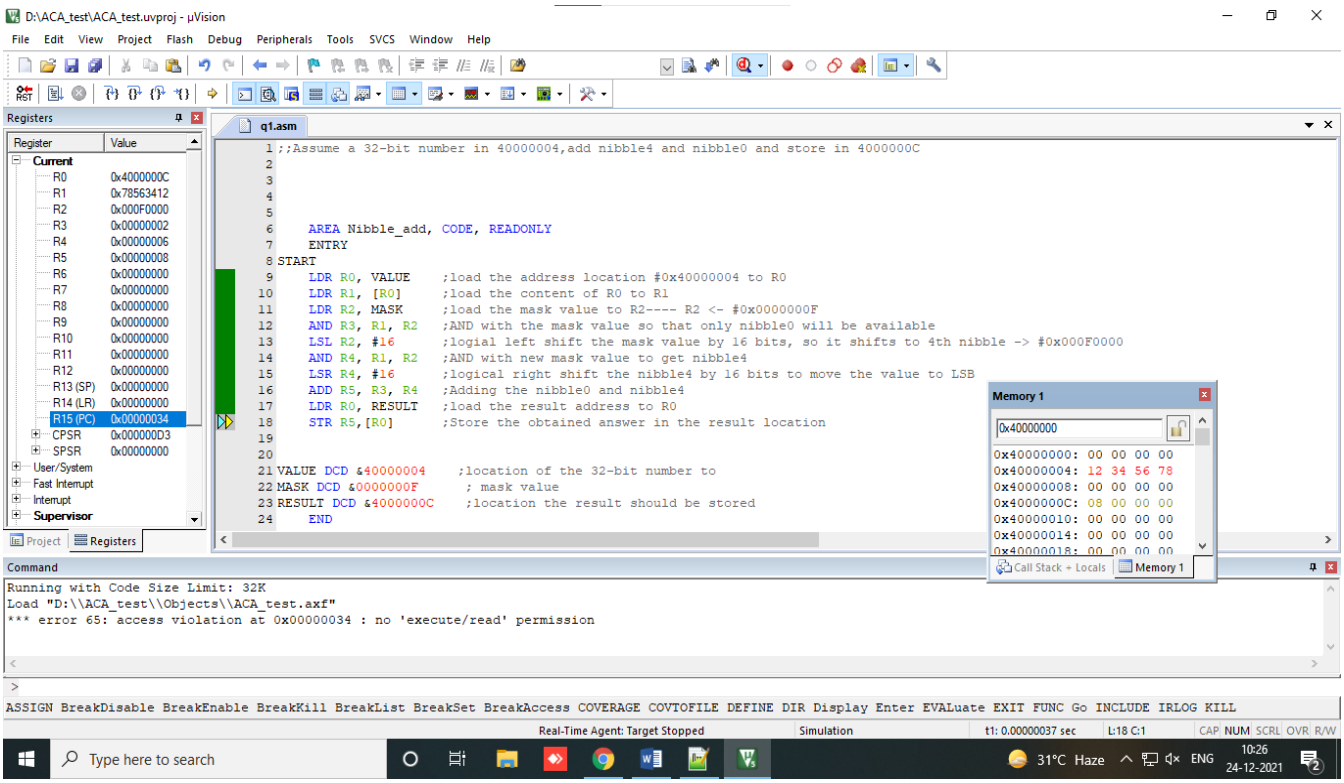
## 211039016

## I.Bharath Simha Reddy

## Program 1:

```
        AREA Nibble_add, CODE, READONLY
        ENTRY
START
        LDR R0, VALUE    ;load the address location #0x40000004 to R0
        LDR R1, [R0]     ;load the content of R0 to R1
        LDR R2, MASK     ;load the mask value to R2---- R2 <- #0x0000000F
        AND R3, R1, R2   ;AND with the mask value so that only nibble0 will be available
        LSL R2, #16      ;logial left shift the mask value by 16 bits, so it shifts to 4th nibble -> #0x000F0000
        AND R4, R1, R2   ;AND with new mask value to get nibble4
        LSR R4, #16      ;logical right shift the nibble4 by 16 bits to move the value to LSB
        ADD R5, R3, R4   ;Adding the nibble0 and nibble4
        LDR R0, RESULT   ;load the result address to R0
        STR R5,[R0]      ;Store the obtained answer in the result location


VALUE DCD &40000004    ;location of the 32-bit number to
MASK DCD &0000000F     ; mask value
RESULT DCD &4000000C    ;location the result should be stored
        END
```




**Output:**

Given the input as 12345678

Number is stored as 78563412

Nibble0 ->2

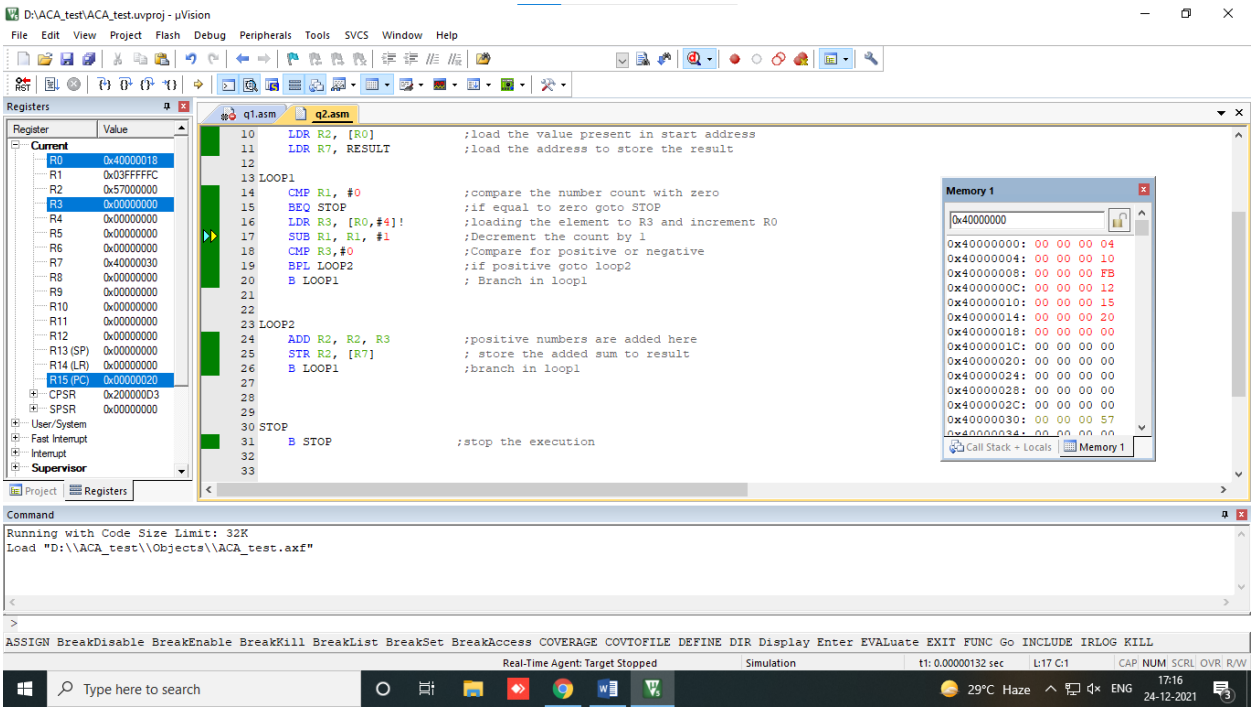
Nibble4 ->6

Output sum is 8

**Program 2.**

```
                    AREA PG2, CODE, READWRITE
                    ENTRY

START
                    LDR R0, COUNT           ; load the count to R0
                    LDR R1, [R0]            ; load the content of R0 to R1
                    LDR R0, ARRAY           ; loading the start
address of the array
                    LDR R2, [R0]            ;load the value
present in start address
                    LDR R7, RESULT          ;load the address to
store the result

LOOP1
                    CMP R1, #0              ;compare the number
count with zero
                    BEQ STOP                ;if equal to zero goto
STOP
                    LDR R3, [R0,#4]!        ;loading the element to R3
and increment R0
                    SUB R1, R1, #1          ;Decrement the count by 1
                    CMP R3,#0               ;Compare for positive
or negative
                    BPL LOOP2               ;if positive goto loop2
                    B LOOP1                 ; Branch in loop1
```

```
LOOP2
                    ADD R2, R2, R3          ;positive numbers are
added here
                    STR R2, [R7]            ; store the added sum
to result
                    B LOOP1                 ;branch in
loop1


STOP
                    B STOP          ;stop the execution


COUNT DCD &40000000
ARRAY DCD &40000004
RESULT DCD &40000030


                    END
```




**Output:**

Given input:

Number count : 5

Array elements :

10

-10

12

15

20

Sum : 57 (-10 is ignored)