2/11/23

Lab-02 - Programs on Arithmatic Instructions

(1) Write a program to add ten 32 bit numbers available in code memory and store in data memory.

```
     AREA RESET, DATA, READONLY
     EXPORT __Vectors
__Vectors
       DCD  0x40001000
       DCD  Reset_Handler
       ALIGN
       AREA mycode, CODE, READONLY
       ENTRY
       EXPORT  Reset_Handler
Reset_Handler
       LDR  R0,=SRC
       LDR  R2,=DST
       MOV R5,#10
       ~~LDR~~ R
LOOP   LDR  R1,[R0],#4
       LDR R3,[R2]
       ADDS R6,R1,R3
       STR  R6,[R2]
       SUBS  R5,R5,#1
       BNE LOOP
STOP B STOP
SRC DCD 0x1,0x2,0x3,0x4,0x5,0x6,0x7,0x8,0x9,0xA
~~Area~~ AREA data, DATA, READWRITE
DST ~~DST~~ DCD 0
       END
```

Output :

```
R0 - 0x0000020  →  0x000000048
R1 - 0x1000000  →  0x10000000
R2 - 0x0000000A →  0x00000000
R4 - 0x000000000 →  0x0000002D
```

② Write a program to add two 128 bit numbers

```
    AREA  RESET, DATA, READONLY
    EXPORT __Vectors
__Vectors
    DCD 0x10001000
    DCD Reset_Handler
    ALIGN
    AREA mycode, CODE, READONLY
ENTRY
      EXPORT Reset_Handler
Reset_Handler
      LDR  R0, =NUM1
      LDR  R1,=NUM2
      LDR  R7,=DST
      MOV R2,#4
LOOP  LDR  R3, [R0], #4
      LDR  R4, [R1], #4
      ADD  R6,R6,R5
      ADDS  R6,R3,R4
      ADC  R5, #0
      SUBS  R2,R2,#1
      STR  R6, [R7], #4
      BNE  LOOP
STOP  B  STOP
NUM1  DCD 1,2,3,4
NUM2  DCD 4,3,2,1
      AREA data, DATA, READWRITE
DST   DCD 0
      END
```

Output

Before 1st

R0 → 0x02C → 0x30 → 0x34 → 0x38 → 0x3C

R1 → 0x03C → 0x40 → 0x44 → 0x48 → 0x4C

R2 → 0x004 → 0x03 → 0x02 → 0x01 → 0x0

R3 → 0x0 → 0x01 → 0x62 → 0x03 → 0x04

R4 → 0x0 → 0x04 → 0x03 → 0x02 → 0x01

R5 → 0x0 → 0x00 → 0x00 → 0x00 → 0x00

R6 → 0x0 → 0x05 → 0x05 → 0x05 → 0x05

R7 → 0x10001000 → 0x1000004 → 0x10001008 → 0x1000100C → 0x10000010

Memory

0x10000000: 0500 00 00 05 000000 05000000 05000000

③ Write a program to subtract two 32 bit numbers

```
    AREA   RESET, DATA, READONLY
    EXPORT __Vectors
__Vectors
    DCD  0X40001000
    DCD  Reset_Handler
    ALIGN
    AREA mycode ,CODE, READONLY
    ENTRY
    EXPORT  Reset_Hander
Reset_Hander
     LDR  R0, =Value1
     LDR  R1, ~~=Value2~~ [R0]
     LDR  R0 , =Value 2
     LDR  R3,[R0]
     SUBS  R6, R1, R3
     LDR  R2 , =RESULT
     STR   R6, [R2]
STOP  B STOP
Value1    DCD  0X00000007
Value 2   DCD  0X00000002
      AREA  data ,DATA, Readwrite
RESULT  DCD  0
       END
```

Output:

```
R0  0x0  ⟶  0x1C
R1  0x0  ⟶  0x07
R2  0x0  ⟶  0x10000000
R3  0x0  ⟶  0x02
R6  0x0  ⟶  0x05
```

(4) Write a program to subtract two 128 bit numbers available in code memory and store result in data memory.

```
        AREA  RESET, DATA, READONLY
        EXPORT __Vectors
__Vectors
        DCD  0X10001000
        DCD  Reset_Handler
    ENTRY
    ~~Reset_Handler~~
        EXPORT Reset_Handler
Reset_Handler
        LDR  R0,=NUM1
        LDR  R1,=NUM2
        MOV  R2,#4
        LDR  R7,=DST
LOOP    LDR  R3,[R0],#4
        LDR  R4,[R1],#4
        ADDS R3,R3,R5
        SUBS R6,R3,R4
        SBC  R5,#0
        STR  R6,[R7],#4
        SUBS R2,#1
        BNE  LOOP
STOP    B  STOP
NUM1    DCD 5,4,0,4
NUM2    DCD 4,3,1,2
        AREA  data, DATA, READWRITE
DST     DCD  0
        END
```

Output

R0 → 0x2C → 0x30 → 0x34 → 0x38 → 0x3C

R1 → 0x3C → 0x40 → 0x44 → 0X48 → 0x4C

R2 → 0x04 → 0x03 → 0x02 → 0x01 → 0x00

R3 → 0x00 → 0x05 → 0x04 → 0x00 → 0x03

R4 → 0x00 → 0x04 → 0x03 → 0x01 → 0x02

R5 → 0x00 → 0x00 → 0x00 → 0xFF⋯F → 0xFF⋯F

R6 → 0x00 → 0x01 → 0x1 → 0xF⋯F → 0x01

R7 → 0x1000000 → 0x1000004 → 0x10000008 → 0x1000000C → 0x1D000010

Memory!

0x100000000: 01 00 00 00 01000000 FF FFFF FF 01 000000