

MAIN PROGRAM:-

```

14  #include <xc.h>
15  #include "guard.h"
16
17  #define _XTAL_FREQ 6000000 //intialize the clock speed
18
19
20
21  void main() //main function
22  {
23      timerinit(); //call timerinit
24
25      while(1)
26      {
27          PORTC=0xFF; //turn on portc
28          delay_ms(500); //delay
29          PORTC=0x00; //turn off portc
30          delay_ms(500); //delay
31      }
32
33  }

```

GUARD PROGRAM:-

```

8
9  // This is a guard condition so that contents of this file are not included
10 // more than once.
11 #ifndef XC_HEADER_TEMPLATE_H
12 #define XC_HEADER_TEMPLATE_H
13
14 #include <xc.h> // include processor files - each processor file is guarded.
15 void timerinit();
16 void __interrupt() isr();
17 #ifdef __cplusplus
18 extern "C" {
19 #endif /* __cplusplus */
20
21 // TODO If C++ is being used, regular C code needs function names to have C
22 // linkage so the functions can be used by the c code.
23
24 #ifdef __cplusplus
25 }
26 #endif /* __cplusplus */
27
28 #endif /* XC_HEADER_TEMPLATE_H */
29

```

TIMMER PRE LOADING

Task program

FUNCTION PROGRAM:

```
1  #include "guard.h"
2  unsigned char count = 0; //declare the global declaration
3  void timerinit() //timerinit function
4  {
5      TRISC=0x00; //set portc as output
6      PORTC=0x00; //clear the portc
7      TRISD=0x00; //set portd as output
8      PORTD=0x00; //clear the portd
9      INTCON = 0xC0; //enable the global interrupt ad pheripheral interrupt
10     PIR1 &= 0xFE; //clear the PIR1 register
11     PIE1=0x01; //turn on time 1 overflow interrupt
12     T1CON=0x01; //enable timer 1 and prescale 1:1
13     TMR1=5475; //set the calibrater value to tmr1
14 }
15
16 void __interrupt() isr() //isr function
17 {
18     if( PIR1 & 0x01) //check the flag
19     {
20         count++; //increment the count
21         if(count == 25) //check the count==25
22         {
23             PORTD=~PORTD; //till the portd
24             count=0; //reset the count
25         }
26         TMR1=5475; //set the timer pre loading value
27         PIR1 &= 0xFE; //clear the flag
28     }
29 }
30
31 }
```

OUTPUT:-

