URBAN EDGE

Lab-04

(1) Convert 32 bit packed BCD to equivalent hexadecimal number

```
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, =SRC
    LDR R1, [R0]
    LDR R2, =DEST
    MOV R3, #1
    MOV R4, #0x0A
    MOV R5, #0
LOOP AND R7, R1, #0x0F
    MLA R5, R7, R3, R5
    MUL R3, R3, R4
    LSR R1, #4
    CMP R1, #0
    BNE LOOP
    STR R5, [R2]
STOP B STOP
SRC DCD 0x123
    AREA mydata, DATA, READWRITE
DEST DCD 0
    END.
```

Output

R0 = 0x 00 → 0x34

R1 0x00 → 0x10000000

R3 0x000 → 0x3E8

R4 0x00 → 0x0A

R5 0x00 → 0x7B

R7 0x00 → 0x01

0x10000000 → 7B 00 00 00

(2) WAP to convert a 16 bit hexadecimal to 32 bit unpacked BCD.

```
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, =SRC
   LDR R1, [R0]
   MOV R2, #10
   LDRB R3, [R0]
   MOV R4, #0
   MOV R5, #1
   MOV R6, #0
   MOV R7, #0
```

```
div10   SUB  R3,R3,R2
        ADD  R6,R6,#1
        CMP  R3,R2
        BHI  div10
        BLS  addto

addto
        ADD  R7,R7,#1
        ADD  R4,R4,R3
        LSL  R4,#4
        MOV  R3,R6
        MOV  R6,#0
        CMP  R3,R2
        BHI  div10
        MUL  R3,R3,R3
        ADD  R4,R4,R3
        MOV  R5,#0

COV     ADD  R1,R4,#0X0F
        ADD  R5,R5,R1
        LSR  R4,#4
        CMP  R4,#0
        BLS  final
        LSL  R5,#4
        BHI  COV

final   LDR  R8,=DEST
        STR  R5,#4
STOP    B    STOP
SRC     DCD  0X7B
        AREA mydata, DATA, READWRITE
DEST    DCD  0
        END
```

URBAN EDGE

Output

R0 → 0x00 → 0x6C

R1 → 0x00 → 0x03

R2 → 0x00 → 0x0A

R3 → 0x00 → 0x01

R4 → 0x00 → 0x00

R5 → 0x0000 → 0x123

R7 → 0x00 → 0x02

R8 → 0x10000000

0x10000000 → 23 01 00 00

(3) Add two 32 bit packed BCD and store results in packed BCD

```
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 R2, [R0]
LDR R0, =NUM2
LDR R3, [R0]
LDR R0, =DEST1
MOV R4, #0
LOOP MOV R7, R2
MOV R8, R3
AND R7, #0x0f
ADDS R7, R4
MOV R4, #0
```

```
        AND R8, #0X0F
        ADDS R7, R4
        MOV R4, #0
        AND R8, #0X0F
        BL ADDN
        LSR R2, #4
        LSR R3, #4
        cmp R2, #0
        BNE
        Cmp R3, #0
        BNE loop
PKIN    MOV R12, R0
        LDR R0, =DEST1
        LDR R1, =DEST2
        SUB R4, R12, R0
UP1     LDRB R9, [R0], #1
        LDRB R10, [R0], #1
        LSL R10, #4
        ORR R10, R9
        STRB R10, [R1], #1
        SUB R4, #2
        Cmp R4, #1
        BGE UP1
STOP    B STOP
ADDN    ADDS R7, R8
        cmp R7, #10
        BLO STORE
        SUB R7, #10
        ADD R6, #01
STORE   STRB R7, [R0], #1
        BX LR
NUM1    DCD 0x35
NUM2    DCD 0x10
        AREA mydata, DATA, READWRITE
```

Dest1 DCD 0

Dest2 DCD 0

END

Output

R0 → 0x00 → 0x10000002

R1 → 0x10000005

R7 → 0x00 → 0x04

R8 → 0x00 → 0x01

R9 → 0x00 → 0x05

R10 → 0x00 → 0x45

R12 → 0x00 → 0x10000002

R

Output :

0x10000000 : 05 04 00 00 45