

[WIP] Lenovo T530 ohne DSDT unter OpenCore

Beitrag von „5T33Z0“ vom 15. September 2020, 14:18

Hallo,

ich habe seit einen Paar Monaten ein Lenovo T530 und von jemandem hier aus dem Forum einen EFI Ordner für OpenCore bekommen, der auch echt super funktioniert für MacOS.

Sobald ich aber Windows damit starten möchte (2. Partition), dann bekomme ich einen ACPI BIOS Error.

Zum Spaß habe ich dann die DSDT mal deaktiviert, die restliche notwendigen .aml für IvyBridge Notebooks dazu gepackt und neugestartet, mit dem Ergebnis:

- Windows 10 Boot funktioniert einwandfrei
- macOS bootet auch, aber ist langsam, die Batterie-Anzeige, Helligkeitskontrolle und Audio funktionieren dann nicht mehr

Ich nehme an, die ganzen Patches dafür befinden sich in der DSDT.

Die frage ist, kann man irgendwas in die DSDT eintragen, damit windows sich Windows beim Booten nicht mehr aufhängt?

Oder soll ich die DSDT rauswerfen und die anderen sachen fixen? Aber wie macht man das - über deviceproperties? Mit IORegistry Explorer und Hackintool?

Beitrag von „apfelnico“ vom 15. September 2020, 18:20

5T33Z0

Wenn du die Patches in der DSDT findest, so können diese in eine if/else-Schleife gepackt werden mit Abfrage auf macOS bzw korrekt "Darwin".

Könnte dann so aussehen:

Code

1. Scope (\BLABLA)
2. {
3. If (_OSI ("Darwin"))
4. {
5. pack hier rein, was macOS glücklich macht
6. }
7. Else
8. {
9. mach sonst das hier
10. }
11. }

Alles anzeigen

Edit:

"SSDT-IMEI.aml" benötigst du nicht, darin wird abgefragt, ob du ein Device "IMEI" hast und wenn nicht, dann wird eines eingetragen. Exakt den Eintrag hast du natürlich schon in der DSDT. Also weg mit dieser SSDT.

"SSDT-EC.aml" ebenfalls. Es gibt kein "EC0" in der DSDT, ein "EC" ist vorhanden und sollte hier bei einem Laptop auch genutzt werden, die SSDT ist also ebenfalls zu entfernen.

"SSDT-XOSI.aml" - ich denke die kann auch raus, ebenso die beiden ACPI-Patches in der config.plist. Der erste Patch bezieht sich auf die "SSDT-XOSI.aml", der zweite Patch ist sehr unsinnig: hiermit wird versucht, das vorhandene Device "EC" in "EC0" umzubenennen. Weg damit ...

Probiere das mal. Dann die zusätzlichen _DSM-Methoden in der SSDT klammern.

Beitrag von „5T33Z0“ vom 15. September 2020, 18:25

Okay, cool. Danke. Ich probier mal, ob ich das hin bekomme.

[apfelnico](#) So, SSDT-IMEI, -EC, -XOSI und patches habe ich rausgeworfen. Aber das mit dem Ausklammern habe ich noch nicht verstanden. Ich sehe zwar Einträge in der DSDT in denen "_DSM" steht, aber ich verstehe ich nicht, wo die Einträge dann jeweils enden. Ist wie für mich.

Gibt's irgendwo n Guide dafür?

Beitrag von „apfelnico“ vom 15. September 2020, 21:07

5T33Z0

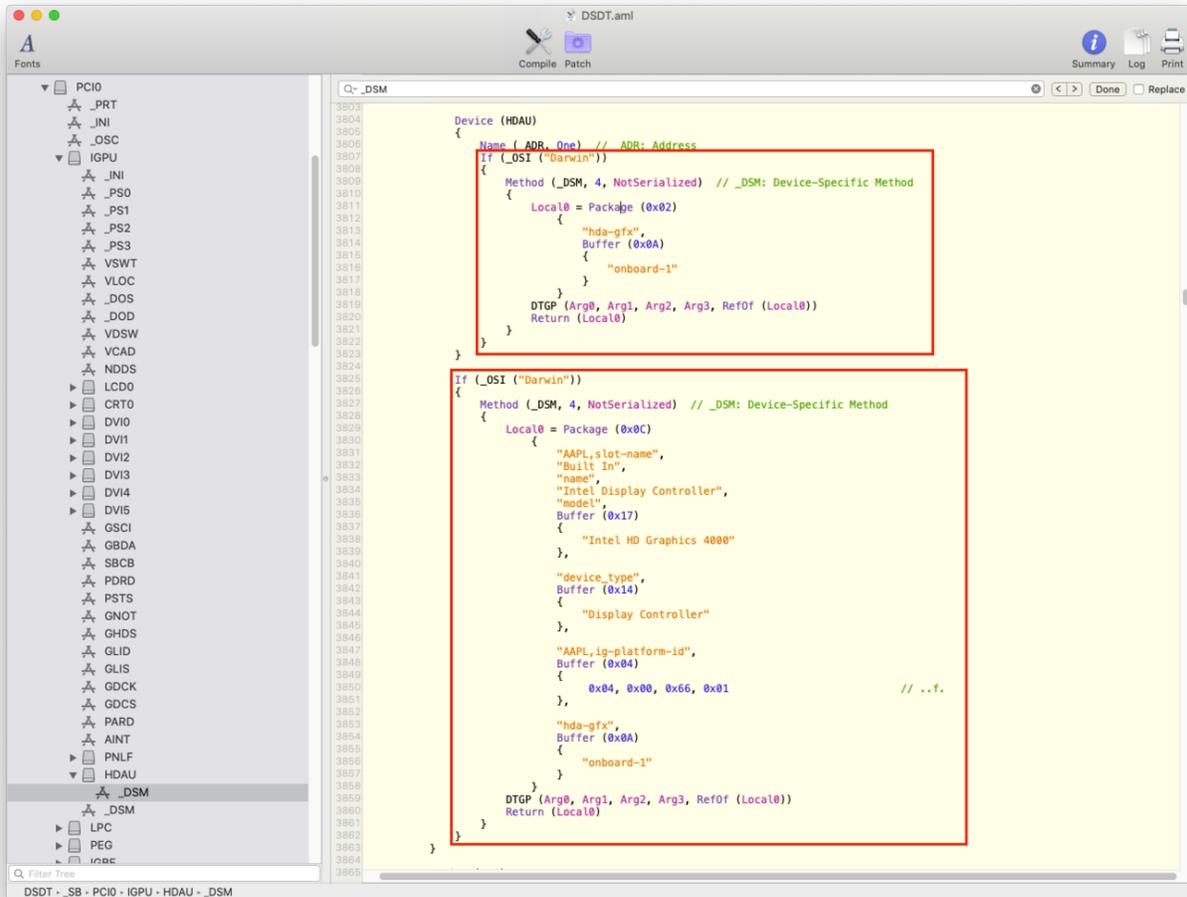
Die "_DSM-Methoden" nur für macOS zu beschreiben, ist recht einfach. ~~Achte darauf, das nicht jede dieser Methoden macOS betreffen.~~ Doch, alle sind "zusätzlich". Die original in der DSDT vorhandenen sind wohl gelöscht worden.

OK, verfare nach diesem Beispiel:

```
3799         0x0609,  
3800         0x0710  
3801     }  
3802 }  
3803  
3804 Device (HDAU)  
3805 {  
3806     Name (_ADR, One) // _ADR: Address  
3807     Method (_DSM, 4, NotSerialized) // _DSM: Device-Specific Method  
3808     {  
3809         Local0 = Package (0x02)  
3810         {  
3811             "hda-gfx",  
3812             Buffer (0x0A)  
3813             {  
3814                 "onboard-1"  
3815             }  
3816         }  
3817         DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))  
3818         Return (Local0)  
3819     }  
3820 }  
3821  
3822 Method (_DSM, 4, NotSerialized) // _DSM: Device-Specific Method  
3823 {  
3824     Local0 = Package (0x0C)  
3825     {  
3826         "AAPL,slot-name",  
3827         "Built In",  
3828         "name",  
3829         "Intel Display Controller",  
3830         "model",  
3831         Buffer (0x17)  
3832         {  
3833             "Intel HD Graphics 4000"  
3834         },  
3835         "device-type",  
3836         Buffer (0x14)  
3837         {  
3838             "Display Controller"  
3839         },  
3840         "AAPL,ig-platform-id",  
3841         Buffer (0x04)  
3842         {  
3843             0x04, 0x00, 0x66, 0x01 // ..f.  
3844         },  
3845         "hda-gfx",  
3846         Buffer (0x0A)  
3847         {  
3848             "onboard-1"  
3849         }  
3850     }  
3851     DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))  
3852     Return (Local0)  
3853 }  
3854  
3855 Device (LPC)  
3856 {  
3857  
3858  
3859  
3860  
3861
```

Wie du siehst, habe ich nach "_DSM" gesucht (Cmd-F). Am Beispiel der "IGPU" mal geschaut, hier siehst du zwei _DSM-Methoden. Die untere ist für das Device "IGPU", die darüber ist für das eingeschobene Device "HDAU" (High Definition Audio, also die Audioeinheit der internen Grafikkarte). Wo die Methoden jeweils enden, siehst du sehr genau durch die wunderbar gestaltete Einrückung des Quellcodes.

Schau dir jetzt an, was ich verändert habe:



Die Einrückung musst du nicht machen, beim sichern, schließen und erneutem öffnen ist der Quellcode perfekt angepasst. Du siehst, dass ich vor der eigentlichen Methode

Code

1. If (_OSI ("Darwin"))
2. {

eingefügt habe, die Klammer ist geöffnet. Entsprechend der Syntax muss ich nach der Methode noch eine geschlossene Klammer hinzufügen. Somit wird der Inhalt, in diesem Fall die Methode "_DSM" nur ausgeführt, wenn die Systemabfrage positiv hinsichtlich "Darwin" erfolgt. Windows wird das nun ab sofort ignorieren.

Das ist eine einfache Übung. Da ich aber nicht die originale DSDT kenne, denke ich, es wird

nicht nur bei diesen beschreibenden Methoden geblieben sein, die Windows auch sicher nicht unbedingt stören. Es sind etliche Devices umbenannt worden, und es fehlen die originalen _DSM-Methoden (die macOS wiederum nicht auswertet, die aber ihren Grund haben). Idealerweise wäre es also, wenn man mit den originalen _DSM-Methoden arbeitet (die macOS gar nicht stören) und zusätzlich die neuen Inhalte für macOS mit einer If-Schleife integriert.

Dann werden da noch weitere "Patches" enthalten sein etc pp.

Dennoch ist eine solche Übung nicht "für die Katz". Ich hoffe, ich konnte dir helfen, viel Spaß.

Edit:

Mal abgesehen von der Übung wäre es in der Tat hilfreich, zu klären, welche Patches denn tatsächlich überhaupt gebraucht werden. Aus verschiedenen Gründen rate ich von einer Bearbeitung der DSDT ab, man sollte schauen, was per SSDT geht und dort dann mittels Darwin-Abfrage arbeiten. ACPI-Patches in OpenCore würde ich auch nicht nutzen wenn irgend möglich, denn auch diese sind systemübergreifend dann vorhanden. Viele dieser Sachen erledigen heute schon Lilu/AppleALC/WhateverGreen – diese benennen Devices selbstständig für macOS genehm um. Und für Windows spielen die natürlich keine Rolle. DSM-Deklarationen (wenn nicht im Original enthalten) gern per SSDT, ansonsten per DeviceProperties über den Bootloader.

Beitrag von „5T33Z0“ vom 15. September 2020, 21:19

[apfelnico](#) Danke für die ausführlichen Erläuterungen.

Ich versuche dann glaube ich die Sachen die nicht funktionieren ohne DSDT einzubinden. Steht ja auch überall, dass man die nicht benutzen soll.

In meinem Clover "Origin"-Ordner hätte ich ansonsten noch die originale ungepatchete DSDT.

Beitrag von „Sascha_77“ vom 15. September 2020, 22:07

Ich hab vorhin mal auf auf Github so rumgeschaut. Da habe ich eine EFI gefunden für Clover. In ACPI waren alles schön in einzelne Dateien aufgesplittet. Booten konnte ich mit OC schon aber es geht noch eine ganze Reihe nicht. Helligkeit, Audio, Batterieanzeige etc. Mit allen ACPI Patches aus Clover aktiviert kommt beim Booten ne Kernelpanic. Auch so kommen ein paar ACPI Errors während des Bootens.

Da ist irgendwo noch der Hase im Pfeffer und bedarf noch einiges an Schliff. Aber vllt. als Ausgangsbasis ganz hilfreich. Habs mal angehängen.

Bei der ig-platform-id musst du ggf aus der 4 ne 3 machen falls dein Display < 1600x900 ist.

Die Cloverconfig hab ich auch mal beigelegt. Da hat derjenige ziemlich wild Haken gesetzt. Das meiste davon dürfte wohl nicht nötig sein.

Beitrag von „apfelnico“ vom 15. September 2020, 22:27

Ist vielleicht nicht so hilfreich, wenn es zum einem um OpenCore geht und zum anderen die Clover-Variante eh völlig kaputt ist ...

Beitrag von „Sascha_77“ vom 15. September 2020, 22:40

In der EFI.zip ist doch OpenCore drin. Von Clover ist lediglich die plist als Muster hier noch extra dran.

Es ging sich mir in erster Linie um die .aml Dateien. So wäre man zum. schonmal die DSDT als solches los. Das war eigtl. der Hintergedanke dabei. Diese Dateien sollten unter OC doch genauso funktionieren oder? Hab hier nochmal etwas mit den ACPI Patches rumprobiert. Habe alle aktiviert die **nicht** mit Battery zu tun haben. Da booted er durch. Ist es zumindest

schonmal etwas eingegrenzt.

Beitrag von „5T33Z0“ vom 15. September 2020, 22:49

Ich hab bei meiner Recherche eine Sammlung von Patches und Kexts für den T530 gefunden. Aber ich glaube, das sind die gleichen von Rehabman, die schon in maciASL enthalten sind, um die DSDT zu patchen, eben nur als Text. Keine Ahnung wie man da irgendwas für OpenCore draus bastelt.

Aber es sind u.a. 2 kexts für Battery und Backlight dabei, die probiere ich gleich mal mit nem boot stick aus. mit deaktivierter DSDT.

Beitrag von „apfelnico“ vom 15. September 2020, 23:13

[Zitat von Sascha 77](#)

In der EFI.zip ist doch OpenCore drin. Es ging sich mir in erster Linie um die .aml Dateien. So wäre man zum. schonmal die DSDT als solches los. Das war eigtl. der Hintergedanke dabei. Diese Dateien sollten unter OC doch genauso funktionieren oder?

Hatte das dann falsch verstanden, hatte nicht reingeschaut. Nur mal grob überflogen, die ACPI-Patches sind hier notwendig, damit die SSDTs greifen. Das eine bedingt das andere. Da die Patches dann aber auch bei Windows aktiv sind, sehe ich in diesem Fall diese Nummer dennoch als die schlechtere.

Dann lieber die vorhandene DSDT mit einer originalen DSDT vergleichen und wo es nötig ist, die gepatchte DSDT mit "OSI-Weichen" ausstatten. Dann ist alles sauber, nur eine DSDT, keine komplexen ACPI-Patches, fertig. Ich denke das ist der bessere Weg hier, die zwei DSDT zu vergleichen kann ja auch nicht so aufwendig sein.

Im übrigen wäre da Clover schon fast ideal, weil dort ganz einfach je Windows und macOS eine eigene DSDT geladen werden kann. Aber es ist auch mit OpenCore hinzubekommen.

Beitrag von „Sascha_77“ vom 16. September 2020, 06:12

Alternativ könnte man aber auch mit F12 das Boot-Menü aufrufen und dann die Win Partition auswählen. Dann wäre es komplett an OC vorbei und es gäbe die DSDT Problematik nicht. Ginge zur Not ja auch.

Beitrag von „5T33Z0“ vom 16. September 2020, 07:47

[Sascha_77](#) ja, das stimmt. So starte Windows aktuell auch. Danke für die Dateien. Ich werde das mal mit meinen vergleichen.

Beitrag von „5T33Z0“ vom 23. Oktober 2020, 17:47

So, ich habe mich der Herausforderung jetzt gestellt und alle Einträge in der DSDT wo "_DSM" drin steht mit If (_OSI ("Darwin")) umklammert. Windows startet jetzt. yay! (mit DSDT_half_gefixt.aml)

Aber: Catalina startet nicht mehr. Nur Logo, kein Ladebalken. Habe ein Foto vom Kernel Panic gemacht.

Könnte sich das jemand ansehen? Ich komme nicht weiter. Vielleicht eine Klammer zuviel oder an der falschen Stelle gesetzt irgendwo? [grt](#) vielleicht?

Bootlog im Anhang. Danke

```
0xffffffff820f212726 : 0xffffffff82127726
Kernel Extensions in backtrace:
  com.apple.driver.AppleACPIPlatform(6.1)[0EF10B66-B44B-32BB-9CE3-5434F4D40FE1]@0xffffffff7fb2126000-0xffffffff7fb21c8fff
    dependency: com.apple.iokit.IORCPIDFamily(1.4)[2956198D-24F2-3790-A982-1EAB9434B906]@0xffffffff7faf67c000
    dependency: com.apple.iokit.IOPCIFamily(2.9)[44472E6F-8DA0-3B46-ADEF-AFF76EC6C60B]@0xffffffff7faf531000
    dependency: com.apple.driver.AppleSMC(3.1.9)[D2F0B610-83F8-3B84-B8BD-D9D8CC5A6971]@0xffffffff7faf68e000

BSD process name corresponding to current thread: Unknown
Boot args: alcid=29 brcafx-country=#a -v

Mac OS version:
Not yet set

Kernel version:
Darwin Kernel Version 19.0.0: Mon Aug 31 22:12:52 PDT 2020; root:xnu-6153.141.2~1/RELEASE_ARM64
Kernel UUID: 85D51A3D-3A87-3FFB-98C3-9CF3B27A3EDD
Kernel slide: 0x00000002e8000000
Kernel text base: 0xffffffff802ea00000
  _HIIB text base: 0xffffffff802e900000
System model name: MacBookPro10,1 (Mac-C3EC7CD22292981F)
System shutdown begun: NO
Panic diags file unavailable, panic occurred prior to initialization

System uptime in nanoseconds: 2089579830

** In Memory Panic Stackshot Succeeded ** Bytes Traced 11280 **
Attempting system restart...MACH Reboot
```

Beitrag von „5T33Z0“ vom 5. Dezember 2020, 22:33

Habe mich jetzt dazu entschlossen, ganz auf die DSDT zu verzichten. Und da seitdem auch irgendwie immer dran rumgebastelt.

Grafik: läuft samt External Monitor Support

Battery Status: läuft.

Fn Keys funktionieren

Was ich bisher immer noch nicht hinbekommen habe, ist Audio aus der Kiste zu bekommen. Es braucht anscheinend mehr, als ein DeviceProperty mit passender Layout ID anzulegen.

Also habe ich mit die gepatchte DSDT angesehen und darin ist ein HDEF Audio Device definiert:

Code

```
1. Device (HDEF)
2. {
3. Name (_ADR, 0x001B0000) // _ADR: Address
4. OperationRegion (HDAR, PCI_Config, 0x4C, 0x10)
5. Field (HDAR, WordAcc, NoLock, Preserve)
6. {
7. DCKA, 1,
8. Offset (0x01),
9. DCKM, 1,
10. , 6,
11. DCKS, 1,
12. Offset (0x08),
13. , 15,
14. PMES, 1
15. }
16.
17. Method (_DSM, 4, NotSerialized) // _DSM: Device-Specific Method
18. {
19. Local0 = Package (0x10)
20. {
21. "AAPL,slot-name",
22. "Built In",
23. "name",
24. "Realtek Audio Controller",
25. "model",
26. Buffer (0x11)
27. {
28. "Realtek ALC269VC"
29. },
30.
31. "device_type",
32. Buffer (0x11)
33. {
34. "Audio Controller"
35. },
36.
37. "layout-id",
38. Buffer (0x04)
39. {
40. 0x1D, 0x00, 0x00, 0x00 // ....
```

```
41. },
42.
43. "PinConfigurations",
44. Buffer (Zero){},
45. "MaximumBootBeepVolume",
46. Buffer (One)
47. {
48. 0x40 // @
49. },
50.
51. "hda-gfx",
52. Buffer (0x0A)
53. {
54. "onboard-1"
55. }
56. }
57. DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))
58. Return (Local0)
59. }
60.
61. Name (_PRW, Package (0x02) // _PRW: Power Resources for Wake
62. {
63. 0x0D,
64. Zero
65. })
66. }
```

Alles anzeigen

Die Frage ist jetzt: kann ich daraus eine "SSDT-HDEF" oder so basteln, damit es dann funktioniert oder was muss ich tun? Oder muss ich noch woander gucken?

Beitrag von „5T33Z0“ vom 11. Dezember 2020, 21:33

So, mittlerweile läuft alles bis auf Sleep/Wake bzw. Clamshell Modus, wenn man das Notebook zuklappt. Ist echt nochmal snappier als mit einer gepatchten DSDT. Aber 20 aml Dateien und knapp 30 patches.

Hab einen chinesischen Github gefunden nur mit ACPI Hotpatches für OpenCore. Sehr nützlich:

<https://ooh3dpsdytm34sfhws63yj...goog/daliansky/OC-little>

Beitrag von „Sascha_77“ vom 11. Dezember 2020, 21:56

Vllt. kann [al6042](#) was zu dem HDEF Thema sagen. Die DSDT fürs T530 die hier im Forum rumschwirrt stammt ursprünglich von ihm. Hätte schon was wenn man auf VoodooHDA verzichten könnte.

Beitrag von „5T33Z0“ vom 11. Dezember 2020, 22:04

Ja, wäre nice. Bei HDEF und bei dem Lid Sleep/Wake Problem bin ich bisher nicht weiter gekommen. Dieses ACPI Gefrickel ist auch echt Neuland für mich.

Was ist eigentlich diese TGT Bridge? Habe gesehen, dass in der Clover Config, die ich habe mehrere Patches drin sind, die das verwenden, die man aber nicht ohne weiteres in OpenCore übertragen kann, da es dieses Feature nicht hat.

Beitrag von „al6042“ vom 11. Dezember 2020, 22:06

Mit den Angaben oben sollte sich auch eine SSDT bauen lassen.

Hier mal ein erster Versuch dafür... 😊

Beitrag von „5T33Z0“ vom 11. Dezember 2020, 22:13

Super, danke dir, al6042. Probiere ich gleich mal aus.

Hab hier noch was zum Thema Sleep und Lid gefunden:
<https://ooh3dpsdytm34sfhws63yj...6%AD%A3%E6%96%B9%E6%B3%95>

Dieser Github ist echt ne Goldrube!

Lieber [al6042](#), vielen Danke für HDEF.aml. Leider funktioniert sie noch nicht. Ist aber nicht so wichtig, da VoodooHDA ja funzt.

Hier ist die aktuellste Version der EFI [Lenovo T530 OC 0.6.5 DSDT-los \(v3\).zip](#)

Es funktionieren bislang:

- Alles, was auf Kexts basiert
- Audio (via VoodooHDA)
- Batterieanzeige
- Sleep Key (Fn+F4)
- Brightness Keys (Fn+F8 u. Fn+F9)
- Backlight Controller (PNLF)
- Speicherung des Helligkeitswerts nach Sleep/Wake/Boot (ALS0)
- Fix der LED des Powerbuttons nach Sleep/Wake (ohne pulsiert sie weiter nach dem Wake)
- Externer Monitor

Wo ich nicht weiterkomme:

- Audio via AppleALC. Device Property für Soundkarte ist vorhanden, aber deaktiviert (#PciRoot(0x0)/Pci(0x1B,0x0))
- Umschalten auf externen Monitor bzw. Sleep, wenn Display zugeklappt wird

Beitrag von „5T33Z0“ vom 13. Dezember 2020, 00:24

Geil, nachdem ich noch "Change OSIF to XSIF" rename hinzugefügt habe, bootet er jetzt auch Windows von derselben Disk direkt aus OpenCore.

Bleibt jetzt nur noch Sleep/Clamshell Modus, wenn die Kiste zugeklappt wird zu lösen [al6042](#)

Beitrag von „Sascha_77“ vom 13. Dezember 2020, 01:33

Saubere Arbeit. 👍😎 Kann die Funktion bestätigen. Bei Fn+F4 ist es noch so, dass nach dem Wake die Lampe weiter pulsiert. Allerdings nutze ich sowie eigentlich nur die Sleepfunktion aus dem Menu heraus und da leuchtet sie nach dem Wake ganz normal.

Was mir direkt positiv auffällt ist, dass nach dem Wakeup der Rechner sofort ansprechbar ist. Vorher mit der DSDT Variante hat er sich schonmal ein paar Gedenksekunden gegönnt.

Beitrag von „al6042“ vom 13. Dezember 2020, 09:38

Ich bin mir wegen dem PciRoot-Pfad für deine Soundkarte überhaupt korrekt ist...

Gestern habe ich einem anderen User und seinem Z390-System geholfen... da hatte er aus einem OC-EFI-Paket auch eine config.plist mit dem PciRoot-Pfad 1B und das war komplett falsch.

Leider habe ich mein T530, sowie die damaligen Files nicht mehr, denn dort konnte ich den ALC269 per AppleALC nutzen.

Was wirft denn das HackinTool als "Device Path" für Audio unter PCIe aus?

Beitrag von „Sascha_77“ vom 13. Dezember 2020, 09:40

Pfad ist der hier:

Code

1. PciRoot(0x0)/Pci(0x1b,0x0)
-

Beitrag von „al6042“ vom 13. Dezember 2020, 09:55

Stimmt...

habe eben mal eine alte EFI meines Boards aus dem Forum geladen und geprüft... 😊

Ebenso habe ich dort aber gesehen, dass als layout-id 0x1C (28) zum Einsatz kam.

Check das doch mal in der SSDT-HDEF.aml aus.

Beitrag von „Sascha_77“ vom 13. Dezember 2020, 10:01

Soeben ausprobiert. Auch kein Ton.

Beitrag von „al6042“ vom 13. Dezember 2020, 10:27

Na ja.. der Eintrag ist 5 Jahre alt... wahrscheinlich wurde im AppleALC die Zuordnung neu gemischt... 🤖

Aktuell findet man auf Github für Lenovos folgende Zuordnungen:

layout-id Comment

18 Hypereitan - ALC269VC for Thinkpad X230 i7

- 23 Custom ALC269VD ThinkPad T430
- 28 Custom ALC269VC Lenovo Z580 by John
- 29 Custom ALC269VC Lenovo V580 by ar4er
- 40 vusun123 - RealtekALC269VC for Lenovo W530
- 55 ALC269VC for Thinkpad X230 with dock4338
- 188 ALC269VC for Lenovo Y500 by BaoStorm

Ich habe Euch zum Test auch nochmal eine neue Variante der SSDT-HDEF dazu gelegt.

Beitrag von „Sascha_77“ vom 13. Dezember 2020, 10:32

Damit mag er leider auch nicht.

Beitrag von „5T33Z0“ vom 13. Dezember 2020, 11:06

[Sascha_77](#) Also seit v3 funzt bei mir die LED wieder, wie sie soll. Die version, die ich dir via dm geschickt hatte war v2 quasi.

[al6042](#) Kann es sien, dass man noch sowas braucht, damit die Soundkarte richtig eingebunden werden kann:

<https://ooh3dpsdytm34sfhws63yj...12-060D%E8%A1%A5%E4%B8%81>

Layout ID 28 und 29 funktionieren beide übrigens via alcid= boot-arg, wenn man die gepatchte DSDT nutzt. Bislang hatte ich immer 29. Aber über Device Property allein geht da nichts. In der gepatchtes DSDT kommt unter dem HDEF Eintrag noch ein Einschub, aber ich weiß nicht, was er bedeutet:

DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))

Return (Local0)

Beitrag von „al6042“ vom 13. Dezember 2020, 11:12

Nope...

dabei geht es meines Erachtens nur um das Verhalten für den Sleep-Kram.

Beitrag von „Sascha_77“ vom 13. Dezember 2020, 11:14

5T33Z0

Nutze deine v3. Gerade nochmal probiert. Mit Fn+F4 pulsiert die LED nach dem Wake. Seltsam das es bei dir klappt. 🤔 Wenn ich danach ohne Reboot ihn nochmal übers Menu schlafen schicke ist mit der LED wieder alles ok.

Beitrag von „5T33Z0“ vom 13. Dezember 2020, 11:16

Stimmt, jetzt pulsiert es wieder. Nerv. Das ging gestern Abend noch. Vielleicht geht's nur einmal. strange.

Hätte noch n codec dump, falls das hilft: [CodecDump_ALC269.zip](#)

Beitrag von „5T33Z0“ vom 16. Dezember 2020, 14:04

[al6042](#) Ich komme mit diesem Lid patch einfach nicht weiter. Funktioniert einfach nicht. Der rename Patch ist drin, aber Ich weiß nicht, welche Pfade/externen Devices und Methods da eingetragen werden müsse.

Das ist der Patch:

Spoiler anzeigen

Das ist der LID Device in der ungepatchten DSDT:

Spoiler anzeigen

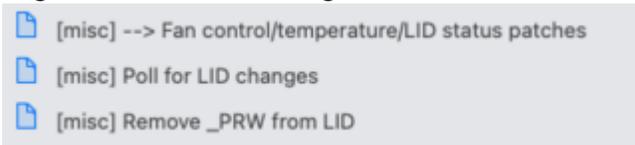
Und hier ist der LID device aus Deiner gepatchten DSDT:

Spoiler anzeigen

Danke

Beitrag von „al6042“ vom 16. Dezember 2020, 16:58

Wenn ich mich nicht allzu arg irre, habe ich damals entweder einen der beiden folgende oder sogar beide Patches eingesetzt:



Beitrag von „5T33Z0“ vom 23. Dezember 2020, 09:06

Vielen Dank. Mal gucken, ob ich damit weiterkomme, um mir ne passende SSDT zu basteln.

Was mich wundert ist, dass die .aml Dateien für Lid Sleep/Wake, die ich aus Clover habe, unter OpenCore nicht funtionieren wollen, trotz der ganzen renames, die ich eingebaut habe.

Beitrag von „5T33Z0“ vom 2. Januar 2021, 18:18

[Sascha 77 al6042](#) Ich habe endlich AppleALC zum Laufen bekommen!

Bin zufällig darüber gestolpert, warum AppleALC nicht funktioniert hat:

<https://github.com/corpnewt/SS.../7#issuecomment-601604756>

Die von SSDTTime generierte SSDT-HPET mit den IRQ Fixes verwendet eine Form, die das Notebook anscheinend nicht "versteht". Nachdem ich es in der DSDT von

Code

1. IRQNoFlags () {0,8,11}

nach

Code

1. IRQNoFlags () {0}
2. IRQNoFlags () {8}
3. IRQNoFlags () {11}

geändert habe, die KEXT und die Device Property wieder aktiviert habe, funzt jetzt Audio. Geil, endlich kein VoodooHDA mehr nötig.

Jetzt fehlt nur noch Lid Sleep/clamshell Mode.

Habs mal bei Github hochgeladen:

<https://github.com/5T33Z0/Lenovo-T530-Hackintosh-OpenCore>

Beitrag von „5T33Z0“ vom 4. März 2021, 15:14

Seit update auf 067 Release Version, geht's jetzt heir nicht mehr weiter nach dem Booten:

```
Security policy loaded: Apple Mobile File Integrity (AMFI)
calling mpo_policy_init for Sandbox
Security policy loaded: Seatbelt sandbox policy (Sandbox)
calling mpo_policy_init for Quarantine
Security policy loaded: Quarantine policy (Quarantine)
calling mpo_policy_init for TMSafetyNet
Security policy loaded: Safety net for Time Machine (TMSafetyNet)
Darwin Image4 Validator Version 2.2.0: Tue Jan 12 22:30:52 PST 2021; root:AppleImage4-61.60.4-3480/Apple
IDAPIC: Version 0x20 Vectors 64:87
ACPI: Executed 2 blocks of module-level executable AML code
ACPI: sleep states S3 S4 S5
HID: Legacy shim 2
HID: Legacy shim 2
pci (buiconsole relocated to 0xf80000000
[ PCI configuration end, bridges 3, devices 14 ]
000001_541557 AppleUSBLegacyRoot@: AppleUSBLegacyRoot::init: enabling legacy matching
VoodooSDHCI ::: an SDHCI driver for Ricoh, TI, and JMicron SD Host Controllers ::: rev 20091008
VoodooSDHCI: initializing SD host controller
ApplePS2Controller: Notification consumer published: ApplePS2Controller
VoodooPS2SynapticsTouchPad: Version 1.9.2 starting on OS X Darwin 19.6.
ApplePS2Controller: Notification consumer published: ApplePS2Keyboard
VoodooPS2House: Version 1.9.2 starting on OS X Darwin 19.6.
VoodooPS2Trackpad starting: Synaptics TouchPad reports type 0x47, version 7.2
ApplePS2Controller: Notification consumer published: ApplePS2SynapticsTouchPad
```

Und da lese ich zu den Update Notes im März:

Zitat

VoodooPS2 received NumLock key support thx to [@chilledHamza](#) and VoodooInput got overall code simplification thx to [@zhen-zen](#).

<https://dortania.github.io/hac...01/acidanthera-march.html>

Scheint kein Zufall zu sein, dass danach was nicht damit nicht mehr funzt.

Beitrag von „Sascha_77“ vom 4. März 2021, 16:35

Habe hier noch VoodooPS2 1.9.2 im Einsatz. Bei Neueren gabs irgendwie immer Probleme.

Beitrag von „5T33Z0“ vom 4. März 2021, 16:50

Habe auch den alten im Einsatz. Habe es aber auch mit dem neuen probiert. Seit dem OpenCore Commit mit HPET-Support geht's nicht mehr - obwohl es deaktiviert ist. Bei allen Vuilids davor ist alles gut.

Beitrag von „5T33Z0“ vom 11. März 2021, 09:24

Habe jetzt die neue VoodooPs2 Extension eingebunden mit einer zusätzlichen SSDT für's Trackpad. Funktioniert für die DSDT-lose Config aber nicht für die mit gepatchter DSDT. Aber ist mir jetzt auch egal. Ich nutze die DSDT eh nicht mehr. Im Zuge dessen auch Windows 10 Bootprobleme mit gepatchter DSDT behoben durch Einbinden aktueller SSDT-XOSI.aml.

Fehlt nur noch LED-blinke fix und Umschlaten beim Schließen des Deckels.