

Monterey bootet nicht mehr seit dem letzten Herunterfahren

Beitrag von „vw440“ vom 4. Januar 2022, 23:24

Hey ihr Lieben,

ich habe nicht am System geändert. Installiert ist aktuell 12.1 und lief bis zu jetzt ohne Probleme.....am System wurden keine Änderungen vorgenommen.....

Im Anhang findet ihr mein EFI und ein Screenshot, wo er hängen bleibt

Vlt kann mir ja jemand behilflich sein

Ich danke euch schon mal im Voraus

Nach jedem Neustart habe ich auch ein zurückgesetztes BIOS 😞

Beitrag von „al6042“ vom 5. Januar 2022, 08:17

Moin...

Achte in deiner config.plist auf den Kernel-Quirk "DisableRtcChecksum".

Der sollte für Z390er-Boards auf "True" stehen -> [AppleRTC Patch für Asus Z390 Board nach Update auf 10.15.4 nicht mehr funktionsfähig](#)

Beitrag von „vw440“ vom 5. Januar 2022, 09:25

ich hab den parameter geändert leider ohne erfolg.....ich sollte vielleicht noch dazu sagen, dass ich windows 11 parallel nutze.....er bleibt nun hier hängen

oder startet gar nicht mehr.....

Beitrag von „al6042“ vom 5. Januar 2022, 09:44

Setzt sich das BIOS weiterhin zurück?

Beitrag von „vw440“ vom 5. Januar 2022, 09:45

hey nein, dass passiert nicht mehr.....das ist gefixt...aber macos hat weiterhin keine lust mehr 😞 ich versteh es halt absolut nicht, da ich nichts am system geändert habe

Beitrag von „al6042“ vom 5. Januar 2022, 09:48

Wie ist das Dual-Dingens eingerichtet?

Pro Betriebssystem eine Platte?

Wenn ja, war die macOS-Platte während der Installation von Win11 im System verfügbar?

Beitrag von „vw440“ vom 5. Januar 2022, 10:01

Als ich windows 10 pro damals installiert habe war sie nicht angeschlossen

beim update auf windoof 11 leider ja

Beide systeme sind auf unterschiedlichen ssa

macos aufm nvme

windows auf sata ssa

Beitrag von „al6042“ vom 5. Januar 2022, 10:09

Ich kann aktuell nicht viel unterstützen, aber anbei findest du meine aktuelle EFI auf Basis von OC 074...

ggf. hilft dir ein Blick auf meine Herangehensweise.

Läuft auch hier als DualBoot, aber mit Win10.

Serials wurden ausge-x-t.

Die DeviceProperties beziehen sich nur auf meine Kiste und sollten bei dir nicht eingesetzt werden.

Beitrag von „MacAbe“ vom 5. Januar 2022, 17:12

in Anwesenheit von SSDT-iMacPo11.aml ist bei dir SSDT-PLUG.aml nicht nötig und wäre besser, wenn du SSDT-iMacPo11.aml mit _OSM Methode schreibst, sonst wie sie ist, wird für Windows und OSX geladen...

"DisableRtcChecksum" auf dem Asus Prime Z390-A ist nicht nötig 😊

Im Anhang die verbesserte SSDT, einfach ersetzen und SSDT-PLUG.aml löschen.

[al6042](#)

Beitrag von „al6042“ vom 5. Januar 2022, 17:43

Ich denke da kann sich der Threadersteller mit auseinandersetzen.

Ich werde an meiner Konfig nichts ändern, denn beide OS laufen einwandfrei.

Beitrag von „MacAbe“ vom 5. Januar 2022, 17:48

Naja, es ist mir egal, ob du sie änderst oder nicht 😊

Beitrag von „al6042“ vom 5. Januar 2022, 17:54

Nun,

nachdem du mich explizit im Beitrag erwähnt hattest, las sich das tatsächlich wie ein Hinweis zur Nutzung.

Beitrag von „cobanramo“ vom 5. Januar 2022, 19:31

┆ [Zitat von vw440](#)

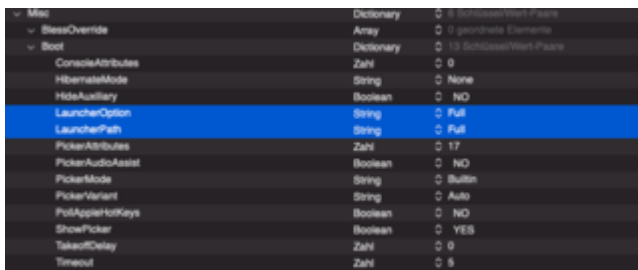

```

20 */
21 DefinitionBlock ( "", "SSDT", 1, "APPLE ", "freqdata", 0x00000001)
22 {
23     External ( _SB_, PR00, DeviceObj) // (from opcode)
24
25     Method ( DTGP, 5, NotSerialized)
26     {
27         If ( LEqual ( Arg0, ToUUID ( "a8b5b7c6-1318-441c-b0c9-fe695eaf949b" )))
28         {
29             If ( LEqual ( Arg1, One) )
30             {
31                 If ( LEqual ( Arg2, Zero) )
32                 {
33                     Store ( Buffer ( One)
34                         {
35                             0x00
36                         }, Arg4)
37                     Return ( One)
38                 }
39                 If ( LEqual ( Arg2, One) )
40                 {
41                     Return ( One)
42                 }
43             }
44         }
45     }

```

Vermutlich auch der grund warum dein Bios zerreisst.

Erst recht wenn du die LauncherOption & LauncherPath falsch konfiguriert hast. Der versucht eben in die NVram (PMCR) zu schreiben.



4. **LauncherOption**
 Type: plist string
 Falseful: Disabled
 Description: Register the **launcher** option in the firmware preferences for persistence.
 Valid values:

- **Disabled** — do nothing.
- **Full** — create or update the top priority boot option in UEFI variable storage at bootloader startup.
 — For this option to work, `RequestBootFromBooting` is required to be enabled.
- **Short** — create a short boot option instead of a complete one.
 — This variant is useful for some older types of firmwares, typically from Intel, that are unable to manage full device paths.
- **System** — create no boot option but assume specified custom option is blessed.
 — This variant is useful when relying on `ForceBooterSignature` quick and OpenCore **launcher** path management happens through `bless` utilities without involving OpenCore.

This option allows integration with third-party operating system installation and upgrades (which may overwrite the `UEFIBOOT/BOOT/EFI` file). The `BOOT/EFI` file is no longer used for bootstrapping OpenCore if a custom option is created. The custom path used for bootstrapping can be specified by using the **LauncherPath** option.

Note 1: Some types of firmwares may have NVRAM implementation flaws, no boot option support, or other incompatibilities. While unlikely, the use of this option may result in boot failures and should only be used exclusively on boards known to be compatible. Refer to `acidanthers/hogwartz#1222` for some known issues affecting Haswell and other boards.

Note 2: While NVRAM resets executed from OpenCore would not typically erase the boot option created in `Bootstrapping`, executing NVRAM resets prior to loading OpenCore will erase the boot option. Therefore, for significant implementation updates, such as was the case with OpenCore 0.8.4, an NVRAM reset should be executed with `Bootstrapping` disabled, after which it can be re-enabled.

5. **LauncherPath**
 Type: plist string
 Falseful: Default
 Description: Launch path for the **LauncherOption** property.
 Default points to `OpenCore.efi`. User specified paths, e.g. `UEFI/Boot/launcher.efi`, can be used to provide custom loaders, which are supposed to load `OpenCore.efi` themselves.

Gruss Coban

Beitrag von „vw440“ vom 9. Januar 2022, 13:09

so ich hab nun denk ich soweit es zum laufen bekommen....einzig mein trackpad verbindet sich nach systemstart nicht mehr direkt.....aber damit kann ich gegebenfalls leben.....ich hab gelesen es gibt generell probleme mit monterey im zusammenspiel mit bluetooth

Beitrag von „RenStad“ vom 9. Januar 2022, 13:49

Für Dein Bluetooth-Problem versuch mal BluetoolFixup.kext