

Erledigt

SSDT Speedstep Anzahl

Beitrag von „Brumbaer“ vom 21. März 2016, 17:17

Der Prozessor entscheidet über die Verwendung von P0 und P1. Das sind die nominal und Turbo Frequenz. P0 und P1 dienen nicht dem Energiesparen per se, sondern eine höhere Leistung als normal so lange zur Verfügung zu stellen bis es "zu gefährlich" für den Prozessor wird. Bei neueren Prozessoren wird dazu ein Modell der Wärmekapazität des Prozessors verwendet. Beim Turbobetrieb wird sie aufgefressen. Ist sie erschöpft gibt es keinen Turbomodus mehr bis sie in Ruhephasen wieder aufgefüllt wurde.

Die höheren States P2 bis Pn verwaltet das OS. Sie dienen dem Stromsparen. Im Wesentlichen schaut das OS in wie weit der Prozessor ausgelastet ist und liegt der Wert unter einer bestimmten Schwellwert wechselt es auf die langsame Stufe.

Ein Programmstück läuft nicht halb. Es läuft oder es läuft nicht. Die Auslastung einer CPU (logischer Prozessor innerhalb eines physikalischen Prozessors. Ein i7 Prozessor hat 4 Kerne und 8 logische CPUs) wird deshalb gemessen indem man schaut in wie vielen Prozent eines Zeitintervalls die CPU ein Programmstück ausführt. Dabei ist es der CPU nicht bekannt ob dieses Programmstück, so schnell wie möglich ausgeführt werden muss, oder ob es eigentlich Zeit hätte.

Im Idealfall wäre die Auslastung immer bei 100%.

Ein Wechsel dauert in der Größenordnung von 20ms bis 30ms. Das ist für einen Prozessor ein Haufen Zeit. Man hinkt den richtigen Werten immer hintennach es sei denn man weiß was in 20ms benötigt wird.

Deshalb ist ein unnötiger Wechsel unter Geschwindigkeitsgesichtspunkten zu vermeiden. Habe ich nun für jedes Prozent Auslastung einen P-State ist man ständig mit Wechseln beschäftigt und immer zu spät.

Also weniger P-States. Wie z.B. Alles oder (fast) nichts. Da man nicht weiss, wie lange ein Programmstück laufen wird und man annehmen kann, dass für gewöhnlich maximale Ausführungsgeschwindigkeit gewünscht ist, ist ein schneller Wechsel von Pn auf P1 wünschenswert. Umgekehrt ist es erstrebenswert den Wechsel von P1 erst auszuführen, wenn man sich ziemlich sicher ist, dass man die Leistung nicht gleich wieder braucht.

Zusätzlich kann das OS durch das Aufteilen von Programmstücken auf die CPUs deren Auslastung steuern, so dass der eine oder andere Kern "wahrscheinlicher" keine Leistung benötigt und niedriger getaktet werden kann. Eine mit vollem Takt und eine mit niedrigem

Takt sind genauso energieeffizient, wie zwei mit mittlerem Takt.

Deshalb hat sich Apple wohl entschieden, dass ein weiterer P-State genug ist.

Das ist so programmiert. Man hätte auch eine Tabelle lesen können und dann je nach Anzahl der Einträge P-States verwalten können. Aber dem ist nicht so.

Es gibt wohl eine gepatchte `AppleIntelCPUPowerManagementInfo.kext` die mehr P-States unterstützt. Verwendung auf eigene Gefahr.

Beim Skylake kann man die Verwaltung der P-States komplett an den Prozessor übergeben (Speed Shift) ob Apple das macht oder machen wird weiß ich nicht.

Ich habe keine Angaben darüber gefunden wie viel Energie jetzt tatsächlich dank P-States gespart wird.

Während ich mir vorstellen kann, dass die Energieersparnis fühlbar ist, bin ich mir nicht sicher ob dies bei zusätzlichen P-States auch der Fall ist.

Das dürfte auch stark damit zusammenhängen wie groß der Unterschied in den Spannungen und Taktraten in den verschiedenen States ist. Bei übertakteten Prozessoren, wird es sich deutlicher bemerkbar machen, da die zusätzliche Taktrate durch unverhältnismäßig mehr Spannung erkaufte wird.

Leider viel könnte und möglich. Vielleicht macht jemand mal den Selbstversuch und definiert den zweiten P-State wie den ersten, also ohne Runterschalten und vergleicht die Leistungsaufnahme.