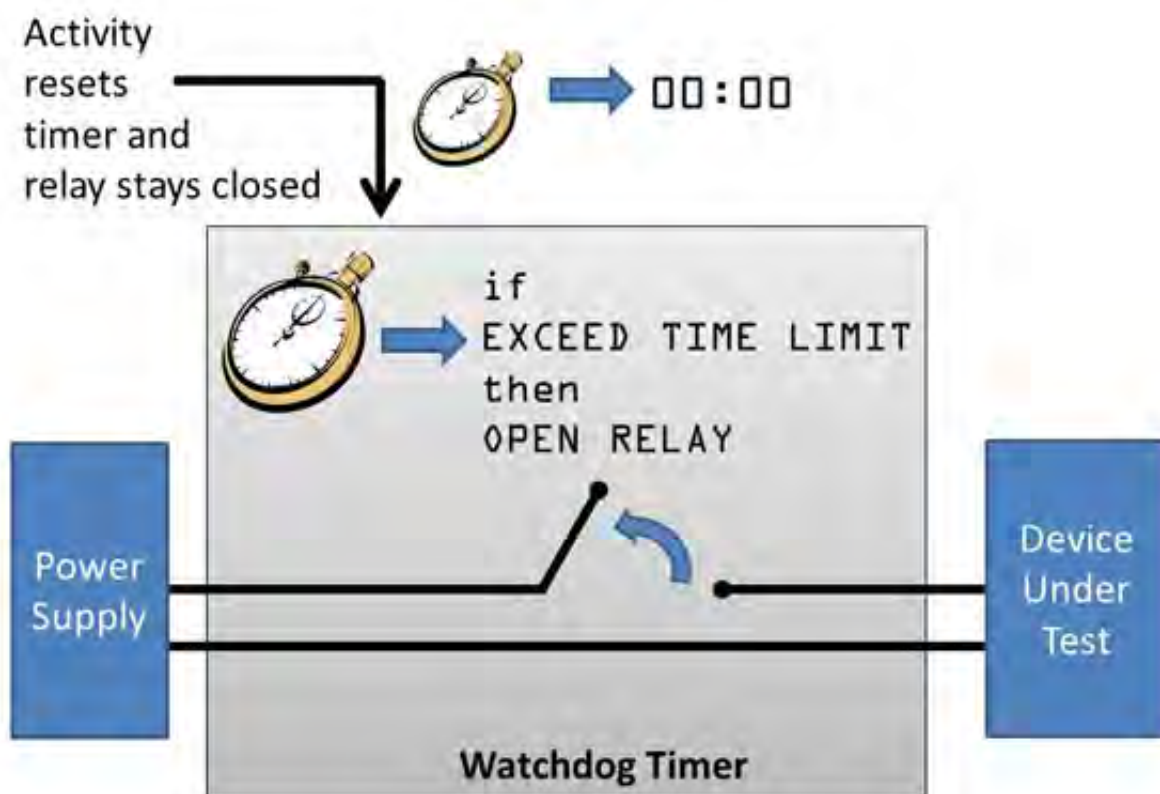


# Watchdog Timers Safeguard Devices from Uncontrolled Power During Test

by Bob Zollo / Keysight Technologies

To protect your device against potentially hazardous uncontrolled ac or dc power, you may want to consider a watchdog timer. A watchdog timer monitors activity; if no activity occurs within a preset amount of time, the timer interrupts the process.



Under normal conditions, when powering (via ac or dc power supply) your device during an automated test, some specific activity occurs within the test system. In turn, the watchdog timer restarts so that it won't elapse, or "time out." If, due to a hardware fault or program error, activity stops and the watchdog timer fails to restart, the timer will elapse and place the power supply in a safe state to avoid uncontrolled power going to the device under test (DUT) (Fig. 1).

Several conditions might warrant using a watchdog timer. For each of these conditions, a hazard could arise if the power supply is left on and powering the DUT:

- **Faulty program:** Despite our best efforts, we write programs with bugs. These bugs can cause the program to terminate prematurely and without proper shutdown, or trigger the program to enter some infinite loop and hang, leaving the power supply uncontrolled.
- **Faulty OS, drivers, or other system software:** Introducing bugs into software isn't solely human error. The operating system itself or the manufacturer-supplied instrument drivers may harbor bugs, causing the system to hang or crash. In addition, within the world of multitasking computers, an application or process might take an excessive amount of time, meaning the ac or dc power supply is left on but not serviced as often one would expect.
- **Faulty communications:** The host computer may be working fine, but if the communications system between the computer and ATE power supply should fail, it once again results in a power supply that's left on and not serviced. The failure could be communications hardware (i.e., bad interface) or a broken interface cable. Now that LAN is popular for instrument control, many parts of the communication system are uncontrollable. For example, routers and switches in your "IT closet" could fail or lose power. Or workers could cut wires or unplug signal paths. Of course, when the LAN is down, your power supply is uncontrolled.
- **Inadvertent user interaction:** The human element comes into play as well. A user could pause the ATE program, shut off power to control PC, or even walk away and never respond to a program prompt, causing the test program to wait indefinitely. All of these will result in an uncontrolled power supply.

### Example Hazards

There are many ways a test program can lose control over the power supply that's powering up your DUT. Potential hazards to the DUT include:

- **DUT is battery:** Most batteries don't like to be over-discharged or over-charged. If you're relying on the test program to terminate when a particular condition occurs (e.g., the battery voltage reaches its maximum or the battery is charged to capacity), and the program loses control over the power supply charging the battery, the power supply will happily continue to apply power to the battery. The end result is a potentially hazardous condition.
- **Temperature-cycling or stress-testing the DUT:** While your test objective could be to stress the DUT, the stress should be controlled. Losing control over the power supply stressing your DUT creates an invalid test and may even damage the DUT. Temperature cycling and stress tests tend to be of long duration (hours or days), and the longer the test, the greater the chance of a loss-of-control fault.
- **DUT is valuable or sensitive:** When testing a one-of-a-kind prototype, you probably want to avoid powering it beyond the specifics of the test plan. Alternatively, when the DUT is costly or can't be subjected to conditions that could cause a latent failure, you want to ensure that control of the power supplies isn't lost during the test.

## Watchdog Timer Implementation

A watchdog timer can be created with a custom-designed and -built hardware circuit, whereby the timer is reset by a digital line. Your test program must be able to assert that line, perhaps by calling a routine that wiggles the I/O line to the watchdog. If the program doesn't get to the timer before it times out, your custom watchdog circuit disables the power supply, either with a remote enable/disable I/O signal or by cutting the ac power to the instrument. It's a sure-fire way to implement an independent watchdog, but it does involve designing, building, and maintaining custom ATE hardware.

Another solution would be to use a power supply with a built-in watchdog function. The watchdog timer should be programmed for its timeout period. If the power supply does not receive any further programming activity before the timer times out, it turns itself off. Programming activity could be additional commands to set voltage or other functions, or commands to measure the power supply's output.



The built-in watchdog holds several advantages over the custom circuit. It requires no additional design or maintenance. It's simpler to interface to the system, requiring no additional hardware or wiring. Lastly, the watchdog is simpler to control, because the watchdog is programmed like any other feature of the supply. Figure 2 shows various examples of power supplies with built-in watchdog timers.

For more information, go to [www.TestEquity.com](http://www.TestEquity.com)