

Erledigt

Clover, DSDT, SSDT..... ich bin überfordert :S

Beitrag von „seefew“ vom 23. November 2017, 17:27

Hi Leutz,

hab meinen Hacki("SkylakeHack") nach etlichen Hürden soweit gebracht, dass er nun von SSD startet, und soweit auch funktioniert.

obwohl´s ja nur ein "Nice2have"-Hacki sein sollte, würde das ja eigtl reichen.

Hab´s aber trotzdem gewagt, mich an der DSDT zu versuchen..... äöhhhhm 🙄🙄🙄

Da lass ich lieber meine Finger von, wer weiß, wo das endet.

Vlt erklärt sich ja jmd bereit, mir dabei zu helfen, würd´s ja gern kapiern,

aber das mit den ganzen Warnungen, Remarks & Oprimizations übersteigt meinen Horizont

BTW: ist die DSDT zwingend, ebenso die SSDT?

Grüße derweil

seefew

Beitrag von „derHackfan“ vom 23. November 2017, 17:31

Hast du schon mal in unserem Wiki gestöbert?

Grundlegende Schritte zum Erstellen einer gepatchten [DSDT - Desktops](#)

Btw: Thread passend verschoben da es sich ja nicht um eine Anleitung von dir handelt. 😊

Beitrag von „kuckkuck“ vom 23. November 2017, 17:50

Für allgemeine Erklärungen und evtl. nötige Optimierungen solltest du auch mal hier vorbeischaun: <https://www.hackintosh-forum.de/...=40-Nach-der-Installation>

Übrigens ist beides nicht Pflicht, du siehst ja wie dein hacky auch ohne funktioniert, aber definitiv nice 2 have!

Beitrag von „seefew“ vom 23. November 2017, 17:57

[@derHackfan](#)

Sorry für den falschen Thread, fiel mir zu spät auf,
in den Link von dir hab ich mich versucht, ehe ich resigniert aufgeben musste.

[@kuckkuck](#)

werd ich mir durchlesen

@all:

wie kann ich Clover dazu überreden, dass kein Auswahlmenü mehr kommt,
sondern gleich in OSX bootet?

Grüße

Beitrag von „grt“ vom 23. November 2017, 18:03

meinem skylake haben dsdt und die ssdt's schon ganz gut getan, es war aber im vergleich zu dsdt's von thinkpads ziemlich harter tobak, mich darin zurechtzufinden.
warnungen kann man eigentlich ignorieren, remarks und optimierungen lasse ich mir gar nicht erst anzeigen (man muss sich ja nicht völlig irre machen)
gibts denn errors?

im clover gibts einen timeout haken, mit dem du einstellst, wie lang das menü angezeigt wird. könntest du auf 0 setzen. dann haste aber keine chance mehr, was auszuwählen, wenn es denn mal sein müsste.

Beitrag von „seefew“ vom 23. November 2017, 18:07

[@grt](#)

nur 2 Fehler, die konnte ich beseitigen,
mein "nicht-kapieren" betrifft eben die Warnungen usw.

Beitrag von „grt“ vom 23. November 2017, 18:28

die warnungen beziehen sich auf unsauberen code, der aber nicht so fehlerhaft ist, dass er nicht ausgeführt werden kann, ist aber manchmal auch total spitzfindig
beispiel:

"*not all controlpaths return a value*" (oder so ähnlich) - kriegste weg, in dem du in der entsprechenden methode an die schliessende geschweifte klammer navigierst, und genau vor diese klammer ein "Return (Zero)" (ohne "") setzt.

die warnung kommt aber auch, wenn alle in der methode vorkommenden "untermethoden" einen rückgabewert haben, und keinerlei "if" bewirkt, dass das fehlende Return Zero überhaupt erreicht wird.

dann gibts noch so eine seltsame warnung, dass irgendwas gesetzt wär, aber nicht benutzt wird (*Method Local is set but never used (Local3)*). da kann man dann eine zeile einfügen:

"And (Local3, Ones, Local3)" (LocalX natürlich entsprechend der warnung)

oder die size mismatch-dinger -> tag is smaller than... da macht man z.b. aus

```
"CreateByteField (CRS3, \_SB.PCI0.LPCB.SIO1._Y1C._HE, IRQS)" ein "CreateBit  
Field (CRS3, \_SB.PCI0.LPCB.SIO1._Y1C._HE, IRQS)"
```

muss man halt gucken, welche grösse angegeben ist - bit, byte, word oder qword - recherchiert, was die einzelnen bezeichnungen in bit bedeuten (oder try and error...) und passt entsprechend der warnung an.

eigentlich könnte man das alles ignorieren, aber schon der übersichtlichkeit halber und natürlich dem bastelehrgeiz geschuldet, versuch ich, möglichst wenige warnungen übrigzubehalten.

meistens findet sich eine lösung, wenn man den text der compilerwarnung googelt.

Beitrag von „Thogg Niatiz“ vom 23. November 2017, 21:11

[Zitat von grt](#)

bit, byte, word oder qword

Fehlt noch DWORD 😊

Beitrag von „grt“ vom 23. November 2017, 23:29

hast recht - 👍

Beitrag von „seefew“ vom 24. November 2017, 02:20



das mal als Erstes !!

Bin nun von den anfänglichen 2 Errors, 118 Warnings auf 0 Fehler und 11 Warnings runtergekommen. 👍

Die 11 sind allesamt gleichlautend:

Missing dependency (Device object requires a _HID or _ADR in same scope)

Tante Google wusste da auch nix zu;
sind zwar noch 95 Remarks, aber wenn die "übersehen" werden können,
dann soll´s so sein.
Aber für jetzt is genug, Zeit für die Horizontale, gut Nacht.
Grüße

Beitrag von „grt“ vom 24. November 2017, 14:26

da warst du ja noch recht fleissig gestern [@seefew](#) 👍

Zitat von seefew

Missing dependency (Device object requires a _HID or _ADR in same scope)

die hab ich auch, und hab auch nix dazu gefunden. mit-der-stange-im-nebel-stochernde tryanderror versuche haben nur zu neuen fehlern geführt, ergo hab ich sie fürs erste da gelassen, wo sie sind.
und wie gesagt, die remarks ignoriere ich von vorne herein, lass ich mir überhaupt nicht anzeigen. sind ja nur "anmerkungen"

Beitrag von „Thogg Niatiz“ vom 24. November 2017, 15:27

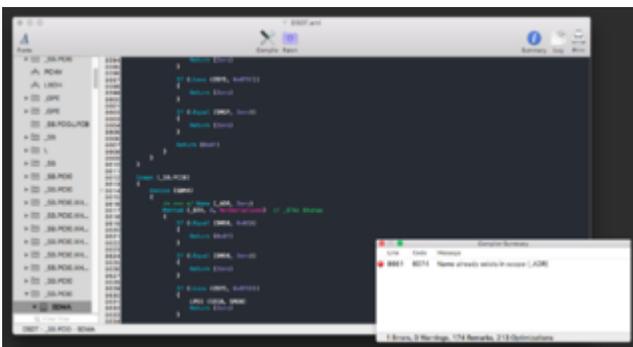
Die Remarks sind von iASL eingefügte Kommentare, wie "// Device Address". Optimizations sind Änderungen wie 0x0 => Zero oder andere Optimierungen, die die Lesbarkeit und Verständlichkeit der C-ähnlichen Interpretation des Assemblercodes verbessern sollen. Beides geht beim Kompilieren in Assembler, also "speichern als ASL File" verloren und wird beim erneuten Öffnen mit MaciASL wieder erzeugt, also gibts da nichts auszubessern.

Bezüglich der dependency issues: Die Adresse ist meist eh vorhanden, nur in einen Scope ausgelagert. Darum ist das auch kein Fehler, sondern wird sozusagen als "bad practice" von iASL bemängelt. Auch daran lässt sich natürlich etwas machen, auch wenn es in der Regel echt

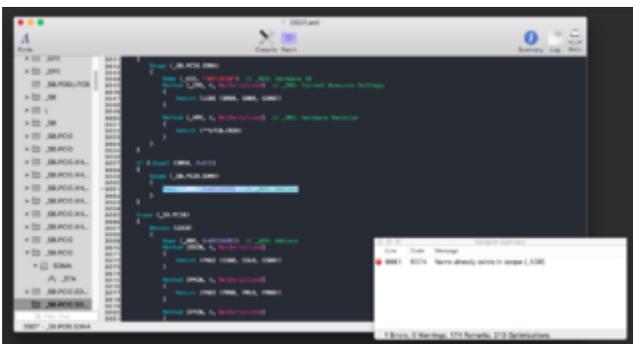
nur kosmetisch für alle ACPI Fetischisten (wie mich 😄) ist, nämlich die Adresse vom Scope direkt ins Device geben. Kleine Bilderstrecke:



Durch Anklicken der Warnung ein betroffenes Device ausfindig machen. Normal sollte darin als erstes die Adresse deklariert werden, was hier offensichtlich nicht der Fall ist. Nun könnte man den Scope mit der Adresse suchen, was aber für viele nicht intuitiv sein dürfte...



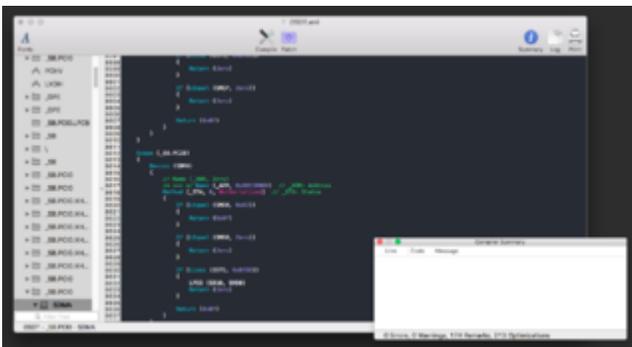
Wenn man einen "Dummy" Adresseintrag macht, wirft der Compiler einen Fehler, da zwei Adressen beim gleichen Device unweigerlich zu Problemen führen würden. Dadurch wird man aber leicht zum gesuchten Scope geführt.



Hier ist die Adresse, welche man gleich ausschneiden und später im Device einfügen kann.



Durch das Ausschneiden ist im Scope nichts mehr enthalten, man kann ihn also komplett entfernen.



Zurück im Device ersetzt man die Adresse 0x0 durch die korrekte Adresse. Kompilieren et voilà - das Mimimi des Compilers hat ein Ende 😊

Beitrag von „seefew“ vom 26. November 2017, 14:47

Soderle, da isser wieder.

[@Thogg Niatiz](#)

hab deine Bilderserie abge"ackert" und konnte die Warnings von 11 auf 3 reduzieren, ob das aber nun so stimmt????

Vlt will ja jmd sich meine DSDT mal ansehen?

Danke vorweg

Grüße

Beitrag von „noEE“ vom 26. November 2017, 15:09

[Zitat von grt](#)

im clover gibts einen timeout haken, mit dem du einstellst, wie lang das menü angezeigt wird. könntest du auf 0 setzen. dann haste aber keine chance mehr, was auszuwählen, wenn es denn mal sein müsste.

Ist nicht ganz richtig. Ich habe meinen Timeout auch auf Null gestellt, damit der Häcki gleich durchbootet. Wenn du zwischen Bios Post und erscheinen des Apfels wiederholt eine beliebige Taste drückst, erscheint Clover für den Fall das was sein sollte.

Beitrag von „Thogg Niatiz“ vom 28. November 2017, 19:50

[@seefew](#) bei mir wurden sogar noch 5 Warnungen angezeigt. Anbei deine Datei zurück. Schau am besten mit diff, was sich gegenüber deiner Datei geändert hat. Dann kannst du es sicher schneller nachvollziehen, als wenn ich das mit 10 Screenshots dokumentiere 😊

Beitrag von „Doctor Plagiat“ vom 28. November 2017, 20:51

Danke [@Thogg Niatiz](#) Ich habe soeben bei meiner DSDT auch allen Warnungen beseitigt. Bringt wahrscheinlich nichts als ein gutes Gefühl, aber das ist ja auch immerhin was. 😄👍

Beitrag von „al6042“ vom 28. November 2017, 21:23

Jupp...habe bei mir auch noch etwas bereinigt..
Da waren es aber andere Warn-Meldungen, nämlich

Code

1. Not all control paths return a value (...)

Das konnte einfach gelöst werden, in dem in das `Return (Zero)` Statement am Ende der jeweils betroffenen Methode eingefügt wurde...

Jetzt ist das "Compiler Summary"-Fenster nackich... 👍

Beitrag von „grt“ vom 28. November 2017, 21:54

ich muss noch ran.. missing dependency gibts noch 11 stück, dann ist hier auch alles sauber geputzt (zumindest die dsdt 😊)

Beitrag von „Doctor Plagiat“ vom 28. November 2017, 21:58

Dank [@Thogg Niatiz](#) nur 'ne kleine Fleissarbeit. 😊

Beitrag von „seefew“ vom 1. Dezember 2017, 08:40



Der Quicksilver-Skyläk-Häck läuft jetzt dank eurer Hilfe fast wie gewünscht; lediglich das Booten direkt in OSX klappt ned, muss nach wie vor Enter drücken. Irgendwie zickt Clover da bissle...

Ein Beitrag im Showroom folgt ASAP.

Besten Dank nochmal
Grüße
seefew

Beitrag von „Ka209“ vom 1. Dezember 2017, 10:03

Mit dem start ist es nur eine Einstellung in der config

entweder auf null setzen den timer oder auf fast boot setzen
und deinen macOS Laufwerk als Default Boot Volumen einsetzen
wie auf dem FOTO läuft es bei mir und bootet durch

EDIT

un wenn es trotzdem nicht geht Theme wechseln
hatte auch das problem mit Theme mal gehabt

Beitrag von „seefew“ vom 1. Dezember 2017, 12:37

Timer verändert & -1 schon probiert, kein Unterschied 😞
Theme & Fastboot werd ich noch antesten.
Besten Dank
Grüße

Beitrag von „al6042“ vom 1. Dezember 2017, 19:59

Wichtig ist, dass der Name deiner OSX-Partition unter "Default Boot Volume" manuell eingetragen wird.

Also genauso, wie der Name der Partition im `diskutil list` angezeigt wird...