So how are P-states related to power management?

Published:08/15/2008 Last Updated:08/15/2008

This relationship between P-states, voltage and frequency is well and good, but how does this relate to power management? Power is literally, energy usage per unit of time. To get the total energy usage, you integrate the instantaneous power over the interval you're interested in, i.e. get the area under the curve. If there are some energy savings to be had, see the previous discussion, we want to reduce the frequency, and so the voltage, such that the CPU is just shy of being 100% utilized. At this voltage, there is (ideally) no increase in the execution time of your application. This is possible because we're only getting rid of the idle time. By minimizing the voltage, we've minimized leakage current and so minimized the instantaneous power. (See comments in my blog entry, "Can P-states save overall energy?" http://software.intel.com/en-us/blogs/2008/07/31/can-p-states-save-overall-energy/.) Integrating over the entire execution time of the application, we find the total energy used is less. (Disclaimer: this applies under ideal circumstances. Actual mileage may vary.)

Similarly, spreading out the work the CPU has to perform over a longer interval reduces the peak power over the interval. As we mentioned earlier, this reduces cooling requirements with the aforementioned reduction in costs.

So in conclusion, we know that in clients, processor usage is very bursty, consisting of large periods of idle punctuated by bursts of furious activity. (The usage profile is different for servers, but then when are you going to be running servers without an A/C tether?) By choosing an appropriate P-state, we can minimize this idle time, reducing peak power, and (potentially) increasing power efficiency.

(This brings up an interesting point. What is the effect of P-states on processor efficiency? By entering a higher p-state (lower frequency), we've effectively downgraded the processor. If we think of processor efficiency as being the amount of work done over a given interval, there is no effect on processor efficiency. Why? Because we're doing the same about of work in a given period -- we're just filling in the idle time.)

Let's look at the type of environments that P-states might be useful within. Mobile environments, of course. But how about environments where you have significant limits in the cost and size of cooling equipment? The embedded environment generally has these constraints. Embedded processors often have to deal with severe space, weight and cooling limitations. Several more advanced embedded and special purpose processors use P-states. (No, I can't really mention what those "other processors" are except to state, unequivocally, that Intel processors are the best thing since sliced bread.)