



PIC18F4550 Watchdog Timer

Introduction to Watchdog Timer

- Watchdog Timer (WDT) can be helpful to automatically reset the system whenever a timeout occurs.
- A system reset is required for preventing the failure of the system in a situation of a hardware fault or program error.
- There are countless applications where the system cannot afford to get stuck at a point (not even for a small duration of time). For example, in a radar system, if the system hangs for 5 minutes, it can result in serious repercussions (an enemy plane or missile may go undetected resulting in huge losses).
- The system should be robust enough to automatically detect the failures quickly and reset itself in order to recover from the failures and function normally without errors.
- One can manually reset the system to recover from errors. But it is not always feasible to manually reset the system, especially once it has been deployed.
- To overcome such problems, a watchdog timer is necessary to automatically reset the system without human intervention.

How does Watchdog Timer work?

- The watchdog timer is loaded with a timeout period which is dependent on the application.
- The watchdog timer starts its counting independent of a system clock i.e. it has a separate internal oscillator to work independently of a system clock.
- The watchdog timer cleared through software each time before the timeout period occurs.
- Whenever software failed to clear the watchdog timer before its timeout period, the watchdog timer resets the system.
- For this purpose, a watchdog timer is used to overcome software failures in real-time applications.
- The watchdog timer is also used to wake up the microcontroller from sleep mode.

Inside of PIC18F4550 Watchdog Timer

- In PIC18F4550, the watchdog timer uses a different 31 kHz INTRC clock and it is independent of a system clock.
- Watchdog Timer can be enabled in two ways through Configuration Register (CONFIG2H) and through WDTCON Register.
- CONFIG2H has a WDTEN bit to enable/disable the watchdog timer.
- WDTCON (WDT control register) has the SWDTEN bit which is used to enable/disable the WDT through software.

Below fig. shows how WDT operates in PIC18F4550.



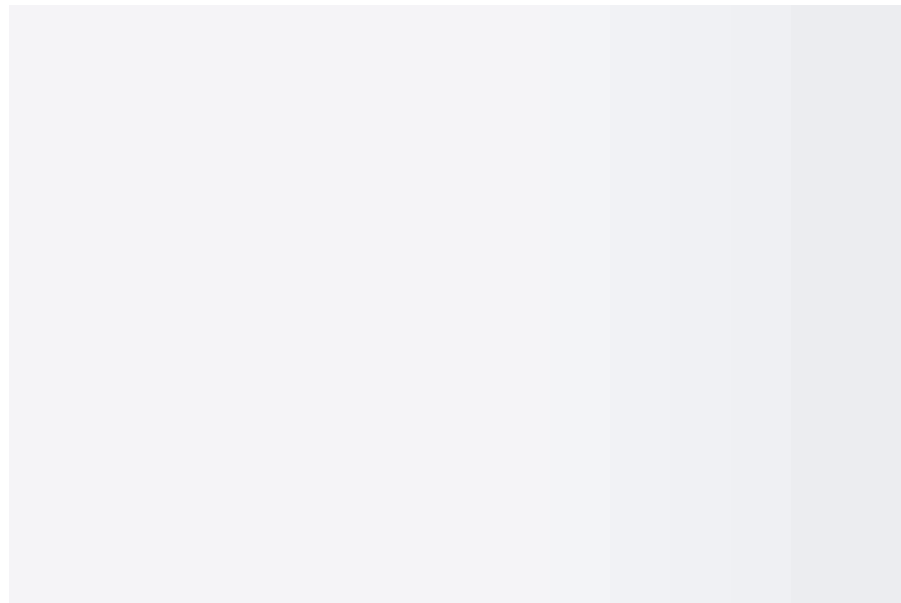
Platforms
(/explore)

Projects
(/projects)

Contests
(/contests)

2. Through WDTCON Register:

WDTCON Register: Watchdog Timer Control Register



Bit 0 – SWDTEN: Software Controlled Watchdog Timer Enable bit

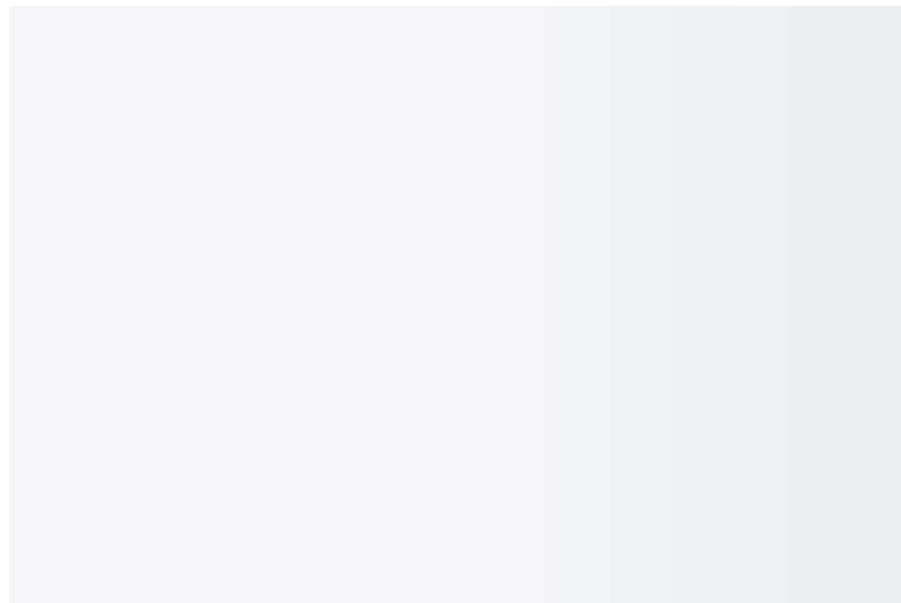
0 = Disable Watchdog Timer

1 = Enable Watchdog Timer

This software controlled watchdog timer can enable watchdog timer only if configuration bit has disabled the WDT.

If the WDTEN (configuration bit) is enabled, then SWDTEN has no effect.

How to calculate the WDT Timeout Period?



WDT Timeout Period Calculation

E.g.

If we configure Post-scaler **WDTPS3:WDTPS0 = 1001** i.e. 512 is Post-scaler then,



Program

- There are two ways for using Watchdog Timer in PIC18F4550. Here, we blink LEDs continuously and they stuck in between.
- Then WDT resets the PIC18F4550 microcontroller after a specific timeout period which starts blinking of LED again. When WDT reset occurs it displayed on LCD16x2.

Note: TO' bit in RCON register gives the status of WDT time-out occurrence.

Configuration Register Controlled WDT

```
/*
  Watchdog Timer in PIC18F4550
  http://www.electronicwings.com
*/

#include "Configuration_Header_File.h"
#include "LCD_16x2_8-bit_Header_File.h"
#include <pic18f4550.h>

#pragma config WDT = ON      /* Watchdog Timer Enable bit (WDT enabled)*/
#pragma config WDTPS = 256   /* Watchdog Timer Postscale Select bits */

void Dummy_Stuck();

#define LED LATB
#define PORT_Direction TRISB

void main(void)
{
    int count;
```

Software Controlled WDT

[+ Project \(/publish/project\)](#)

[Platforms](#)
[\(/explore\)](#)

[Projects](#)
[\(/projects\)](#)

[Contests](#)
[\(/contests\)](#)

Watchdog Timer in PIC18F4550

<http://www.electronicwings.com>

```
#include "Configuration_Header_File.h"
#include "LCD_16x2_8-bit_Header_File.h"
#include <pic18f4550.h>

#pragma config WDT = OFF      /* Watchdog Timer Enable bit (WDT disabled (c
#pragma config WDTPS = 512    /* Watchdog Timer Postscale Select bits */

void Dummy_Stuck();

#define LED_LATB
#define PORT_Direction TRISB

void main(void)
{
    int count;
```

Video

Platforms
[\(/explore\)](#)

Projects
[\(/projects\)](#)

Contests
[\(/contests\)](#)


(https://www.mouser.in?utm_source=electronicswing&utm_medium=display&utm_campaign=mouser-componentslisting&utm_content=0x0)


Powered By

Components Used

PICKit 4 MPLAB
PICKit 4 MPLAB


X 1


 (https://www.mouser.in/ProductDetail/Microchip-Technology/PG164140?qs=r5DSvIrkXmLKDuYNJlmlWw%3D%3D&utm_source=electronicswing&utm_medium=display&utm_campaign=mouser-componentslisting&utm_content=0x0)

 [Datasheet \(/components/pickit-4-mplab/1/datasheet\)](#)

PIC18f4550
PIC18f4550

X 1

 (https://www.mouser.in/ProductDetail/Microchip-Technology/PIC18F4550-I-P?qs=oKK8NaWdAJs8nLDXBGwMXw%3D%3D&utm_source=electronicswing&utm_medium=display&utm_campaign=mouser-componentslisting&utm_content=0x0)

 [Datasheet \(/components/pic18f4550/1/datasheet\)](#)

EW

ElectronicWings

(/)

Q

MOUSER

+ Project Electronics (/project)

(https://www.mouser.in?utm_source=electronicswing&utm_medium=display&utm_campaign=mouser-componentslisting&utm_content=0x0)

Platforms

(/explore)

Projects

(/projects)

Contests

(/contests)


Components Used

Powered By




LCD16x2 Display

LCD16x2 Display

X 1

 (https://www.mouser.com/ProductDetail/Adafruit/1447?qs=XAKIUOoRPe6ACImsjw7y7g%3D%3D&utm_source=electronicswing&utm_medium=display&utm_campaign=mouser-componentslisting&utm_content=0x0)

Downloads

<div></div> <div>PIC18F4550 Watchdog Timer Project File</div>	<div>Dow (/api/download/platform-attachment/180) d</div>
<div></div> <div>PIC18F4550 Software controlled WDT Project File</div>	<div>Dow (/api/download/platform-attachment/181) d</div>
<div></div> <div>PIC18F4550 WDT Simulation</div>	<div>Dow (/api/download/platform-attachment/182) d</div>

Comments



Comment

dilshankasthuriarachchi


[\(/users/dilshankasthuriarachchi/profile\)](#)
[2020-08-22 13:02:14](#)

Could you please What is the compiler has been in here?

[+ Project \(/publish/project\)](#)


Platforms
[\(/explore\)](#)

Projects [Reply](#) [Like](#) [Contests](#)
[\(/projects\)](#) [\(/contests\)](#)

[About Us \(/about\)](#)

[Business Offering \(/business-services\)](#)

[Host Platform \(/launch-platform\)](#)

[Contact Us \(/contactus\)](#)

[Terms of Service \(/terms-of-service\)](#)

[Cookies Policy \(/cookie-policy\)](#)

[Privacy Policy \(/privacy-policy\)](#)

Connect On:

Facebook(<https://www.facebook.com/electronicwings>)

LinkedIn(<https://www.linkedin.com/company/electronicwin>)

Youtube(<https://www.youtube.com/channel/UCNdqkukBtk4>)

Instagram(https://www.instagram.com/electronicwings_coigshid=1cip10jjttko)

ElectronicWings

© 2023