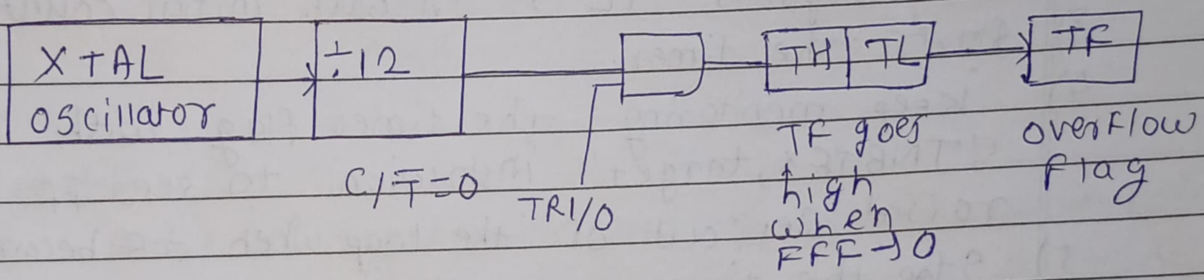
Experiment NO:- 2.

Aim:- Assignment exploiting times & its applications PWM generation in 8051

part A::
Write a program to generate square wave with 1KHZ frequency on p2.3, use Timer0 mode 1

Theory:
Mode 1:



1) It is a 16 bit timer therefore it allows values of 0000 to FFFF H to be loaded into the timers registers TL & TH.
2) After TH & TL are loaded with a 16 bit initial value the timer must be started this is done by "SETB TRO" for timer 0 & "SETB TR1" for Timer 1.
3) After the timer is started it starts to count up It counts up until it reaches its limit of FFFFH when it rolls over from FFFF H to 0000, it sets high a flag monitored when this timer flag is raised one option would be to stop the timer with the instructions "CLR TRO" or "CLR TR1" for timer 0 & Timer 1 respectively. Again it must be noted that each timer has its own timer flag TF0 for Timer0 & TF1 for Timer 1

4) After the timer reaches its limit and rolls in order to repeat the process the registers - TL must be reloaded with the original value. TF must be reset to 0.

Steps to program in Mode 1:

1) load the TMOD value Register ~~intial~~ indicating which Timer is to be used & which timer mode is selected
2) load registers TL & TH with initial count values
3) Starts the timer.
4) Keep monitoring the timer flag with the ~~"JNB~~ "JNB TFx, target" instruction to see if it is raised. Get out of the loop when TF becomes high.
5) Stop the timer
6) Clear the TF flag for the next round.
7) go back to step 2 to load TH & TL again.

Calculation:

For 1 KHz Square wave:

$$T = 1/F = \frac{1}{1\times10^3} = 0.001\ ms = 1000\ \mu s.$$

$$\therefore \frac{1000\ \mu s}{2} = 500\ \mu s.$$

$$\therefore \frac{500\ \mu s}{1.085} = 460$$

65536 − 460 = 65076

$(85076)_{10} \rightarrow (34FE)_{16}$

TLO = OFEH , THO = 34H.

- program

```
          ORG   0000H
Next: Mov  TMOD , # 01H
~~Rax~~     Mov  TH0 , # 0FEH
          Mov  TL0, # 34H
          SETB  TR0.
Back : JNB  TF0, Back
          CLR  TR0
          CPL   P2.3
          CLR   TF0
          SJMP  Next
           END.
```

part B :-

Write a program to generate square wave form of 1KHZ of 70% duty cycle on port 0 using Timer 0 Mode1.

- Algorithm :

1) Initialize counter to 0000H.
2) TMOD is loaded
3) FFEC is loaded into TH0 & TL0 respectively
4) call Delay.
5) After Delay again load ED7B into TH0 & TL0
6) Then clear TR0 & TF0
7) RET.

- Calculation:-

1KHZ 70% duty cycle timer 0 mode 1

70% duty cycle = 70% ON period 50% OFF period

for ON period = $\frac{70}{100} \times 1000\ \mu s = 700\ \mu s$.

$T = \frac{1}{F} = \frac{1}{1KHZ} = 1ms = 1000\ \mu s$.

for off period = $\frac{30}{100} \times 1000\ \mu s = 300\ \mu s$.

Timer reg. value calculation

for ON Time = $\frac{700}{1.085} = 645$.

$65536 - 645 = 64891$

$(64891)_{10} = (FD7BH)_{16}$

For off time = $\frac{300}{1.085} = 276$

$65536 - 276 = 65260$

$(62260)_{10} = (FEECH)_{16}$

- program:

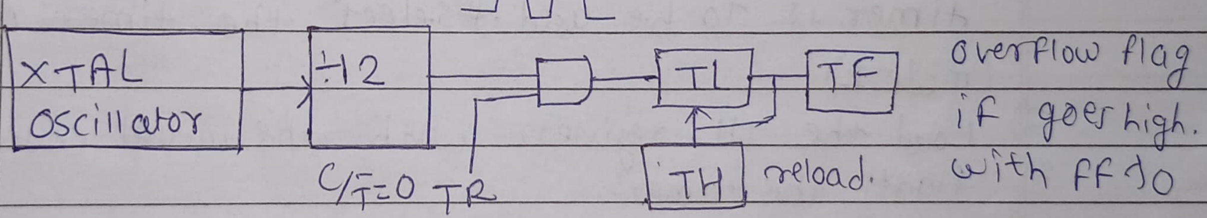
```
        ORG   0000H
        MOV   TMOD, #01H
Back !  MOV   TH0, #0FEH
        MOV   TL0, #0ECH
        MOV   P0, #00H
        ~~MOV~~ ACall Delay
        MOV   TH0, #0FDH
        MOV   TL0, #07BH
        MOV   P0, #0FEH.
```

```
          ACALL Delay.
          SJMP  BACK.
Delay :   SETB  TRO
AGAIN :   JNB   TFO, AGAIN
          CLR   TRO
          CLR   TFO
          RET
          END.
```

- PORT C

Write a program to generate square wave with 5kH frequency on p1.0 using timer1 mode 2.

- Theory :
- Mode 2 :



1) It is an 8 bit timer, therefore it allows only values of 00 to FFH to be loaded into the timers registers TH.
2) After TH is loaded with the 8 bit value the 8051 gives a copy of it to TL. Then the timer must be started This is done by the instructions "SETB TRO" for Timer 0 & "SETB TR1" for timer 1 This is just like mode 1.
3) After the timer is started it starts to count. up by incrementing the TL Register It counts up.

until by incrementing the TL register It ro
over from FFH to 00, it sets high the TF
(timer flag), If we are using timer 0, TF0
goes high if we are using Timer 1. TF1 is raised.

4) When the TL Register rolls from FFH to 0 & TF i
set to 1. TL is reloaded attomatically with the
original value kept by the TH register to repeat
the process we must simply clear TF & let
it go without any need by the programmer (
to reload the original value. This makes mode 2
an autto-reload in contrast with mode 1 in
which the programmer has to reload TH & TL.

- Steps to program in mode 2:

1) load the TMOD value register indicating which timer is to be used & select the timer mode (mode 2).

2) Load the TH registers with the initial count value.

3) Starts the timer.

4) keep monitoring the timer flag (TF) with the ("JNB" TFX, target" instruction to see ~~whatever~~ whether it is raised set out of the Loop when TF goes high.

5) Clear the TF flag.

6) Go back to step 4, since mode 2 is auto reload.

○ Calculation:

Fequency = 5KHz

$T = 1/F = 1/5KHz = 0.2ms = 200\ \mu s$.

For ON & off time.

$\frac{200}{2} = 100\ \mu s$.

$\therefore \frac{100\mu s}{1.085\mu s} = 92$

$256 - 92 = 164$

$(164)_{10} = (A4)_{16}$

○ Algorithm:

1) Initialize program counter to 0000H.
2) TMOD is loaded.
3) A4 is loaded into TH1
4) clear pin P1.0
5) Start timer.
6) After rolling out timer flag it is cleared.
7) Again repeat the process..

○ program:

```
    ORG 0000H
    MOV TMOD, #20H
    MOV TH1, #0A4H
    ~~Mov~~ CLR P1.0
    SETB TR1
BACK: JNB TF1, BACK.
    CPL P1.0
    CLR TF1
    SJMP BACK.
```

o conclusion:

Hence we can generate pulse waveform of duty cycle & 70% duty cycle using timer 0 model & also using timer 1 modes

C

23/4/22