

Erledigt

Kein Sleep mehr, auf ein neues ...

Beitrag von „locojens“ vom 14. September 2018, 10:47

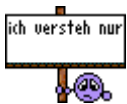
Ich hatte meinen Hack soweit das er mit High Sierra und auch mit der Mojave Beta in den Sleepmodus ging. Dann fiel mir auf das meine USB 3.0 Ports nicht liefen.

Letzteres Problem habe ich behoben es funktionieren Alle 3.0er . Nun wollte ich den "Kleinen" wieder so einstellen das er in ungenutzten Zeiten wieder in den Sleep geht. Pustekuchen.

Der Rechner rödelt ne Weile wenn der Bildschirm schon im Standby ist, nehme mal an er bastelt am Sleepimage, dann geht er aus und wacht sofort wieder auf.

Ich habe schon im Bios sämtliche Einstellungen überprüft und auch in den SYS-Einstellungen z.Bsp. Wake from Bluetooth etc. ausgestellt, aber er mag nicht.

Kann es wirklich an den 3.0 Ports liegen ???



Beitrag von „CMMChris“ vom 14. September 2018, 11:21

Hast du deine internen Ports auch als solche markiert? Ohne diesen Schritt wird Sleep nicht funktionieren. Wenn du zum Beispiel eine WLAN + Bluetooth Karte nutzt wo Bluetooth an einem internen USB Header hängt, musst du diesen in einer SSDT-UIAC auf 255 (0xff) setzen.

Beitrag von „Sascha_77“ vom 14. September 2018, 11:39

Bzgl. deines Sleeps lege ich dir auch mal diese 5 Zeilen nahe falls noch nicht ausgeführt:

Code

1. sudo pmset -a hibernatemode 0
2. cd /private/var/vm/
3. sudo rm sleepimage
4. sudo touch sleepimage
5. sudo chflags uchg /private/var/vm/sleepimage

Nutzt DU eine DSDT? Dann könntest du noch den USB3 Power Patch aus den Repos von MaciASL ausführen. Das war bei meinem Z97 nämlich das Problem. Der ist auch direkt nach dem Schlafen aufgewacht. Nach dem Powerpatch war Ruhe.

Aber ggf. dann die 3 _DSM Einträge ganz unten aus dem Patchfile entfernen.

Beitrag von „locojens“ vom 16. September 2018, 20:14

So ich habe das nun verifiziert es liegt an dem Bluetooth-Teil der "Flughafen Karte" wenn ich das Kabel abziehe schläft er nun wie ein Baby.

Hatte leider keine ruhige Minute gefunden dieses WE mich ausführlicher mit dem Thema SSDT-UIAC.aml zu befassen.

War heute eher wie Knete im Kopf...

Da muss ich mich wohl oder übel dann die Woche damit befassen.

Beitrag von „CMMChris“ vom 16. September 2018, 21:45

Wenn du da nicht weiter kommst kann ich dir dabei helfen. Können wir dann über ne Team Viewer Sitzung gemeinsam machen. Ist im Grunde ne Sache von 15 bis 20 Minuten.

Beitrag von „Sascha_77“ vom 17. September 2018, 07:12

Mit diesem Tool kannst Du Bluetooth an- und abschalten per Terminal:

<https://github.com/toy/blueutil>

Um das nicht manuell machen zu müssen kann ich Sleepwatcher:

<https://www.bernhard-baehr.de/>

wärmstens empfehlen. Man kann damit Kommandos definieren die bei Sleep bzw. Wakeup ausgeführt werden sollen.

Beitrag von „locojens“ vom 19. September 2018, 12:00

Heute habe ich mal etwas Zeit und auch Ruhe um mich mit der Materie zu befassen, gibt es denn eine Möglichkeit die Veränderung eines Ports herauszufinden wenn man den Versuch des Patches wagt. Ich meine jetzt nicht den Rechner in den sleep zu schicken, da er nach dem wieder aufwachen aus einem missglückten Sleep zwar läuft aber nur nen schwarzen Bildschirm anzeigt... ?

[@Sascha 77](#) das mit dem sleepwatcher wäre natürlich eine Option, jedoch wollte ich das Problem lieber auf der Bootloader Ebene angehen da der mit dem Clover ja dann quasi ohne weitere Eingriffe in das OS als solches auch direkt nach einer Neuinstallation laufen würde.

Ich hab mal die Datei eben noch angehängt, wenn mal jemand Zeit hätte sich die anzusehen, wäre es super.

Denn ich habe Null Plan bis jetzt ob das überhaupt was taugt was ich da verbrochen habe.

Hat hier noch jemand eine Idee dazu? Ich hab schon mit der SSDT-UIAC gebastelt, bin aber

noch nicht mal so richtig Sicher ob die als AML oder DSL da rein muß ??

Beitrag von „Harper Lewis“ vom 19. September 2018, 12:41

.aml wäre richtig. Welcher Controller ist denn verbaut? Kannst du mal eine Screenshot vom DPCIManager > PCI List machen? Welche Ports aktiv sind hast du auch schon herausgefunden?

Beitrag von „locojens“ vom 19. September 2018, 13:07

Aktiv sind Alle. Der problematische Port ist der HS12 denn da steckt der Bluetoothteil der "Flughafen Karte" dran d.h. wenn die abgezogen ist kein Bluetooth aber Sleep.

Hab nochmal die DSL-Datei angehängt und die AML aber da stehe ich irgendwie auf dem Schlauch.

Beitrag von „Harper Lewis“ vom 19. September 2018, 13:13

Du musst herausfinden, welche Ports aktiv sind, wie diese bezeichnet sind und ob es sich um USB-2, USB-3 (oder neuer) oder interne Ports handelt. Hilfreich dabei ist [diese Anleitung](#) von [@kuckkuck](#). Das ist die Voraussetzung, erst dann kann man optimieren.

Beitrag von „CMMChris“ vom 19. September 2018, 14:03

[@locojens](#) Ich sagte dir ja schon dass ich dir da per Team Viewer Sitzung assistieren kann. Melde dich dazu einfach per PN wenn du magst.

Beitrag von „locojens“ vom 19. September 2018, 16:14

Moin [@CMMChris](#) Würde mich schon reizen deine Hilfe anzunehmen. Sollte ich da der Einfachheit nicht einige der Benötigten Programme direkt auf den Schreibtisch packen. da ich die auf extra Laufwerken verstreut habe?! Denn das muss bei mir nicht alles im Programme-Ordner rumfliegen.

Beitrag von „CMMChris“ vom 19. September 2018, 16:20

Wir brauchen nur IOJones, ggf. Clover Configurator und MacIASL. Außerdem ein USB 2.0 und ein USB 3.0 Gerät / Stick.

Beitrag von „locojens“ vom 21. September 2018, 16:53

So das USB rein/raus Spielchen habe ich nun mal durch beim XHC@14 habe ich folgendes:

Code

1. Es handelt sich um den intel USB-Controller 8086:A2AF
- 2.
- 3.
4. HS01@14100000 frei
5. HS02@14200000 frei
6. HS03@14300000 hinten USB3.1 mit USB2.0 Geschwindigkeit bei der LAN Buchse
7. HS04@14400000 hinten USB3.1 mit USB2.0 Geschwindigkeit bei der LAN Buchse
8. HS05@14500000 Front USB3.0 mit USB2.0 Geschwindigkeit
9. HS06@14600000 Front USB3.0 mit USB2.0 Geschwindigkeit
10. HS07@14700000 hinten USB3.1 mit USB2.0 Geschwindigkeit beim HDMI
11. HS08@14800000 hinten USB3.1 mit USB2.0 Geschwindigkeit beim HDMI
12. HS09@14900000 hinten USB2.0 beim PS/2 Anschluss
13. HS10@14a00000 hinten USB2.0 beim PS/2 Anschluss
14. HS11@14b00000 intern USB2.0 daran ein Cardreader und USB2.0 HUB an der Front
15. HS12@14c00000 intern USB2.0 daran der Bluetooth-Teil der "Flughafen Karte"

16. HS13@14d00000 intern USB2.0 daran ein USB2.0 Frontanschluss
17. HS14@14e00000 intern USB2.0 daran ein USB2.0 Frontanschluss
- 18.
- 19.
- 20.
- 21.
22. SS01@15100000 frei
23. SS02@15200000 frei
24. SS03@15300000 hinten USB3.1 bei der LAN Buchse
25. SS04@15400000 hinten USB3.1 bei der LAN Buchse
26. SS05@15500000 intern USB3.0 Frontanschluss in Gebrauch
27. SS06@15600000 intern USB3.0 Frontanschluss in Gebrauch
28. SS07@15700000 hinten USB3.1 beim HDMI
29. SS08@15800000 hinten USB3.1 beim HDMI
30. SS09@15900000 frei
31. SS10@15a00000 frei
- 32.
- 33.
- 34.
- 35.
36. USR1@14f00000 Platzhalter?
37. USR2@15000000 Platzhalter?

Alles anzeigen

Beitrag von „Harper Lewis“ vom 21. September 2018, 17:23

Das sind aber sehr viele Ports. Sicher, dass die alle über den XHC(I)-Controller angebunden sind? Das Board hat ja noch einen weiteren Controller verbaut: ASMedia 3142 USB 3.1 Gen2 with USB Type-C™ and Type-A. Magst du mal einen IOREG-Dump zur Verfügung stellen?

Nachtrag: Ok, doch alle unter XHC. Kannst du ein paar Ports opfern? Maximal 15 sollten es sein. USR1 und USR2 so wie alle inaktiven Ports kann man eh kicken.

Beitrag von „locojens“ vom 21. September 2018, 18:08

PXSX@0 ist der ASMedia Controller. HS09@14900000 und HS10@14a00000 tauchen auch beim PXSX auf. Da hängt hinten extra in rot ein USB3.1 Gen.2 Type A und ein USB3.1 Typ C Gen. 2 dran. Ich glaube das der interne USB 3.1 Typ C Anschluss auch am ASMedia hängt ... !?

Beitrag von „Harper Lewis“ vom 21. September 2018, 18:52

RP05.PXSX ist der ASMedia-Controller (pci1b21,2142). Kannst du die Ports unter XHC auf insgesamt 15 zusammenstreichen? Siehe auch Nachtrag In Beitrag #15.

Nachtrag: Ich sehe auch gerade, dass bei dir X86PlatformPlugin nicht geladen wird. Abhilfe: In der config.plist Acpi > SSDT > Generate Options > PluginType.

Beitrag von „locojens“ vom 21. September 2018, 20:46

Also einsparen könnte ich HS07, HS08, HS09 wenn auch schöner wäre wenn alle laufen würden.

Zum X86PlatformPlugin er taktet aber ganz normal und nicht so wirr wie mit PluginType.

Beitrag von „CMMChris“ vom 21. September 2018, 20:47

Man kann mit FakePCIID-XHCIMux eventuell ein paar USB 2.0 Ports auf EHCI schieben, womit diese aus dem Port Limit für XHC rausfallen.

Beitrag von „locojens“ vom 21. September 2018, 23:46

Das wären bei mir ja 6 an der Zahl. HS09@14900000 , HS10@14a00000 ,HS11@14b00000 , HS12@14c00000 der mit dem Bluetooth , HS13@14d00000 und HS14@14e00000

Das heisst ich muss die Adresse der Ports im IOReg rausfischen und in die XHCIMUX plist reintragen bei IOPCIPrimaryMatch.

Nachtrag: Nee geht ja sooo einfach scheinbar nicht.

Wenn kleine Kinder spielen...! Ich habe zur Zeit ja viel um die Ohren wie schon weiter oben erwähnt. Aber dieser PXSX hat mich neugierig gemacht.

Jetzt habe ich die "Flughafen Karte" einfach mal an zum Testen an den ASMedia Controller bzw dessen Anschlüsse auf der Rückseite angesteckt, bei dem USB2.0 Port welcher unter XHC@14 HS09@14900000 läuft geht der Sleep nicht! Wenn ich jedoch den USBc Anschluss USB3.1 Gen2 nutze welcher wie schon erwähnt unter PXSX steckt und demnach nicht unter XHC auftaucht, schläft die Kiste besteht eine Möglichkeit den HS09 aus XHC rauszubekommen damit der nurnoch unter PXSX auftaucht ?

MfG Jens

Nachtrag: habe es nun mehrfach probiert egal ob automatisch oder selbst in den Ruhezustand geschickt... er schläft wie ein Baby und er wacht auch ohne Murren wieder auf. 🙄🤪

Beitrag von „Harper Lewis“ vom 22. September 2018, 10:16

FakePCIID-XHCIMux bringt hier leider nichts, weil das Board keinen EHCI-Controller hat. Der Ruhezustand sollte funktionieren, wenn man den Port, an dem Bluetooth hängt, als intern konfiguriert (z.B. über die SSDT-UIAC). Aber wie gesagt, es sollten dann maximal 15 Ports unter XHC aktiv sein.

Hier mal als Beispiel, wenn Bluetooth unter HS12 auftaucht:

Code

1. "HS12",
2. Package (0x04)
3. {
4. "UsbConnector",
5. 0xFF,
6. "port",
7. Buffer (0x04)
8. {
9. 0x0C, 0x00, 0x00, 0x00 //
10. }
11. }

Alles anzeigen

Beitrag von „locojens“ vom 22. September 2018, 12:36

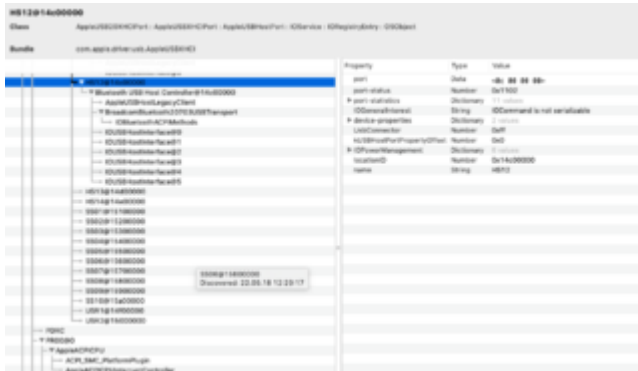
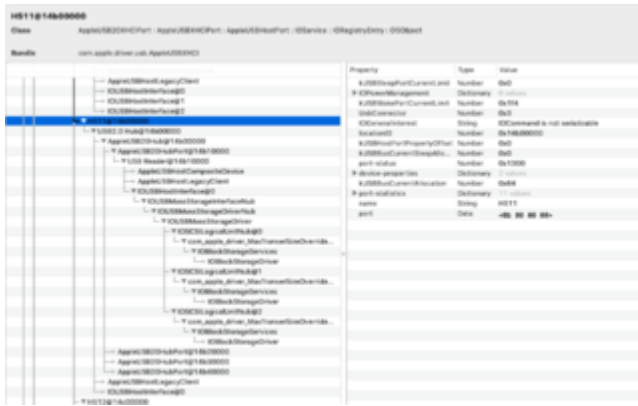
quick and dirty!

0xFF oder 255 war gerade eben auch nochmal in meinem Hirn rumgegangen. Ich bin zur Zeit etwas ooO um diese DSDT's etc. zusammen zu basteln, verliere dauernd den Faden. Da dachte der Jens sich das es doch wesentlich einfacher gehen MUSS.

Der Port war mir ja nun schon bekannt. HS12@14c00000 nennt sich der Gute. Also habe da jetzt gaaaanz einfach mal die USBInjectAll.kext bzw. deren Plist auf mein Platine und das von mir verwendete SMBIOS angepasst.

Ich habe alle nicht zutreffenden Einträge entfernt außer die passenden eben und damit die Liste schon wesentlich schlanker bekommen.

Nun habe ich an der passenden Stelle (dem HS12) den Port Typ einfach von 3 auf 0xff (255) geändert. Siehe da er taucht nun in einer neuen Form in IOReg. auf.



1. Bild normaler USB 2. Bild der "BLUETOOTH" USB

Ergebnis: Der Rechner geht in den Sleepmodus (ob über das Menü oder über die Energieeinstellungen) und er wacht auch nicht mehr von alleine auf! Wake über Bluetooth (haben wir ja in einem anderen Thread) geht nicht. Das ist aber egal da meine Tastatur wie gewöhnlich ja direkt vor mir liegt.

Hier ist noch die von mir angepasste Plist.... [Info.plist](#)

PS: falls Interesse besteht erkläre ich es nochmal etwas schöner/besser, was ich hier verbrochen habe.

Mfg Jens!

Beitrag von „Harper Lewis“ vom 22. September 2018, 12:41

So kann man es natürlich auch machen. Oder auch [gleich auf USBInjectAll](#) verzichten.
Nachtrag: USBPower.kext mit dem FBPatcher zu erstellen wäre auch eine Variante.

Beitrag von „locojens“ vom 22. September 2018, 13:45

intel FB-Patcher läuft bei mir nicht so wie er es sollte... Mojave und Coffee Lake mag der definitiv NICHT.

Beitrag von „NoirOSX“ vom 22. September 2018, 13:49

Was kommt da als Fehler, bzw was läuft da nicht um einen USB.kext zu erzeugen

Beitrag von „Harper Lewis“ vom 22. September 2018, 14:01

Statt USBInjectAll könntest du mal die Kext im Anhang ausprobieren. Ich weiß allerdings nicht, ob das so zusammen mit dem Portlimit-Patch funktioniert. Auf jeden Fall muss dieser bei 18 Ports aktiv sein. Bitte prüfe auch nochmal, ob ich die Ports richtig konfiguriert habe. HS11 und HS12 habe ich mal als intern konfiguriert, den Rest entweder als USB-2- oder USB-3-Ports.

Beitrag von „locojens“ vom 22. September 2018, 14:23

[@NoirOSX](#) der Zeigt einfach nichts an, Coffee Lake auswählen aktualisieren ... nix ,voreingestellt ist ja Kaby Lake

[@Harper Lewis](#) was macht die anders als die USBInjectAll ?

Beitrag von „Harper Lewis“ vom 22. September 2018, 14:45

Siehe [hier](#). Schonmal ausprobiert?

Beitrag von „Noir0SX“ vom 22. September 2018, 14:56

[@locojens](#) USB könnte aber trotzdem gehen, auch wenn die Auswahl der Intel HD dort nicht geht

Beitrag von „Harper Lewis“ vom 24. September 2018, 09:42

Und? Füllwort Füllwort.

Beitrag von „locojens“ vom 24. September 2018, 10:19

Also der FB-Patcher bringt mir nichts, das andere mit der USB-Power.kext werde ich jetzt mal testen. Obwohl das ja auch nichts anderes bewirkt wie die angepasste InjectALL.

Beitrag von „Harper Lewis“ vom 24. September 2018, 11:01

Könnte sein, aber USBInjectAll direkt zu bearbeiten fühlt sich für mich nicht sauber an und ist

auch nicht updatesicher. Genau dafür ist ja das Konfigurieren der Ports über die Whitelist gedacht (SSDT-UIAC).

Oder eben die oben angehangene Kext ohne USBInjectAll.

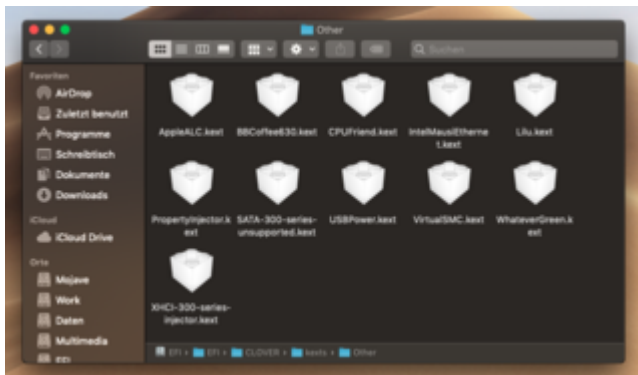
Beitrag von „locojens“ vom 24. September 2018, 11:32

habe jetzt die USBPower.kext und die von mir angepasste umgemodelte XHCI-300-series-injector.kext (war mal die 200er) nur Port 11 steht nun wieder auf 0 statt 255 in der USBPower.kext...

Es funktioniert nun und bedarf nun wie du ja sagtest keine Neuanpassungen in der InjectALL bei Updates etc.

Danke dafür [@Harper Lewis](#) .

Somit sieht es im Other-Ordner nun so aus ...



Beitrag von „Harper Lewis“ vom 24. September 2018, 11:43

Besten Dank für die Rückmeldung. Das ist allerdings seltsam, denn HS11 und HS12 sind als

intern konfiguriert:

▼ HS11	Dictionary	↕ 2 items
UsbConnector	Number	↕ 255
port	Data	↕ <0b000000>
▼ HS12	Dictionary	↕ 2 items
UsbConnector	Number	↕ 255
port	Data	↕ <0c000000>

Welche device-id hat der Controller im Original und wie wird das mit mit XHCI-300-series-injector.kext umgebogen? Ich bin jetzt von 0xa2af8086 ausgegangen, XHCI-300-series-injector sollte eigentlich nicht nötig sein.

Beitrag von „locojens“ vom 24. September 2018, 12:14

Hast Recht hab ich garnicht drauf geachtet. Der ist nun auch raus. Also nun auch ohne XHCI-300-series-injector.kext.

Zum 11er hab den geändert, obwohl das eigentlich völlig egal ist.

Also nochmal neu... :

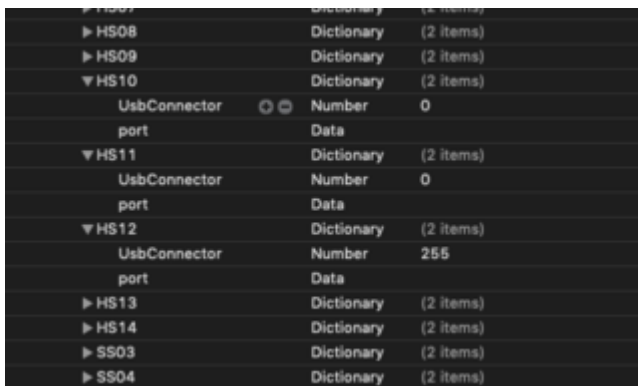


Beitrag von „Harper Lewis“ vom 24. September 2018, 12:20

Also passt jetzt alles? Im Anhang da oben hat HS11 schon 255 als UsbConnector, daher wundert mich das irgendwie... Du müsstest jetzt unter XHC > XHC im IOReg auch unter IOPersonalilyPublisher den Wert "com.Headsoft.USBPower" sehen.

Beitrag von „locojens“ vom 24. September 2018, 12:30

Noch ein Nachtrag, die Plist sieht so aus.



Beitrag von „Harper Lewis“ vom 24. September 2018, 12:38

Viele Fragezeichen. Ich habe mir die Datei aus Beitrag #26 jetzt nochmal heruntergeladen und da ist HS11 = 255. Kann doch nicht sein... 😞

Beitrag von „locojens“ vom 24. September 2018, 12:39

Ich hab die geändert. 😁

Beitrag von „Harper Lewis“ vom 24. September 2018, 12:41

Ah, jetzt ist der Groschen gefallen 😊

Beitrag von „locojens“ vom 27. September 2018, 12:34

Juhu??? 😞

Nun habe ich meine RX580 eingebaut und das Drama geht in eine neue Runde. Den USB/Bluetooth kann ich schonmal ausschliessen, der läuft weiterhin auf "intern".
In wie weit kann mir die Grafikkarte beim Sleep dazwischenfunken?

Hat sich geklärt Problem gelöst.