

Technical Article

Exploring Watchdog Timer Applications



Kelvin Odom

In my last post, “[What is a watchdog timer, and why is it important?](#)” I discussed how watchdog timers (WDTs) work and how you can implement them in your system. In this post, I’ll talk about why you should use a watchdog timer in your application, how a window watchdog timer works and some additional watchdog timer features that make them a great addition to many different applications.

Why Should You Use a Watchdog Timer in Your Application?

Watchdog timers have a home in a wide variety of applications, from drones and grid metering to motor control and beyond. For all of these different applications, the watchdog timer provides the same basic functionality: monitoring the processor for errant operation and issuing a corresponding signal.

However, the utility of this monitoring can vary based on the type of application. For example, it’s impractical to expect someone to manually reset a drone whose processor hangs up when it’s hundreds of feet in the air. In smart meters, this same issue could cause the device to be unable to read or record information. Meters and sensors in remote locations can also experience software failure causing the system to be unable to read and record information and no one to manually reset it. Code failure in systems that control motors and other mechanical components is especially alarming as faulty operation of these systems could cause physical harm to operators and nearby persons. Fortunately, using a watchdog timer can add redundancy to catch these errors and help prevent these problems from occurring.

How Does a Window Watchdog Timer Work?

A window watchdog timer, like that in [Figure 1](#), is a special type of watchdog timer that monitors not only if the signal from the processor fails to send before the watchdog timeout ends (for example, upper watchdog boundary) but also if the processor sends the watchdog input (WDI) signal before the watchdog timeout window begins (for example, lower watchdog boundary). This functionality is illustrated in [Figure 2](#).

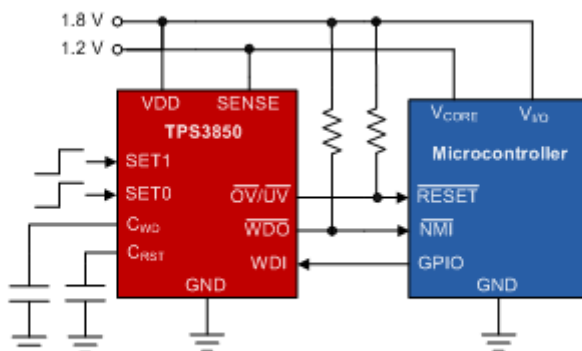


Figure 1. A Window Watchdog Timer (TPS3850) Monitoring a Microcontroller

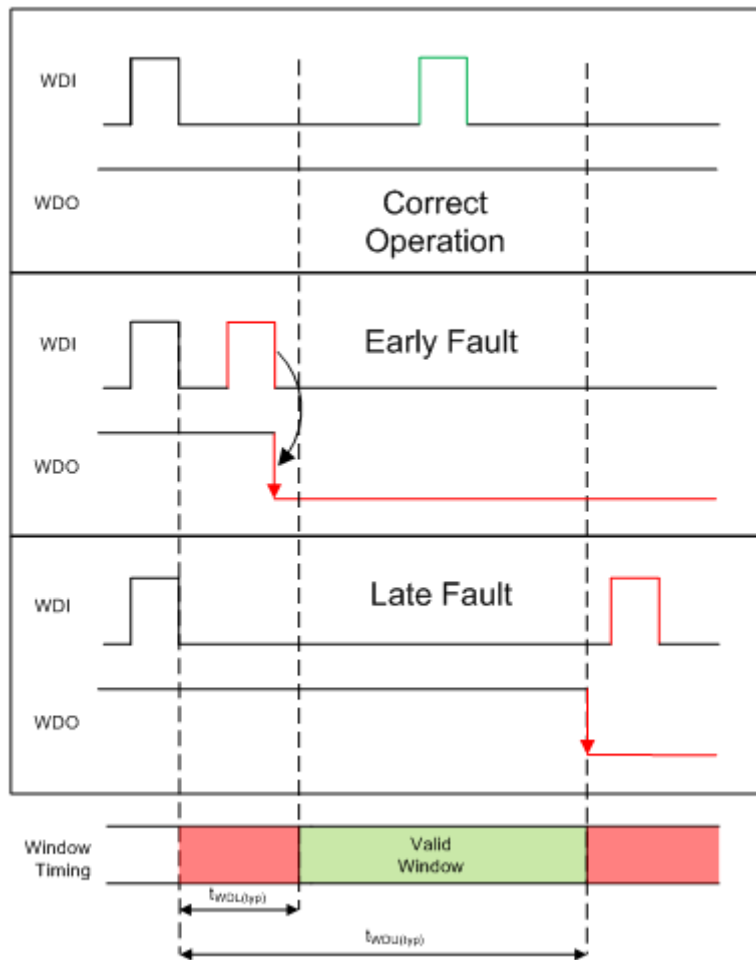


Figure 2. Window Watchdog Functionality

This is unlike a standard watchdog timer which only checks to see if the pulse comes before the timeout ends. Being able to monitor for an early WDI signal is useful as it can detect and halt undesirable overclocking.

What Are Some Additional Watchdog Timer Features?

The window watchdog timer is just one type of additional functionality in watchdog timers. Several additional features can ensure the robustness and functionality of a system:

- **Adjustable watchdog timeout** – Different systems and applications require different timeout periods for their WDI signal. An external capacitor and SET pins can be used to adjust the watchdog timeout anywhere from a few milliseconds to over a minute.
- **Watchdog timer disable function** – Some watchdog timers can be disabled using a pin on the device. This feature is useful during system initialization and debugging, when you don't want to reset the processor.

TI's [TPS3850](#), [TPS3851](#) and [TPS3852](#) family of devices is available in both regular and window watchdog options and contains all of the features described below in [Figure 3](#).

	TPS3850	TPS3851	TPS3852
Watchdog Type	Window	Standard	Window
Threshold Accuracy	±1%		
Quiescent Current	10uA (typ)		
Output Type	Open-Drain		
WDO Output	Yes		
Watchdog Timeout	Fixed & Adjustable		
RST & WDT Accuracy	±15%		
WD Disable	Yes		
RST Delay	Fixed & Adjustable	Fixed	Fixed
UV Fault Detection	✓	✓	✓
OV Fault Detection	✓		
SENSE Input	✓		
MR Input		✓	✓
Package	3x3 DFN-10	3x3 DFN-8	3x3 DFN-8

Figure 3. TI's TPS3850 Family

Now that you've learned more about the importance of a watchdog timer, how a watchdog timer works, and some additional features, consider the [TPS3850](#) family when designing your next application.

Additional Resources

- See more TI supervisors with [watchdog timers](#).
- Read my last blog post "[What is a watchdog timer and why is it important?](#)"

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on [ti.com](https://www.ti.com) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2023, Texas Instruments Incorporated