**ALU OPERATIONS**

; ALU Operations Program

ORG 0H ; Origin of the program

; Initialize registers

MOV R0, #0AH ; Initialize R0 to 10H

MOV R1, #05H ; Initialize R1 to 05H

MOV R5, #05H ; Initialize R5 to 05H (multiplier)

MOV R7, #0AH ; Initialize R7 to 0AH (dividend)

; Addition operation

ADD A, R0 ; Add R0 to accumulator

ADD A, R1 ; Add R1 to accumulator

MOV R2, A ; Store result in R2

; Subtraction operation

MOV A, R0 ; Load R0 into accumulator

SUBB A, R1 ; Subtract R1 from accumulator

MOV R3, A ; Store result in R3

; Multiplication operation (using repeated addition)

MOV R4, #00H ; Initialize R4 to 00H (result register)

LOOP1:

ADD A, R0 ; Add R0 to accumulator

MOV R4, A ; Store result in R4

DJNZ R5, LOOP1 ; Decrement multiplier, jump to LOOP1 if not zero

; Division operation (using repeated subtraction)

MOV R6, #00H ; Initialize R6 to 00H (quotient register)

LOOP2:

CLR C ; Clear carry flag

SUBB A, R1 ; Subtract R1 from accumulator

JNC CONT ; Jump to CONT if no carry

INC R6 ; Increment quotient

CONT:

MOV R7, A ; Store result in R7

JZ END\_LOOP2 ; Jump to END\_LOOP2 if zero

AJMP LOOP2 ; Jump to LOOP2

END\_LOOP2:

MOV A, R6 ; Load quotient into accumulator

END ; End of the program

***Output***

