1. **Develop and simulate ARM ALP for Data Transfer, Arithmetic and Logical operations (Demonstrate with the help of a suitable program).**

# AREA calculator, CODE, READONLY ; name the code block

# ENTRY ; marker of first executable instruction

# START ; label

# MOV R0, #08 ; put the value 08 in register0

# MOV R1, #06 ; put the value 06 in register1

# ADD R3, R0, R1 ; R3 = R0 + R1

# SUB R4, R0, R1 ; R4 = R0 - R1

# MUL R5, R0, R1 ; R5=R0\*R1;

# S B S ; Branch to Label S

# END

# Logical Instructions: Logical instructions perform bitwise logical operations on the two source registers.

# AREA logical, CODE, READONLY ; name the code block

# ENTRY ; marker of first executable instruction

# START ; label

# MOV R0, #0x08 ; put the value 08 in register0

# MOV R1, #0x0A ; put the value 10 in register1

# AND R3, R0, R1 ;Logical Bitwise AND: R3 = R0 & R1

# ORR R4, R0, R1 ; Logical Bitwise OR: R3 = R0 | R1

# EOR R5, R0, R1 ; Logical Bitwise EXOR: R3 = R0 ^ R1

# BIC R6, R0,R1 ; Logical Bit Clear ( AND ~): R3 = R0 & ~ R1

# S B S ; Branch to Label S

# END

**3. Develop an ALP to multiply two 16-bit binary numbers.**

AREA MULTIPLY, CODE, READONLY

ENTRY ; Mark first instruction to execute

START

MOV R1, #6400 ; STORE FIRST NUMBER IN R0

MOV R2, #3200 ; STORE SECOND NUMBER IN R1

MUL r3, r1, r2 ; MULTIPLICATION

back B back

END ; Mark end of file

; **Program to multiply two 16 bit numbers defined in memory and display the result in register**

AREA MULTIPLY, CODE, READONLY ; NAME THE CODE BLOCK

ENTRY

START

LDR R0, =VALUE1

LDRH R1,[R0]

LDR R0,=VALUE2

LDRH R2,[R0]

MUL R3,R2,R1

BACK B BACK

VALUE1 DCW &BBBB ; OR 0XBBBB

VALUE2 DCW &CCCC ; OR 0XCCCC

END