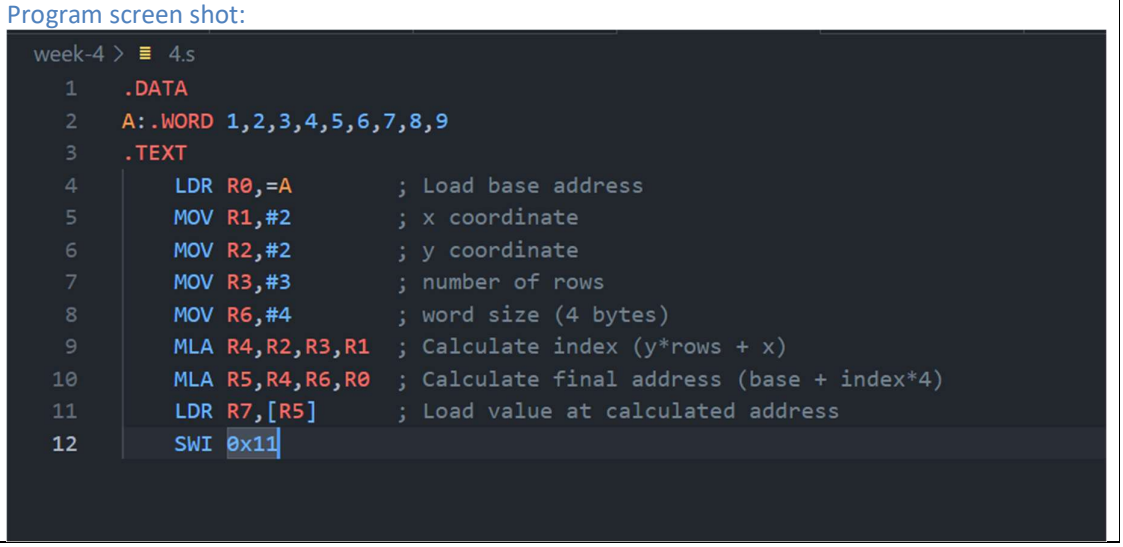
4 Write an ALP using ARM7TDMI, for the given matrix arranged in Column major order, find the index of an element if coordinates of a matrix is given and also find the address of the indexed element. (Using MLA instruction)

. DATA

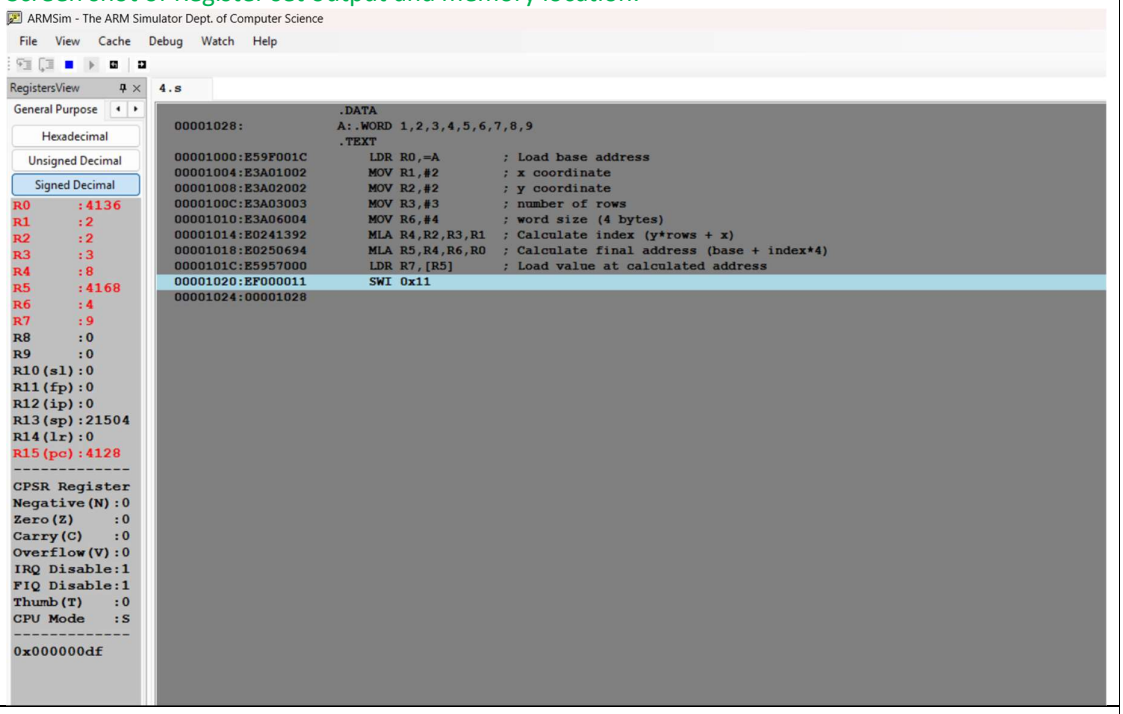
A:.WORD 1,2,3,4,5,6,7,8,9

.Index for the column major= y*no of rows+x

Program screen shot:



Screen shot of Register set output and memory location:



Assignments Questions

5 Write an ALP using ARM7TDMI to reverse the elements stored in location A with location B

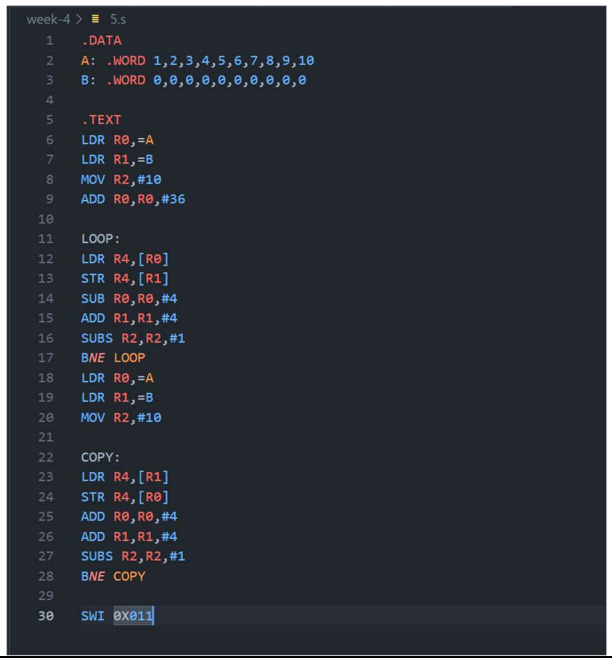
Before:

A:.word 1,2,3,4,5,6,7,8,9,10

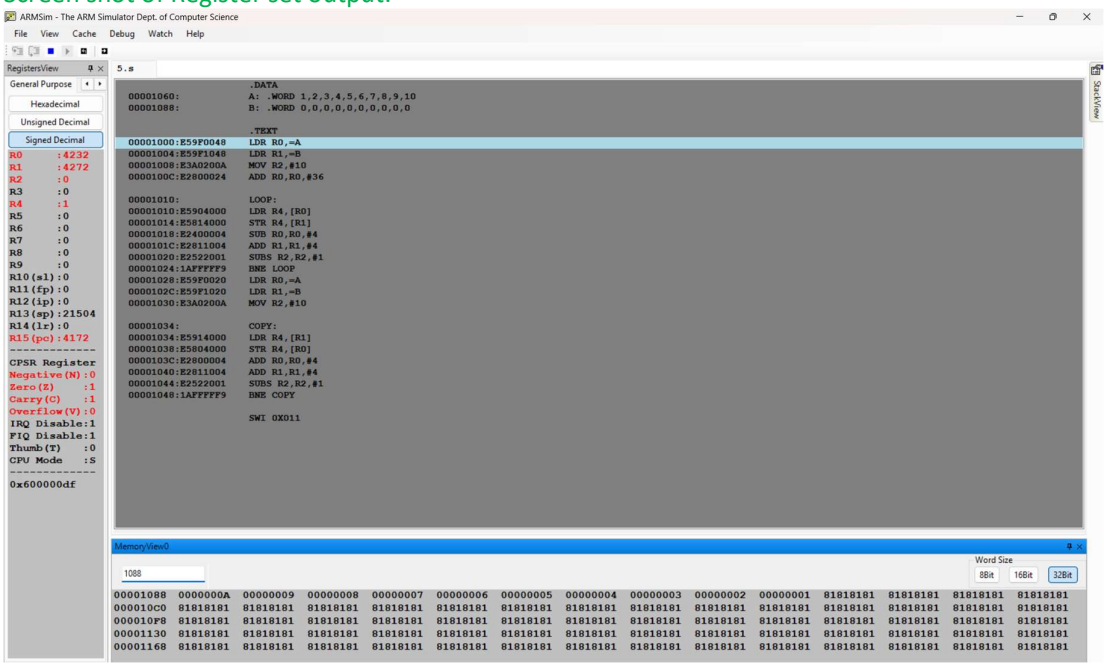
After :

A:.word 10,9,8,7,6,5,4,3,2,1

Program screen shot:



Screen shot of Register set output:



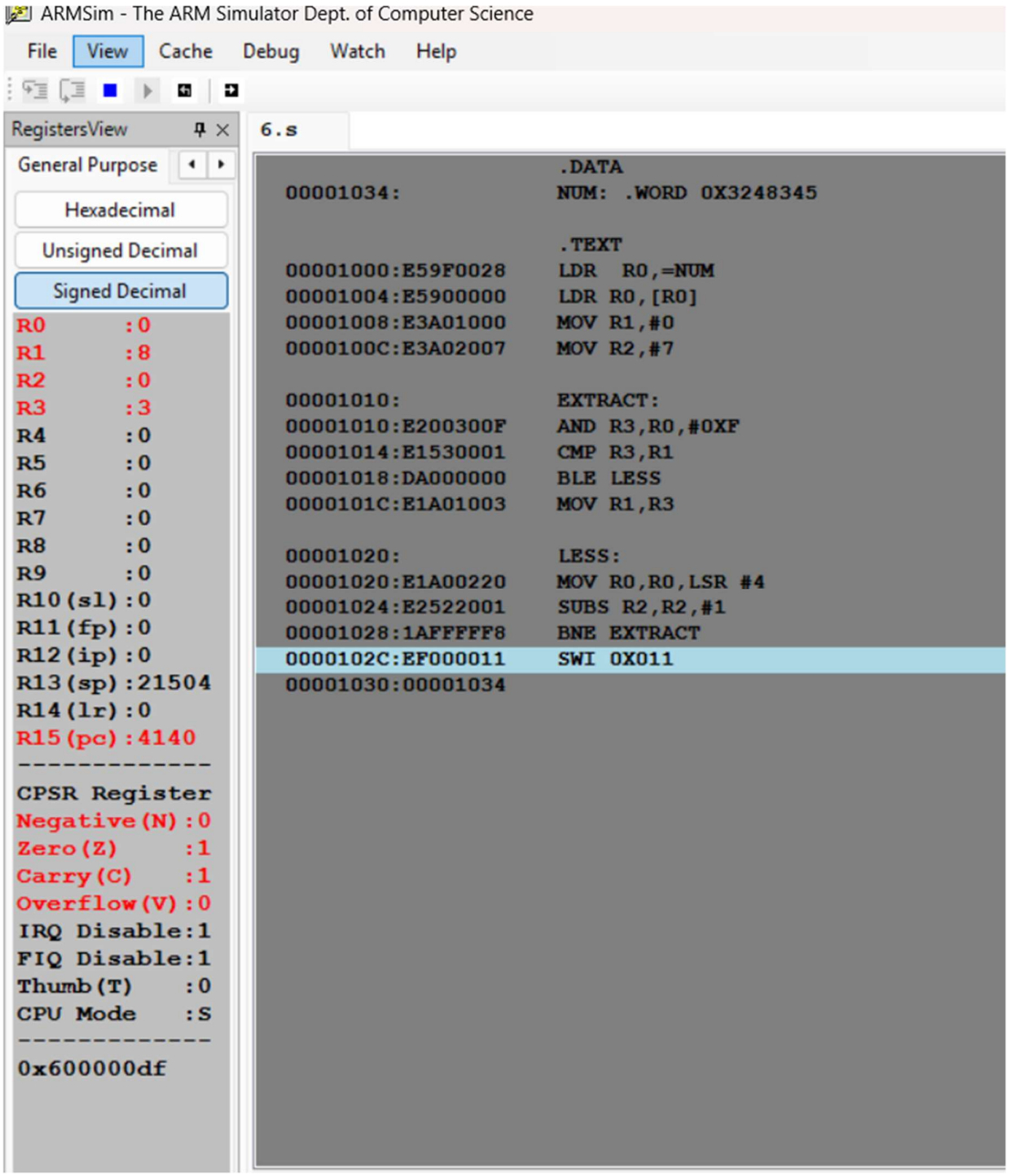
6 Write an ALP using ARM7TDMI to find the largest of all the BCD digits of a given 32bit number. (hint:If R1=17845374 the largest digit is 8

Program screen shot:

week-4 > 6.s

```
1   .DATA
2   NUM: .WORD 0X3248345
3
4   .TEXT
5   LDR  R0,=NUM
6   LDR R0,[R0]
7   MOV R1,#0
8   MOV R2,#7
9
10  EXTRACT:
11  AND R3,R0,#0XF
12  CMP R3,R1
13  BLE LESS
14  MOV R1,R3
15
16  LESS:
17  MOV R0,R0,LSR #4
18  SUBS R2,R2,#1
19  BNE EXTRACT
20  SWI 0X011
```

Screen shot of Register set output:



Extra
Write an ALP using ARM7TDMI to copy a block 400 bytes of data from location A to location B if the rate of data transfer rate is 40 bytes, LDM and STM instructions. and For the same transfer the block with auto-indexing.