Q.P Code :- 25077

Q.1 (e)

i) MOV A, @R0 :- Move indirect RAM to accumulator.

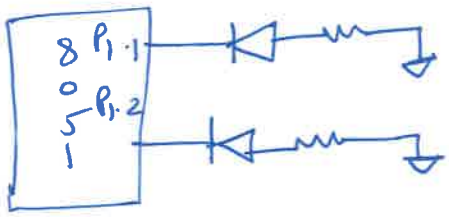
ii) SETB bit :- Set direct bit

iii) JNC rel :- Jump if Carry not set

iv) MOVC A @A+DPTR :- Move code byte relative to DPTR to accumulator

v) CPL C :- Complement Carry.

Q.1 (c)



Program

```
again:- SETB  P1.1
        CALL  DELAY
        CLR   P1.1
        SETB  P1.2
        CALL  DELAY
        CLR   P1.2
        SJMP  again
```

1ms DELAY :-

```
        MOV  R6,#250D
        MOV  R7,#250D
Label1: DJNZ R6, Label1
Label2: DJNZ R7, Label2
        RET.
```

Note:- (any other way/method of generating delay could be acceptable).

Q.1 (d) 8051 program to find factorial of no. 05H.

```
        ORG 000H
        MOV R0,#5
        MOV A,R0
        ACALL FACT
FACT:   DEC R0
        CJNE R0,#01,rel;  value of R0 is compared with 1.
        SJMP stop
rel :   MOV B,R0
        MUL AB
        ACALL FACT
stop :  END.
```

Q.3 (a) Assuming XTAL = 11.0592 MHz, 50Hz square wave

→ $T = \frac{1}{50Hz} = 20ms$ period of square wave.

1/2 of it for high & 1/2 for low = 10 ms.

$\frac{10ms}{1.085\mu s} = 9216$ & $65536 - 9216 = 56320(d)$.

$= DC00H$. ∴ TL = 00H, TH = DCH.

Using Timer 1, mode 1.

```
        MOV TMOD,#10H ; Timer 1, mode 1.
again:  MOV TL1,#00.    — lower byte
        MOV TH1,#0DCH   — Higher byte count
        SETB TR1        start timer 1
back :  JNB TF1, Back   stay until timer rolls.
        CLR TR1         stop timer 1
        CPL P2.3        complement P2.3 to get hi, lo
        CLR TF1.        clear timer flag
        SJMP again      reload count
```

Q46. Program for serial communication.

```
        MOV TMOD,#20H  ; timer 1, mode 2
        MOV TH1,#-3    ; 9600 baud
        MOV SCON,#50H
        SETB TR1
Again : MOV A,#"B"     ; Tranfer "B"
        ACALL Trans
        MOV A,#"E"
        ACALL Trans
        MOV A,#"S"
        ACALL Trans
        MOV A,#"T"
        ACAL Trans
        SJMP again
Trans : MOV SBUF,A     ; load SBUF
here  : JNB TI,here    ; wait for last bit
        CLR TI         ; to tranfer
        RET            ; send next.
```

Q5 a. i) ADD R3, R2, R1 :- add contents of R1 & R2 add store in R3

ii) CMP R8, R2 ;- Compare contents of R2 & R8, check flags (Test)

iii) TST R2, R5 :- set cc on R2 & R5, bit test.

iv) STMFD R13! {R0-R2, R14} - use stack to save/restore the return address & registers.

v) ~~STR~~ CMN R1, R2 ; compare negated.

Q.5 (b) Interfacing.

i) 16 KB RAM using 8 kb devices

ii) 16 kB ROM using 8 kb devices.

Addresses for RAM & ROM.

for 8 KB — no. of address lines = 13
i.e A_0 to A_{12}

RAM S.A device 1 = 0000H
& ROM E.A —11— = 1FFFH

S.A device 2 = 2000H.
Ending address 2 = 3FFFH.

remaing 3 lines for chip select i.e A_{13} to A_{15}