

Hilfe! Meine CPU taktet uns bis 3,4 GHz, obwohl sie 3,6 GHz können sollte

Beitrag von „Torpor“ vom 13. Februar 2021, 22:38

Laut Intel Power Gadget taktet meine CPU nur auf 3,4 GHz, obwohl sie 3,6 GHz können sollte und ist damit im Geekbench-Vergleich sogar langsamer als mein alter Prozessor.

Ich hab einen

Intel Xeon E3-1230V2 @ 3.30 GHz

zu einem

Intel Xeon E3-1280V2 @ 3.6 GHz

getauscht. Wird der maximale Takt irgendwie durch die dsdt gebremst?

Beitrag von „LetsGo“ vom 13. Februar 2021, 22:56

[Torpor](#)

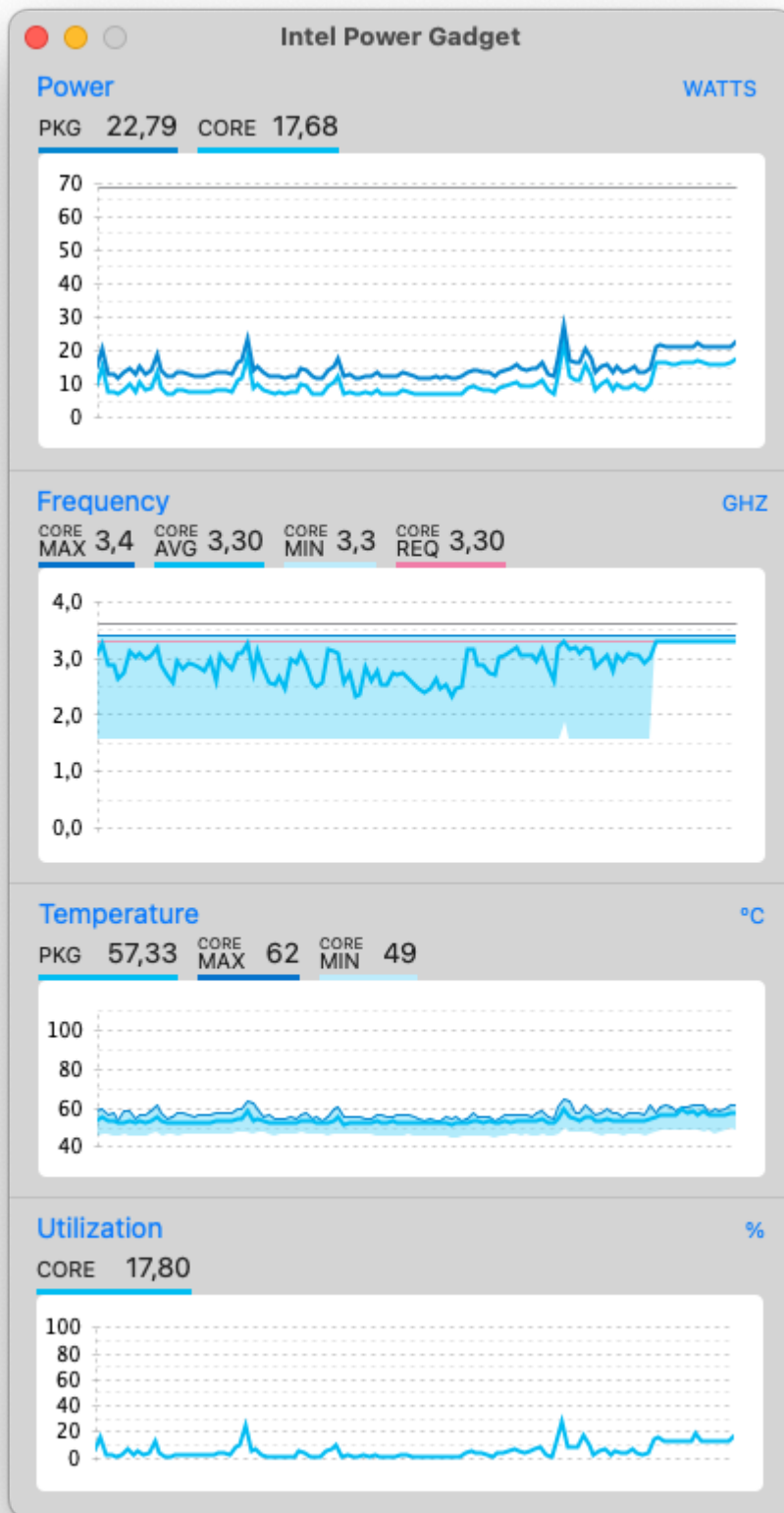
Hast du mit IntelPowerGadget den MaximumFrequency Test gemacht?

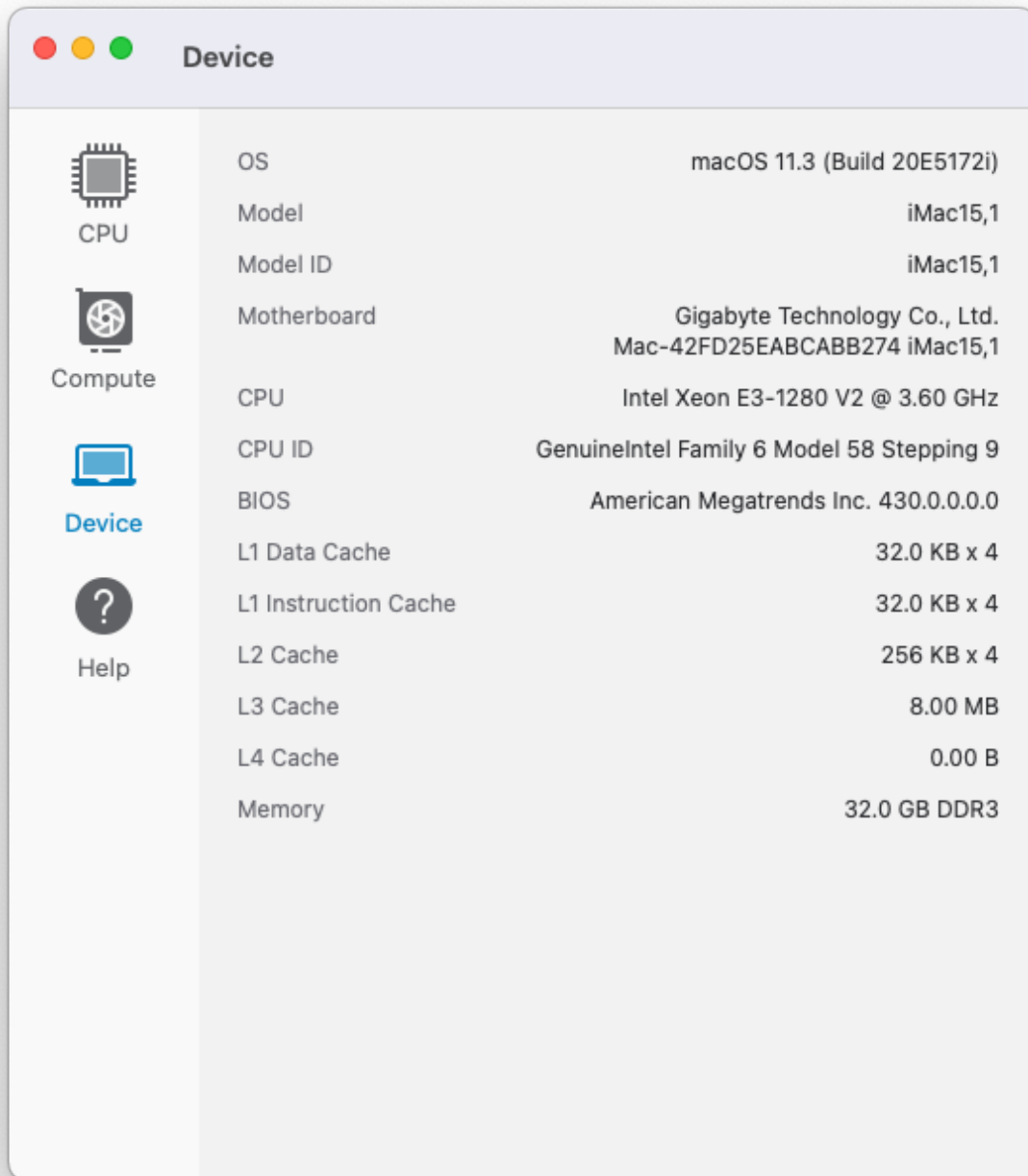


Beitrag von „Torpor“ vom 13. Februar 2021, 23:25

Ja, 3,4 GHz bekomme ich angezeigt. Mein Prozessor sollte aber 3,6 können.







Beitrag von „LetsGo“ vom 13. Februar 2021, 23:59

[Torpor](#)

Fällt mir im Moment auch nichts dazu ein. Aber meine CPU erreicht auch nur max. 4,6 GHz anstatt der 4,7 GHz wenn das System warm ist. Ich denke das der Maximaltakt dementsprechend sinkt, wenn mehrere Kerne höher takten. Und ich bin weit von kritischen Temperaturen entfernt. Kannst ja mal [HWMonitorSMC2](#) installieren. Da siehst du dann, wie die einzelnen Kerne takten.

Ansonsten könntest du eventuell noch mit FrequencyVetoren spielen, da der iMac15,1 ja i5 bzw. i7 Prozessoren verbaut hat. Ich weiß jetzt aber nicht welches SMBIOS deiner CPU am Nächsten kommt.

Beitrag von „Torpor“ vom 14. Februar 2021, 00:14

SMBIOS, ja!

Der MacPro6,1 hat ja Ivy Bridge XEON Prozessoren verbaut wie mein Rechner. Vielleicht wäre das einen Versuch wert.

Ist schon eine Weile her, dass ich meinen Open Core Boot Stick erstellt habe. Und ich habe keine wirklich Ahnung von dem ganzen Hackintosh-Forum-Ding und hab dazu noch das Talent, immer wenn ich mein System zum Laufen gebracht habe, alles was ich mir angelesen habe, wieder zu vergessen.

Wie kann ich jetzt auf dem kürzesten und einfachsten Weg aus dem iMac15,1 ein MacPro6,1 machen?

Einfach

```
<key>SystemProductName</key>
```

```
<string>iMac15,1</string>
```

zu

<key>SystemProductName</key>

<string>Mac Pro 6,1</string>

ändern oder wie?

Beitrag von „5T33Z0“ vom 14. Februar 2021, 00:17

[Torpor](#) Mit [ssdtPRGen](#) eine SSDT-PM.aml erzeugen und in die Config einbinden. SSDT-PLUG.aml entfernen (falls vorhanden), da diese CPU zur IvyBridge CPU Family gehört und daher nicht das neuere X86PlatformPlugin für CPU Power Management verwenden kann, sondern nur das Legacy Plugin. ssdtPRGen verfügt über die Daten für diese CPU und schreibt sie in diese SSDT-PM. Danach sollte die CPU dann weiter runter takten im Idle, aber auch höher boosten.

Die Datei nach EFI > Clover > ACPI > patched kopieren (Name SSDT.aml)

Beitrag von „Torpor“ vom 14. Februar 2021, 00:20

Danke. Also erstmal iMac15,1 lassen?

Bei ssdtPRGen steht, wie ich es herunterlade, die Dateigröße überprüfe und die permissions ändere. Aber was mache ich danach? Verstehe es nicht...

Beitrag von „5T33Z0“ vom 14. Februar 2021, 00:25

Idealerweise ein SMBIOS wählen, was zur CPU passt *UND* vom aktuellen OS unterstützt wird. Aber nach erstellter SSDT sollte das schon funktionieren.

Beitrag von „Torpor“ vom 14. Februar 2021, 00:27

Bei ssdtPRGen steht, wie ich es herunterlade, die Dateigröße überprüfe und die permissions ändere. Aber was mache ich danach? Verstehe es nicht...

Und SMBIOS ändere ich einfach, wenn ich zum Beispiel

```
<key>SystemProductName</key>
```

```
<string>iMac15,1</string>
```

zu

```
<key>SystemProductName</key>
```

```
<string>Mac Pro 6,1</string>
```

ändere oder gehört da mehr dazu?

Beitrag von „5T33Z0“ vom 14. Februar 2021, 07:58

<https://www.youtube.com/watch?v=OEYi0fNU8oo>

Als Namen der Datei SSDT.aml verwenden für Clover.

SystemProductNames werden immer ohne Leerzeichen geschrieben. Also **MacPro6,1** statt

Mac Pro 6,1. Vorher natürlich checken, ob das vom verwendeten macOS noch unterstützt wird. Ansonsten kommt ein Stoppschild beim Bootversuch. Daher EFI am besten auf einen in FAT32 formatierten USB Stick kopieren und damit starten.

Beitrag von „mitchde“ vom 14. Februar 2021, 08:42

Würde das SMBIOS komplett auf den anderen MACTyp umstellen.

Dazu erstmal dein - du nutzt Clover ? - config.plist mal wo anderes hin kopieren.

Dann mit CloverKonfigurator deine bestehende config .plist mit dem Konfigurator öffnen und auch die vorher (Kopie) config.plist.

In der Kopie änderst du das Mac Modell per Auswahlliste auf MacPro6,1.

Dann kopierst du noch aus der parallel geöffneten SMUUID und die Seriennummer in die geänderte (MacPro) config.plist (teilweise werden diese bei Apps für Lizenz Aktivierung gebraucht).

WICHTIG: Bei solche Änderungen wie beim Powermanagement ist ein **USB Stick** mit aktuell funktionierendem Clover) echt gut! Falls noch nicht vorhanden besser mal anlegen oder du hast den evtl. schon auf ner anderen HD , das geht auch. So kannst du, wenn die Änderungen früh zur Kernelpanic führen, vom USB Stick starten und das rückgängig machen. **Oder** andersrum du machst diese Änderungen erstmal nur auf dem USB Stick, bootest von da um das zu prüfen 😊

Damit andere dir evtl. besser helfen können wäre ein Upload deiner config.plist nützlich.

PS: Normalerweise funktioniert die generate PSTATES/CSTATES Funktion in Clover gut.

Wenn du dir mit dem SSDTGEN eine eigene SSDT.aml generierst musst du diese generate Funktion UND auch Plugintype deaktivieren UND bei dem Plugintype als Auswahlliste leer lassen.

BEides parallel (SSDT..aml und Clovers Generate States, PluginType wird Murks 😊

EDIT: Gerade gesehen dass laut Dortania Guides diese Ivy Bridge-E(HEDT/server) CPU doch XPCM

kann (also Plugintype aktiviert in Clover nutzt man kein SSDTGEN .aml

Note: *SSDT-PLUG is only compatible with Intel's Haswell and newer CPUs, Sandy Bridge and Ivy Bridge will need to follow the [ssdtPRgen method \(opens new window\)](#)(in post-install))*

- **Ivy Bridge-E**(HEDT/server) however is **supported by XCPM**

<https://dortania.github.io/Get...-ACPI/Universal/plug.html>

SSDTGEN:

<https://github.com/Piker-Alpha/ssdtPRGen.sh>

Das Ergebnis ist ein .dsl (Lesbar mit TExtedit) und die .aml (die du in Clover nutzt).

Sollte das mit der Aussage oben (**Ivy Bridge-E kann XCPM**) stimmen, müsste in der lesbaren dsl im Text

"plugin-type", One auftauchen bzw. schon im Header der .dsl , welche SSDTGEN erzeugt

"machdep.xcpm.mode.....: 1"

So sieht der HEader bei meiner IVY Normal CPU aus.

```

DefinitionBlock ("", "SSDT", 1, "APPLE ", "CpuPm", 0x00021500)
{
    External (_PR_.CPU0, DeviceObj)
    External (_PR_.CPU1, DeviceObj)
    External (_PR_.CPU2, DeviceObj)
    External (_PR_.CPU3, DeviceObj)

    Scope (\_PR.CPU0)
    {
        Method (_INI, 0, NotSerialized) // _INI: Initialize
        {
            Debug = "ssdtPRGen version.....: 21.5 / Mac OS X 10.15.2 (19C57)"
            Debug = "custom mode.....: 0"
            Debug = "host processor.....: Intel(R) Core(TM) i5-3570K CPU @ 3.40GHz"
            Debug = "target processor.....: i5-3570K"
            Debug = "number of processors...: 1"
            Debug = "baseFrequency.....: 1600"
            Debug = "frequency.....: 3400"
            Debug = "busFrequency.....: 100"
            Debug = "logicalCPUs.....: 4"
            Debug = "maximum TDP.....: 77"
            Debug = "packageLength.....: 25"
            Debug = "turboStates.....: 6"
            Debug = "maxTurboFrequency.....: 4000"
            Debug = "machdep.xcpm.mode.....: 1"
        }

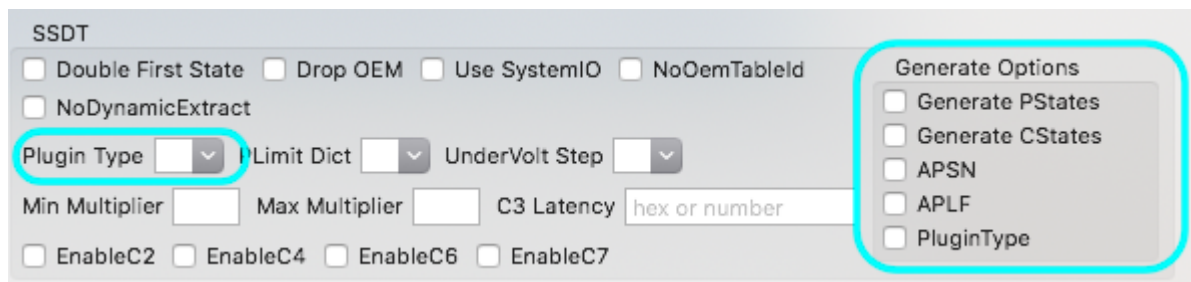
        Name (APLF, Zero)
        Name (APSN, 0x06)
        Name (APSC, Package (0x10)

```

PS: Bei obigem Bild, Header der SSDTGEN erzeugten .dsl für meine I5-3570K - normaler Turbo Takt 3.8 Ghz habe ich beim Aufruf der shell die max Turbofrequenz auf **4.0 GHz** verändert. So ist das gleich wie im BIOS eingestellt. Ohne Parameter erstellt der SSDTGEN das für den Prozessor normal nötige dsl., nur bei OC schon im BIOS sind Parameter nötig.

Beitrag von „5T33Z0“ vom 14. Februar 2021, 08:48

Guter Hinweis [mitchde](#) Ich nutze Clover gar nicht mehr, von daher vergesse ich es manchmal. Die SSDT zu generieren ist auf jeden Fall eine Gute Idee (erleichtert Wechsel zu OpenCore). 😄



Beitrag von „Torpor“ vom 14. Februar 2021, 09:45

Danke für die Tipps. Ich nutze aber OpenCore. Sorry.

Wie generiere ich denn eine SSDT für OpenCore? Kann ich dafür auch den Clover Configurator nutzen oder gibt es etwas entsprechendes für OpenCore?

Beitrag von „mitchde“ vom 14. Februar 2021, 10:51

Nun du generierst die SSDT (powermanagement CPU plus plugintype) **gleich** egal ob für Clover oder OC!

Bei Clover gibts eben Zusatzfunktion Plugintype und Generate C/Ptstates was OC nicht hat, das erspart manchen diese SSDT Sache.

Die gleiche von ssdtgen erzeugte .aml kann insofern bei beiden (Clover und OC) genutzt werden (kopiert entsprechenden OC / Clover UnterOrdner) UNTERSCHIED bei OC ist nur, dass du zudem diese .aml noch in OC config.plist eintragen musst, sonst beachtet OC die NICHT. Clover nutzt alle .aml automatisch die im entsprechenden Unterordner sind, OC nicht.

Beitrag von „5T33Z0“ vom 14. Februar 2021, 11:14

[Torpor](#) Generier erstmal die SSDT und probier sie aus. Danach kannst du immernoch am SMBIOS rumspielen. Vorher in der Config unter ACPI > Delete diese beiden Patches einbauen und aktivieren:

Removing CpuPm:

Key	Type	Value
All	Boolean	YES
Comment	String	Delete CpuPm
Enabled	Boolean	YES
OemTableId	Data	437075506d000000
TableLength	Number	0
TableSignature	Data	53534454

Removing Cpu0Ist:

Key	Type	Value
All	Boolean	YES

Key	Type	Value
Comment	String	Delete Cpu0Ist
Enabled	Boolean	YES
OemTableId	Data	4370753049737400
TableLength	Number	0
TableSignature	Data	53534454

Nachdem die SSdT generiert und eingebunden wurde, die Patches aber wieder deaktivieren! Das übersehen viele, weil der config checker das als eine falsche Einstellung bewertet (was nicht stimmt).

Beitrag von „Torpor“ vom 14. Februar 2021, 13:29

Wenn ich diese ssdtPRGen.sh-Sache versuche, bleibe ich immer hier hängen:

```

Last login: Sun Feb 14 13:24:38 on ttys000
phil@PhilWs-Mac-Pro ~ % curl -o - /ssdtPRGen.sh https://raw.githubusercontent.com/Piker-Alpha/ssdtPRGen.sh/Beta/ssdtPRGen.sh
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 153k  100 153k    0     0 1281k      0 --:--:-- --:--:-- --:--:-- 1281k
phil@PhilWs-Mac-Pro ~ % chead -x - /ssdtPRGen.sh
phil@PhilWs-Mac-Pro ~ % curl -o - /library/ssdtPRGen.zip https://codecloud.github.com/Piker-Alpha/ssdtPRGen.sh/zip/Beta
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 347k  100 347k    0     0 348k      0 @:00:01 @:00:01 --:--:-- 348k
phil@PhilWs-Mac-Pro ~ % unzip -qu - /library/ssdtPRGen.zip -d - /library/
phil@PhilWs-Mac-Pro ~ % mv - /library/ssdtPRGen.sh-Beta - /library/ssdtPRGen
phil@PhilWs-Mac-Pro ~ % rm - /library/ssdtPRGen.zip
phil@PhilWs-Mac-Pro ~ % $ - /ssdtPRGen.sh
zsh: command not found: $
phil@PhilWs-Mac-Pro ~ %

```

Was mache ich falsch?

Product Type auf Mac Pro 6,1 ändern war einfach, hat aber nichts gebracht.

Beitrag von „LetsGo“ vom 14. Februar 2021, 14:06

[Torpor](#)

Das `curl -o ~/ssdtPRGen.sh https://raw.githubusercontent.com/Piker-Alpha/ssdtPRGen.sh/Beta/ssdtPRGen.sh` hat die Datei `ssdtPRGen.sh` im Home Directory erzeugt.

Und der Befehl zum ausführen des Scripts lautet `./ssdtPRGen.sh`.

Beitrag von „Torpor“ vom 14. Februar 2021, 15:31

[LetsGo](#)

Okay. So hat es funktioniert. Danke.

Jetzt habe ich einen Ordner 'ssdtPRGen' mit einigen Dateien.

Welche davon muss/ müssen jetzt in den ACPI-Ordner von OpenCore?

ssdt.dsl (die so aussieht wie in [@mitchde](#) Screenshot) oder ssdt.aml (von der in dem YouTube-Video die rede ist, das 5T33Z0 verlinkt hat)?

Und wie verweise ich darauf in meiner config.plist ?

Beitrag von „LetsGo“ vom 14. Februar 2021, 15:43

[Torpor](#)

ssdt.aml in SSDT-PM.aml umbenennen. Die SSDT-PM.aml in den ACPI Ordner deiner EFI kopieren und in der config.plist einfügen.

Die beiden ACPI/Delete Einträge wieder auf NO setzen. Also einfach so machen, wie es im Guide oder im Bild beschrieben wird.

<https://dortania.github.io/Opency-bridge-power-management>

Remember to now add this to both EFI/OC/ACPI and your config.plist, I recommend renaming it to SSDT-PM to find it more easily.

Finally, we can disable our previous ACPI -> Delete entries(Enabled set to NO):

Key	Type	Value
All	Boolean	YES
Comment	String	Drop CpuPm
Enabled	Boolean	NO
OemTableId	Data	437075506d000000
TableLength	Number	0
TableSignature	Data	53534454

Key	Type	Value
All	Boolean	YES
Comment	String	Drop Cpu0Ist
Enabled	Boolean	NO
OemTableId	Data	4370753049737400
TableLength	Number	0
TableSignature	Data	53534454

Beitrag von „Torpor“ vom 14. Februar 2021, 16:09

Danke schonmal an alle! Jetzt zeigt mir Intel Power Gadget sogar 3,7 GHz CORE MAX an und CORE MIN geht runter auf 1,6.

Bei Geekbench habe ich jetzt einen dem Prozessor angemessenen Score!

Super!

Diese ACPI -> Delete Geschichte checke ich aber noch nicht.

1. Was bringt das?

und

2. Wie mache ich das?

Es hat bestimmt irgendwas damit zu tun:



Aber wie kann ich da etwas hinzufügen und muss ich alle 2x6x3 Zellen eintragen aus den Tabellen irgendwo manuell reinschreiben?? 🤔

Beitrag von „LetsGo“ vom 14. Februar 2021, 16:27

[Torpor](#)

Eigentlich hätten die laut Guide vor der Nutzung von ssdtPRGen auf YES gesetzt werden müssen. Ich habe diese Einträge z.B. in meiner config.plist drin, weil ich eine OC Snapshot mit propertree erstellt habe. Sind die auf False haben sie eh keine Bedeutung und können auch aus der config.plist gelöscht werden, aber du hättest sie eben vor der Nutzung von ssdtPRGen auf yes setzen müssen. Aber anscheinend funktioniert die Geschichte jetzt auch so bei dir.

```

-ADP
+ADP
+ Dictionary 6 key/value pairs
+ Array 6 entries
-8
+ Dictionary 3 key/value pairs
+ String SSDT_AKMC.amt
+ Boolean True
+ Boolean True
-1
+ Dictionary 3 key/value pairs
+ String SSDT_EC.amt
+ Boolean True
+ Boolean True
-2
+ Dictionary 3 key/value pairs
+ String SSDT_GPU_SPOOF.amt
+ String SSDT_GPU_SPOOF.amt
+ Boolean True
+ Boolean True
-3
+ Dictionary 3 key/value pairs
+ String SSDT_PLUS.amt
+ Boolean True
+ Boolean True
-4
+ Dictionary 3 key/value pairs
+ String SSDT_POWER_DISABLE.amt
+ Boolean True
+ Boolean True
-5
+ Dictionary 3 key/value pairs
+ String SSDT_PMC.amt
+ Boolean True
+ Boolean True
+ Boolean True
-Delete
+ Array 2 entries
+ Dictionary 6 key/value pairs
+ Boolean False
+ String Delete Cpuid
+ Boolean False
+ Data <037130 4973400>
+ Number 0
+ Data <0304454>
-1
+ Dictionary 6 key/value pairs
+ Boolean False
+ String Delete Cpuid
+ Boolean False
+ Data <037130 4973400>
+ Number 0
+ Data <0304454>

```

Nutze doch lieber einen plist Editor (z.B. Propertree, Xcode...) anstatt des OC Konfigurators, finde ich besser, da man exakt sieht was man tut.

Anbei habe ich dir eine Datei namens delete.plist angehängt, die die beiden Einträge enthält. Kannst du dann einfach bei dir in die config.plist unter ACPI/Delete kopieren. Solltest diese Einträge dann auch im OC Configurator sehen.

Danach kannst du die Prozedur wie im Dortania Guide wiederholen, falls du das möchtest.

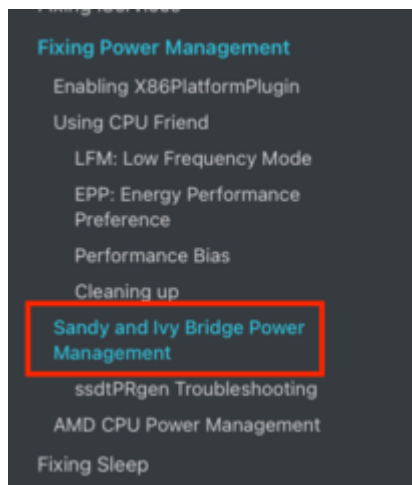
Beitrag von „Torpor“ vom 14. Februar 2021, 16:58

Danke. was genau bewirken die?

Beitrag von „LetsGo“ vom 14. Februar 2021, 17:10

[Torpor](#)

Ich habe dir den Dortania Guide verlinkt. Unter dem Punkt "Sandy und Ivy Bridge Power Management" findest du Folgendes. Mehr kann dir auch nicht sagen. Falls du mehr erfahren möchtest solltest du vielleicht mal die Begriffe "CpuPM" und Cpu0Ist" googlen.



Initializing with OpenCore's setup in the Ivy Bridge section, we recommended users drop their CpuPm and Cpu0Ist to avoid any issues with AppleIntelCPUPowerManagement.kext. But dropping these tables have the adverse affect of breaking turbo boost in Windows. So to resolve this, we'll want to keep our OEM's table but we'll want to add a new table to supplement data only for macOS. So once we're done creating our CPU-PM table, we'll re-add our OEM's CPU SSDTs.

To start, grab your config.plist then head to ACPI -> Delete and ensure both of these sections have Enabled set to YES:

Beitrag von „5T33Z0“ vom 14. Februar 2021, 17:19

[Torpor](#) Wie ich zuvor bereits geschildert habe, müssen diese Patches in die Config eingebaut

werden unter ACPI > Delete und aktiviert werden, bevor man ssdtPRGen verwendet, damit die richtigen Werte für die CPU ausgelesen und in die SSDT geschrieben werden können. Nache Einbau der Patches neustarten, dann SSDTPRGen verwenden.

[LetsGo](#) Die Hierarchie in deiner Plist ist falsch, was zu Irritationen führt und auch zu nicht funktionierenden patches und zu Fehlermeldungen führt.

Beitrag von „LetsGo“ vom 14. Februar 2021, 17:22

5T33Z0

~~Verstehe jetzt nicht ganz, was du damit meinst?~~

Achso du meinst weil diese unter Root/Delete liegen. War weil ich die aus meiner config.plist mittels copy/paste in eine Neue gepackt habe. Da hätte ich die ganze Struktur mitnehmen sollen. Hast Recht, das kann zu Irritationen führen.

Beitrag von „5T33Z0“ vom 14. Februar 2021, 17:27

[Torpor](#) Nicht verstehen, einfach machen!

- Die beiden Patches in die Config unter ACPI > DELETE einbauen und einschalten.
- Config speichern
- Neustarten.
- SSDTPrGen ausführen >> SSDT generieren
- SSDT.aml in EFI > OC > ACPI kopieren
- Snapshot der Config erstellen mit PropeTtree
- Die beiden Patches wieder deaktivieren.
- Neustarten
- freuen.

Beitrag von „LetsGo“ vom 14. Februar 2021, 17:32

5T33Z0

Dir ist aber schon klar, dass ich eine CFL CPU habe. Die Ausführung galt nur für den TE. Für mich ist die ssdtPRGen Geschichte irrelevant!

Beitrag von „5T33Z0“ vom 14. Februar 2021, 17:47

[LetsGo](#) Ich habe von dem Pfad in deiner Delete.plist gesprochen - von nichts anderem! Und der ist falsch. Da ist einfach nur ein Array Delete mit den beiden Patches drin. Die Hierarchie ist aber ACPI > Delete. Und wenn man sie einfach so in die Root der Config kopieren würde, wie in deiner Plist gäbe es a) Fehler und b) würde es nicht funktionieren

Beitrag von „LetsGo“ vom 14. Februar 2021, 17:55

5T33Z0

Hast ja Recht, ist mir im Nachhinein auch aufgefallen. Dachte aber auch, dass es der TE eigentlich zusammenbringt die beiden Patches mit Copy/Paste unter Root/ACPI/Delete einzufügen.

[Zitat von 5T33Z0](#)

Nicht verstehen, einfach machen!

Und das klang eben so, als ob ich die Patches anwenden wollte.

Aber alles gut!

Beitrag von „5T33Z0“ vom 14. Februar 2021, 17:59

War auf Toror bezogen das "einfach machen"

Obwohl, da fällt mir ein, ich kann die ja auch generieren per Terminal command und dem passenden Modifizier 😁

EDIT:

Terminal-Befehl: `sudo ./ssdtPRGen.sh -p 'E3-1280 v2'` funktioniert:

Spoiler anzeigen

Leider ist XCPM Mode noch aus.

Weiß nicht, was ich hinten dran hängen muss an den Befehl, damit XCPM aktiviert wird.

Die Hilfe ist da jetzt auch nicht wirklich hilfreich. Ausprobiert habe ich:

-xcpm mode: 1 = invalid argument,

-xcpm = invalid argument.

-xcpm:1 = invalid argument

-x:1 = invalid argument

-x=1 invalind argument

Beitrag von „mitchde“ vom 14. Februar 2021, 21:18

-x 1

Ohne = oder :

Obs geklappt hat siehst du in der .DSL (Textversion der -AML)

"....

Store ("machdep.xcpm.mode.....: 1", Debug)

....."

Beitrag von „5T33Z0“ vom 14. Februar 2021, 23:06

[mitchde](#) Ja, hat geklappt. Danke

[Torpor](#) Kannst Du mal testen [SSDT-PM.aml](#)

Beitrag von „Torpor“ vom 15. Februar 2021, 10:20

@[5T33Z0](#) kann ich gerne heute Abend mal testen.

Ich würde mich trotzdem außerordentlich freuen, wenn mir jemand kurz erklären könnte, was sich dadurch zusätzlich verbessern soll, nachdem mein Prozessor ja bereits zu funktionieren scheint wie er soll nach ssdtPRGenohne die Patches.

Beitrag von „5T33Z0“ vom 15. Februar 2021, 13:29

Ich dachte Du hättest es nicht hinbekommen. Aber wenns läuft, dann brauchst die Datei

natürlich nicht.