

Erledigt

ASRock X299 Extreme4 Anleitung und Frage zu Dualboot und Bootfehler

Beitrag von „hackschlafosx“ vom 4. Juni 2020, 19:07

Hallo!

Dank der [Anleitung](#) von DSM2 habe ich es ziemlich schnell geschafft, ein lauffähiges System aufzusetzen. Vielen Dank nochmal für die Mühe die Infos in Anleitungen und Videos festzuhalten.

Da es im Netz kaum Hackintosh-Infos für das ASRock Extreme4 x299 gibt, hier mal meine bisherigen Schritte und im Anhang meine EFI - dazu sei gesagt, dass ich überhaupt kein Experte bin und mir das Wissen überall im Netz zusammengesucht und versucht habe logisch auf meine Hardware anzuwenden - ich freue mich daher über Input und Verbesserungsvorschläge.

Mein System:

i7 7800x Prozessor mit Scythe Ninja 5 CPU Lüfter

ASRock X299 Extreme4 Mainboard - Firmware 1.6*

32GB DDR4 Corsair 3200Mhz RAM

ATI Radeon RX 570 Grafikkarte 8GB

600W BeQuiet Pure Power 11 Netzteil

Zyxel USB WLAN Stick NWD660S - bringt Treiber für OSX mit, läuft ohne Probleme nativ

Kingston A400 480GB SSD -> OSX Mojave 10.4.6

Samsung 850 EVO 500GB SSD -> Win 10

Samsung 470 SSD 60GB -> OSX Sierra (für Final Cut Pro 7)

*auf die aktuelle Firmware, die auch Cascade Lake-X CPUs unterstützt, habe ich noch nicht ge-

updated - wegen der beschriebenen Probleme hier: [Cascade Lake X Board ? Cascade Lake Refresh Firmware macht Probleme mit macOS? Kein Thema!](#)

BIOS-Einstellungen - zurücksetzen auf Optimized Settings (F9) und dann prüfen, ob folgende Einstellungen gesetzt sind:

OC Tweaker -> CPU Configuration -> Intel Speed Shift Technology -> bei "Out Of Band" bootet das System nicht mit meiner Konfi, deswegen auf Native Mode

OC Tweaker -> DRAM Tweaker -> Load XMP Setting

OC Tweaker -> DRAM Tweaker -> DRAM Frequency auf DDR4-3200 einstellen

Advanced -> CPU Configuration -> CPU C States Support auf Manual stellen und sichergehen, dass CFG Lock auf "Disabled" steht -> das ist wohl das Gleiche wie „MSR 0xE2“

Advanced -> USB Configuration XHCI Hand-off auf "Enabled"

Boot -> CSM "disabled"

Dann habe ich nach der Anleitung von DSM2 Clover auf einem Bootstick installiert - hier muss man die Clover-Version 5103 nehmen, da in den neueren OCQuirks nicht enthalten ist. Ich habe dann im Nachhinein, erst Clover 5104 und jetzt 5118 installiert. Das aktuelle OCQuirks (mit OCQuirks.efi, OcQuirks.plist und OpenRuntime.efi) habe ich hier runtergeladen: <https://github.com/ReddestDream/OcQuirks/releases>

Die alte OCQuirks.efi und FwRuntimeServices.efi müssen gelöscht/überschrieben werden.

Die [TSCAdjustReset.kext](#) habe ich an meinen Prozessor angepasst.

Bis dahin hat alles ziemlich gut funktioniert.

USB-Anschlüsse waren erstmal nicht alle verfügbar. Mit Hilfe von Hackintool, USBInjectAll.kext und dem Port Limiter Patch (10.4.4+ von hier: [USB Port Limit Patches \(Zusammenfassung\)](#)) habe ich das in den Griff bekommen (siehe DSM2-Anleitung). Alle Ports funktionierten jetzt, auch die vom externen ASMedia Controller (ASM3142).

Das einzige, woran ich dann ein wenig verzweifelt bin: Der zusätzliche USB3.1 10Gbps Controller ASM3142 von ASMedia wird in Hackintool fälschlicherweise als ASM2142 angezeigt und mit dem Kürzel SLT1 aufgeführt (statt XHCI oder ähnlichem). Und in der USB-Port-Liste tauchen diese Ports nicht auf. Der Intel-USB 3.0 Controller taucht nativ als XHCI auf und die Ports werden auch alle aufgelistet.

Über die DSDT-ACPI-Patches in Clover habe ich SLT1 in XHC, XHC_, XHC2 oder XHCI umbenannt (der Hex-Konverter im Clover Konfigurator hilft dabei). Leider hat das alles nichts gebracht. Die Ports waren in Mojave da, aber nicht im Hackintool. In der Kombination XHC1 für den ASMedia und XHCI für den Intel Controller erschienen die ASMedia Ports zwar, allerdings nur als HS02, HS03 und HS04 und am USB Typ-C Port gab es dann nur noch USB 2.0 Geschwindigkeit - in Hackintool wird entsprechend auch 480Mbps Dev Speed angezeigt. Die Benennung des USB-C Ports in "TypeC+Sw" hat leider auch nicht geholfen. Der von Hackintool exportierte USB-Kext wird zwar geladen, aber diese Angabe leider nicht.

Nach sehr langer Suche und sehr vielen reboots nehme ich das jetzt so hin - ohne die ACPI renames funktionieren die Ports ja dennoch mit voller Geschwindigkeit.

Evtl. probier ich irgendwann mal noch eine andere Methode bzw. lese mich mal noch ausführlicher in das USB-Thema ein, mal sehen. Kennt ihr diese Phänomene? Weiß jemand vielleicht eine Lösung?

Nach einem Blick in das Boot.log von Clover habe ich nur noch diesen **DSDT-Patch** aktiv gelassen:

PC00 -> PCI0

~~Bei den anderen hieß es: Pattern not found, already patched.~~

EDIT 7.6.20: Im Bootlog gibt es anscheinend mehrere Stellen, an denen die Patches ausprobiert werden - ich poste mal ausführlich, welche gefunden werden und welche nicht.

npci=0x2000 braucht's als Boot-Argument, sonst hängt's beim Booten.

Stand-By scheint zu funktionieren, liegt wahrscheinlich am Boot-Argument darkwake=0. Allerdings werden beim Sleep die USB-Laufwerke ausgeworfen - ich habe irgendwo eine Lösung für dieses Problem gelesen, muss ich nochmal nachschauen.

Noch eine Beobachtung: Gebe ich im Clover Configurator bei CPU -> Type 0x0a05 oder 0x0a02 ein, hängt das System beim Boot. Dafür steht bei Kernel and Kext Patches bei FakeCPUID jetzt 0x050654, so dass bei "Über diesen Mac" auch "Intel Core i7" steht und nicht "unbekannt". Ich habe auf einen schnellen Blick nicht wirklich herausgefunden, was die Sektion "CPU" eigentlich so macht. Die Zahlen oben habe ich in einer Anleitung für den 7800X u.a. hier gefunden: [High Sierra 10.13 - Installations Problem\(e\) ; Gigabyte X299 UD4 - i7-7800X - GeForce® GTX™ 1080 Ti](https://www.hackintosh-forum.de/forum/thread/48815-asrock-x299-extreme4-anleitung-und-frage-zu-dualboot-und-bootfehler/)

Ansonsten habe ich gestern den halben Tag damit gekämpft **Dualboot** und den TimeOut im Clover Boot Menu einzurichten. Um die richtige SSD anzusprechen und sie nach Wunsch zu benennen, muss man für OSX im Clover Configurator bei GUI einen Custom Entry anlegen und hier die „Partition UUID“ (kann man im Hackintool nachschauen oder "diskutil info") eingeben - mit Häkchen bei "Title/FullTitle". Und um das Boot-Laufwerk von OSX anzusprechen muss es die UUID des AppleAPFS Partitions-Containers sein und nicht das des darin liegenden Haupt-Laufwerks. Um Windows zu booten muss man die Windows EFI-Partition als Start-Laufwerk in Clover auswählen.

Damit im Clover Boot Menü dieses Laufwerk standardmäßig ausgewählt ist, gibt man eigentlich bei "Boot" -> "Default Boot Volume" die gleiche UUID wie bei "GUI" ein, oder? Leider funktioniert das bei mir nicht. Egal, in welcher Syntax ich das eingebe:

```
3AE9B5F8-62D3-4A52-D2C1-2E86418B9D16
```

```
HD(2,GPT,3AE9B5F8-62D3-4A52-D2C1-2E86418B9D16,0x64028,0x2C3CE1E8)
```

```
PciRoot(0x0)\Pci(0x17,0x0)\Sata(0x1,0xFFFF,0x0)\HD(2,GPT,3AE9B5F8-62D3-4A52-D2C1-2E86418B9D16,0x64028,0x2C3CE1E8)
```

oder einfach nur "Mojave" - so heißt das Volume

Überseh ich irgendwas? Oder ist die Clover Version 5118 buggy? Ich hatte das schonmal mit ner Clover-Version, die plötzlich die Icons im Boot-Menü nicht mehr anzeigen wollte.

Normalerweise müsste jetzt, wenn man unter "Boot" einen Timeout ausgewählt hat, ein Timer runterzählen und dann automatisch vom Default Boot Volume starten. In meinem alten System ging das ohne Probleme. Hier nicht. Auch hier habe ich einiges ausprobiert. Meine letzte These: Mein Mainboard unterstützt wohl kein natives NVRAM, irgendwie hat das wohl damit zu tun?

Ich habe dann bei einer neuen Re-Installation von Clover die RC-Skripte installiert und die EmuVariableUefi.efi in den Clover -> Driver kopiert. NVRAM scheint jetzt zu gehen, allerdings leider nicht der Timeout beim Boot-Menü. Wenn ich als Default-Boot-Volume die Mojave SSD auswähle geht's.

Hat jemand eine weitere Idee? Liegt es wirklich am NVRAM oder an etwas anderem?

Was manchmal beim (Re-)Booten auftritt ist folgende Meldung:

```
Error allocating 0x11ced pages at 0x00000000247ac000 alloc type 2  
Couldn't allocate runtime area
```

Das hat wohl mit der Speicher-Adressierung zu tun, oder? Passiert sowohl mit OCQuirks als auch AptioMemoryFix...

[Hier](#) empfiehlt [CMMChris](#) bei einem ähnlichen Problem:

Zitat

Wenn schon OcQuirks vorgeschlagen wird dann bitte auch komplett...

Für Z390er Boards ist folgendes nötig:

- In der Config Datei von OcQuirks "DevirtualiseMMIO" auf true setzen
- SSDT-PM.aml nach /EFI/CLOVER/ACPI/patched
- EmuVariableUefi muss raus und auch sämtliche anderen Aptiofixe müssen entfernt werden
- Den Haken bei KernelXCPM kannst du dir sparen, das hat mit dem Problem gar nichts zu tun

Nun läuft nativer NVRAM und die Memory Allocation sollte funktionieren. Vor dem Start im Clover Boot Menü F11 drücken damit der NVRAM leer ist.

Alles anzeigen

Allerdings ist das für ein Z390 Board und in der DSM2-Anleitung wurde "DevirtualiseMMIO" nicht auf true gesetzt.

Anscheinend hilft dieses sehr komplexe Vorgehen: https://www.reddit.com/r/hacki...couldnt_allocate_runtime/ oder kennt ihr eine andere Lösung?

So, das war jetzt ne Menge Text, ich hoffe, mir kann jemand bei meinen verbliebenen Problemen helfen - würde mich sehr freuen!

Und, meldet euch gerne, wenn ihr Fragen habt - ich helfe gerne, soweit ich kann...

Im Anhang noch mein EFI-Ordner, das USB-Kext für das Mainboard und ein paar Diagnosedateien.

Beitrag von „DSM2“ vom 7. Juni 2020, 01:07

Der Cascade Lake X thread ist gleichzeitig die Lösung für das MSR Problem als auch die ACPI Geschichte.

Ich muss das ganze noch aktualisieren, da ich mittlerweile das Vollständige Puzzle gelöst habe, jedoch aufgrund von fehlender Zeit nicht festgehalten habe.

Die Abweichungen bei der Installation bzw Inbetriebnahme liegen einfach darin das sich die Boards unterscheiden

und sozusagen jedes Board eine etwas andere Behandlung benötigt, weshalb du auch meinen Guide nicht 1 zu 1 übernehmen kannst.

Bei den X Plattformen ist es einfach so und diese benötigen nun mal eine besondere Herangehensweise als 0815 Plattformen.

In Bezug auf ASM3142:

Das ist nichts verkehrtes oder unübliches, wenn mehrere Controller auf dem Board liegen, dann wird nicht jeder unbedingt über XHC angebunden sein,

was grundsätzlich auch nicht tragisch ist, der Controller fällt dadurch nicht unter das Port Limit und muss auch nicht zwingend in einer Kext hinterlegt werden.

Ausnahme wäre wenn dieser Devices nicht annehmen würde, dann müsste man gegebenenfalls nachhelfen aber dies ist hier nicht der Fall.

Hintergrund ist der das macOS den ASM3142 durchaus kennt und das ganze nativ schon korrekt läuft.

PC00-PCI0 ist schon lange nicht notwendig und wird in meinem Guide eigentlich auch nicht empfohlen.

npci=0x2000 - anteilige Prefetch Memory deaktivierung - Board spezifisch, kann unter OpenCore zum Beispiel gar nicht notwendig sein.

In Bezug auf Time Out sollte es eigentlich keine Probleme geben solange das richtige Laufwerk hinterlegt ist,

da ich jedoch nichts mit Clover zu tun habe, kann ich dir da auch leider nichts zu sagen ob es ein Bug ist oder so.

Demnächst wird auch mein X299 Guide komplett auf OpenCore umgeschrieben, spätestens dann gibt es auch keinen Support für Clover von mir.

Das Bios kann man grundsätzlich unlocken falls das CFG Lock im Bios nicht disabled werden kann, der Flashvorgang ist aber etwas umständlich sein bei ASRock.

Du kannst theoretisch aber auch den Bios support kontaktieren und falls die Option nicht vorhanden ist, nach einer Bios Version fragen die Unlocked ist.

Ansonsten gibt es die weitere Möglichkeit den neuen Efi Treiber von Brumbaer zu nutzen, welcher der Unlock des MSR ohne Bios patchen ermöglicht :

[CFG Lock](#)

Beitrag von „hackschlafosx“ vom 7. Juni 2020, 16:58

Hallo, danke für deine ausführliche Antwort, ich weiß das zu schätzen!

CFG-Lock disabled kann ich bei meinem Mainboard auswählen, ob das noch bei nem Update auf die aktuelle Version (Cascade Lake X support) noch möglich sein wird, weiß ich nicht.

Zitat

Der Cascade Lake X thread ist gleichzeitig die Lösung für das MSR Problem als auch die ACPI Geschichte.

Meinst du mit ACPI Geschichte diese Fehlermeldung?

Code

1. Error allocating 0x11ced pages at...

Ich habe vorhin mal einen Open Core Bootstick samt SSDTs [erstellt](#), und komme zumindest bis ins Bootmenu und ein wenig darüber hinaus - genau genommen bis:

Code

1. 47:467 00:107 OCSMC: SmcReadValue Key 4D535463 Size 1
2. 47:548 00:081 OCSMC: SmcReadValue Key 4D534163 Size 2

Danach kommen ganz viele <NUL><NUL><NUL>...

Ich schau mir morgen nochmal meine DSDTs an und gehe nochmal die config.plist durch. Evtl. muss ich bei den ACPI-Patches auch schauen, ob ich etwas vergessen habe, ich habe bisher nur EC und Plug.

Ich bin sehr gespannt auf den überarbeiteten Guide! Danke!

Beitrag von „hackschlafosx“ vom 10. Juni 2020, 21:58

Kurzes Update: OpenCore mit dem Asrock Extreme4 geht!

Standardeinstellungen wie im [Guide für Skylake-X](#) beschrieben + Vorgehensweise um NVRAM zu aktivieren. Den [Sanity Checker](#) fand ich auch sehr praktisch.

Es kann sogar sein, dass es die angepassten ACPI-Files nicht braucht. Woran erkennt man eigentlich, ob die ACPI-Files etwas "bringen"?

Nur das Bootargument: npci=0x2000 brauchts, sonst hängt's bei:

```
OCSMC: SmcReadValue Key 4D535463 Size 1
```

```
OCSMC: SmcReadValue Key 4D534163 Size 2
```

In diversen Foren war geschrieben worden, dass man mit folgenden Schaltern herumprobieren

sollte, hat aber bei mir nichts gebracht:

ProtectUefiServices - true/false

SetupVirtualMap - true/false

DevirtualiseMmio - true/false

Dual Boot mit Timer funktioniert jetzt, USB Sleep leider immer noch nicht, es werden beim Sleep sogar Verbindungen zu Netzwerkspeichern getrennt...

Ich werde mein EFI im entsprechenden Thread posten, sobald ich etwas weiter bin.