

BOOTVORGANG STOPPT BEI IOCONSOLEUSERS: GIOSCREENLOCKSTATE

Beitrag von „dav3c137“ vom 15. Dezember 2020, 09:41

Möchte gern Mojave auf einem FSC NB Lifebook U772 zum Laufen zu bringen.

Das OS habe ich mit dem Terminal von einer anderen MacOS Installation heraus auf einen USB Stick erstellt, anschließend habe ich Clover installiert.

Im Forum hier habe ich mich durch das Wiki gelesen und bin auf [INSTALL-STICK FÜR DIVERSE SYSTEM-VARIANTEN \(HIGH SIERRA 10.13.6, MOJAVE 10.14.6 UND CATALINA 10.15\)](#) gestoßen. Habe dann den CLOVER-Ordner für meine CPU (Ivy) in den EFI Ordner auf den Stick kopiert und die config.plist mit dem Clover Configurator die Grafikchip-Einstellungen editiert da das Zielsystem ein NB ist.

Clover bootet soweit vom Stick, allerdings stoppt der Bootvorgang bei der Meldung "IOConsoleUsers: gIOScreenLockState 3, hs 0, bs 0, now 0, sm 0x0" (siehe auch Bild im Anhang).

Den EFI-Ordner hänge ich mit an.

Habe ich das richtig verstanden das die Ursache für die Meldung mit fehlerhaften Einstellung in Bezug auf den Grafikchip zusammenhängt?

Bin für jede Unterstützung dankbar 😊.

PS: Ist mein erster Eintrag hier. Bitte um Nachsicht wenn Infos, Screenshots, usw. fehlen.

Beitrag von „5T33Z0“ vom 15. Dezember 2020, 11:36

Du verwendest das falsche SMBIOS: "iMac13,2" ist für IvyBridge Desktops. Du hast aber ein

IvyBridge Notebook mit nem i5 Prozessor, wenn ich es richtig sehe.
<https://preisvergleich.check24...jitsu-lifebook-u7723.html>

Dafür verwendet man ein Notebook SMBIOS, am besten "MacBookPro10,2" für i5 Prozessorren mit Intel HD4000 on-board Grafk und 10,1 für i7. Also bitte unter "SMBIOS" aus dem Dropdown Menü rechts in der Ecke den entsprechenden Eintrag auswählen, um ein neues SMBIOS zu generieren, speichern und nochmal versuchen.

Du verwendest zudem eine ältere Version von Clover, die du unter BigSur nicht verwenden kannst. Dafür wäre ein Udate auf Version mit Quirks notwendig (ab Clover 5123). Details dazu in meiner Signatur.

Beitrag von „dav3c137“ vom 15. Dezember 2020, 19:27

Vielen Dank für deine Rückmeldung!

Ich habe die Einstellung im SMBIOS auf MacBookPro10,2" für i5 Prozessorren mit Intel HD4000 geändert und auch Clover upgedatet.

Ich habe mich dann in den [\[GUIDE\] CLOVER UPDATEN \(AB V5123\) INKL. OPENRUNTIME UND QUIRKS](#) eingelesen und die Treiber die dort erwähnt werden gelöscht. Anschließend habe ich unter Quirks die Einstellungen wie für "LAPTOP (IvyBridge)" beschrieben gesetzt.

Wenn ich das OS nun boote hängt er noch immer an der gleichen Stelle.

Hänge den aktuellen Screenshot und die EFI-zip nochmal an.

Wäre toll wenn du oder jemand im Forum noch einen Tipp hätten.

Beitrag von „5T33Z0“ vom 15. Dezember 2020, 19:55

- Quirks sehen gut aus
- Kexts sehen gut aus
- Treiber sehen soweit gut aus. DataHubDxe.efi, EmuVariableUefi.efi, FSInject.efi hätte ich vorerst drin gelassen. aber ich denke nicht, dass es einen Unterschied macht, da die Fehlermeldung ja dieselbe bleibt.
- ig-platform-id sieht gut aus
- SMBIOS sieht gut aus.
- ACPI > da fehlen glaube ich noch n Paar Sachen: <https://dortania.github.io/Ope...plist/ivy-bridge.html#add>

- Ist CSM deaktiviert im BIOS?

- Hat das Gerät noch ne zusätzliche diskrete Grafikkarte außer der onboard HD4000? Falls ja, muss sie deaktiviert werden

Es ist auf jeden Fall eine der nerfigen Fehlermeldungen, bei der man nicht immer sofort weiß, woran es liegt.

Beitrag von „dav3c137“ vom 16. Dezember 2020, 00:14

Bezgl. den ACPI Einstellungen habe ich mich mal durch den Link gearbeitet. Verstehe ich das richtig das man eine neue config.plist mit ProperTree baut?

In jedem Fall habe ich das mal gemacht und die Einstellungen aus dem Link übernommen.

Der Bootvorgang bricht jetzt sofort mit Verbotssymbol ab nachdem ich MacOS installieren will. Hänge mal die aktuelle config.plist an.

Was ich noch nicht verstanden habe ist inwieweit man die Einstellungen aus der alten config.plist in die neue übernehmen kann bzw. ob das überhaupt notwendig ist.

CSM oder auch andere UEFI-Settings finde ich im BIOS gar nicht.

Das NB hat nur einen Grafik-Chip.

Beitrag von „5T33Z0“ vom 16. Dezember 2020, 02:04

Es ging eigentlich nur darum, die fehlenden ACPI Dateien zu ergänzen. Von OpenCore IvyBridge Config bauen und damit dann Clover zu starten, war keine Rede. Dass das nicht geht, sollte klar sein. 😊

Allerdings könntest Du die OpenCore Config könntest Du sogar verwenden - in OpenCore natürlich und dann auch nur mit Änderungen: **AppleXcpmCfgLock = no setzen**, falsche Platforminfo >> MacBookPro10,2 immernoch, nicht iMac19,1, es ist keine Grafikkarte eingetragen unter Device Properties eingetragen. PciRoot(0x0)/Pci(0x2,0x0) und die Daten für **AAPL,ig-platform-id 03006601** müssten da rein. Bewahr sie erstmal auch für einen anderen Zeitpunkt...

Aber hier hast Du erstmal ne Clover cconfig mit ein Paar laptop-spezifischen Settings. Ich habe die Umbenennung der Grafikkarte (change GFX0 to IGPU) mal rausgenommen, weil Whatevergreen schon von sich aus macht. Falls die Karte systemintern aber ne andere Bezeichnung haben sollte als die, über die whatevergreen verfügt, dann wird sie nicht von macOS erkannt.

Dewegen am besten unter Windows Im Geräte-Manager checken, wie die Karte im BIOS bezeichnet ist. Dazu den "BIOS-Pfad" der Grafikkarte anzeigen lassen (aus Dropdown-Menü auswählen). Sollte sie nämlich nicht GFX0 heißen, dann benötigst Du eine andere Umbenennung, damit sie erkannt und angesprochen werden kann, wenn macos startet.

[clover config.plist](#) >> In config.plist unbenennen nicht vergessen.

Beitrag von „EdD1024“ vom 16. Dezember 2020, 09:48

Ich würde mit einer ganz rudimentären config anfangen, kein WLAN, Bluetooth oder Netzwerk und würde wie im Post #1 auch auf Grafik tippen. Bei meinen Versuchen lädt WEG nicht

korrekt oder gar nicht, meist gibt es einige Meldungen dazu zwischendurch.

Beitrag von „dav3c137“ vom 16. Dezember 2020, 20:30

[Zitat von 5T33Z0](#)

Es ging eigentlich nur darum, die fehlenden ACPI Dateien zu ergänzen. Von OpenCore IvyBridge Config bauen und damit dann Clover zu starten, war keine Rede. Dass das nicht geht, sollte klar sein. 😊

Allerdings könntest Du die OpenCore Config könntest Du sogar verwenden - in OpenCore natürlich und dann auch nur mit Änderungen: **AppleXcpmCfgLock = no setzen**, falsche Platforminfo >> MacBookPro10,2 immernoch, nicht iMac19,1, es ist keine Grafikkarte eingetragen unter Device Properties eingetragen. PciRoot(0x0)/Pci(0x2,0x0) und die Daten für **AAPL,ig-platform-id 03006601** müssten da rein. Bewahr sie erstmal auch für einen anderen Zeitpunkt...

Aber hier hast Du erstmal ne Clover cconfig mit ein Paar laptop-spezifischen Settings. Ich habe die Umbenennung der Grafikkarte (change GFX0 to IGPU) mal rausgenommen, weil Whatevergreen schon von sich aus macht. Falls die Karte systemintern aber ne andere Bezeichnung haben sollte als die, über die whatevergreen verfügt, dann wird sie nicht von macOS erkannt.

Dewegen am besten unter Windows Im Geräte-Manager checken, wie die Karte im BIOS bezeichnet ist. Dazu den "BIOS-Pfad" der Grafikkarte anzeigen lassen (aus Dropdown-Menü auswählen). Sollte sie nämlich nicht GFX0 heißen, dann benötigst Du eine andere Umbenennung, damit sie erkannt und angesprochen werden kann, wenn macos startet.

[clover_config.plist](#) >> In config.plist umbenennen nicht vergessen.

Alles anzeigen

Puh, hat jetzt bissl gedauert.

Merci für die Config-Datei. Leider wirft die einen anderen Fehler (IMG_0814) und kurz darauf (IMG_0815).

Auf dem NB lief Debian, ich musste erst Windows drauf installieren um den BIOS-Gerätenamen auszulesen. Der heißt GFX0 (IMG_0813), deswegen hatte ich in deiner Konfig nichts angepasst.

Beitrag von „5T33Z0“ vom 16. Dezember 2020, 23:07

Oh ha. Habe gerade die Kexts aus deiner EFI geprüft. die sind fast alle nicht mehr aktuell. Bitte updaten. Eine Fehlerquelle weinger dann.

Beitrag von „dav3c137“ vom 17. Dezember 2020, 10:37

OK. Habe die Kext-Updates durchgeführt.

Denke der Bootvorgang stoppt nun an einer anderen Stelle. Hab den aktuellen EFI Ordner und die Meldung nochmal in den Anhang gepackt.

Beitrag von „EdD1024“ vom 17. Dezember 2020, 10:42

Die SSDT-EC.dsl in ACPI/Patched müsste eine .aml-Datei sein. Mit MaciASL kompilieren und die SSDT-EC.dsl ersetzen.

Beitrag von „dav3c137“ vom 17. Dezember 2020, 11:18

OK, aber in diesem Verzeichnis liegt ja schon eine .aml-Datei die aktueller ist als die .dsl-Datei.

Beitrag von „5T33Z0“ vom 17. Dezember 2020, 11:53

[dav3c137](#) Die .dsl löschen am besten. Nimm ansonsten mal wieder Deine ursprüngliche Clover Config.plist und versuch es damit. Vielleicht passen die Ergänzungen, die ich hinzugefügt habe einfach nicht.

Beitrag von „EdD1024“ vom 17. Dezember 2020, 12:52

Die EC-Datei wird gebraucht, das ist die Ursache der Fehler 1349 oder so ähnlich...

Nur als aml, muss im Machinencode sein. Kannst einfach die Version von Dortania nehmen....

Alles in Patched muss .aml sein, das weiß man doch mittlerweile... ;)

Beitrag von „5T33Z0“ vom 17. Dezember 2020, 13:20

[EdD1024](#) Er hat die SSDT-EC.aml Datei. Ich habe davon gesprochen, die .dsl zu löschen

[dav3c137](#) Lies mal deine DSDT mit SSDT-Time unter Windows aus und generiere dir damit SSDT-EC und SSDT-Plug, SSDT-HPET etc. Dann hast du die schon mal passend für dein System. Und dann siehst Du auch gleich ob die neue SSDT für den Embedded Controller (EC) benötigt. Bei meinem Notebook zum Beispiel ist der schon vorhanden und definiert.

<https://github.com/corpnewt/SSDTime>

Beitrag von „EdD1024“ vom 17. Dezember 2020, 14:23

Passt!

Bzgl. SSDTTime, ein super Tip, vor allem weil man Linux vom Stick problemlos booten kann, wo Windows doch manchmal zickt.

(Warum habe ich das Tool eigentlich vergessen? Ich frage mich nur, wie gut es unter macOS funktioniert? Aber das ist off-topic...)

Beitrag von „dav3c137“ vom 18. Dezember 2020, 11:45

[Zitat von 5T33Z0](#)

[EdD1024](#) Er hat die SSDT-EC.aml Datei. Ich habe davon gesprochen, die .dsl zu löschen

[dav3c137](#) Lies mal deine DSDT mit SSDT-Time unter Windows aus und generiere dir damit SSDT-EC und SST-D-Plug, SSDT-HPET etc. Dann hast du schon mal passend für dein System. Und dann siehst du auch gleich ob die neue SSDT für den Embedded Controller (EC) benötigt. Bei meinem Notebook zum Beispiel ist der schon vorhanden und definiert.

<https://github.com/corpnewt/SSDTTime>

Das war etwas holprig, wenn ich die DSDT unter Windows erstellen will kommt die Meldung: "Failed to decompile DSDT.aml". Das selbe in Linux hat dann geklappt, ich hänge die Dateien an.

Hab dann die SSDT-EC ersetzt und nochmal gebootet, leider noch die gleiche Meldung.

Beitrag von „EdD1024“ vom 18. Dezember 2020, 12:47

Nun ja, um die SSDT.aml-Datei zu öffnen, braucht es iasl und das ist ja nicht drauf, es sei denn Du lädst es vorher auch runter und kopierst es in das SSDTTime-Verzeichnis.

Nochmal, im patched-Ordner haben sich nur *.aml-Dateien zu befinden, keine anderen.

Die letzte Zeile sagt etwas boot-id-media, also wäre mein nächster Tip zu schauen, ob USBInjectAll.kext im Kexts-Ordner ist, in der config eingebunden ist und der Parameter XHCIPortLimit gesetzt ist.

Alternativ versuche einen anderen USB-Port.

BTW. Du solltest deine Hardware im Profil eintragen, macht es einfacher nachzuvollziehen, über was man gerade redet...

PS: Vielleicht eine grundlegende Klarstellung: Die DSDT.aml ist im Computer drin. Sie wurde vom Hersteller erstellt und da rein geladen, mit dem BIOS. Deswegen ändert sich die DSDT auch mit jedem BIOS-Update. Tools wie SSDTTime (Clover kann das auch) entladen sie nur, sie befindet sich dann im binärformat auf der Platte/Stick. AML steht für ACPI Machine Language (Binary). Um aus ihr wieder Code zu machen braucht es das Programm iasl. Das macht eine de-Kompilierung und erstellt daraus eine lesbare Programmier-Datei. Wir pfuschen dann darin rum, um das Ganze an die MacOS-Erwartungen anzupassen. Perfekt wird das nicht. Aber als eine gute Annäherung reicht das. Die aml-Dateien in patched werden von OpenCore vor dem Start anstatt des eigentlichen Code geladen. Auch die Patches in MaciASL machen das Gleiche. Das sind alles verschiedene Möglichkeiten, das gleiche zu erreichen.

Beitrag von „dav3c137“ vom 18. Dezember 2020, 14:19

iasl hatte ich im SSTTime gesehen. Die De-Kompilierung hat unter Linux soweit auch funktioniert, hänge die lesbare Datei mal mit an.

Im Patchordner lag bzw. liegt nur die SSDT-EC.aml Datei.

USBInjectAll.kext sehe ich in der aktuellen Version in der Config und auch im kexts-Ordner im Unterverzeichnis "Other". XHCIPortLimit ist gesetzt (hänge ich auch mal als Screenshot zusammen mit der aktuellen Config an).

Einen anderen USB-Port habe ich auch probiert, mit selben Ergebnis (immer noch die ursprüngliche Fehlermeldung). Ich hatte bisher immer nur denselben Port benutzt weil das NB nur einen USB 2.0 Port hat und ich gelesen hatte das USB 3.0 Probleme beim booten machen kann.

Die Signatur habe ich eben erweitert.

Vielen Dank für deine Erläuterung in Sachen DSDT, wieder was gelernt 😊

Beitrag von „5T33Z0“ vom 18. Dezember 2020, 14:23

[dav3c137](#) So, du löschst jetzt aus EFI > Clover > ACPI > patched alle Dateien. Dann packst Du diese 4 aus deinem Result Ordner rein:

- SSDT-EC.aml
- SSDT-HPET.aml
- SSDT-Plug.aml
- SSDT-PMC.aml

In den Results sollte auch noch ne Plist namens patches_Clover.plist dabei gewesen sein. Die Einträge daraus musst du in deine Clover Config einbauen. Am besten beides nebeneinander öffnen und copy /paste an die richtigen stellen. ist einfacher als das mit clover configurator zu machen. Und dann weiter sehen.

Beitrag von „dav3c137“ vom 19. Dezember 2020, 13:46

Ich habe die *.aml-Dateien nun in den patched-Ordner kopiert und die Einstellungen aus der patches_Clover.plist in meine config.plist übernommen.

Nun stoppt der Bootvorgang an anderer Stelle (hänge einen aktuellen Screenshot an).



Beitrag von „EdD1024“ vom 19. Dezember 2020, 14:03

Zum Installieren brauchst Du eigentlich nur die SSDT-EC.aml, evtl. noch die HPET, je nach dem. Entferne die anderen aml-Dateien erstmal, oder deaktiviere sie erstmal.

Beitrag von „5T33Z0“ vom 19. Dezember 2020, 16:33

Deaktiviere mal die SSDT-PLUG.aml in der Config, denn IvyBridge CPUs unterstützen das X86PlatformPlugin nicht. Und dann probiers nochmal. Für IvyBridge benötigt man eine andere Datei dafür, die kann man aber später noch erstellen mit ssdtPRGen.

[EdD1024](#) Kannst Du mal aufhören, mir permanent dazwischen zu grätschen? Das nervt.

Beitrag von „dav3c137“ vom 19. Dezember 2020, 16:49

[Zitat von 5T33Z0](#)

Deaktiviere mal die SSDT-PLUG.aml in der Config, denn IvyBridge CPUs unterstützen das X86PlatformPlugin nicht. Und dann probiers nochmal. Für IvyBridge benötigt man eine andere Datei dafür, die kann man aber später noch erstellen mit ssdtPRGen.

[EdD1024](#) Kannst Du mal aufhören, mir permanent dazwischen zu grätschen? Das nervt.

Es ist die SSDT-PLUG.aml die den Fehler aus #21 verursacht.

Wenn ich nur die Datei deaktiviere, kommt aber wieder die ursprüngliche Meldung.

Beitrag von „5T33Z0“ vom 19. Dezember 2020, 17:28

Ich habe in deiner DSDT nachgeschaut. Da ist der Embedded Controller bereits definiert. Dweswegen benötigst Du die SSDT-EC nicht. Also auch deaktivieren.

Und ansonsten alle Kexts außer Lilu, VirtuakSMC und WhateverGreen vorerst deaktivieren. Sind die kexte alle aktuell? Wenn nicht, updaten und dann noch mal versuchen.

Welches OS version soll installiert werden?

Beitrag von „dav3c137“ vom 19. Dezember 2020, 19:09

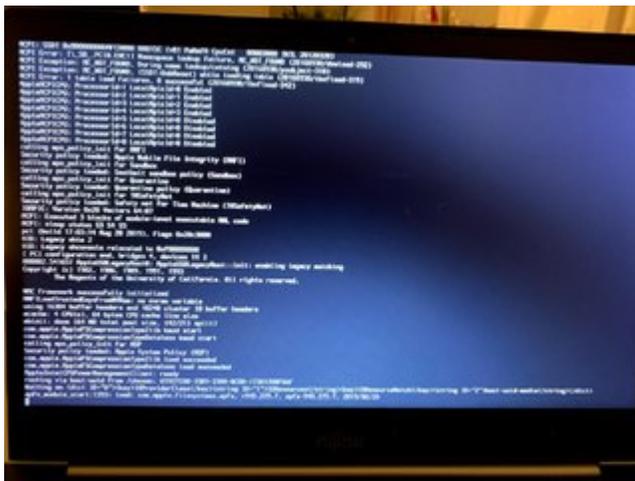
Hab jetzt ohne die SSDT-EC gebootet, vorher alle Kexts außer Lilu, VirtualSMC und WEG rausgeschmissen.

Die Kexts sind alle aktuell.

Es soll Mojave installiert werden.

Anbei Screenshots der jetzigen Fehlermeldung.

BTW: vielen Dank für Deine Geduld und die klasse Unterstützung!



Beitrag von „EdD1024“ vom 19. Dezember 2020, 19:32

[Zitat von 5T33Z0](#)

[EdD1024](#) Kannst Du mal aufhören, mir permanent dazwischen zu grätschen? Das nervt.

Spinner

Beitrag von „5T33Z0“ vom 19. Dezember 2020, 20:21

[dav3c137](#) Die eigentliche Fehlermeldung ist auf dem Bild, wo alles so krisselig ist, mit dem durchgestrichenen Kreis, die kann ich nicht erkennen. Ein Durchgedrichener Kreis ist ein Indiz für ein falsches bzw. nicht mehr unterstütztes SMBIOS. Probier mal MacBookPro9,1 oder 9,2 ansonsten.

Beitrag von „al6042“ vom 19. Dezember 2020, 20:53

Ähem...

Wenn das der Bootvorgang mit Install-Stick ist, kommt der "Garbled Screen" auch gerne, wenn die Verbindung zum USB-Stick verloren geht.

Das passiert gerne, wenn der USB-Port-Limit-Patch nicht aktiv ist.