

Das leidige Thema USB und seine Auswirkungen

Beitrag von „G.com“ vom 28. März 2021, 15:52

Moin,

die letzten zwei Wochen habe ich mich relativ tief in das Thema OC eingearbeitet und habe heute einen deutlich besseren Hacki, als je zuvor.

Das Thema USB - an für sich keine Raketenwissenschaft, aber dennoch ein Becken voller kleiner Stolperfallen.

Warum ich das erzähle. Nun ja, ich hatte seit Clover so einige kleine Problemchen.

- Sleep/Wake Reboots
- Nach manuellem Ruhezustand jeden zweiten Boot kein BT
- Festplatten wurden plötzlich nur noch am HS Port initialisiert, obwohl Sie den 3.0 Standard erfüllen...

Ich habe mit vielen Methoden mein USB gemappt.

Mir haben hier etliche wirklich kompetente User mit der DSDT und Anpassungen im Bios geholfen.

Nichts, hat das gelöst.

Nun ja, heute weiß ich mehr.

Wichtig ist, genau zu lesen.

Ab hier setze ich OC 0.6.7 und die Kenntnis des Selben voraus.

Folgende Vorbereitungen:

- USBInject all
- . Kernel - Quirks - XHCIPortLimit enabled
- die zum SMBios und zum Board passenden Rename [HIER](#)

Einsetzen und booten. Wichtig hierbei ist, Dortonia spricht z.B. davon man brauche die meisten Renames erst ab Skylake bzw. für das iMacPro1,1 SMBios nur den XH1 to SHCI Rename. Stimmt nur bedingt, bei meinem Board benötige ich alle drei, da ich beide Chipsätze habe und alles sauber gepatcht haben möchte/muss.

Warum das ganze? Nun ja, kurz gesprochen ab Catalina patcht Apple die XHCI Ports sogar selber, und das kann bei nicht vorhandnen Renames dazu führen, dass die Ports von OSX falsch gemappt werden, das führt zu Problemen. Siehe oben.

Dann alle Ports im Terminal testen:

Code

1. `ioreg -l -p IOService -w0 | grep -i XHC1`

Code

1. `ioreg -l -p IOService -w0 | grep -i EHC1`

Code

```
1. ioreg -l -p IOService -w0 | grep -i EHC
```

Es sollte keine Ausgabe kommen, wird irgendetwas angezeigt - spätestens dann die Renames nutzen.

Erst dann ist man bereit, den bekannten Weg mit Hackintool zu gehen um eine USBPort.kext zu erstellen.

BTW - der Dummy Kext bei Dortonia funktioniert auch, nutzt aber eine andere Adressierung, der Kext aus Hackintool spricht die direkte Adresse des Gerätes an. Hat viele Vorteile.

Mit dem gewonnen Wissen konnte ich schlussendlich sogar meine Inateck Karte voll patchen. Hierzu habe ich den einfach die entsprechenden Ports in den Kext integriert.

Mein Beispiel anbei.

WICHTIG: Ich hatte zunächst die Personalities für die Stromversorgung direkt im Kext, IMO funktioniert das aber besser mit der separaten SSDT-EC-USBX.aml, denn da werden dann alle Ports unabhängig vom Kext gepatcht.

Tja warum ich das schreibe, zum einen für alle mit ähnlichen Problemen, zum Anderen zu zeigen, dass das Ganze kein Hexenwerk ist und es nur Geduld und Interesse braucht. Ausserdem ich den Moderatoren vor Jahren ein Versprechen gegeben habe am Sleep dranzubleiben und sollte ich eines Tages eine Lösung finden, dann hier zu berichten.

[griven](#) [al6042](#) [kuckkuck](#) Melde gehorsamst Vollzug.

Die Erklärung zum Hackintool spare ich mir hier mal, ist ja oft genug vorhanden.

Meine Probleme sind gelöst und ich glücklich.

Viel Erfolg.

P.S. Kleiner Nebenerfolg - warum auch immer zusammen mit darkwake = 3 laufen die Lüfter nicht mehr heiß beim Wake 😊

UPDATE: Bin eben durch einen anderen Post erinnert worden, WICHTIGE Anmerkung.

Trotz Rename kann es sein, dass Ports nicht richtig gemappt werden. So z.B. hatte ich einen Port der nur als HS angezeigt wurde. Dies Problem hatte ich nur beim der Intel 9 Series Chipsatz. Die Inateck brauchte das nicht. Warum - das weiß nur der liebe Gott - oder vielleicht

Muahaha!!

Apple.



Ursache ist auch hier ein Catalina Problem. Die IOClass hat sich verändert.

```
MacPro1,1-@101
  IOPlatformInfo          < Dictionary < key,value pairs
  IOPlatformIdentifier    < Dictionary < key,value pairs
  IOClass                 < String      com.apple.driver.AppleUSBHostHub
  IONameMatch             < String      USB
  IOCFPropertyMatch      < String      IODataHS
  IOProbeClass            < Number     10000
  IOProviderClass         < String     AppleUSBHCXHCI
  IOPlatformDeviceProperty < Dictionary < key,value pairs
  port-count              < Data       <IO000000>
  *ports                  < Dictionary < key,value pairs
    *HS00                 < Dictionary < key,value pairs
      IODataConnector     < Number     0
      name                < String     HS00
      port                 < Data       <IO000000>
  model                   < String     MacPro1,1
```

Da musste ich dann wissend, dass mein SSP06 korrekterweise eingebunden werden muss, ihn einfach mit einbinden und dann im Kext die IOClass manuell auf die alte Variante abändern.

```
MacPro1,1-@101
  IOPlatformInfo          < Dictionary < key,value pairs
  IOPlatformIdentifier    < Dictionary < key,value pairs
  IOClass                 < String     com.apple.driver.AppleUSBHostTargetProperties
  IONameMatch             < String     USB
  IOCFPropertyMatch      < String     IODataHS
  IOProbeClass            < Number     10000
  IOProviderClass         < String     AppleUSBHCXHCI
  IOPlatformDeviceProperty < Dictionary < key,value pairs
  port-count              < Data       <IO000000>
  *ports                  < Dictionary < key,value pairs
    *HS00                 < Dictionary < key,value pairs
    *HS01                 < Dictionary < key,value pairs
    *HS02                 < Dictionary < key,value pairs
    *HS03                 < Dictionary < key,value pairs
    *HS04                 < Dictionary < key,value pairs
    *SSP0                 < Dictionary < key,value pairs
    *SSP1                 < Dictionary < key,value pairs
    *SSP2                 < Dictionary < key,value pairs
    *SSP3                 < Dictionary < key,value pairs
    *SSP4                 < Dictionary < key,value pairs
    *SSP5                 < Dictionary < key,value pairs
    *SSP6                 < Dictionary < key,value pairs
  IODataConnector        < Number     0
  name                   < String     HS06
  port                   < Data       <IO000000>
```

Voilà, SSP06 wurde korrekt eingebunden nach Neustart.

Sorry dafür das hier nachzuschieben. Lag mir aber am Herzen.

Beitrag von „griven“ vom 28. März 2021, 20:49

Sehr geil das es Dir gelungen ist und meine absoluten Respekt für Deine Beharrlichkeit da kann man nur den Hut ziehen 🙌👏

Beitrag von „G.com“ vom 28. März 2021, 22:25

Mhh, danke für die Blumen. Aber seit Geburt der Tochter war auch lange nix möglich. 2 Jahre OC verpennt.

Ich hatte Nachholbedarf

Nun, Corona hatte dann sein Übriges dazu beigetragen.

Beitrag von „macdream“ vom 28. März 2021, 23:04

Sehr schön, jetzt habe ich auch gesehen WAS du geschrieben hast.

Beitrag von „kuckkuck“ vom 28. März 2021, 23:40

[Zitat von G.com](#)

zum einen für alle mit ähnlichen Problemen, zum Anderen zu zeigen, dass das Ganze kein Hexenwerk ist und es nur Geduld und Interesse braucht. Ausserdem ich den Moderatoren vor Jahren ein Versprechen gegeben habe am Sleep dranzubleiben und sollte ich eines Tages eine Lösung finden, dann hier zu berichten.

[griven](#) [al6042](#) [kuckkuck](#) Melde gehorsamst Vollzug.

Wow! Genau so wünscht man sich das! Danke für den Beitrag!



Beitrag von „G.com“ vom 31. März 2021, 15:45

So, ein kurzes Update. CPUFriend hat nun meine allerletzte Baustelle geschlossen. Nun läuft das System ohne lästige CPU-PLL Einstellung im Bios und Darkwake=3 mit PowerNap laufen auch. Es ist wirklich komisch. Wenn man nur will und sich tief einarbeitet - dann kann man sich echt einen 100% lauffähigen Rechner bauen.

Repektive ist am wichtigsten IRQ Korrekturen, SMBus Patch, USB Patch, Prozessor PM systemspezifisch. Alles andere, wie z.B. SMBios ist für die zur Verfügung gestellten Funktionen z.B. DRM, VDA Support etc. wichtig.

DANKE an alle hier, die mir all Ihre Wissen über die Jahre zur Verfügung gestellt haben.

Ich hoffe das bald alles weitergeben zu können. Wenn ich hier Hilfesuchende finde, stehe ich gerne zur Verfügung.

Beitrag von „G.com“ vom 17. April 2021, 08:11

OK, vorsichtige Anmerkung. Ich hatte ja berichtet, das es plötzlich wieder zu Problemen kam. Hatte da so einige Vermutungen.

Derzeit scheint sich herauszustellen, was der Unterschied ist. Nach Erstellung der EFI nach Dortania hatte ich gemäß Vorgaben alles im Sleep eingerichtet, was vorgeschlagen war. Danach aber die Defaults wiederhergestellt.

Es scheint so zu sein, dass autopoweroff hier reinspielt.

Was macht das genau...

Um mit den Regularien der EU zum Stromverbrauch konform zu gehen, stellt der Mac sich nach einiger Zeit von Standby auf den Low-Energy Betrieb um. Hierbei sollte der Inhalt des RAM auf die Platte geschrieben werden. Ähnlich oder sogar genau wie beim Hibernation 3 bzw. 25. Und genau da könnte die Ursache liegen. Denn beim Zurückschreiben in den RAM entsteht ein Fehler, weil Bereiche geslided werden oder weiß der Henker.

Also folgenden Befehl erachte ich für korrekten Sleep auf meiner Maschine für extrem wichtig:

Code

1. `sudo pmset -a autopoweroff 0`

Alles andere scheint geringeren Einfluß zu haben.

Was tut das, es verhindert, dass OSX den Low-Energy Betrieb abschaltet. Somit entstehen dann auch keine Probleme nach long Sleep - denn er schaltet hier nicht um. Das Gerät bleibt im Standby.

Sicher ist der Stromverbrauch etwas höher, aber damit kann ich leben.

Vielleicht, aber auch nur ganz vielleicht habe ich damit nach viel Experimentieren herausgefunden, wo der Unterschied zwischen "läuft" und "läuft nicht" liegt.

Ich berichte weiter.

Beitrag von „bluebyte“ vom 17. April 2021, 09:38

[G.com](#) das bedeutet, man kann auf Hibernate 3 wechseln, das Hibernate-File wieder beschreibbar machen und Sleep funktioniert? Schreibt er jetzt das RAM auf die Disk? Ist das Vorgehen auch bei anderen Boards umsetzbar?

Beitrag von „G.com“ vom 17. April 2021, 13:27

[bluebyte](#) Nein, zumindest teste ich das gerade noch nicht. Mein Ansatz ist erst einmal zu verhindern, dass er schreiben will im long sleep. Da das Image ja nicht beschreibbar ist, kann er auch keine Daten zurückschreiben. Daher vermute ich meine Probleme nach long sleep. Aber, wie gesagt. Derzeit experimentiere ich und werde das Ergebnis zeitnah mitteilen.