

```
//Description:
//
//
//This example shows the delay function by using the library <delay.h>
//In PIC18F4450,one instruction will take 4 clock cycles.
//This means if one delay is executed, that will delay 4/48MHz = 1/12 us

/*Delay calculation examples.

    delay functions:  TCY: instruction cycle. 1 TCY = 4/(48M Hz) = 1/12 us.

    Delay1TCY(10);      delay for (1 instruction cycle)*10. Delay time = (1/12)*e-6 *10 = 0.83 us

    Delay10TCYx(100);  delay for (10 instruction cycles)*100.
                       // In this command, x is standing for multiply.Delay time = 10 *(1/12)*e-6 *
100 = 83 us

    Delay100TCYx(10);  delay for (100 instruction cycle)*10. Delay time = 100 *(1/12)*e-6 * 10 =
83us

    Delay1KTCYx(3);    delay for (1000 instruction cycle)*3. Delay time = 1000 *(1/12)*e-6 * 3 =
0.25 ms

    Delay10KTCYx(30);  delay for (10000 instruction cycle)*30. Delay time = 10000 *(1/12)*e-6 * 30
= 25 ms

    notice the range of variables in brackets is from 1 to 255. A value of 0 will make the variable
in to 256.
*/

//This example using the delay function to make the LED (A1) on for 2 seconds and off for 0.5
second.

#include <p18f4550.h>          //Standard library for 18F4550
#include <delays.h>           //Library with delay functions

//Bootloader code: never delete this part!
extern void _startup (void);
#pragma code _RESET_INTERRUPT_VECTOR = 0x000800
void _reset (void)
{
    _asm goto _startup _endasm
}
#pragma code

#pragma code _HIGH_INTERRUPT_VECTOR = 0x000808
void _high_ISR (void)
{
    ;
}
#pragma code

#pragma code _LOW_INTERRUPT_VECTOR = 0x000818
void _low_ISR (void)
{
    ;
}
#pragma code

void Delay1mS(int x)
{ //Pre: Delay library is included.
  //Post: Delay for x* 1 ms.
  int i;
  for (i=0; i<x; i++) Delay1KTCYx(12);
}

void Delay10mS(int x)
```

```
{ //Pre: Delay library is included.
  //Post: Delay for x* 10 ms.
  int i;
  for (i=0; i<x; i++) Delay10KTCYx(12);
}

void Delay100mS(int x)
{ //Pre: Delay library is included.
  //Post: Delay for x* 100 ms.
  int i;
  for (i=0; i<x; i++) Delay10KTCYx(120);
}

void Delay1S(int x)
{ //Pre: Delay library is included.
  //Post: Delay for x* 1 s.
  int i;
  for (i=0; i<10*x; i++) Delay10KTCYx(120);
}

void main(void)
{
  //Pre: The delay library is included, some extra functions of delay are defined.
  //Post: A0 will be on for 2 second and off for 0.5 second.

  TRISA=0b11111101;    //Channel A1 as output.
  LATA= 0 ;           // initialize PortA. All the outputs are set to 0.

  while(1)
  {
    LATAbits.LATA1 = 1;
    Delay1S(2);      // delay 2 seconds.
    LATAbits.LATA1 = 0;
    Delay100mS(5);  // delay 500ms.
  }
}
```