

Erledigt

Nach Installation DW1560 Wlan Bluetooth Karte startet PC sofort wieder von alleine nach dem Herunterfahren

Beitrag von „witjojo“ vom 17. Februar 2018, 03:50

Hallo,

ich habe ein Lenovo ThinkCentre M93 Tiny wo ich Sierra drauf installiert habe. Es läuft alles soweit.

Mit Hilfe von Harper Lewis habe ich es geschafft eine WLAN Karte DW1560 von Dell zu installieren ([hier](#)).

Diese läuft nun auch perfekt. Danke noch mal @ Harper Lewis.

ABER: Nachdem ich alles fertig habe startet jetzt mein Rechner sofort wieder von alleine nach dem Herunterfahren.

Das war zuvor nicht der Fall.

Laut Aussage von Harper Lewis liegt das am USB-Port, an dem die Karte hängt und dieser USB-Port müsste als intern definiert werden.

Ich nutze USBInjectAll.kext (Vers. 0.6.2). FixShutdown in Clover ist bereits aktiviert. Und ich nutze keine DSDT.

Kann mir jemand dabei helfen?

Gruß
witjojo

Beitrag von „Harper Lewis“ vom 17. Februar 2018, 10:38

Du müsstest den internen USB-Port, an dem der Bluetooth-Teil der Karte hängt, entsprechend konfigurieren. [@kuckkuck](#) hat [hier](#) mögliche Lösungswege zusammengefasst.

Beitrag von „kuckkuck“ vom 17. Februar 2018, 11:15

Irgendetwas bringt den PC dazu neu hochzufahren, nachdem er runtergefahren wurde. An diesem Punkt sind jedoch Treiber von Apple nicht mehr aktiv (im Gegensatz zu ACPI Vorgängen oder UEFI Services). Einen bestimmten port als intern zu definieren, bringt häufig was, wenn der Rechner aus dem Sleep direkt wieder aufwacht, da bestimmte Treiber Informationen falsch verarbeiten und Signale der Karte als Aufwachsignal eines externen Geräts interpretieren. Ich glaube aber nicht, dass uns das als intern definieren hier weiterhilft, da diese Treiber am besagten Punkt garnicht mehr aktiv sind 😊

Manchmal hilft es bereits bestimmte [BIOS Einstellungen](#) zu verändern. Hier zB als Beispiel:

[Zitat von Peter Pan](#)

Ich habe das Shutdown Problem lösen können. Das war auch nervig
Ich bin wirklich alle Punkte im Bios durchgegangen und habe den Punkt -Ein/Aus-Wiedergabe gefunden-
dieser war aktiviert und soll dazu da sein um MP3-Player anzuschließen wenn der PC aus ist um Musik zu hören.

Zudem gibt es verschiedene Möglichkeiten einen shutdown in den ACPI Tabellen zu verhindern. Der FixShutdown aus Clover ist hierbei nur ein 😊 Wir können also ebenfalls noch ein paar andere mithilfe deiner DSDT probieren.

Ist das Shutdown Problem nur vorhanden wenn die Karte eingebaut is?

Beitrag von „Harper Lewis“ vom 17. Februar 2018, 11:41

[@kuckkuck](#): Bei meinem Lenovo E460 hat genau das geholfen, den doppelten Neustart zu verhindern. Zusätzlich braucht es noch einen weiteren Fix, damit dies nicht mit angeschlossenen USB-Geräten passiert.

Beitrag von „kuckkuck“ vom 17. Februar 2018, 16:38

Sehr interessant, hätte ich nicht erwartet 😊 Na dann lass mal sehen ob [@witjojo](#) nachdem er sich eine UIA SSDT erstellt hat ebenfalls keine Neustart Probleme mehr hat 👍

[Zitat von Harper Lewis](#)

Zusätzlich braucht es noch einen weiteren Fix, damit dies nicht mit angeschlossenen USB-Geräten passiert.

Meinst du Rehabmans radikalen "USB3 _PRW 0x0D (instant wake)" Patch?

Beitrag von „Harper Lewis“ vom 17. Februar 2018, 18:03

Der Patch ist auch von Rehabman, aber ein anderer: *[sys] Shutdown restart (XHC.PMEE)* Fix. Ich habe nicht die DSDT direkt gepatcht, sondern habe die Methode _PTS in Clover umbenannt und nutze den Fix in einer SSDT. Auf meinem Skylake-Desktop funktioniert das ebenfalls prima.

Beitrag von „kuckkuck“ vom 17. Februar 2018, 18:16

Ok, dann übergeb ich dir mal das Kommando und lass dich mal mit witjojo versuchen 👍

Beitrag von „Harper Lewis“ vom 17. Februar 2018, 18:24

Würde ich gerne tun, nur bei Erstellen einer SSDT-UIAC werde ich leider nicht helfen können (wobei es dafür hier ja auch Anleitungen gibt). Diese Lösung habe ich selbst noch nie genutzt und stattdessen USB-Kexts nach [@Brumbaer](#) im Einsatz. Der o.g. Fix hilft auch nur, wenn XHC.PMEE in der DSDT vorhanden ist. Ich sehe auch gerade, dass es den erst seit vier Tagen im Repository von Rehabman gibt. Natürlich versuche ich trotzdem gerne, hier weiterzuhelfen.

Beitrag von „witjojo“ vom 18. Februar 2018, 04:27

Hi,

ich sehe schon das es jetzt kompliziert wird. Ich bin nicht so der Profi aber werde zuerst einmal im BIOS nachschauen ob da noch irgendwelche Einstellungen sind, wie Kuckkuck sagt. Dann lese ich mir mal die Anleitungen von Brumbaer durch.

Und zu dem 2. Fix für die angeschlossenen USB Geräte kann ich sagen, das ich ihn nicht benötige. Es reicht mir aus wenn der PC nur nicht neu startet ohne angeschlossene USB Geräte.

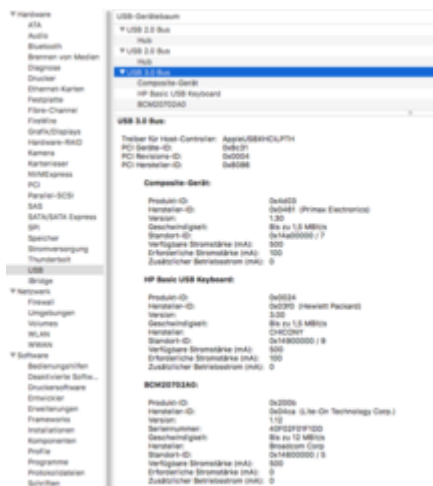
EDIT:

Ich habe jetzt die [Bios Einstellungen](#) überprüft. WOL war auf Automatic.

Ich habe es auf Disabled gestellt und der PC macht jetzt keinen eigenständigen Reboot mehr nach dem Herunterfahren.

Jetzt kann ich aber kein WOL mehr nutzen, also habe ich WOL wieder eingeschaltet. Es liegt ja eindeutig an der WLAN Karte.

Ich habe jetzt die WLAN Karte mit IORegistryExplorer gefunden.



Was ich bereits sagen kann:
 Ports HS01 und HS02 sind die Frontanschlüsse.
 Ports HS09 und HS10 sind die Rückanschlüsse.
 Port HS06 ist der interne wo die WLAN Karte dran ist.

Wie kann ich jetzt genau die Karte als intern definieren. Kann mir jemand helfen?
 Ich verstehe bei dem Text von Brumbaer echt gesagt nicht viel.
 Da wird immer nur erklärt wie man Ports deaktiviert und nicht wie man einen Port als intern definiert.

Beitrag von „witjojo“ vom 22. Februar 2018, 11:59

Hat hier keiner eine Idee?

Beitrag von „kuckkuck“ vom 22. Februar 2018, 13:32

Ich geb dir jetzt einfach mal einen Rausch an links...

Diese Anleitung befolgen und durcharbeiten, damit USBInjectAll richtig läuft und dir dabei eine Liste anfertigen, in der genau steht welchem physischen Port welcher virtuelle Port entspricht.

Beispiel:

USB 3.0 Port - Case vorne links - HS04 und SSP2

Interner USB Port - Mit Wifi Karte verbunden - HS06

Und so weiter: [Anleitung: USB 3.0 El Capitan / \(High-\) Sierra für jedermann \(Zukunftssicher für Updates und komplett gepatcht\)](#)

Danach kannst du eine SSDT erstellen, die dem USBInjectAll Informationen gibt, was er machen soll, darunter auch die Info, dass HS06 intern ist, also einen USBConnector von 255 oder 0xff bekommt.

Hier alle Guides zum erstellen:

- [USBInjectAll - Elmaccis Guide mit Beispiel- und Erläuterungs-SSDTs](#)

- [Brumbaers USB-SSDT Guide](#)

Ab "Sei doch nicht so negativ"

- [Konkrete Erläuterung zur Erstellung](#)

- [Rehabmans Beispiel-SSDT](#)

alle möglichen USB Definitionen

Beitrag von „witjojo“ vom 4. März 2018, 05:46

Also ich habe mir jetzt mal die SSDT-USB-Z170XG5.aml von elmacci als Vorlage genommen.

• [USBInjectAll - Elmaccis Guide mit Beispiel- und Erläuterungs-SSDTs](#)

Dann habe ich dort meine 5 vorhandenen Ports definiert.

Ports HS01 und HS02 sind die Frontanschlüsse.

Ports HS09 und HS10 sind die Rückanschlüsse.

Port HS06 ist der interne wo die WLAN Karte dran ist.

Aber ich habe noch Probleme mit den ersten Daten:

```
DefinitionBlock ("iASLSPBIBQ.aml", "SSDT", 1, "sample", "USBFix", 0x00003000)
{
Device (UIAC)
{
Name (_HID, "UIA00000")
Name (RMCF, Package (0x02)
{
"8086_a12f",
Package (0x04)
{
"port-count",
Buffer (0x04)
{
0x19, 0x00, 0x00, 0x00 <----- hier
},

"ports",
Package (0x20) <----- hier
{
```

ab hier kommen dann die Ports die ich definiert habe (siehe Anhang SSDT-USB.aml).

[SSDT-USB.aml](#)

Ich habe auch nur 5 Ports. Ich habe mit IORegistryExplorer und einem USB alle Ports durchgetestet.

Warum haben manche 15 oder mehr Ports? Mein Macbook Pro hat z.B. nur noch 2 USB Ports. Und muss ich die anderen ungenutzten Ports in Clover noch deaktivieren?

Beitrag von „Harper Lewis“ vom 4. März 2018, 09:26

Hast du die Ports auch mal mit einem USB3-Stick getestet?

Beitrag von „kuckkuck“ vom 4. März 2018, 13:00

[Zitat von witjojo](#)

Dann habe ich dort meine 5 vorhandenen Ports definiert.

[Zitat von witjojo](#)

Ich habe auch nur 5 Ports.

Du musst in der SSDT jeden einzelnen Port definieren, nicht einfach nur 5 von 14, oder so.

[Zitat von witjojo](#)

```
"port-count",  
Buffer (0x04)  
{  
0x19, 0x00, 0x00, 0x00 <----- hier
```

[Problem mit Sleep/Wake](#)

Bei deiner bisherigen SSDT war der "höchste Port" HS10 mit 0x0A, 0x00, 0x00, 0x00, in diesem Fall müsstest du also auch 0x0A, 0x00, 0x00, 0x00 nach port-count schreiben, also anstatt 0x19, 0x00, 0x00, 0x00.

Zitat von witjojo

```
"ports",  
Package (0x20) <----- hier
```

Einfach den Inhalt der Klammer löschen, also zu `Package ()` machen, der Wert wird in diesem Fall dann von selbst gesetzt.

Zitat von witjojo

Ich habe mit IORegistryExplorer und einem USB alle Ports durchgetestet.

Du solltest 1. sowohl einen USB 2.0 als auch 3.0 Stick verwenden und 2. im IOReg nachschauen, ob die Connector-Types korrekt aus der SSDT übernommen wurden.

Zitat von witjojo

Warum haben manche 15 oder mehr Ports?

Man kann das Port Limit auf über 15 anheben, das brauchst du aber nicht und solltest es deswegen am besten auch lassen 😊 Stattdessen musst du einfach nur alle benutzten Ports in die SSDT schreiben.

Beitrag von „witjojo“ vom 18. März 2018, 21:33

Vielen Dank an Harper Lewis und Kuckkuck.

Ich habe zuerst alle Port nochmal mit einem USB3.0 Stick durchprobiert. Und jetzt hatte ich auch 4 weitere SSPx Ports gefunden.

Auch diese habe ich nun in der SSDT definiert. Somit insgesamt 9 Ports. Die anderen habe ich in Clover mit uia_exclude ausgeschlossen.

Den port-count habe ich wie Kuckkuck schrieb definiert. Bei Ports habe ich den Inhalt in der Klammer weggelassen (wird von selbst gesetzt).

Zuerst habe ich das alles unter Sierra gemacht. Das funktionierte nicht. Trotz der richtigen SSDT hatte ich immer mehr Ports als eingetragen.

Und beim Herunterfahren gab es auch den bekannten erzwungenen Neustart beim Herunterfahren.

Dann habe ich auf einer Test-SSD auf "High Sierra" upgedatet. Alles gleich gelassen, und siehe da, die richtigen Ports sind nur noch da und alle funktionieren auch. Und der automatische Neustart beim Herunterfahren ist auch weg.

Warum es unter Sierra nicht funktioniert kann ich nicht sagen aber unter High Sierra läuft es bestens.

Ich kann jetzt mit High Sierra sehr gut leben, habe jedoch mit Absicht noch kein APFS Dateisystem im Einsatz.

Anbei noch meine SSDT-USB, die Screenshots der Clover einstellungen (exclude) und die Ports welche nun richtig definiert sind.

Vielleicht hilft das hier jemand anderem weiter.

Gruß
witjojo

Beitrag von „kuckkuck“ vom 18. März 2018, 22:59

Sehr schön, und danke für die ausführliche Rückmeldung!

Wenn du willst werfe ich morgen nochmal einen Blick auf deine erstellte USB-SSDT...

Beitrag von „Harper Lewis“ vom 19. März 2018, 09:27

Prima! Da du mit der SSDT-USB ja bereits die aktiven Ports als Positivliste definiert hast, benötigst du das Bootarg uia_exclude (Negativliste) gar nicht mehr. Der Bluetooth-Chip der DW1560 meldet sich unter HS06, wenn ich das richtig sehe.

Beitrag von „witjojo“ vom 19. März 2018, 18:59

[@kuckkuck](#)

Meist du auf die USB-SSDT draufscheuen im Bezug das es unter Sierra nicht läuft?
Sonst ist ja alles ok unter High Sierra.

[@Harper](#) Lewis

Ich meine das Bootarg uia_exclude (Negativliste) muss definiert werden.

Irgendwie werden die nicht benötigten Ports sonst automatisch erkannt und dann fehlern die SSPx Ports, weil ja nur 15 möglich sind.

Ich denke man kann die nicht benötigten Ports auch in der SSDT als nicht benötigt definieren, dann benötigt man natürlich nicht mehr das Bootarg uia_exclude.

Gruß
witjojo

Beitrag von „Harper Lewis“ vom 19. März 2018, 19:08

Ich nutze USBInjectAll nicht mehr, aber meines Wissens sollten in der SSDT nur die aktiven

Ports definiert werden. Nachzulesen z.B. auch [hier](#).

Beitrag von „kuckkuck“ vom 20. März 2018, 17:22

[Zitat von witjojo](#)

Meist du auf die USB-SSDT draufscheuen im Bezug das es unter Sierra nicht läuft?

Nein, einfach generell zum überprüfen. In so einer SSDT mogelt sich mal schnell ein Fehler unter...

[Zitat von witjojo](#)

Ich meine das Bootarg uia_exclude (Negativliste) muss definiert werden.

Und ich glaube in deiner SSDT ist auch ein Fehler gelandet und deswegen funktioniert das ganze auch nicht so wie es soll... Ich bräuchte aber bitte mal einen IOReg Dump.

Beitrag von „witjojo“ vom 23. März 2018, 07:43

Anbei der Dump und noch ein Bild des USB Teils.

Aber wie bereits gesagt funktionieren alle Ports richtig unter High Sierra.

Beitrag von „Harper Lewis“ vom 23. März 2018, 09:38

Hast du denn nun die Positivliste (SSDT), die Negativliste (uia_exclude) oder beides im Einsatz? Der Dump zeigt, dass die SSDT nicht oder nicht richtig geladen wurde. HS06 sollte ein interner Port sein, wird aber als USB3-Port angezeigt.

Nachtrag: Versuche es mal bitte mit der SSDT im Anhang.

Beitrag von „witjojo“ vom 23. März 2018, 12:29

Ich habe jetzt deine SSDT-USB.aml verwendet.
Und anbei davon der ioreg Dump.
Woran erkenne ich jetzt ob HS06 als interner Port definiert ist?
Evtl. an dem Build-In = True (siehe Bild)? Ist es denn jetzt korrekt?

Noch eine Frage zur SSDT-USB, Was ist das für eine Änderung die du vorgenommen hast?

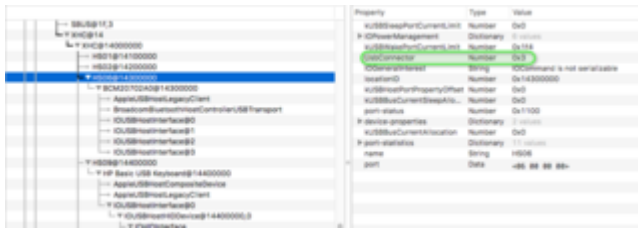
```
....  
"8086_8xxx",  
Package (0x04)  
....
```

```
vorher  
....  
"8086_a12f",  
Package (0x04)  
....
```

Beitrag von „Harper Lewis“ vom 23. März 2018, 12:52

8086_8xxx ist der passende Eintrag, 8086_a12f kommt bei dir gar nicht vor (das siehst du im IOReg).

Wenn alles passt, solltest du jetzt unter *USBConnector* den Wert *0xff* statt *0x3* sehen:



Beitrag von „witjojo“ vom 23. März 2018, 13:09

Ja jetzt passt es.

Und noch ne kurze Frage. Wo finde ich genau den Eintrag 8086_8xxx im ioreg?
Hier im 2. Bild?

Beitrag von „Harper Lewis“ vom 23. März 2018, 13:23

Sehr gut!

Den Eintrag findest du gar nicht, dafür aber direkt unter XHC:

```
device-id: <31 8c 00 00>
vendor-id: <86 80 00 00> (Intel)
IOName: pci8086,8c31
```

Nachtrag: Sieht man ja auf deinem 2. Screenshot.

In der config.plist von USBInjectAll gibt es keinen direkten Eintrag für pci8086_8c31. Wäre es so, sollte man diesen benutzen. Ansonsten eben wie bei dir nun 8086_8xxx, der pci8086_8c31 und weitere ähnliche Devices abdeckt.

Beitrag von „witjojo“ vom 23. März 2018, 14:07

HILFE

Ich habe mein USB Ports jetzt alle nochmal getestet.

1. Ich habe keinen SS06 mehr unter ioreg. Und wenn ich einen USB 3.0 Stick in den zuvor definierten Port SS06 stecke wird er jetzt unter HS01 angezeigt. Kein USB 3.0 mehr (Bild vorher_SS06_jetzt_HS01_USB2.0.png). Der Vergleich mit SS01 USB 3.0 welcher noch geht (Bild SS01_USB3.0.png)
2. Wenn ich einen USB 3.0 Stick in dem vorher definierten Port SS02 stecke wird er unter HS09 angezeigt. Kein USB 3.0 mehr.

Port SS02 und SS06 funktionieren jetzt nicht mehr.

Ist da mit der Nummerierung durch den internen Port etwas in der SSDT-USB durcheinander geraten?

Beitrag von „Harper Lewis“ vom 23. März 2018, 15:58

Probiere mal bitte die SSDT im Anhang aus. Die ist 1:1 der Vorlage von Rehabman entnommen.

Beitrag von „witjojo“ vom 30. März 2018, 15:20

So jetzt habe ich deine SSDT-USB nochmal getestet.

1. Da ist ein Fehler in HS10 ---> muss 0x0A sein.

2. Es werden gar keine SSPx Ports angezeigt und wenn ich einen USB 3.0 Stick reinstecke wird der unter den HSxx Ports angezeigt mit USB 2.0 Geschwindigkeit.

Beitrag von „Harper Lewis“ vom 30. März 2018, 17:25

Dann solltet sich vielleicht jemand mit mehr Ahnung als ich mal die SSDT anschauen, [@kuckkuck](#) z.B.

Beitrag von „kuckkuck“ vom 30. März 2018, 18:07

Na dann, bitteschön:

Edit: Wir reden hier doch nicht vom GA-Z97-D3H, oder? Wie viele Ports hat das eigentliche Ding?

Beitrag von „witjojo“ vom 31. März 2018, 15:00

[@kuckkuck](#) Danke Danke Danke.

Jetzt funktioniert alles wie es soll. Und auch alle USB 3.0 Ports sind vorhanden.

Ich wusste nicht das die Port-Nummerierung von SS01 bei 10 anfängt:

```
"SS01",  
Package (0x04)  
{  
"UsbConnector",
```



```
0x03,  
"port",  
Buffer (0x04)  
{  
0x10, 0x00, 0x00, 0x00 <----- hier  
}  
},
```

Ich hätte es in der mit Clover erzeugten DSDT.aml eigentlich sehen müssen:

```
Device (SSP1)  
{  
Method (_ADR, 0, Serialized)  
{  
If (LEqual (And (CDID, 0xF000), 0x8000))  
{  
Return (0x10) <----- hier  
}  
Else  
{  
Return (0x0A)  
}  
}
```

Ich habe es jetzt aber nachvollzogen und weiß nun wie es funktioniert. Vielleicht für das nächste Mal.

Gruß
witjojo

Beitrag von „Harper Lewis“ vom 31. März 2018, 17:34

Interessanter Hinweis. Ich dachte, die passende Zuordnung der Portnummern würde USBInjectAll übernehmen.

Beitrag von „kuckkuck“ vom 31. März 2018, 19:16

Kein Problem 😊

Die PortNummern sind in der SSDT ein ausschlaggebender Identifikator des Ports. Die vorhandene PortNummer entnehme ich immer am liebsten dem IOReg.

Zitat von Harper Lewis

In der config.plist (Info.plist?) von USBInjectAll gibt es keinen direkten Eintrag für pci8086_8c31.

Die Regelung mit 8xxx war mir bisher nicht so bekannt, deswegen habe ich egal ob vorhanden oder nicht immer die richtige ProductID des Controllers eingetragen. Das ganze hatte sich bewährt, deswegen habe ich es genauso bei witjojo gemacht. Ob 8xxx jetzt einen Unterschied machen würde weiß ich nicht 😊

Beitrag von „Harper Lewis“ vom 1. April 2018, 10:34

Die Beispieldatei SSDT-UIAC-ALL ist dann eher verwirrend, als hilfreich. 🤔