**Program 1 :- ADD TWO 16 BITS NUMBERS AND STORE THE RESULT ON LOCATION 11H & 10H**

org 0000h

mov a,#33h

add a,#46h

mov 11h,a

mov a,#42h

addc a,#32h

mov 10h,a

end

**Program 2 :- Find Factorial of Number 5**

org 0000h

mov a,#01h

mov r0,#05h

x:mov b,r0

mul ab

djnz r0,x

end

**Program 3 :- To find number of one(1) in byte**

org 0000h

mov a,#76h

mov r0,#08h

mov r1,#00h

b1:rrc a

jnc a1

inc r1

a1:djnz r0,b1

end

**Program 4: TO ADD FIRST 10 NUMBERS AND STORE THE RESULT ON LOCATION 10H.**

org 0000h

mov r1,$10

a1:add a,r1

djnz r1,a1

end

**Program 5: TO MOVE BLOCK OF 10 DATA FROM LOCATION 30H TO 40H.**

org 0000h

mov r2,$10

mov r0,#30h

mov r1,#40h

a1:mov a,@r0

mov @r1,a

inc r0

inc r1

djnz r2,a1

end

**Program 6 blinking led:**

org 0000h

mov p1,#00h

setb p1.5

setb p1.7

clr p1.4

clr p1.6

end

**Program 7 To scroll led**

org 0000h

mov p1, #00h //output port

again: mov p1,#10000000b

acall delay

mov p1,#01000000b

acall delay

mov p1,#00100000b

acall delay

mov p1,#00010000b

acall delay

ajmp again

delay: mov R0, #0ffh

a1: mov r1, #0ffh

b1: djnz r1, b1

c1: djnz R0, c1

djnz R2, a1

ret

end

**Program 8 To generate 1ms delay using timer 0**

org 0000h

again:mov p1,#00h

acall delay

mov p1,#0ffh

acall delay

jmp again

delay: mov tmod,#01h

mov tl0,18h

mov th0,0fch

setb tr0

a1:jnb tf0,a1

clr tr0

clr tr0

ret

end

**Program 9 :- To generate 10ms delay using timer 0**

org 0000h

again:mov p1,#00h

acall delay

mov p1,#0ffh

acall delay

jmp again

delay: mov tmod,#10h

mov tl0,0f0h

mov th0,0d8h

setb tr0

a1:jnb tf0,a1

clr tr0

clr tr0

ret

end

**Program 10 :- To display 0 to 9 on 7 segment**

org 0000h

mov p1,#11h

mov p1,#0f9h

acall delay

mov p1,#0a4h

acall delay

mov p1,#0b0h

acall delay

mov p1,#099h

acall delay

mov p1,#092h

delay:mov r1,#0ffh

a2:djnz r1,a2

ret

end

**Program 11 :- Display 5 to 0 on seven segment**

org 0000h

mov p1,#092h

acall delay

mov p1,#099h

acall delay

mov p1,#0b0h

acall delay

mov p1,#0a4h

acall delay

mov p1,#0f9h

acall delay

mov p1,#0c0h

acall delay

delay:mov r1,#0ffh

a2:djnz r1,a2

ret

end

**Program 12 :- DISPLAY RK UNIVERSITY ON LCD**

ORG 0000H

MOV A,#38H

ACALL CMD

MOV A,#0EH

ACALL CMD

MOV A,#01H

ACALL CMD

MOV A,#06H

ACALL CMD

MOV A,#80H

ACALL CMD

MOV A,#'R'

ACALL DATAa

MOV A,#'K'

ACALL DATAa

CMD:MOV P1,A

CLR P3.0

setb p3.2

CLR P3.2

ACALL DELAY

RET

DATAa:MOV P1,A

SETB P3.0

SETB P3.2

CLR P3.2

ACALL DELAY

RET

DELAY:MOV R0,#0FFH

A1:DJNZ R0,A1

RET

END

**Program 13 DISPLAY YOUR NAME IN 1ST LINE AND 5CED IN 2ND LINE**

org 0000h

mov a, #38h

acall cmd

mov a, #0eh

acall cmd

mov a, #01h

acall cmd

mov a, #06h

acall cmd

mov a, #80h

acall cmd

mov a, #'N'

acall data

mov a, #'E'

acall data

mov a, #'T'

acall data

mov a, #'R'

acall data

mov a, #'Y'

acall data

mov a,#0C0H

acall cmd

mov a, #'C'

acall data

mov a, #'O'

acall data

mov a, #'M'

acall data

mov a, #'P'

acall data

mov a, #'U'

acall data

mov a, #'T'

acall data

mov a, #'E'

acall data

mov a, #'R'

acall data

mov a, #' '

acall data

mov a, #'E'

acall data

mov a, #'N'

acall data

mov a, #'G'

acall data

mov a, #'.'

acall data

mov a, #'.'

acall data

cmd: mov p1, a

clr p3.0

setb p3.2

clr p3.2

acall delay

ret

data: mov p1,a

setb p3.0

setb p3.2

clr p3.2

acall delay

ret

delay: mov r0, #255

here: djnz r0, here

ret

end

**Program 14 To interface Dc motor with 8051 and rotate it in clockwise direction.**

org 0000h

setb p3.1

clr p3.0

end

**Program 15 :- To interface Dc motor with port 1 of 8051 and rotate it in anti clockwise direction**

org 0000h

setb p1.1

clr p1.0

end

**Program 16 To interface Switch using external intrrupt 0.**

org 0000h

ajmp start

org 0020h

start: mov IE,#81h

here: ajmp here

org 0003h

setb p1.1

clr p1.0

reti

end

**Program 17 :- To interface Switch using external interrupt 1.**

org 0000h

ajmp start

org 0020h

start: mov IE,#84h

here: ajmp here

org 0013h

mov p1,#00h

reti

end