EE2016F22 Microprocessor Lab
Experiment 5 Lab Report

Ameya (EE21B014)

22 October 2022

Contents

List of Figures

1 Aim

To

1. learn the architecture of ARM processor
2. learn basics of ARM instruction set, in particular the ARM instructions pertaining to computations
3. go through example programs and
4. write assembly language programs for the given set of (computational) problems

2 Equipments, Hardwares / Softwares Required

1. KEIL 5 IDE for ARM
2. Flashmagic software for programming flash memory
3. ARM7 hardware kit
4. USB to serial converter
5. Serial cross cable

3 Tasks to be Performed

3.1 Compute the factorial of a given number using ARM processor through assembly programming

The ARM code used to implement the given task has been given below :

```
  AREA Program, CODE, READONLY
Entry
  LDR R0, NUM1
  SUB R1, R0, #1
label
  CMP R1, #0
  BEQ done
  MUL R2, R1, R0
  MOV R0, R2
  SUB R1, R1, #1
done
  LDR R3, =Result
  STR R0, [R3]
  SWI &11
NUM1 DCW &00003
```

```
align
  AREA DataRam, DATA, READWRITE
Result DCD 0
END
```

ARM code

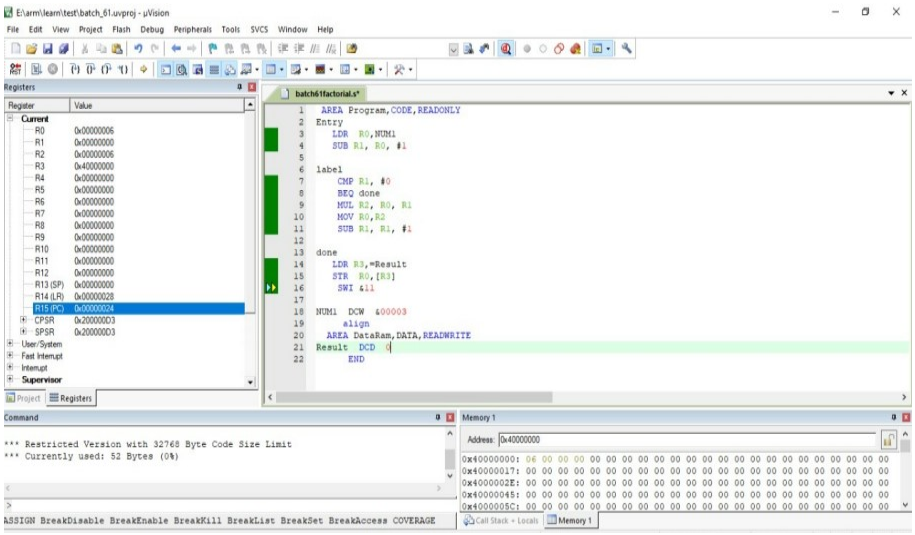

Figure 1: Output on debugging

3.2 Combine the low four bits of each of the four consecutive bytes beginning at LIST into one 16-bit halfword. The value at LIST goes into the most significant nibble of the result. Store the result in the 32-bit variable RESULT.

The ARM code used to implement the given task has been given below :

```
  AREA Program,CODE,READONLY
Entry
  LDR R0, =LIST
  MOV R5, #0
  MOV R6, #0
  MOV R9, #16
Loop
  CMP R6, #4
  BEQ done
  LDR R1, [R0]
```

```
  MOV R2, #15
  AND R3, R2, R1
  MOV R12, R5
  MUL R5, R12, R9
  ADD R5, R5, R3
  ADD R0, R0, #4
  ADD R6, R6, #1
B Loop
done
  LDR R7, =Result
  STR R5, [R7]
  SWI &11
 LIST DCD &2D3F, &5F53, &1234, &0987
align
AREA DataRam,DATA,READWRITE
Result DCD 0
END
```

ARM code

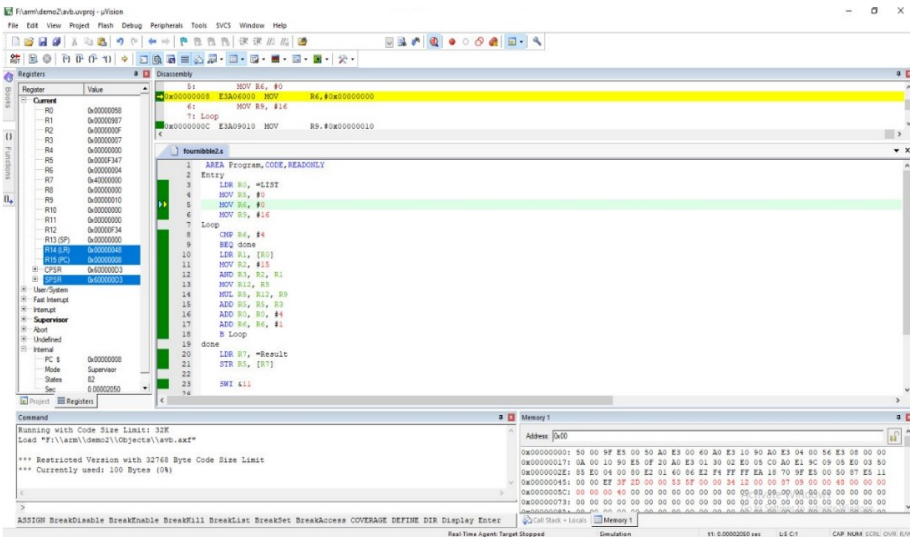

Figure 2: Final Output

3.3 Given a 32 bit number, identify whether it is an even or odd (without division)

The ARM code is given below

```
  AREA Program, CODE, READONLY
Entry
```

```
  LDR R0, NUM1
  MOV R1, #1
  AND R2, R1, R0
  CMP R2, #0
  BEQ even
  MOV R3, #1
  B done
even
  MOV R3, #0
done
  LDR R4, =Result
  STR R3, [R4]
  SWI &11
NUM1 DCD &00003
align
  AREA DataRam, DATA, READWRITE
Result DCD 0
  END
```

ARM code

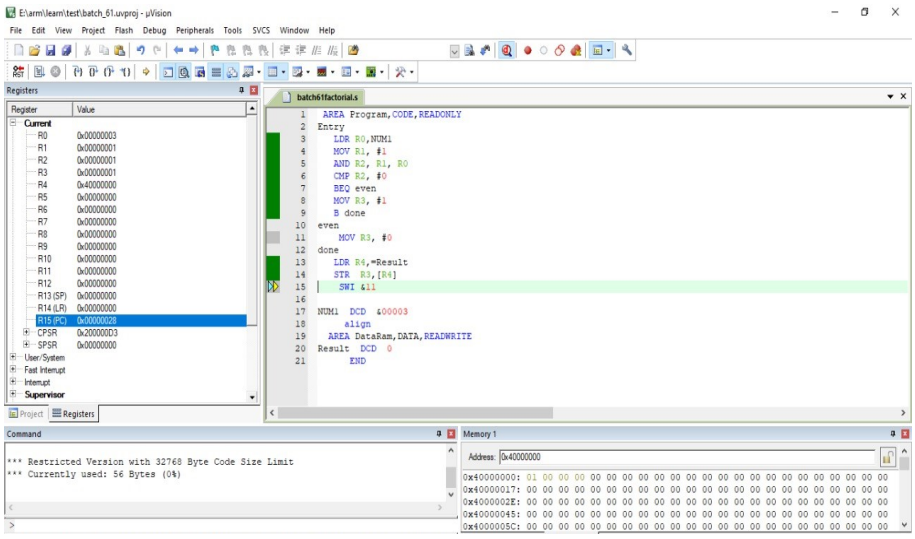

Figure 3: KEIL Output