Benha University **Department:Electrical engineering**

Benha Faculty of Engineering **Time:**2Hours.

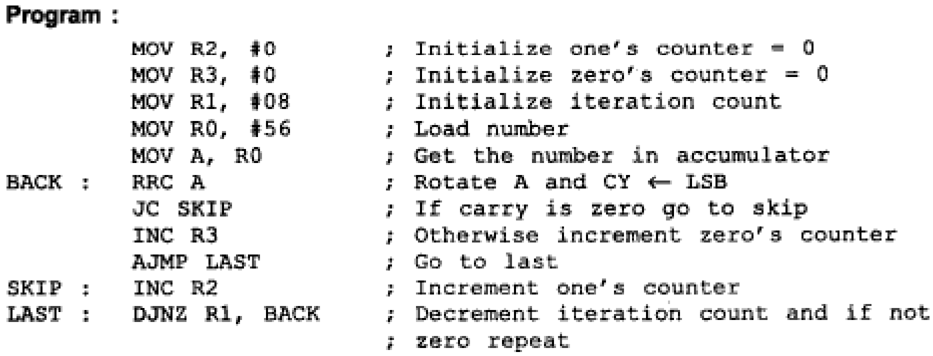
April, 2017 3^rd^**Year Course Exam** 3^rd^

Exam (Midterm**)** **Subject**:Microprocessor Based Systems (b)

Examinar:**Dr. Tamer Omar**

Question 1(8marks)

Write assembly language program for 8051 to count number of one's and zeros's in number N stored in memory location 30H

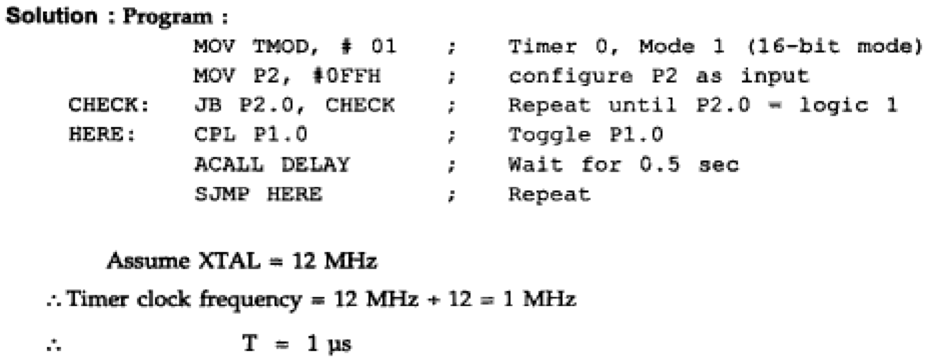


Program :

```
        MOV  R2,  #0        ; Initialize one's counter = 0
        MOV  R3,  #0        ; Initialize zero's counter = 0
        MOV  R1,  #08       ; Initialize iteration count
        MOV  R0,  #56       ; Load number
        MOV  A,  R0         ; Get the number in accumulator
BACK :  RRC  A              ; Rotate A and CY ← LSB
        JC  SKIP            ; If carry is zero go to skip
        INC  R3             ; Otherwise increment zero's counter
        AJMP  LAST          ; Go to last
SKIP :  INC  R2             ; Increment one's counter
LAST :  DJNZ  R1,  BACK     ; Decrement iteration count and if not
                            ; zero repeat
```

***Questin**2(**8marks**)*

Assume XTAL=12 MHz, write assembly language program for 8051 such that LED connected to port P1.0 flash at 0.5 sec rate when line P2.0 goes high. Use timer 0 for generating delay.



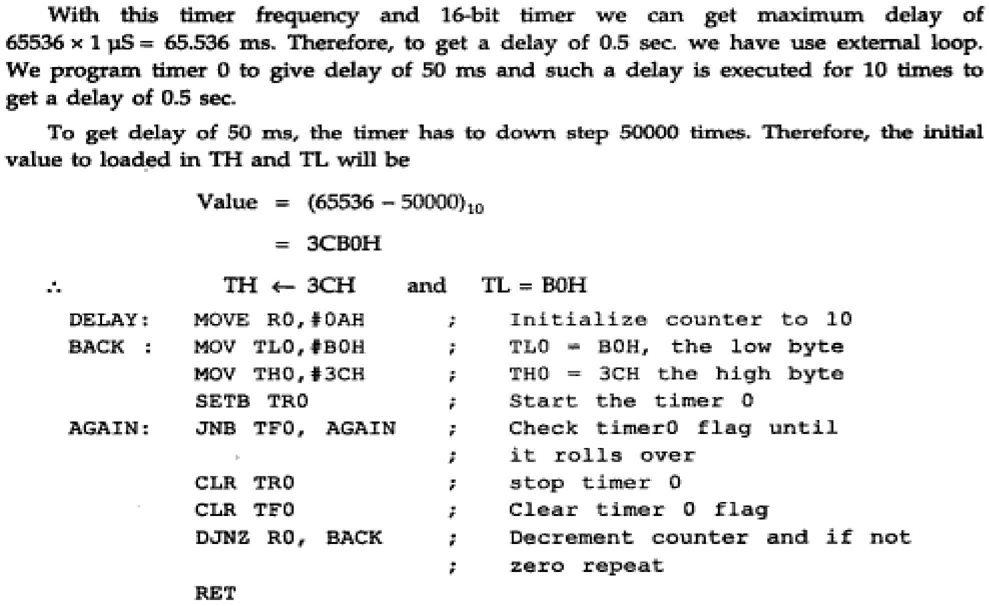
Solution : Program :

```
            MOV  TMOD,  # 01    ;    Timer 0, Mode 1 (16-bit mode)
            MOV  P2,  #0FFH     ;    configure P2 as input
    CHECK:  JB  P2.0,  CHECK    ;    Repeat until P2.0 = logic 1
    HERE:   CPL  P1.0           ;    Toggle P1.0
            ACALL  DELAY        ;    Wait for 0.5 sec
            SJMP  HERE          ;    Repeat
```

Assume XTAL = 12 MHz

∴ Timer clock frequency = 12 MHz + 12 = 1 MHz

∴ T = 1 μs



With this timer frequency and 16-bit timer we can get maximum delay of 65536 × 1 μS = 65.536 ms. Therefore, to get a delay of 0.5 sec. we have use external loop. We program timer 0 to give delay of 50 ms and such a delay is executed for 10 times to get a delay of 0.5 sec.

To get delay of 50 ms, the timer has to down step 50000 times. Therefore, the initial value to loaded in TH and TL will be

$$\text{Value} = (65536 - 50000)_{10}$$

$$= 3CB0H$$

$$\therefore \quad TH \leftarrow 3CH \quad \text{and} \quad TL = B0H$$

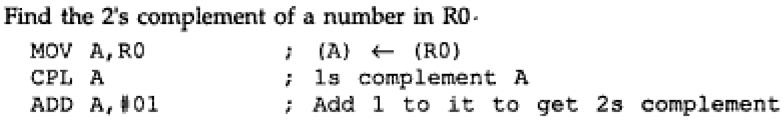
```
DELAY:    MOVE R0,#0AH      ;   Initialize counter to 10
BACK :    MOV TL0,#B0H      ;   TL0 = B0H, the low byte
          MOV TH0,#3CH      ;   TH0 = 3CH the high byte
          SETB TR0          ;   Start the timer 0
AGAIN:    JNB TF0, AGAIN    ;   Check timer0 flag until
                            ;   it rolls over
          CLR TR0           ;   stop timer 0
          CLR TF0           ;   Clear timer 0 flag
          DJNZ R0, BACK     ;   Decrement counter and if not
                            ;   zero repeat
          RET
```

Question*3**(6 marks***)

Write programs that will accomplish the desired tasks listed below, using a few lines of code.

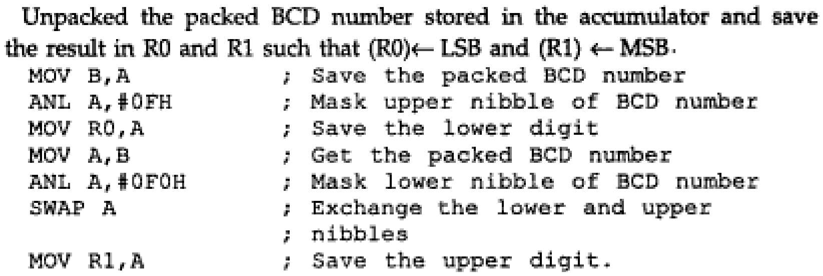
1. Find the 2's complement of a number in R0
2. Unpacked the packed BCD stored in the accumulator and save the result in R0 and R1.
3. Subtract the contents of Bank0 from the contents of R0 of Bank2
4. Add the following data and store the result in RAM location 30H

```
            ORG 200H
MYDATA:     DB 06, 09, 02, 05, 07
```



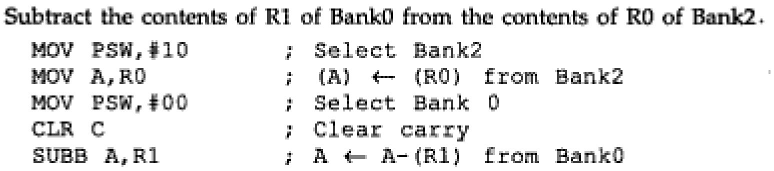
Find the 2's complement of a number in R0.

```
MOV  A,R0          ;  (A)  ←  (R0)
CPL  A             ;  1s  complement  A
ADD  A,#01         ;  Add  1  to  it  to  get  2s  complement
```



Unpacked the packed BCD number stored in the accumulator and save the result in R0 and R1 such that (R0)← LSB and (R1) ← MSB.

```
MOV  B,A           ;  Save  the  packed  BCD  number
ANL  A,#0FH        ;  Mask  upper  nibble  of  BCD  number
MOV  R0,A          ;  Save  the  lower  digit
MOV  A,B           ;  Get  the  packed  BCD  number
ANL  A,#0F0H       ;  Mask  lower  nibble  of  BCD  number
SWAP  A            ;  Exchange  the  lower  and  upper
                   ;  nibbles
MOV  R1,A          ;  Save  the  upper  digit.
```



Subtract the contents of R1 of Bank0 from the contents of R0 of Bank2.

```
MOV  PSW,#10       ;  Select  Bank2
MOV  A,R0          ;  (A)  ←  (R0)  from  Bank2
MOV  PSW,#00       ;  Select  Bank  0
CLR  C             ;  Clear  carry
SUBB  A,R1         ;  A  ←  A-(R1)  from  Bank0
```

Question4(***8 marks***)

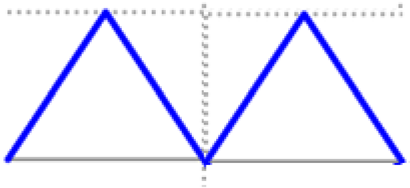
Assume XTAL=11.0592 MHz Write software delay subroutine to generate

a. 2KHz square wave on P1.0.

```
ORG 000H

MOV P1,#00000000B

MOV A,#00000000B

MAIN: MOV R6,#220D

      MOV R7,#183D

LOOP1:DJNZ R6,LOOP1

LOOP2:DJNZ R7,LOOP2

      CPL A

      MOV P1,A

      SJMP MAIN
```

```
        END
```

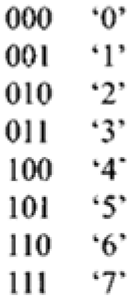
b. Triangular wave of period 1 ms



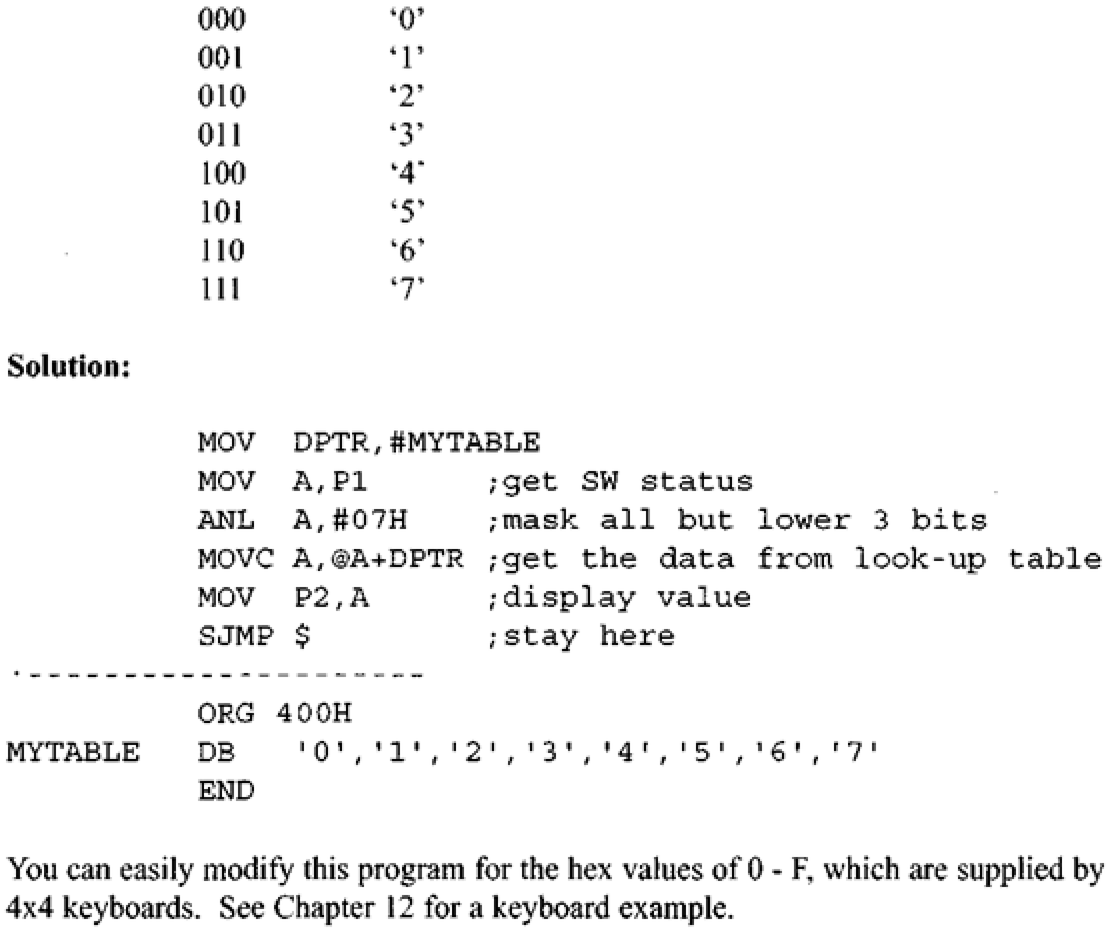
```
ORG 0

CLR A

UP:INC A

MOV P2,A

CJNE A,#0FFH,UP

LCALL DELAY

DOWN: DEC A

MOV P2,A

CJNE A,#00,DOWN

LCALL DELAY

SJMP UP

END
```

Question5(***8 marks***)

Assume that the lower three bits of PI are connected to three switches. Write a program to send the ASCII characters to P2 based on the status of the switches.



000	'0'
001	'1'
010	'2'
011	'3'
100	'4'
101	'5'
110	'6'
111	'7'



000	'0'
001	'1'
010	'2'
011	'3'
100	'4'
101	'5'
110	'6'
111	'7'

Solution:

```
            MOV   DPTR,#MYTABLE
            MOV   A,P1        ;get SW status
            ANL   A,#07H      ;mask all but lower 3 bits
            MOVC  A,@A+DPTR   ;get the data from look-up table
            MOV   P2,A        ;display value
            SJMP  $           ;stay here
;---------------------
            ORG  400H
MYTABLE     DB    '0','1','2','3','4','5','6','7'
            END
```

You can easily modify this program for the hex values of 0 - F, which are supplied by 4x4 keyboards. See Chapter 12 for a keyboard example.