

Opencore statt Clover - und es klappt beharrlich nichts ...

Beitrag von „griven“ vom 26. Oktober 2023, 23:01

Cool das es jetzt endlich klappt 😊

Ich habe im Endeffekt nur den ACPI->Delete Eintrag nochmal neu in die config.plist eingefügt diesmal aber den Wert für TableLength auf 0 anstatt der vorherigen 224 gesetzt und zudem den Wert für OEMTableID entfernt und das scheint der Schlüssel zum Erfolg (ein Teil davon) gewesen zu sein...

Wie [mhaeuser](#) ja schon erwähnt hat ist es relativ unwahrscheinlich das Du das Problem auf Deiner geplanten neuen Plattform auch bekommen wirst von daher wirst Du nicht wirklich tiefer in die Materie einsteigen müssen dennoch schadet es vielleicht nicht zu verstehen was da eigentlich passiert ist oder warum es passiert ist. Im groben gibt es im ACPI zwei Stellen mit Informationen zum Prozessor/den Prozessoren einmal eben die APIC.aml und die DSDT wobei die DSDT die realen Werte der installierten Prozessoren (real vorhandene Kerne/Threads sowie deren ID's) beinhaltet. Idealerweise sollten die Informationen in der APIC Tabelle mit dem übereinstimmen was in der DSDT zu finden ist allerdings ist das eben aber nicht immer der Fall sondern manchmal sind in der APIC Tabelle eben alle möglichen Werte/Konfigurationen für alle möglichen Ausbaustufen des Boards enthalten. MacOS bezieht seine Informationen zum Prozessor bzw. zu den Kernen und Threads aus der APIC und nimmt einfach stumpf an das was da drin steht auch vorhanden ist. Nehmen wir mal konkret Deine originale APIC Tabelle hier steht gleich zu Anfang folgendes:

Code

1. [02Ch 0044 1] Subtable Type : 00 [Processor Local APIC]
2. [02Dh 0045 1] Length : 08
3. [02Eh 0046 1] Processor ID : 00
4. [02Fh 0047 1] Local Apic ID : 02
5. [030h 0048 4] Flags (decoded below) : 00000001
6. Processor Enabled : 1
7. Runtime Online Capable : 0

Hier wird also ein Prozessor (Kern/Thread) mit der Prozessor ID 00 deklariert. Wenn man das jetzt mit dem ersten Eintrag aus der DSDT zum Prozessor vergleicht fällt auf...

Code

1. Processor (C001, 0x80, 0x00000410, 0x06)
- 2.
3. Der zweite Wert ist die ProzessorID also die 0x80

Unser erster Prozessor hat also gar nicht die ID 00 sondern die 80 und hier liegt der Hase buchstäblich im Pfeffer. Wie schon geschrieben geht macOS davon aus das alles was in der APIC.aml deklariert ist auch vorhanden ist und versucht nun also den Prozessor mit der ID 00 zu aktivieren was natürlich nicht klappen kann weil den gibt es in unserem System ja überhaupt nicht. Mit dem Wissen ausgestattet kommt nun also die Fleißarbeit die Aufgabe ist es nun die Prozessor ID für jeden Kern aus der DSDT zu friemeln und in die APIC.aml zu übertragen so wird zum Beispiel im ersten Eintrag aus der 00 eine 80 usw...

Wenn man weiß was und warum also eigentlich kein Hexenwerk 😄