**Experiment 9**

**Aim:**

To write an ARM Assembly Language program for arranging the number in to ascending and descending order.

**Tool Used:**

Keil uVision4

**Theory:**

LDR loads the register with some value. One register can be used as a counter. The first number can be subtracted by second number. On every loop the counter register is incremented on to the result.

**Code(ascending):**

 AREA PROGRAM, CODE, READONLY

 ENTRY

MAIN

 MOV R0, #9

LOOP1    LDR R1, =0X00001000

         ADD R2, R1, #1

         MOV R3, R0

LOOP2    LDRB R4, [R1]

         LDRB R5, [R2]

         CMP R4, R5

         STRCSB R4, [R2]

         STRCSB R5, [R1]

         ADD R1, R1, #1

         ADD R2, R2, #1

         SUBS R3, R3, #1

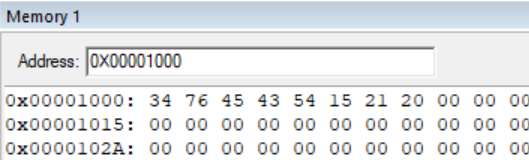
         BNE LOOP2

         SUBS R0, R0, #1

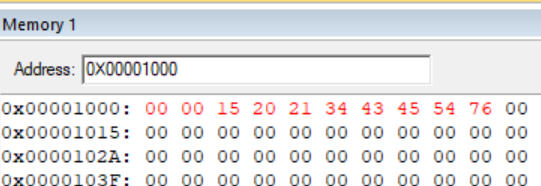
         BNE LOOP1

         END

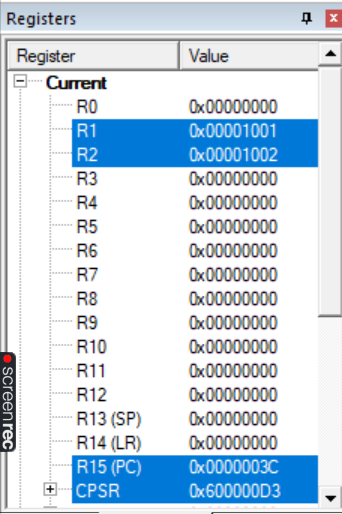
**Input:**

****

**Output:**

****

**Register Content**

****

**Code(descending):**

 AREA PROGRAM, CODE, READONLY

 ENTRY

MAIN

 MOV R0, #9

LOOP1    LDR R1, =0X00001000

         ADD R2, R1, #1

         MOV R3, R0

LOOP2    LDRB R4, [R1]

         LDRB R5, [R2]

         CMP R4, R5

         STRCCB R4, [R2]

         STRCCB R5, [R1]

         ADD R1, R1, #1

         ADD R2, R2, #1

         SUBS R3, R3, #1

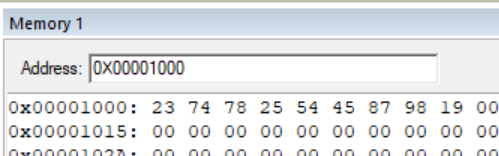
         BNE LOOP2

         SUBS R0, R0, #1

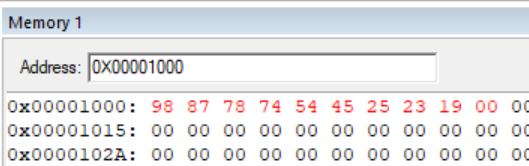
         BNE LOOP1

         END

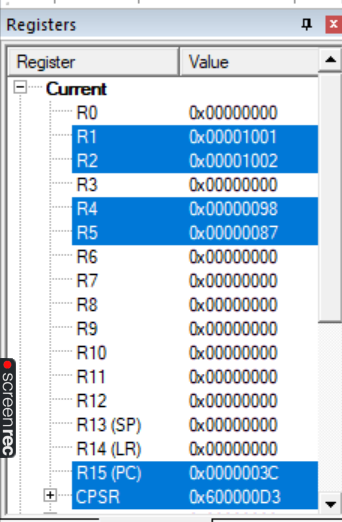
**Input:**

****

**Output:**

****

**Register Content:**

****

**Result:**

The experiments on arranging the number in to ascending and descending order have been performed and verified to be correct.