

8051 Assembly Language Programs - All Experiments

Exp 1: Arithmetic Operations (Add, Sub, Mul, Div)

ORG 0000H

; Addition

MOV A, #15H

MOV B, #10H

ADD A, B

MOV 40H, A

; Subtraction

MOV A, #15H

MOV B, #10H

SUBB A, B

MOV 41H, A

; Multiplication

MOV A, #05H

MOV B, #07H

MUL AB

MOV 42H, A

MOV 43H, B

; Division

MOV A, #24H

MOV B, #05H

DIV AB

MOV 44H, A

MOV 45H, B

END

Exp 2: Logical Operations (AND, OR, XOR, NOT)

```
ORG 0000H  
MOV A, #35H  
MOV R0, #25H  
ANL A, R0  
MOV R3, A  
MOV 40H, R3
```

```
MOV A, #35H  
MOV R0, #25H  
ORL A, R0  
MOV R3, A  
MOV 41H, R3
```

```
MOV A, #35H  
MOV R0, #25H  
XRL A, R0  
MOV R3, A  
MOV 42H, R3
```

```
MOV A, #35H  
CPL A  
MOV 43H, A  
END
```

Exp 3: Move Block of Data

```
ORG 00H
MOV R0, #20H
MOV R1, #40H
MOV R2, #0AH
UP: MOV A, @R0
    MOV @R1, A
    INC R0
    INC R1
    DJNZ R2, UP
EXIT: SJMP EXIT
END
```

Exp 4: Exchange Data Blocks

```
ORG 0000H
MOV R0, #20H
MOV R1, #40H
MOV R2, #05H
EXCHANGE:
MOV A, @R0
MOV B, @R1
MOV @R0, B
MOV @R1, A
INC R0
INC R1
DJNZ R2, EXCHANGE
EXIT: SJMP EXIT
END
```

Exp 5: Add Bytes Using Indirect Addressing

```
ORG 0000H
MOV R0, #34H
MOV A, @R0
MOV R1, #35H
ADD A, @R1
MOV R5, A
MOV R6, #00H
JNC NEXT
INC R6
NEXT: SJMP NEXT
END
```

Exp 6: Subtract Bytes

```
ORG 0000H
MOV R0, #34H
MOV A, @R0
MOV R1, #35H
CLR C
SUBB A, @R1
MOV R5, A
MOV R6, #00H
JNC NEXT
INC R6
NEXT: SJMP NEXT
END
```

Exp 7: Multiply 8-bit Numbers

```
ORG 0000H
MOV R0, #30H
MOV A, @R0
INC R0
MOV B, @R0
MUL AB
INC R0
MOV @R0, A
INC R0
MOV @R0, B
END
```

Exp 8: Divide 8-bit Numbers

```
ORG 0000H  
MOV A, 30H  
MOV B, 31H  
DIV AB  
MOV 32H, A  
MOV 33H, B  
END
```


Exp 9: Separate Positive and Negative Numbers

```
ORG 0000H
MOV R0, #30H
MOV R1, #50H
MOV R2, #05
BACK1: MOV A, @R0
JB ACC.7, SKIP1
MOV @R1, A
INC R1
SKIP1: INC R0
DJNZ R2, BACK1
MOV R0, #30H
MOV R1, #60H
MOV R2, #05
BACK2: MOV A, @R0
JNB ACC.7, SKIP2
MOV @R1, A
INC R1
SKIP2: INC R0
DJNZ R2, BACK2
HERE1: SJMP HERE1
END
```

Exp 10: Separate Even and Odd Numbers

```
ORG 0000H
MOV R0, #30H
MOV R1, #50H
MOV R2, #05H
BACK3: MOV A, @R0
JB ACC.0, SKIP3
MOV @R1, A
INC R1
SKIP3: INC R0
DJNZ R2, BACK3
MOV R0, #30H
MOV R1, #60H
MOV R2, #05H
BACK4: MOV A, @R0
JNB ACC.0, SKIP4
MOV @R1, A
INC R1
SKIP4: INC R0
DJNZ R2, BACK4
HERE2: SJMP HERE2
END
```