

The ongoing story of seconds on the taskbar

 devblogs.microsoft.com/oldnewthing/20250421-00/?p=111095

April 21, 2025



Over a decade ago, I noted that [early beta versions of the taskbar clock showed seconds, and sometimes even blinked the colon like some clocks do](#), but it was removed because the blinking colon and updating time were ruining Windows 95's benchmark numbers due to the need to keep all of the code paths related to text rendering in memory, as well as the stack of the thread in the Explorer process that updates the clock.

Even though computers are not under the same tight memory constraints as Windows 95, the taskbar still does not show seconds because Terminal Services would have hundreds of updating clocks, and even on single-user systems, the energy efficiency team [gives the side-eye to any timer that runs faster than once per minute](#).

But finally, in 2023, the Taskbar added an option to show seconds. The option is however disabled by default, and it comes with the warning "(uses more power)". How much power is it using?

The Taskbar team ran power consumption tests to try to convince the energy efficiency team that even though there was an extra cost, that extra cost was not exorbitant.

A member of the Taskbar team told me that their preliminary measurements showed that Explorer consumed 0.417 mW of energy in its default configuration, but the energy usage went up to 5.42 mW if seconds were enabled on the taskbar. This is over a factor of ten, so that sure seems like a big jump.

I'm having trouble finding information on how much energy a laptop screen consumes. [This Web site](#) gives a range of 200 mW to 1100 mW active power consumption for a 16-inch screen at 100 nits. Do I believe these numbers? No. Will I report them anyway? I guess I just did.

If we accept these numbers, then an additional 5 mW doesn't seem quite so bad overall.¹ It's an extra 2.5% on the high end, or 0.05% on the low end. That's a loss of 3 to 15 minutes over a 10-hour period.

¹ The energy efficiency team looks at the system as a whole. I mean, sure, Explorer decreasing battery life by 15 minutes over a 10-hour period might not sound like much, but suppose 10 other components come to the same conclusion. Now you've lost over two hours. They also look at the aggregate impact of Windows on the global environment and make recommendations like [having Windows Update perform its work when there are more low-carbon energy sources available](#). You can use the [PowerGridForecast](#) class to use this information in your own programs. There's even [sample code](#).