**THUMB Instruction Related Programs**

**;Program to demonstrate thumb operation**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area mycode,code,readonly

entry

export Reset\_Handler

Reset\_Handler

start

mov r1,#4 ; Encoded into its equivalent 32-bit opcode (F04F0104)

add r1,#4 ; Encoded into its equivalent 32-bit opcode (F1010104)

add r1,#2 ; Encoded into its equivalent 32-bit opcode (F04F0102)

code16

start1

mov r2,#4 ;Encoded into its equivalent 16-bit opcode (2204)

add r2,#4 ;Encoded into its equivalent 16-bit opcode (3204)

add r2,#2 ;Encoded into its equivalent 16-bit opcode (3202)

;addeq r3,r3,#0 ; not thumb instruction

stop b stop

end

**;ARM ALP to demonstrate division operation using thumb instruction**

area reset,data,readonly

export \_\_Vectors

\_\_Vectors

dcd 0

dcd Reset\_Handler

area mycode,code,readonly

entry

export Reset\_Handler

Reset\_Handler

mov r8,#50

mov r9,#16

mov r11,#0

loop subs r8,r8,r9

addge r11,r11,#1

bge loop

add r10,r8,r9

code16

start1

mov r0,#50

mov r1,#16

mov r3,#0

loop1 sub r0,r1

blt loop2

add r3,#1

b loop1

loop2 add r0,r1

ldr r4,=data1

ldr r5,=data2

ldrh r6,[r4]

strh r6,[r5]

;addeq r3,r3,#0 ; not thumb instruction

stop b stop

data1 dcw 0x1234

area mydata,data,readwrite

data2 space 0

end