

Erledigt

USB XHC ohne USBInjectAll.kext oder USB-Portlimit Patch zum Laufen bringen / Diskussionen

Beitrag von „anonymous_writer“ vom 6. Dezember 2018, 15:02

Hallo Leser,

habe versucht hier eine Anleitung zu schreiben alle XHC USB Ports auf einem Rechner zu aktivieren ohne USBInjectAll.kext und USB-Portlimit Patch.

Das ganze ist noch in der Testphase. Die Anleitung ist auch noch ausbaufähig.

Trotzdem hier der Hinweis auf diese Anleitung falls sich jemand am Verbessern der Anleitung oder am Testen dieser beteiligen möchte.

Bitte alles dazu in diesen Thread schreiben.

[USB XHC ohne USBInjectAll.kext oder USB-Portlimit Patch zum Laufen bringen](#)

Beitrag von „pgr69“ vom 12. Oktober 2020, 22:16

wow - hat noch nie einer was dazu gefragt ???

Ich bin auf das Thema gestoßen, weil ich seit einiger Zeit Probleme mit meinem USB's habe.

Derzeit habe ich das Gigabyte X299 Designare EX laufen.

In der Front des Gehäuses habe ich 2 USB3 und 2 USB2 Anschlüsse - ich bin der Meinung die haben immer alle funktioniert.

In einem der USB2 steckt mein Receiver der Logitech Maus.

Wenn ich jetzt einen USB3-Stick in eine oder USB3-Anschlüsse stecke wird er nicht erkannt.

Stecke ich ihn in den USB2-Anschluss wird er erkannt und im IORegExploer erscheint ein Eintrag wie "SanDisk 3.2 Gen1@14a00000"

Dieser Eintrag erscheint unter "AppleUSBXHCI Root Hub Simulation@14000000"

Darunter erscheint auch "USB2.0 Hub@14800000" wo mir dann alle USB 2.0 Geräte angezeigt werden wie meine Tastatur und die Webcam die Rückseitig am Board angeklemt sind.

Seltsamerweise kann ich vorne meinen Maus-Receiver einstecken wo ich will er erscheint immer unter dem XHCI-Zweig.

Da ich im Hinblick auf meine neuen Build gerne mal wieder was dazu lernen möchte würde ich gerne ein paar Fragen stellen.

Die dieser Diskussion zugrunde liegende Anleitung ist ganz nett aber ich bräuchte ein paar Erläuterungen dazu.

z.B.

Code

```
1. Method (GUPC, 1, Serialized) // Method XUS3
2. {
3. Name (PCKG, Package (0x04)
4. {
5. 0xFF,
6. 0x03,
7. Zero,
8. Zero
9. })
10. PCKG [Zero] = Arg0
11. Return (PCKG)
12. }
```

Alles anzeigen

Meine Programmierzeiten sind schon sehr lange her.

Meine Vermutungen (bitte korrigieren):

- Ich würde mal sagen diese Methode ist sowas wie eine Funktion nur eben Objektorientiert mit dem Namen GUPC
- Alles hinter // ist wohl Kommentar
- die 1 könnte bedeuten es gibt ein Argument das beim Aufruf übergeben werden kann.
- serialized.....keine Ahnung
- da drinnen das PCKG schein ein Array mit 4 Werten (0x04) zu sein die in diesem Falle 0xFF, 0x03, Zero, Zero zu sein scheinen wobei Zero nach einem Platzhalter aussieht
- der von mir vermutete Platzhalter Zero wird mit PCKG [Zero] = Arg0 durch den Wert ersetzt der der Methode als Argument übergeben wurde was aber nicht wirklich stimmen kann das bei den anderen Methoden PCKG[One] zum Einsatz kommt aber es keinen Platzhalter "One" gibt sondern dort auch wieder "Zero" steht.

Also bitte erleuchtet mich.

Offensichtlich haben die 4 Hex-Werte in diesem PCKG Abschnitt etwas damit zu tun ob ein Port ein USB3 ein USB2 ist und ob er intern oder extern ist oder ob er gar ganz ausgeschaltet ist.

Aber welcher wert an welchem der 4 Plätze hat welche Bedeutung?


In der Beispiele DSL erkennt man ja dass mit diesen Methoden die USB-Anschlüsse eingestellt werden (USB2 UBS3 intern extern OFF). Allerdings stellt sich mir die Frage woher weiß ich welche von den im Beispiel benutzen USBs gibt es bei mir und wie heißen sie bei mir. Im IOREGExplorer sehe ich ja nur die oben genannten Werte. ich brauche aber sowas wie SB.PCI0.XHC.RHUB.HS01

Fragen über Fragen aber ich glaube die Cracks unter Euch werden mir helfen können.

Ig

Peter

Beitrag von „Raptortosh“ vom 12. Oktober 2020, 22:33

Liege ich falsch, oder ist das nur eine alte Variante hiervon?  [Patching in Deutsch](#)

[Hackintool Anleitung USB-](#)

Beitrag von „apfelnico“ vom 12. Oktober 2020, 23:07

[pgr69](#)

Bei mir sehen die Ports so aus, in einer eigenen SSDT beschrieben:

Code

1. Scope (_SB.PC00.XHCI.RHUB.HS01)
2. {
3. Name (_UPC, Package (0x04) // _UPC: USB Port Capabilities
4. {
5. 0xFF,
6. 0x03,
7. Zero,
8. Zero
9. })
10. }

Dabei steht in dem Package der erste Wert für den "Status" des Devices, lässt sich auch extra mit der Methode _STA festlegen. "0xFF" ist komplett an, "0x00" komplett aus, es gibt noch feinere Abstufungen dazwischen, die hier aber nicht benötigt werden.

Der zweite Parameter ist die Beschreibung um was für einen Port es sich handelt. "0x00" (beziehungsweise "Zero") bedeutet USB2, wogegen "0x03" USB3 festlegt (beides klassische Type A Buchse). Dann gibt es noch "0xFF" für einen internen USB2-Port (es gibt offenbar KEINEN "internen" USB3). "Intern" meint damit nicht unbedingt die Ports, die direkt auf einem Mainboard stecken. Denn diese können ja auch herausgeführt werden an "Gehäuseports". Diese sind somit je nach USB2 oder USB3 zu bezeichnen. Vielmehr bedeutet "Intern" zum einen für direkt auf dem Mainboard verdrahtete Geräte (herstellereigenes Bluetooth, RGB-Lichtsteuergedöhns etc), oder auch selbst eingesetzte interne Bluetoothkarte mit USB2-Uplink.

Wichtig: auch die USB2-Ports innerhalb eines USB3-Ports (der ja immer aus diesen zwei Komponenten besteht), wird als USB3, also "0x03" deklariert. Lediglich einzelne exklusive USB2-Ports als "0x00". Dann gibt es noch USB-C (USB3.1/USB3.2). Hier gibt es wiederum zwei Varianten: "0x9" für diejenigen Ports, an denen man einen USB-C Gerät egal wie rum anstecken kann (der Stecker ist "Verdrehsicher", heißt er kann um 180° ebenfalls eingesteckt werden). Eine weitere Variante erlaubt letztendlich für den Nutzer das gleiche Verhalten, nur wird intern je nach Steckerdrehung ein zweiter eigener Port genutzt – in dem Fall ist es "0x0A". Letzteres findet man oft bei den "Type-E" Buchsen auf dem Mainboard (für Gehäuseports USB-C). Erkennen kann man das wunderbar mittels IORegistryExplorer. Auch hier gilt, die zugehörigen USB2-Ports – sofern vorhanden – werden genau so deklariert.

Dann gibt es noch ein exotisches Sonderformat von USB-C, welches man aber nicht unbedingt am Computer findet: USB2 exklusiv über USB-C. Dieses wird dann als "0x08" deklariert.

Die unteren beiden "Zero" haben sich mir auch noch nicht erschlossen.

Warum gibt es überhaupt nur 15 Ports (je Controller) bei Apple? Nun, direkt in Apples Geräte sind ja wesentlich weniger eingebaut, also ist "15" als Obergrenze mehr als ausreichend gewesen. Warum aber nun gerade "15", und nicht etwa "37"? 😊

Die Ports werden, unabhängig von ihrer tatsächlichen Adresse, von "1 bis maximal 15" in macOS durchnummeriert. Also von "0x01" bis max. 0x0F". HEXADEZIMAL halt, sind eben 15 ...

Beitrag von „kuckkuck“ vom 12. Oktober 2020, 23:56

Sehr schön erklärt 😊

[Zitat von apfelnico](#)

Die unteren beiden "Zero" haben sich mir auch noch nicht erschlossen.

Reserviert für weitere mögliche Spezifikationen in der Zukunft 😊

Hier aus dem ACPI Spec die Return Package Values:

1. Connectable If this value is non-zero, then the port is connectable. If this value is zero, then the port is not connectable.

	Connector Typ:
	0x00: Type 'A' connector
	0x01: Mini-AB connector
	0x02: ExpressCard
	0x03: USB 3 Standard-A connector
	0x04: USB 3 Standard-B connector
2. Typ	0x05: USB 3 Micro-B connector
	0x06: USB 3 Micro-AB connector
	0x07: USB 3 Power-B connector
	0x08: Type C connector - USB2-only
	0x09: Type C connector - USB2 and SS with Switch
	0x0A: Type C connector - USB2 and SS without Switch
	0x0B - 0xFE: Reserved
	0xFF: Proprietary connector
3. Reserved0	This value is reserved for future use and must be zero.
4. Reserved1	This value is reserved for future use and must be zero.

Beitrag von „pgr69“ vom 13. Oktober 2020, 07:48

Danke Ihr Beiden das ist doch schon mal eine tolle Übersicht.

Aber nochmal zur Frage IOREgExplorer

Dort sehe ich beim eistecken von USB Geräten wie die auftauchen. Beim entfernen wieder der beim einstecken generierte Eintrag ROT und DURCHGESTRICHEN+

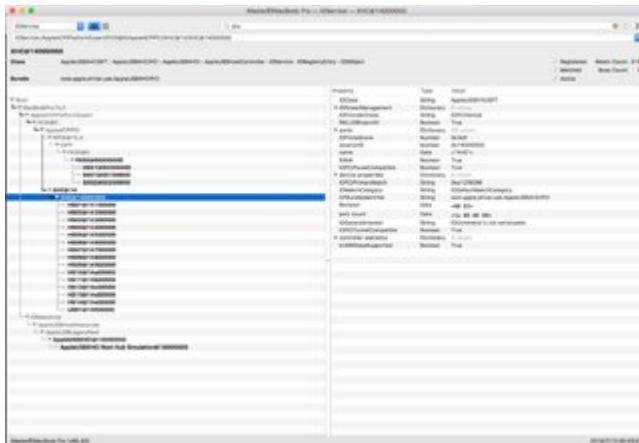
Woher aber weiß ich passend dazu wie die Dinger heißen.....also sowas wie
_SB.PC00.XHCI.RHUB.HS01

Andere Frage.....wenn die USBInjectAll usw Kexte raus schmeisse und dafür eine eigene SSDT verwenden will.....muss die einen bestimmten Namen haben ? Abgelegt wird sie ja im EFI Ordner irgendwo unter PATCHED ?

Ig

Beitrag von „anonymous_writer“ vom 13. Oktober 2020, 09:51

HS01 solltest du direkt beim roten Eintrag sehen. Sieht das nicht so Ähnlich aus wie in diesem Bild?



Der Name der Datei ist egal. Wichtig ist das die Datei im richtigen Format abgespeichert wird, also als *.aml.

Die Anleitung ist die ursprüngliche Idee wie man seine USB-Anschlüsse richtig patcht War noch vor der Zeit vom Hackintool.

Hackintool macht da einiges einfacher. Aber angesichts dessen das die Entwicklung vom Hackintool gerade stockt kann es durchaus sein das diese Variante bald wieder sehr aktuell wird.

Beitrag von „DSM2“ vom 13. Oktober 2020, 11:56

[pgr69](#) warum gehst du das ganze so umständlich an? Mach das ganze doch via Hackintool.

Beitrag von „pgr69“ vom 13. Oktober 2020, 12:37

hoppla.....das kenn ich gar nicht

Beitrag von „DSM2“ vom 13. Oktober 2020, 12:41

[pgr69](#) Check mal meinen X299 Guide, USB wird darin ebenfalls angesprochen:

[X299 Tutorial - ASUS WS X299 Sage/10G](#)

Das du ein Designare nutzt ist irrelevant 😊

Beitrag von „pgr69“ vom 13. Oktober 2020, 14:44

Hi DSM2

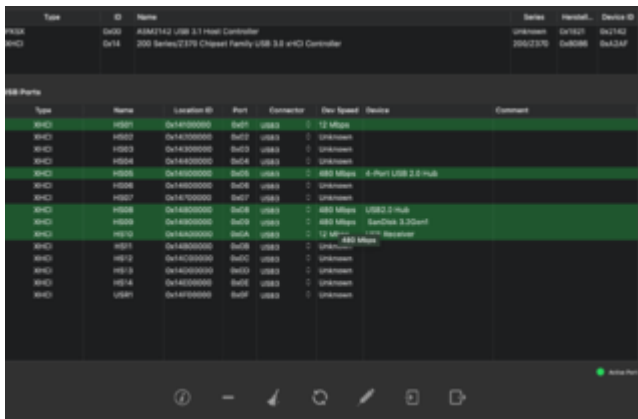
Den Guide werde ich mir die Tage eh ansehen müssen weil ich damit beginne die Kiste zusammen zu bauen.

lg

Habe jetzt mal das Hackintool geladen und ausgeführt.

Im Bild der HS1 funktioniert nur mit USB2-Geräten. Ein USB3-Stick funktioniert darin GAR NICHT also nicht das er nur mit USB2 laufen würde...er wird gar nicht erst angezeigt.

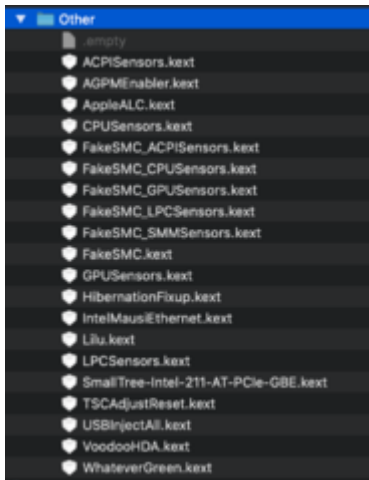
Schaut das Bild.....in HS1 steckt der gleiche Stick wie in HS9. HS9 ist einer von 2 USB2-Ports in der Front meines Rechner. In einem davon steckt immer mein Receiver der Maus.



Stecke ich in HS1 ein USB2 Gerat dann geht das dort.

Grrrrrrrr....hilfe

Das sind meine kexte....



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

[Zitat von pgr69](#)

Im Bild der HS1 funktioniert nur mit USB2-Geräten. Ein USB3-Stick funktioniert darin GAR NICHT also nicht das er nur mit USB2 laufen würde...er wird gar nicht erst angezeigt.

Völlig normal bei deinem jetzigen Stand. HS1 funktioniert grundsätzlich nur mit USB2-Geräten, ist ja auch ein USB2-Port. Bevor hier Begriffsverwirrung aufkommt, Port im Sinne das der an deinem Controller hängt und technisch so angesprochen wird. NICHT im Sinne von mechanischem Port, an dem du dein Gerät einsteckst. Denn dieser mechanische USB3-Port enthält zwei logische Ports, in diesem Fall "HS01" für USB2, und gleichzeitig "SS01" für USB3. Die SS0x-Ports siehst du derzeit nicht im Hackintool, weil eben das 15PortLimit von macOS greift. Das siehst du sehr gut an der "Location-ID". Der XHCi-Controller hat die Adresse "0x14", die Ports werden laut ACPI-Tabelle in Hex von "01" bis "0F" hochgezählt.

In der ACPI-Tabelle sind alle Ports aufgeführt, die an dem Controller möglich sind. Diese lauten:

HS01

HS02

...

HS14

USR1

USR2

SS01

SS02

...

SS10

wobei SS07-SS10 selten genutzt sind. Die USR1/2 können auch entfernt werden, auch diesen sind in der Regel auf dieser Plattform keine physischen Ports zugeordnet. Damit du vorerst mehr als 15 Ports in Hackintool sehen kannst, empfehle ich ein sogenannten PortLimitPatch anzuwenden. Damit wird das PortLimit deutlich erhöht (nicht aufgehoben), und du kannst alle

Ports durchtesten um zu entscheiden, auf welche du verzichten möchtest. Dauerhaft den Patch zu nutzen, um sich diese Arbeit hier zu sparen, kann ich nicht empfehlen. Zumal der Patch auch mit "Nachkomma-Updates" gern mal nicht funktioniert und erst wieder neu gefunden werden muss.

HS01 bis HS06 sowie SS01 bis SS06 sind jeweils sechs USB3-Ports mit USB2 enthalten, diese werden als USB3 deklariert (auch die USB2). HS07 bis HS14 sind reine USB2-Ports, wovon einige auch interne sind.

Ich nehme an, dass HS01/SS01 bis HS04/SS04 die auf dem Board vorhandenen Anschlüsse für die Gehäusefront sind, die vier blauen hinten am I/O-Shield könnten letztendlich nur HS05/SS05 sowie HS06/SS06 über internen Realtek USB-Hub sein. Dann gibt es auf dem Board noch einen Type-E ASMedia-Controller (PXSX) für Gehäuse USB-C, der spielt erstmal keine Rolle (vielleicht diesen vorerst im BIOS ausschalten, damit du keine Ports beim ausmisten verwechselst).

Edit:

Dein Kext-Ordner sieht ja furchtbar aus. Erst mal alle "Sensors" raus (kann später gezielt wenn eine einzelne benötigt wird wieder rein), "voodoo" ebenso, "Hibernation" und "Usbinjectall" kann auch weg. Die USB-Controller benötigen letztere nicht.

Beitrag von „pgr69“ vom 14. Oktober 2020, 14:34

[apfelnico](#) Okay - werde mal heute abend aufräumen

Nochmal zu den USB. Der HS1 den ich ja identifizieren kann indem ich ein USB2-Stick einstecke und dann sehe dass dieses Gerät im Hackintool angezeigt wird ist physikalisch auf meinem BOARD ein USB3-Anschluss. Also dieser breite mit den "feinen" PINS. Ist auf dem Board auch gelabelt als USB3_1

Die hinteren Ports reagieren auch ALLE nicht auf USB3-Sticks

Aber ich werde heute Abend mal wieder testen.

Ich würde auch gerne auf das USBinjectAll verzichten und lieber die SSDT-Methode verwenden sobald ich weiss welche Ports was sind

Ig und Danke

Beitrag von „apfelnico“ vom 14. Oktober 2020, 16:33

[pgr69](#)

Nichts anderes schrieb ich dir. Schau auf dein verlinkten Bild von Hackintool. Du siehst 15 Ports, mehr stellt macOS an diesem Controller (XHCI) nicht zur Verfügung. Es sind alle theoretisch möglichen USB2-Ports aktiv und ein USB1. Hast du den zweiten physischen Port auf dem Mainboard (USB3_2) genutzt? Wenn nicht, kannst du hier schon mal vier Ports deaktivieren, das wären HS03/HS04/SS03/SS04. Wenn du dann noch die beiden USB1/2 deaktivierst, musst du nur noch einen USB2 (HS07-HS14) opfern, dann bist du im Limit und kannst alle weiteren nutzen, USB3 geht dann auch komplett.



Beitrag von „pgr69“ vom 14. Oktober 2020, 18:39

[apfelnico](#)

Hi und danke. Okay ich verstehe - meine SS Ports sind die die derzeit hinter den jetzigen 15 liegen und deshalb im System nicht vorhanden sind. Wenn ich jetzt die besagten Ports abschalte rücken quasi von hinten die SS Ports nach ? In den SS-Ports gehen auch USB2-Geräte oder nur USB3-Geräte ?

Ich schalte die Ports ab in dem ich sie im Hackintool lösche und dann dort die SSDT exportiere. Das sind 3 Dateien

Die .aml kommt ja nach CLOVER/ACPI/PATCHED, die USBPorts.kext kommt in meinen Kext-Ordner nehme ich mal an.

Die .dsl ist die Quelldatei für die .aml.....nur zum reinschauen ?

Was ist mit dem USBInjectAll....soll der jetzt raus wegen der Port Ermittlung ? Denn im X299 Guide steht drin das die rein soll.....

Grüße

Beitrag von „apfelnico“ vom 14. Oktober 2020, 20:35

Die USBInjectAll wird überhaupt nicht benötigt. Alle Controller laufen nativ. Nur der XHCI hat oft mehr Ports als macOS zulässt. Die neue Kext kommt in patched, mehr nicht. Die SSDT wäre für USBInjectAll, welche du ja auch nicht benötigst.

[Zitat von pgr69](#)

ich verstehe - meine SS Ports sind die die derzeit hinter den jetzigen 15 liegen und deshalb im System nicht vorhanden sind. Wenn ich jetzt die besagten Ports abschalte

rücken quasi von hinten die SS Ports nach ?

Richtig.

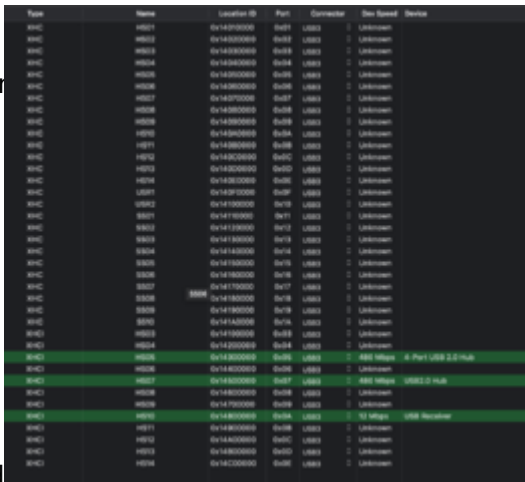
Sinnvoll wäre es aber gewesen, vorerst ein PortLimitPatch anzuwenden (temporär), damit du _ALLE_ Ports auf einmal siehst und mit einem Durchgang entscheiden kannst, welche Ports du rauswerfen willst.

Beitrag von „pgr69“ vom 14. Oktober 2020, 20:54

Die neue KEXT kommt in patched ? Kommt die nicht unter Efi/clover/kexts/other

Die generierte SSDT habe ich nach efi/clover/acpi/patched gelegt aber du sagst ja die brauche ich nicht

Die Ports die ich entfernt habe (HS01,HS02, USB1) fehlen jetzt auch im Hackintool. Allerdings ist nichts nachgerückt.



Type	Name	Location ID	Port	Connector	Bus Speed	Status
HS02	HS01	0x14000000	HS01	USB3	1	Lebendigen
HS02	HS02	0x14000000	HS02	USB3	1	Lebendigen
HS02	HS03	0x14000000	HS03	USB3	1	Lebendigen
HS02	HS04	0x14000000	HS04	USB3	1	Lebendigen
HS02	HS05	0x14000000	HS05	USB3	1	Lebendigen
HS02	HS06	0x14000000	HS06	USB3	1	Lebendigen
HS02	HS07	0x14000000	HS07	USB3	1	Lebendigen
HS02	HS08	0x14000000	HS08	USB3	1	Lebendigen
HS02	HS09	0x14000000	HS09	USB3	1	Lebendigen
HS02	HS10	0x14000000	HS10	USB3	1	Lebendigen
HS02	HS11	0x14000000	HS11	USB3	1	Lebendigen
HS02	HS12	0x14000000	HS12	USB3	1	Lebendigen
HS02	HS13	0x14000000	HS13	USB3	1	Lebendigen
HS02	HS14	0x14000000	HS14	USB3	1	Lebendigen
HS02	HS15	0x14000000	HS15	USB3	1	Lebendigen
HS02	HS16	0x14000000	HS16	USB3	1	Lebendigen
HS02	HS17	0x14000000	HS17	USB3	1	Lebendigen
HS02	HS18	0x14000000	HS18	USB3	1	Lebendigen
HS02	HS19	0x14000000	HS19	USB3	1	Lebendigen
HS02	HS20	0x14000000	HS20	USB3	1	Lebendigen
HS02	HS21	0x14000000	HS21	USB3	1	Lebendigen
HS02	HS22	0x14000000	HS22	USB3	1	Lebendigen
HS02	HS23	0x14000000	HS23	USB3	1	Lebendigen
HS02	HS24	0x14000000	HS24	USB3	1	Lebendigen
HS02	HS25	0x14000000	HS25	USB3	1	Lebendigen
HS02	HS26	0x14000000	HS26	USB3	1	Lebendigen
HS02	HS27	0x14000000	HS27	USB3	1	Lebendigen
HS02	HS28	0x14000000	HS28	USB3	1	Lebendigen
HS02	HS29	0x14000000	HS29	USB3	1	Lebendigen
HS02	HS30	0x14000000	HS30	USB3	1	Lebendigen
HS02	HS31	0x14000000	HS31	USB3	1	Lebendigen
HS02	HS32	0x14000000	HS32	USB3	1	Lebendigen
HS02	HS33	0x14000000	HS33	USB3	1	Lebendigen
HS02	HS34	0x14000000	HS34	USB3	1	Lebendigen
HS02	HS35	0x14000000	HS35	USB3	1	Lebendigen
HS02	HS36	0x14000000	HS36	USB3	1	Lebendigen
HS02	HS37	0x14000000	HS37	USB3	1	Lebendigen
HS02	HS38	0x14000000	HS38	USB3	1	Lebendigen
HS02	HS39	0x14000000	HS39	USB3	1	Lebendigen
HS02	HS40	0x14000000	HS40	USB3	1	Lebendigen
HS02	HS41	0x14000000	HS41	USB3	1	Lebendigen
HS02	HS42	0x14000000	HS42	USB3	1	Lebendigen
HS02	HS43	0x14000000	HS43	USB3	1	Lebendigen
HS02	HS44	0x14000000	HS44	USB3	1	Lebendigen
HS02	HS45	0x14000000	HS45	USB3	1	Lebendigen
HS02	HS46	0x14000000	HS46	USB3	1	Lebendigen
HS02	HS47	0x14000000	HS47	USB3	1	Lebendigen
HS02	HS48	0x14000000	HS48	USB3	1	Lebendigen
HS02	HS49	0x14000000	HS49	USB3	1	Lebendigen
HS02	HS50	0x14000000	HS50	USB3	1	Lebendigen
HS02	HS51	0x14000000	HS51	USB3	1	Lebendigen
HS02	HS52	0x14000000	HS52	USB3	1	Lebendigen
HS02	HS53	0x14000000	HS53	USB3	1	Lebendigen
HS02	HS54	0x14000000	HS54	USB3	1	Lebendigen
HS02	HS55	0x14000000	HS55	USB3	1	Lebendigen
HS02	HS56	0x14000000	HS56	USB3	1	Lebendigen
HS02	HS57	0x14000000	HS57	USB3	1	Lebendigen
HS02	HS58	0x14000000	HS58	USB3	1	Lebendigen
HS02	HS59	0x14000000	HS59	USB3	1	Lebendigen
HS02	HS60	0x14000000	HS60	USB3	1	Lebendigen
HS02	HS61	0x14000000	HS61	USB3	1	Lebendigen
HS02	HS62	0x14000000	HS62	USB3	1	Lebendigen
HS02	HS63	0x14000000	HS63	USB3	1	Lebendigen
HS02	HS64	0x14000000	HS64	USB3	1	Lebendigen
HS02	HS65	0x14000000	HS65	USB3	1	Lebendigen
HS02	HS66	0x14000000	HS66	USB3	1	Lebendigen
HS02	HS67	0x14000000	HS67	USB3	1	Lebendigen
HS02	HS68	0x14000000	HS68	USB3	1	Lebendigen
HS02	HS69	0x14000000	HS69	USB3	1	Lebendigen
HS02	HS70	0x14000000	HS70	USB3	1	Lebendigen
HS02	HS71	0x14000000	HS71	USB3	1	Lebendigen
HS02	HS72	0x14000000	HS72	USB3	1	Lebendigen
HS02	HS73	0x14000000	HS73	USB3	1	Lebendigen
HS02	HS74	0x14000000	HS74	USB3	1	Lebendigen
HS02	HS75	0x14000000	HS75	USB3	1	Lebendigen
HS02	HS76	0x14000000	HS76	USB3	1	Lebendigen
HS02	HS77	0x14000000	HS77	USB3	1	Lebendigen
HS02	HS78	0x14000000	HS78	USB3	1	Lebendigen
HS02	HS79	0x14000000	HS79	USB3	1	Lebendigen
HS02	HS80	0x14000000	HS80	USB3	1	Lebendigen
HS02	HS81	0x14000000	HS81	USB3	1	Lebendigen
HS02	HS82	0x14000000	HS82	USB3	1	Lebendigen
HS02	HS83	0x14000000	HS83	USB3	1	Lebendigen
HS02	HS84	0x14000000	HS84	USB3	1	Lebendigen
HS02	HS85	0x14000000	HS85	USB3	1	Lebendigen
HS02	HS86	0x14000000	HS86	USB3	1	Lebendigen
HS02	HS87	0x14000000	HS87	USB3	1	Lebendigen
HS02	HS88	0x14000000	HS88	USB3	1	Lebendigen
HS02	HS89	0x14000000	HS89	USB3	1	Lebendigen
HS02	HS90	0x14000000	HS90	USB3	1	Lebendigen
HS02	HS91	0x14000000	HS91	USB3	1	Lebendigen
HS02	HS92	0x14000000	HS92	USB3	1	Lebendigen
HS02	HS93	0x14000000	HS93	USB3	1	Lebendigen
HS02	HS94	0x14000000	HS94	USB3	1	Lebendigen
HS02	HS95	0x14000000	HS95	USB3	1	Lebendigen
HS02	HS96	0x14000000	HS96	USB3	1	Lebendigen
HS02	HS97	0x14000000	HS97	USB3	1	Lebendigen
HS02	HS98	0x14000000	HS98	USB3	1	Lebendigen
HS02	HS99	0x14000000	HS99	USB3	1	Lebendigen
HS02	HS100	0x14000000	HS100	USB3	1	Lebendigen

Wenn ich die Spritze (Inject) gehe bekomme ich die Liste aus dem

Bild

Was den LimitPatch angeht habe ich im Clover die aus dem folgenden Bild genommen - sind die richtig ?

com.apple.Iokit.IOSEHostFamily	83E30FD3 E341090F	83E33FD3 E341090F
com.apple.Iokit.IOSEHostFamily	83FB0FCF 870B0400 00	83FB3FCF 870B0400 00
com.apple.driver.usb.AppleUSBXHCI	4183FFCF DF839404 0000	4183FF3F DF839404 0000

Den USBInjectAll habe ich auch schon rausgenommen.

EDIT:

Jetzt wird es aus meiner Sicht lustig.

Stecke ich am Backpanel einen USB2 Stick ein erscheint der im Finder und auch IORegExplorer.

Allerdings nicht im Hackintool.....bzw. im Hackintool steht an der Stelle "4-Port USB2.0 HUB" und ist grün

Es ist der HS05 an Adresse 14300000

Ig und Danke

Beitrag von „Nightflyer“ vom 14. Oktober 2020, 20:57

Jeep, der kommt in Efi/clover/kexts/other

Beitrag von „apfelnico“ vom 14. Oktober 2020, 21:16

Ja natürlich. War nicht ganz bei der Sache und hatte an die SSDT gedacht, die nicht benötigt wird. 😊

[Zitat von pgr69](#)

Jetzt wird es aus meiner Sicht lustig.

Stecke ich am Backpanel einen USB2 Stick ein erscheint der im Finder und auch

IORegExplorer.

Allerdings nicht im Hackintool.....bzw. im Hackintool steht an der Stelle "4-Port USB2.0 HUB" und ist grün

Es ist der HS05 an Adresse 14300000

Irgendwie liest du nicht mit. Das habe ich dir bereits vorausgesagt und ist völlig normal. Du hast hinten vier USB3, diese sind teilweise an einem internen Hub angeschlossen (eigentlich sind es zwei, USB2 und USB3), die sich wiederum an HS05/SS05 befinden. Das steht in den technischen Unterlagen zu deinem Board, dass dort ein interner REALTEK-HUB benutzt wird ...

Zitat von apfelnico

Ich nehme an, dass HS01/SS01 bis HS04/SS04 die auf dem Board vorhandenen Anschlüsse für die Gehäusefront sind, die vier blauen hinten am I/O-Shield könnten letztendlich nur HS05/SS05 sowie HS06/SS06 über internen Realtek USB-Hub sein.

Wieso sind in deinem Hackintool nun XHCI sowie XHC zu sehen? Das ist Unfug.

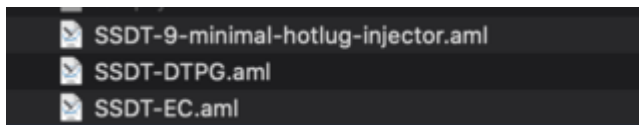
Beitrag von „pgr69“ vom 14. Oktober 2020, 21:21

das kommt wenn ich dieses Spritzensymbol drücke (inject)

stimmt....das mit dem internen HUB hast du oben geschrieben - SORRY.

So - das abschalten von Ports (siehe HS01 und HS02) funktioniert ja - aber wie bekomme ich dafür andere nachgerückt ?

In patched habe ich übrigens noch andere SSDTs liegen von denen ich nicht weiß wozu sie sind.



Beitrag von „apfelnico“ vom 14. Oktober 2020, 21:33

Noch mal von vorn. Du nutzt einfach einen für dein System passenden FUNKTIONIERENDEN PortLimitPatch.

DANN siehst du bei erneutem Starten des Rechners in Hackintool unter USB (gern auf "Aktualisieren" drücken, da hier gern der "letzte" Stand angezeigt wird) nun ALLE Ports. Es muss nichts nachrücken. DU siehst ALLE und es funktionieren ALLE.

Warum lässt man es dann nicht so? Weil der PortLimitPatch etwas macht, wofür macOS nicht vorgesehen ist und weil er gern im entscheidenden Moment versagt - wenn man ein neues Update von macOS einspielt, worin ein neuer PortLimitPatch erst noch gefunden werden muss.

Also klickst du nun die Ports weg, die du nicht benötigst. Sollte es der zweite interne Port vom Mainboard sein, gehen hier schon vier Ports weg (HS03/04, SS03/04). Dann klickst du USB1/2 weg, und wenn vorhanden, SS07-SS10. Dann sollte noch ein einziger USB2-Port weg (HS07-HS14), und du hast nur noch 15Ports. Du erstellst dir die Kext, packst die in kexte/others und

löscht wieder den PortLimitKext. Neustart, fertig.

Nochmal schreib ich's nich.

Edit:

Du hast also IRGENDEINEN EFI-Ordner und daher weißt du nicht, was das alles soll. Mir reicht's ...

Beitrag von „pgr69“ vom 14. Oktober 2020, 21:47

Hi

Ich habe Deine letzte Ausführung verstanden und kann das auch nachvollziehen wie ich vorgehen soll. Du musst es nicht nochmal schreiben.

Im BIOS gab es keine Option unter USB um den ASMEDIA abzustellen es sei denn der ist dort unter "USB Mass Storage Control" oder so ähnlich angegeben.

ich habe den EFI-Ordner seinerzeit mit Hilfe von hier erstellt. Ich weiß HEUTE nicht mehr wofür diese SSDTs waren.

Tut mir leid das ich nicht so "erfahren" bin wie du - ich lerne ja gerne dazu.

Ich habe den Portlimit-Patch aus dem Forum hier für Catalina im CloverConfigurator eingetragen.

Der sieht so aus

com.apple.iokit.IOUSBHostFamily	83FB0F0F	83FB3F0F	USB Port Limit P...	10.15.x
com.apple.driver.usb.AppleUSBXHCI	83F90F0F	83F93F0F	USB Port Limit P...	10.15.x

Dennoch sehe ich nur die HS-Einträge.

Ig

EDIT:

Habe die unbekanntes SSDTs rausgeschmissen - ich glaube die hatten was mit Thunderbolt zu tun. TB habe ich nicht also raus.

Die zuvor generierte USBPorts.kext habe ich auch entfernt sowie im BIOS diesen USB Mass Storage Driver abgeschaltet.

Ergebnis:

- Der Asmedia ist immer noch da - lässt sich wohl nicht abschalten.
- Die entfernten SSDTs hatten keine negativen Effekt den ich derzeit erkennen kann

UND

- jetzt endlich sehe ich ALLE meine USB-Ports jetzt werde ich die von apfelnico vorgeschlagen Ports löschen.

Werde gleich über das Ergebnis berichten

thx

EDIT2:

Nun endlich laufen meine USB korrekt großen DANK an APFELNICO

unnötige SSDTs entfernt, Kexte entrümpelt, USB-Ports sortiert

Und wieder viel dazu gelernt

DANKE

Es wird beim ASUS SAGE 10G Build nicht schaden.

Beitrag von „apfelnico“ vom 14. Oktober 2020, 22:48

Danke das du dich noch mal gemeldet hast, wollte nicht unhöflich sein, Sorry.

Beitrag von „pgr69“ vom 15. Oktober 2020, 09:27

Zitat von apfelnico

Danke das du dich noch mal gemeldet hast, wollte nicht unhöflich sein, Sorry.

Warum sollte ich mich nicht mehr melden? Mach Dir keinen Kopf - ich weis aus meinem Beruf wie das ist wenn man das Gefühl hat man redet gegen eine Wand.

Ich bin auf jeden Fall froh dass du es mir so genau erklärt hast - so lernt man dazu und zukünftig sollte das Thema kein Problem mehr darstellen.

Also nochmal DANKE

lg

Beitrag von „bluebyte“ vom 15. Oktober 2020, 22:13

[apfelnico](#) nur mal eine Frage am Rande. Ich habe ein Problem an meinem XHCI-Ports am Front-Panel.

Meine externe Platte MyBook von WD und externer Stromversorgung läuft da ohne Probleme. Meine SSD in der Icybox wird dort fortlaufend getrennt. Hackintool ist an dem Anschluss

dauernd am Blinken. Am Rear-Panel gibt es keine Probleme. Die SSD ist von Sandisk und zieht laut Typenschild 5Volt mit 1.5 Ampere.

Daher meine Frage: "Haben die Anschlüsse am Rear-Panel mehr "Saft?"

Viele Grüße von bluebyte

Beitrag von „apfelnico“ vom 15. Oktober 2020, 22:56

[bluebyte](#)

An sich nicht. Sollten gleichberechtigt sein. Hast du ein "USBX" per SSDT integriert?

Code

```
1. Scope (\_SB)
2. {
3. Device (USBX)
4. {
5. Name (_ADR, Zero) // _ADR: Address
6. Method (_STA, 0, NotSerialized) // _STA: Status
7. {
8. If (_OSI ("Darwin"))
9. {
10. Return (0x0F)
11. }
12. Else
13. {
14. Return (Zero)
15. }
16. }
17.
18. Method (_DSM, 4, NotSerialized) // _DSM: Device-Specific Method
19. {
20. If ((Arg0 == ToUUID ("a0b5b7c6-1318-441c-b0c9-fe695eaf949b")))
21. {
22. Local0 = Package (0x08)
```

```
23. {
24. "kUSBSleepPowerSupply",
25. 0x13EC,
26. "kUSBSleepPortCurrentLimit",
27. 0x0834,
28. "kUSBWakePowerSupply",
29. 0x13EC,
30. "kUSBWakePortCurrentLimit",
31. 0x0834
32. }
33. DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))
34. Return (Local0)
35. }
36.
37. Return (Zero)
38. }
39. }
40. }
```

Alles anzeigen

Beitrag von „bluebyte“ vom 16. Oktober 2020, 00:15

[apfelnico](#) herzlichen Dank für deine schnelle Antwort. Ich hatte die SSDT-EC-USBX.aml, den USBPorts.kext (info.plist), ohne Renames, XHCIPortlimit=No, UsbInjectAll=No. Habe mich genau an die Anleitung von Faust gehalten.

Im Moment fahre ich ohne SSDT, mit UsbInjectAll und Renames EHC1->EH01 und EHC2->EH02. Am Rear-Panel an allen Anschlüssen kein Problem. An beiden Front-USB 2.0 kein Problem. Am Front-USB 3.0 SS01 steigt die SSD in der IcyBox aus. MyBook mit ext. Netzeteil gibt es kein Problem. USB-Sticks an allen Ports kein Problem. Habe schon mit grit den ganzen Tag geschattet. Einen Defekt der SSD kann ich ausschließen. Der Port SS01 ist bei mir jetzt problematisch. Ich gehe jetzt ins Bett. Eilt nicht.

Viele Grüße von bluebyte

Beitrag von „maschinenwart“ vom 20. Oktober 2020, 11:50

[Zitat von apfelnico](#)

Bei mir sehen die Ports so aus, in einer eigenen SSDT beschrieben:

Hallo [apfelnico](#),

ich habe auch mal versucht meine USB Ports in einer SSDT zu beschreiben. Das funktioniert scheinbar auch, nur der interne Port 14 wird im Hackintool als "normaler" USB2 Port angezeigt.

Könntest du dir bei Gelegenheit mal meine SSDT anschauen und mir gegebenenfalls einen Tip geben, wie man das eleganter lösen kann?

[X299-USB-MW.aml](#)

Beste Grüße

mw

Beitrag von „apfelnico“ vom 20. Oktober 2020, 12:03

[maschinenwart](#)

habe alle Ports drin und die nichtbenötigten mittels "Zero" statt "=0xFF" ausklamüsert. Könnte bei dir so aussehen (vergiss nicht, die originale SSDT in der ACPI zu unterdrücken):

Beitrag von „hObelware“ vom 20. Oktober 2020, 13:59

[bluebyte](#)

ich hab auch schon immer das Problem, dass bus-powered devices an SS01 (am internen Frontpanel-Header) permanent aussteigen .. sowohl beim Z77 als auch beim Z97, das ganze aber auch unter Windows

ich hab irgendwo mal gelesen, dass das "normal" sei und wohl ein Designfehler bei den Intel XHC Controllern bei der Stromversorgung vorliegt

.. kann das sonst noch wer bestätigen?

Beitrag von „bluebyte“ vom 20. Oktober 2020, 15:04

[hObelware](#) vielen herzlichen Dank für deine Rückmeldung. Also bin ich nicht allein mit dem Problem. Komisch ist nur, dass sich noch niemand, außer mir und jetzt du, zu diesem Problem geäußert hat.

Bin mal gespannt, ob sich jemand meldet.

Beitrag von „maschinenwart“ vom 21. Oktober 2020, 15:39

[Zitat von apfelnico](#)

[maschinenwart](#)

habe alle Ports drin und die nichtbenötigten mittels "Zero" statt "=0xFF" ausklamüsert. Könnte bei dir so aussehen (vergiss nicht, die originale SSDT in der ACPI zu unterdrücken):

...vielen Dank, dass ging ja flott! Hat auch auf Anhieb geklappt. Die originale SSDT hatte ich schon nach deiner [Anleitung](#) unterdrückt. ~~Beim Herumprobieren habe ich festgestellt, dass Hackintool wohlmöglich die USB-C Ports falsch bezeichnet. Die Standard USB-C Ports werden als "switchable" angezeigt und die Schaltbaren als Standard.~~

Beitrag von „pgr69“ vom 25. Oktober 2020, 16:07

G E L Ö S T

Mit Hackintool alle grünen Ports den Connector-Typ eingestellt. Die nicht grünen raus geworfen neue KEXT erstellt und ins OTHERS-Verzeichnis im EFI gelegt und nun wird bei jedem Port gefragt ob es an eine VM gehen soll oder an den MAC.

Ich kann jetzt nur nicht sagen ob die Lösung das Einstellen der Connector-Typen war oder das rauswerfen der unbenutzten Ports - denn unter 15 war ich ja.

Hallo Leute

[apfelnico](#): es tut mir echt leid.....meine Probleme sind anscheinend noch nicht ganz behoben.

Zumindest weiß ich jetzt dank dir so einigermaßen wie das funktioniert.

Also ich habe im Thread [Probleme Parallels und USB-Geräte](#)

beschrieben das ich Probleme habe das Parallels mich nicht fragt wo die USB-Sticks die ich einstecke eingebunden werden sollen.

Also Hackintool angeworfen und IOREGExplorer und dann habe ich mir die Arbeit gemacht in jeden Slot einmal ein UBB3-Gerät und einmal ein USB2-Gerät eingesteckt habe.

Dabei hat sich herausgestellt dass ich beim letzten benutzen von Hackintool wohl einen USB-Port (HS08 schätze ich mal) entfernt habe der auf der Rückseite liegt. Darin funktioniert derzeit gar nichts. Im daneben liegenden Port (HS07) geht alles.

Aber das nur so nebenbei.

Es geht ja um die Parallels-Problematik.....

Also bei der Testerei ist aufgefallen das Parallels immer dann korrekt anspringt wenn ich einen der 4 Ports am IOPanel hinten benutze die über den 4-Port-USB3-Switch am Port SS05 (respektive HS05 für USB2) angeschlossen sind.

Jetzt stellt sich mir die Frage WARUM - warum geht es am Port (SH)S05 und an den anderen nicht?

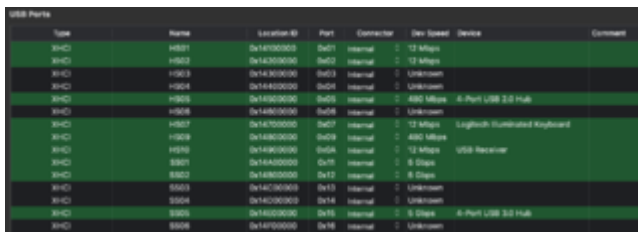
NEBENFRAGE: Die jetzt noch nicht grün gewordenen Ports 3,4 und 6 kann ich wohl entfernen die scheinen nicht angeschlossen zu sein bzw. gehören zu den beiden Anschlüssen auf dem Mainboard die ich nicht belegt habe.

EDIT1:

Sollte oder muss ich im Hackintool etwas bei Connector einstellen?

Vorab vielen Dank

Ig



Type	Name	Location ID	Port	Connector	Dev. Speed	Device	Comment
XHC	4801	0x14700000	0a01	Internal	12 Mbps		
XHC	4802	0x14700000	0a02	Internal	12 Mbps		
XHC	4803	0x14700000	0a03	Internal	12 Mbps		
XHC	4804	0x14700000	0a04	Internal	12 Mbps		
XHC	4805	0x14700000	0a05	Internal	480 Mbps	4-Port USB 3.0 Hub	
XHC	4806	0x14700000	0a06	Internal	12 Mbps		
XHC	4807	0x14700000	0a07	Internal	12 Mbps	Logitech Unimouse Keyboard	
XHC	4808	0x14700000	0a08	Internal	480 Mbps		
XHC	4809	0x14700000	0a09	Internal	12 Mbps	USB Receiver	
XHC	4810	0x14700000	0a10	Internal	8 Stages		
XHC	4811	0x14700000	0a11	Internal	8 Stages		
XHC	4812	0x14700000	0a12	Internal	8 Stages		
XHC	4813	0x14700000	0a13	Internal	12 Mbps		
XHC	4814	0x14700000	0a14	Internal	12 Mbps		
XHC	4815	0x14700000	0a15	Internal	8 Stages	4-Port USB 3.0 Hub	
XHC	4816	0x14700000	0a16	Internal	12 Mbps		

Beitrag von „pgr69“ vom 2. November 2020, 17:26

Hallo [apfelnico](#)

noch etwas Nerven für USB 😊

Also beim Sage 10G mit OpenCore benutze ich glaube ich den USB.Kext von Dir.

Alle physikalischen Ports funktionieren auch mit allem was ich da so reinstecken kann.

Ich bekomme in Hackintool alles grün bis auf einen Port. Es ist der Port 4 des XHC3...Adresse ist glaube ich 0x03400000 und der Name HS02. Weisst DU welcher physikalische Anschluss das bei dir ist?

Ich habe dadurch keinerlei Einschränkungen und könnte den Port auch raus schmeißen. Dachte nur da du ihn drin hast muss es ihn ja auch irgendwo physikalisch geben.

Ig und Danke

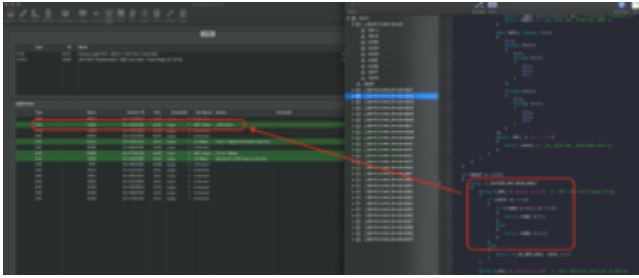
Beitrag von „apfelnico“ vom 2. November 2020, 17:48

[pgr69](#)

habe dir gerade eine persönlich Nachricht geschickt. Bei mir ist das alles über SSDT gelöst, ohne USB-Kext. Die wird nicht benötigt. Bin gerade an einem neuen Guide, im alten steht das noch anders drin. Schau mal in deine "Konversationen" ...

Beitrag von „atl“ vom 6. Februar 2021, 01:13

Ich habe mir das USB-Patchen per SSDT auch mal angesehen und ausprobiert. Soweit scheint (äußerlich) alles zu funktionieren. Ob es vom System aber intern richtig erkannt wird, da bin ich mir nicht sicher. Zumindest zeigt das Hackintool mir etwas anderes an, als konfiguriert / gepatcht ist:



Der Port HS02/SS02 ist ein USB3.1 Type-C Port am Gigabyte Z390 M Gaming. Deshalb habe ich HS02 und SS02 in der SSDT als XUSC definiert. Das Hackintool zeigt den Board aber als USB2-Port an.

Ähnliches ist übriges mit dem HS10, welcher als "internal" gekennzeichnet ist, aber vom Hackintool als normaler USB2-Port angezeigt wird.

Ist das ein Anzeigeproblem vom Hackintool, oder habe ich da etwas falsch gemacht in der SSDT? 🤔

Beitrag von „kuckkuck“ vom 6. Februar 2021, 12:40

Nutz am besten IOReg zum überprüfen. Aber wahrscheinlich liegt der Fehler hier auf deiner Seite. Wird denn überhaupt die SSDT korrekt geladen und sind die Pfade richtig?

Beitrag von „atl“ vom 7. Februar 2021, 01:48

[Zitat von kuckkuck](#)

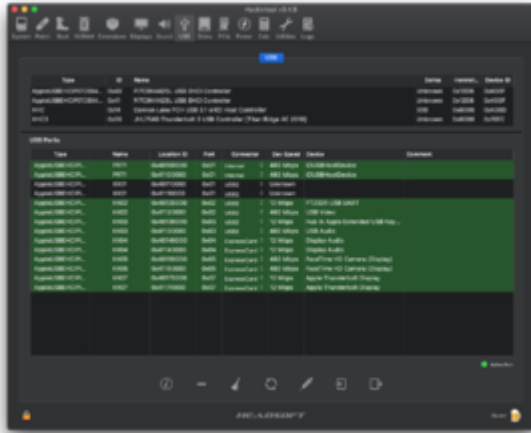
Aber wahrscheinlich liegt der Fehler hier auf deiner Seite.

Davon gehe ich aus. 🤔

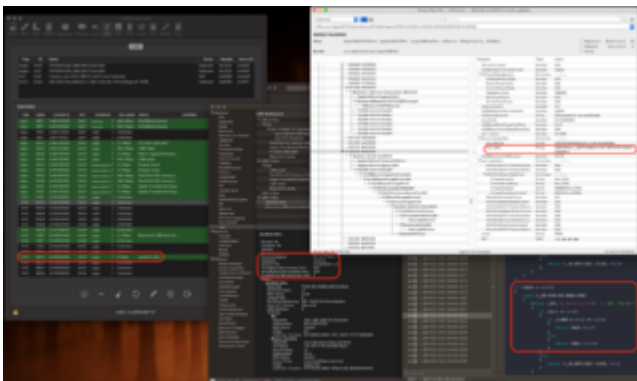
[Zitat von kuckkuck](#)

Wird denn überhaupt die SSDT korrekt geladen und sind die Pfade richtig?

Wenn ich die SSDT deaktiviere, dann sieht es im Hackintool wie folgt aus, weshalb ich mir sicher bin, dass sie geladen wird.



Stecke ich einen USB3-Stick in den USB-C Port, sieht das Ganze im IORegistryExplorer so aus:



Ein Test mit "Blackmagic Disk Speed Test" bringt Lesegeschwindigkeiten bis zum 95MB/s, so dass es kein USB2-Port sein kann. Ich glaube langsam tatsächlich, dass das Hackintool die Anzeige nicht hinbekommt. Ein `ioreg` im Terminal sagt mir auch, dass der Port der Klasse "`AppleUSB30XHCIPort`" zugeordnet ist:

Code

1. `ioreg | grep AppleUSB30X`
2. `||| +o SS01@14b00000 <class AppleUSB30XHCIPort, id 0x10000038c, registered, matched, active, busy 0 (0 ms), retain 14>`
3. `||| +o SS02@14c00000 <class AppleUSB30XHCIPort, id 0x10000038d, registered, matched, active, busy 0 (210 ms), retain 19>`
4. `||| +o SS03@14d00000 <class AppleUSB30XHCIPort, id 0x10000038e, registered, matched, active, busy 0 (0 ms), retain 14>`

5. | | | +-o SS04@14e00000 <class AppleUSB30XHCIPort, id 0x10000038f, registered, matched, active, busy 0 (0 ms), retain 14>
6. | | | +-o SS05@14f00000 <class AppleUSB30XHCIPort, id 0x100000390, registered, matched, active, busy 0 (0 ms), retain 14>