**Code Convertor related programs**

**;ARM ALP to convert hexadecimal to decimal**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area hello,code,readonly

entry

export Reset\_Handler

Reset\_Handler

**;8-bit (0x00 t0 0x63)**

;mov r0,#0x63

;mov r5,#10

;udiv r4,r0,r5

;mul r1,r4,r5

;sub r2,r0,r1 ;remainder (r2=r0-(r4\*r5)

;add r7,r2,r4,lsl#4

**;8-bit (0x64 t0 0xff)**

mov r0,#0x65 ; hexadecimal number

mov r1,#100

udiv r2,r0,r1 ;r2=0x00000001

mov r3,r2,lsl#8 ;r3=0x00000100

mul r4,r2,r1 ;r4 = Qoutient\*Divider

sub r5,r0,r4 ;remainder = dividend -(Qoutient\*Divider)(r2=r0-(r4\*r5)

mov r6,#10

udiv r7,r5,r6

mul r8,r7,r6

sub r9,r5,r8

add r10,r9,r7,lsl#4

add r11,r3,r10 ; r11 = 101 equivalent decimal value of 0x65

stop b stop

end

**;ARM ALP to convert packed bcd to unpacked bcd**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area mycode,code,readonly

entry

export Reset\_Handler

Reset\_Handler

ldr r0,num ; packed bcd number

and r0,r0,#0xf0

mov r1,r0,lsr#4

ldr r0,num

and r2,r0,#0x0f

stop b stop

num dcd 0x00000072

end

**;ARM ALP to convert un-packed bcd to packed bcd**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area mycode,code,readonly

entry

export Reset\_Handler

Reset\_Handler

**; 0702 = 0x00000072**

ldrb r0,nums ; r0 = 0x07

mov r1,r0,lsl#4 ; r1 = 0x00000070

ldrb r2,nums+1 ; r2 = 0x02

orr r3,r1,r2 ; r3 = 0x00000072

stop b stop

nums dcb 0x07,0x02

end

**;ARM ALP to convert binary to ascii**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area mycode,code,readonly

entry

export Reset\_Handler

Reset\_Handler

**; Binary(0x01 to 0x09) = Ascii(0x30 to 0x39)**

**; Binary (0x0a to 0x0f)= Ascii(0x41 to 0x46)**

mov r0,#0x0c ; binary number

cmp r0,#0x0a

bne nxt

beq nxt1

nxt blt nxt2

nxt1 add r2,r0,#0x37 ; add 0x37 if morethan 9

b stop

nxt2 add r2,r0,#0x30 ;add 0x30 if lessthan a

stop b stop

end