

Erledigt

Eigene USB SSDT - ASUS HERO VIII - Review und ein paar Detailfragen

Beitrag von „Maddeen“ vom 25. September 2018, 22:52

Hallo zusammen,
nach dem auch bei diesem Update erneut das USB3 Thema bei mir aufgetreten ist und wieder per manuellem Eingriff gelöst werden musste,
wollte ich jetzt mal das Thema mit der eigenen USB-SSDT angehen, da es ja offensichtlich die einzige dauerhafte Lösung ist.

Mir geht es jetzt einfach nur um einen Quercheck bzw. 4-Augen-Prinzip der SSDT-USB und ein paar Fragen zu dem ganzen Thema an sich 😊
Daher wird es jetzt auch was länger - aber ich würde halt auch gerne etwas mehr verstehen 👍

Ich habe also jeden Port meines ASUS HERO VIII durchgecheckt und die folgende Übersicht erstellt (siehe Screenshot ASUS_Hero_VIII.jpeg)
Auf Basis dieser Erkenntnisse dann die *SSDT-USB-Maximus-VIII-Hero.dsl* gebastelt (siehe Anlage) - die dann ja nach dem compilen zu einer SSDT-USB.aml wird und in den Ordner **///Volumes/EFI-SYS/EFI/CLOVER/ACPI/patched/** kommt.
Danach setze ich in der config.plist via CloverConfigurator die excludes mit dem Boot-Arg --> `uia_exclude=HSXX;HSXX;HSXX`
Wobei ich mindestens so viele excluden muss, dass die Gesamtzahl aller Ports ≤ 15 ist
Ich würde daher auf Grund meiner Portprüfung folgende excluden - insgesamt sind es 22 Ports (inkl. USR1/2).
Somit muss ich mindestens 7 excluden, kann aber auf Grund eines komplett nicht angeschlossenen USB3_Headers (HS01/HS02/SS01/SS02) sogar 8 excluden.

Sprich das Boot-Arg wäre in meinem Fall dann =
`uia_exclude=USR1;USR2;HS01;HS02;HS13;HS14;SS01;SS02`

Hoffe der Ablauf ist schonmal korrekt 😊?

Jetzt mehr zum Inhalt der SSDT-USB-Maximus-VIII-Hero.dsl

- Ist die soweit ok? Sieht ggf. jmd. einen Fehler?
Lt. Internet werden die USB1 und USB2 erst gar nicht in der SSDT definiert - ist das korrekt?
Was ist mit den Ports SS07;SS08;SS09;SS10? Laut ioReg sind die vorhanden? Sind diese durch das Weglassen in der SSDT somit automatisch "excluded"? Wüsste gar nicht welche physischen Ports das wären - das Bord hat "nur"
4x USB3 direkt auf Mainboard = HS01; HS02; HS03; HS04; SS01; SS02; SS03; SS04
8x USB2 (4 auf Mainboard / 4 Rückseite) = HS07; HS08; HS09; HS10; HS11; HS12; HS13; HS14
2x USB3 (Rückseite - AsMedia-Controller) = HS05; HS06; SS05; SS06
2x USB3.1 (Rückseite - AsMediaController) = AppleUSB30XHCIPort@00100000;
AppleUSB30XHCIPort@00200000; AppleUSB30XHCIPort@00300000;
AppleUSB30XHCIPort@00400000
Somit kann es amS. gar keine SS07-SS10 geben - woher sollten die 4 Ports noch kommen?
- Was ist mit dem Port HS11 - dieser ist Teil des USB2-Headers auf dem Mainbord. Der andere Teil des Headers ist der HS12.
Wie man sieht, ist dort meine Broadcom WLAN/BT PCI-E Karte dran. Lt. IOREG nutzt diese aber nur den HS12 - sprich ich kann den HS11 eigentlich auch excluden? Und ist die Definition des Ports als "0" auch korrekt?
Vielleicht doof gefragt, aber mich verwirrt halt die "1-Stecker aber zwei Ports-Geschichte"



Und jetzt noch ein paar generell Themen

Wie man im Bild sieht, habe ich noch zwei weitere USB-Ports - diese hängen an einem USB3.1 Controller von ASMedia (Rückseite Mainboard)

Diese kann ich aktuell mit der USBInjectall auch nutzen - jedenfalls den Typ-A - zu Typ-C kann ich nichts sagen, da ich kein USB-C Device fürs Testing habe.

Was mich aber jetzt interessiert ist folgendes

- Zählen diese Ports zu den o.a. 15 dazu? Ich nehme an, dass **JA**, somit müsste ich die ja auch excluden, damit ich nicht über 15 Ports habe.
- Diese Ports werden aber NICHT als HS oder SS erkannt - sondern als "AppleUSB30XHCIPort@00100000" (USB3-Typ-A-Device) bzw "AppleUSB20XHCIPort@00300000" (USB2-Typ-A-Device).
Zudem gibt es (vermutlich) für den Typ-C dann noch "AppleUSB30XHCIPort@00200000" bzw "AppleUSB20XHCIPort@00400000" - als "nicht erkannt" im IOReg-Screenshot1 ersichtlich.

Ergeben sich also zwei neue Fragen daraus:

- Wie definiere ich diese in der SSDT-USB?
- Wie exclude ich diese im Boot-Arg? Brauche beide aktuell nicht!

Schon mal herzlichen Dank für alle, die mich hier unterstützen können. Natürlich habe ich auch einen aktuellen ioREG angehängen.

Ggf. ist es ja auch für andere ASUS HERO VIII User interessant - denn (wenn ich das richtig verstehe) kann man sich dann einfach die SSDT kopieren und via Boot-Args auf seine eigenen Anschlüsse simple anpassen.

Dazu habe ich ja mit dem Schaubild schon eine schöne Übersicht gemacht, die das Anpassen somit deutlich verkürzt.

Schönen Abend
Maddeen

Beitrag von „kuckkuck“ vom 25. September 2018, 23:03

[Zitat von Maddeen](#)

Danach setze ich in der config.plist via CloverConfigurator die excludes mit dem Boot-Arg --> uia_exclude=

Jeder Port, der in der SSDT nicht aufgezählt wird, wird automatisch excluded. SSDT+bootarg ist nicht nötig.

[Zitat von Maddeen](#)

Ist die soweit ok? Sieht ggf. jmd. einen Fehler?

Lt. Internet werden die USB1 und USB2 erst gar nicht in der SSDT definiert - ist das korrekt?

Sieht soweit ganz gut aus. USB1+2 kannst du raus lassen.

Zitat von Maddeen

Lt. iOREG nutzt diese aber nur den HS12 - sprich ich kann den HS11 eigentlich auch excluden? Und ist die Definition des Ports als "0" auch korrekt?

HS11 brauchst du dann in der SSDT nicht zu definieren. Ansonsten muss der Port an dem intern was hängt (die Wifi Karte) als 0xff/255 definiert werden.

Zitat von Maddeen

Zählen diese Ports zu den o.a. 15 dazu?

Nein, denn sie laufen über einen anderen Controller. Das 15 Port Limit gilt immer nur für einen Controller, nicht für alle zusammen.

Beitrag von „Maddeen“ vom 25. September 2018, 23:28

Danke [@kuckkuck](#) für die Erklärungen.

Zitat

Jeder Port, der in der SSDT nicht aufgezählt wird, wird automatisch excluded.
SSDT+bootarg ist nicht nötig.

Ok, aber es macht mich flexibler oder?

Sprich bei einer volldefinierten SSDT (also alle HS01-HS14 und SS01-SS06) und einem Exclude der nicht benötigten Ports kann ich bei technischen Problemen mit einem Port (z.B Defekt) einfach das Exclude ändern und alles ist in Butter.

Während ich bei einer Nichtdefinition des Ports diese in der dsl bzw. aml komplett neu schreiben müsste? Korrekt?

Kann ich dann eigentlich die USBInjectAll.kext löschen? Müsste ich ja, weil die ja durch die SSDT überschrieben wird, korrekt? 😊

Dank dir nochmal!

Beitrag von „kuckkuck“ vom 25. September 2018, 23:45

Wenn es dir leichter fällt das uia_exclude zu bearbeiten und nicht die SSDT, dann könnte man sagen du bist mit deiner Lösung flexibler...

Zitat von Maddeen

Kann ich dann eigentlich die USBInjectAll.kext löschen? Müsste ich ja, weil die ja durch die SSDT überschrieben wird, korrekt?

Nein, die SSDT und das uia_exclude Argument geben lediglich Informationen an USBInjectAll, die daraufhin von der Kext verarbeitet werden. Ohne USBInjectAll macht die SSDT nichts. Wenn du eine Lösung ohne USBInjectAll willst, gibt es da andere Möglichkeiten, die eigene Kext schreiben, oder ACPI (DSDT) anpassen.

Beitrag von „Maddeen“ vom 26. September 2018, 00:37

Super, dank dir.

Dann werde ich das morgen mal testen und Feedback geben.

GN8

UPDATE: Läuft einwandfrei! Endlich bin ich diese nervige "Nachpatches" los!! Vielen Dank noch mal fürs drüber gucken, [@kuckkuck](#)

Beitrag von „kuckkuck“ vom 26. September 2018, 13:33

Kein Problem, und freut mich 👍

Beitrag von „pgr69“ vom 2. Oktober 2018, 10:29

Hallo kuckkuck

Du scheinst dich ja gut mit der Materie auszukennen.

Deshalb würde ich gerne ein paar grundsätzliche Dinge fragen.

- Wenn ich die mit Clover extrahierte DSDT.aml unverändert in den patched-Ordner(EFI/CLOVER/ACPI/patched) lege - habe ich doch erstmal keine Veränderung - oder?
- Muss dem Clover eigentlich irgendwie gesagt werden das ich dort eine DSDT.aml hinterlegt habe?
- Wenn das soweit klappt könnte ich doch dann meine DSDT.aml nach und nach anpassen.
- Wenn ich die DSDT.aml compiliere.....entsteht daraus eine andere Datei (DSTD.???) und wenn ja wo finde ich diese?

Wenn ich es richtig verstanden habe könnte ich dann in dieser DSDT.aml die Patches vornehmen, die auch im Clover gemacht werden wie z.B. GFX0 zu IGPU umbenennen usw.

Würde mich über eine Antwort freuen.

Ig
Peter

Beitrag von „kuckkuck“ vom 2. Oktober 2018, 16:48

[Zitat von pgr69](#)

Wenn ich die mit Clover extrahierte DSDT.aml unverändert in den patched-Ordner(EFI/CLOVER/ACPI/patched) lege - habe ich doch erstmal keine Veränderung - oder?

Kommt drauf an wie genau du jetzt haben willst. Tendenziell hast du keine Veränderung, wenn man auf technischer Ebene komplett ins Detail geht gäbe es da schon 2-3 Nachteile, die entstehen würden, wenn man eine unveränderte DSDT nach ACPI/patched legt.

[Zitat von pgr69](#)

Muss dem Clover eigentlich irgendwie gesagt werden das ich dort eine DSDT.aml hinterlegt habe?

Alles was in ACPI/patched liegt, wird auch injected...

[Zitat von pgr69](#)

Wenn das soweit klappt könnte ich doch dann meine DSDT.aml nach und nach anpassen.

Kannst du natürlich machen, wenn du dich nach und nach einarbeiten willst.

[Zitat von pgr69](#)

Wenn ich die DSDT.aml compiliere.....entsteht daraus eine andere Datei (DSTD.???)
und wenn ja wo finde ich diese?

Eine .AML ist bereits kompiliert. Decompiled wäre es DSDT.dsl. Um sie zu aml zu kompilieren, müsstest du in MaciASL Speichern unter wählen und sie als Assemble (ASL) abspeichern.

[Zitat von pgr69](#)

Patches vornehmen, die auch im Clover gemacht werden wie z.B. GFX0 zu IGPU umbenennen usw.

Im Prinzip ja. Wo du aber konkret Renames ansprichst ist es wichtig zu wissen, dass es mehrere ACPI Tabellen gibt, zB wird die DSDT durch SSDTs erweitert. Renames sollten also auch in allen Tables gemacht werden. Clover macht das automatisch und macht Renames somit direkt im ganzen ACPI. DSDT patchen lohnt sich vorallem, wenn man Code erweitern oder kürzen will. Manche patches sind außerdem auch nur über DSDT möglich.

Beitrag von „al6042“ vom 2. Oktober 2018, 17:02

„Alles was in ACPI/patched liegt, wird auch injected...“
Das ist nicht ganz korrekt.

Die DSDT muss in der config.plist unter „Acpi“ mit genauem Namen eingetragen werden, ansonsten nimmt die vorliegende DSDT.aml, wenn eine drinnen liegt.

Somit kann man mehrere DSDTs ablegen.

Bei SSDTs wiederum werden immer alle reingepumpt und man kann nur Einfluss auf die Reihenfolge nehmen.

Beitrag von „kuckkuck“ vom 2. Oktober 2018, 17:16

Liegt nur eine DSDT.aml in ACPI/patched, muss sie nicht eingetragen werden um injected zu werden.

Gut, wenn man jetzt mehrere DSDTs drinnenliegen hat, kann man eine davon in der config definieren die injected wird. Es können ja nicht mehrere Tables mit der gleichen TableID gleichzeitig injected werden, eine TableID kann ja im ACPI auch nur ein mal vorhanden sein. Gleiches gilt also auch prinzipiell für SSDTs, hat man mehrere SSDTs mit der gleichen TabelID in Acpi/patched, wird auch nicht alles injected...