

# 8051 LAB MANUAL

## 1. Addition and Subtraction of 8-bit numbers

ORG 00h MOV A,31h ; move the content in location 31H to A register MOV B,41h ; move the content in location 41H to A register ADD A,B ; Adds the content of A and B, and stores in A register JNC LOOP1 ; if no carry jump to loop1 INC R2 ; increment R2 content by 1 LOOP1: MOV 51H,A ; move the content of A(sum) register to location 51H MOV 50H, R2 ; move the content of R2(carry) register to location 50H END	ORG 00h MOV A,31h MOV B,41h subb A,B JNC LOOP1 INC R2 LOOP1: MOV 51H,A MOV 50H, R2 END
--	--

## 2. Addition and Subtraction of 16-bit numbers

ORG 00h MOV R0,#00H MOV R1,#00H CLR C MOV A,32H MOV B,42H ADD A,B JNC LOOP1 INC R0 LOOP1: MOV 52H,A MOV 21H,R0 MOV A,31H MOV B,41H ADDC A,B JNC LOOP2 INC R1 LOOP2:MOV 51H,A MOV 50H,R1 END	ORG 00h MOV R0,#00H MOV R1,#00H CLR C MOV A,32H MOV B,42H SUBB A,B JNC LOOP1 INC R0 LOOP1: MOV 52H,A MOV 21H,R0 MOV A,31H MOV B,41H SUBB A,B JNC LOOP2 INC R1 LOOP2:MOV 51H,A MOV 50H,R1 END
---	--

## 3. Addition of N-8bit Numbers

ORG 00h MOV R1,#03H ; count MOV R0,#30H ; Series CLR C MOV A,@R0 INC R0 BACK:MOV B,@R0 ADD A,B JNC LOOP1 INC R2 LOOP1: INC R0 DJNZ R1,BACK MOV 40H,A MOV 41H, R2 END
--

#### 4. Generation of Fibonacci series up to limit

```
ORG 00h
MOV R1,#0AH ;limit
MOV R0,#40H ; RESULT
MOV A,#00H
MOV @R0,A
INC R0
DEC R1
MOV B,#01H
MOV @R0, B
LOOP: ADD A,B
DA A
INC R0
MOV @R0,A
XCH A,B
DJNZ R1,LOOP
END
```

#### 5. Finding Minimum and Maximum in a given series

<pre>ORG 00H MOV R0,#07h ;series limit MOV R1,#30H ;series data MOV A,@R1 MOV R2,A BACK:INC R1 MOV B,@R1 MOV A,R2 SUBB A,B JC NEXT MOV R2,B NEXT: DJNZ R0,BACK MOV 40H,R2 ; Result END</pre>	<pre>ORG 00H MOV R0,#07h ;series limit MOV R1,#30H ;series data MOV A,@R1 MOV R2,A BACK:INC R1 MOV B,@R1 MOV A,R2 SUBB A,B JNC NEXT MOV R2,B NEXT: DJNZ R0,BACK MOV 40H,R2 ; Result END</pre>
--	---

6. Arrange the given series in increasing or decreasing order

<pre>ORG 00H MOV R4,#09H AGAIN:MOV R3,#09H     MOV R0,#30H     CLR C BACK: MOV A,@R0     MOV R1,A         INC R0         MOV A,@R0         SUBB A,R1     JNC SKIP         MOV A,@R0         DEC R0     MOV @R0,A         MOV A,R1         INC R0         MOV @R0,A SKIP: DJNZ R3,BACK     DJNZ R4,AGAIN     END</pre>	<pre>ORG 00H MOV R4,#09H AGAIN:MOV R3,#09H     MOV R0,#30H     CLR C BACK: MOV A,@R0     MOV R1,A         INC R0         MOV A,@R0         SUBB A,R1     JC SKIP         MOV A,@R0         DEC R0     MOV @R0,A         MOV A,R1         INC R0         MOV @R0,A SKIP: DJNZ R3,BACK     DJNZ R4,AGAIN     END</pre>
---	--

7. Data block interchange

<pre>ORG 00H MOV R2,#05H MOV R0,#30H MOV R1,#40H BACK:MOV A, @R0 MOV B,@R1 MOV @R1,A MOV @R0,B INC R0 INC R1 DJNZ R2,BACK END</pre>
---

8. Toggle P1.0 every 10ms using timer mode1

```
ORG 00H
LOOP: CLR P1.0
CALL DELAY
SETB P1.0
CALL DELAY
SJMP LOOP

DELAY : MOV TMOD,#10H
MOV TH1,#0DCH
MOV TL1,#00H
SETB TR1
BACK:
JNB TF1,BACK
CLR TR1
CLR TF1
RET
END
```

Steps to calculate delay

Crystal frequency = **11.0592 MHz**

$$11.0592/12 = 921.6 \text{ KHz}$$

$$\text{Machine cycle} = 1/921.6 \text{ KHz} = 1.085\mu\text{s} [\mu\text{s}=\text{microsecond}]$$

Divide the time delay with timer clock period.

$$\text{NNNN} = \text{time delay}/1.085\mu\text{s}$$

Subtract the resultant value from 65536.

$$\text{MMMM} = 65536 - \text{NNNN}$$

Convert the difference value to the hexa decimal form.

$$\text{MMMMd} = \text{XXYYh}$$

Load this value to the timer register.

$$\text{TH} = \text{XXh}$$

$$\text{TL} = \text{YYh}$$

9. Finding how many times data occur in a given series

```
ORG 00H
MOV R2,#09H
MOV R0,#30H
MOV R1,#40H
BACK:CLR C
MOV A,@R1
MOV B,@R0
CJNE A,B,FORWARD
INC R3
FORWARD:INC R0
DJNZ R2,BACK
END
```

#### 10. Finding ones and zeros in 8 and 16-Bit numbers

```
ORG 00H
MOV R0,#30H ;DATA
MOV R1,#40H ;RESULT ADDRESS
SJMP FIRST
REPEAT:INC R0
CLR 00H
FIRST:MOV R4,#08H ;COUNT
MOV A,@R0
BACK:RLC A
JNC ZERO
INC R2
SJMP FORW
ZERO:INC R3
FORW:DJNZ R4,BACK
JB 00H,REPEAT
MOV A, R2
MOV @R1,A
INC R1
MOV A,R3
MOV @R1,A
END
```