

# Why is Windows using only even-numbered processors?

 [devblogs.microsoft.com/oldnewthing/20230620-00](https://devblogs.microsoft.com/oldnewthing/20230620-00)

June 20, 2023



Raymond Chen

A customer had an 8-core system with 16 logical cores, and they noticed in Task Manager that Windows was using only the even-numbered cores. Are even-numbered processors faster than odd-numbered processors? Do odd-numbered cores have cooties? Does Windows simply dislike odd numbers?

What you're seeing is the effect of hyperthreading-aware scheduling.

Hyperthreading works by having two threads of execution running on a single CPU core: When one thread stalls, say, because it is waiting for data to arrive from memory, the CPU can execute instructions from the other thread. Basically, you are taking advantage of computational capacity that would otherwise go wasted because the CPU is stuck twiddling its thumbs waiting for some external resource to respond. The CPU has a tiny pre-emptive scheduler inside itself, so it can act like two virtual processors.

Windows understands the relationship between logical and physical cores, and it will try to spread work out among physical cores. If you have two threads, putting them both on the same physical core will cause them to compete for shared resources, like processor execution units and caches. If you have two physical cores available, then you want to put each thread on a separate physical core so they don't have to share these CPU resources with each other.

Suppose you have a dormitory with eight floors, two rooms on each floor: Rooms 1 and 2 on the first floor, rooms 3 and 4 on the second floor, and so on. Suppose that each floor has a single kitchen that is shared by the two rooms on that floor. If you have only eight occupied rooms, you would probably put one person on each floor, so that each one has exclusive access to a kitchen. Somebody who didn't understand that pairs of rooms share a kitchen would wonder why you are assigning people only to odd-numbered rooms.

That's what the Windows scheduler is doing. If there are only eight threads that need to run, the scheduler will spread them out so that each one goes onto a separate physical core.

**Bonus chatter:** Now, there are other factors in play. For example, if a core is parked to save energy, it is taken out of consideration for scheduling, at least until the system decides to unpark it to add additional computing capacity.

In our dormitory analogy, that would be like leaving the top floor empty and doubling up two occupants on a lower floor so you can turn off the electricity on the top floor.