

# Projekt Asus X99 Deluxe 2 in NEU und OC

Beitrag von „MacGrummel“ vom 25. September 2024, 20:42

## Einen Schritt weiter:



Ob ich die Lösung der System-Einfrier-Probleme bei meinem X99er und X299er wirklich gefunden habe, weiß ich nicht. Aber manchmal hilft halt die Recherche im eigenen Archiv am Besten weiter:

a.) die beiden Dicken sind mit Sonoma ne ganze Weile ohne Probleme mit einem iMacPro-SmBIOS gelaufen. Auch mit der (wieder-)Umstellung auf MacPro 7,1 ging das Einschlafen auch nicht sofort los: erst mit der 6. oder 7. Sequoia-Beta-Version im August und den zeitgleichen Sonoma-Updates. Da ist also irgendwas in macOS umgestellt worden.

Also hab ich beide wieder auf iMacPro umgestellt. Und sie laufen stabiler. Aber zumindest der 299er nicht stabil genug. Etwa 10 Minuten nach dem Anspringen des Sequoien-Waldes (Bildschirmschoner mit Uhr) bleibt er zuverlässig stehen, der X99er nicht. Also am X299 erstmal wieder Ventura..



b.) ich hab das gepatchte X99-BIOS wieder gefunden mit Version 2101, MSR Unlock & Microcode-Update.

c.) früher war mehr Lametta, äh nein: der X99er ist jahrelang mit seinem Broadwell-6-Kerner i7-6850k ohne TSC-Sync-Kexte gelaufen. Wer lesen kann, ist eindeutig...



Macht er seit heute früh auch wieder, leider bin ich mit meinen typischen Photoshop- und Netzwerkarbeiten für heute schon auswärts durch, muss mal künstlich für Stress sorgen ..

Mit dem Stress kommt der alte X99er recht gut zurecht. Ich darf also recht eindeutig den CpuTscSync.kext zur Ursache der Abstürze erklären, denn der ist ja jetzt abgeschaltet als einzige Veränderung mit dem MacPro-SmBIOS.

d.) wenn es so bleibt, braucht ja nur der 299er einen BIOS-Patch für den Skylake-X-10-Kerner i9-9900X. Der Broadwell wird ja schon richtig erkannt -- und Broadwell-E 6-Kerner sehe ich nicht in der Patch-Liste. Wenn ich den aktuellen Patch richtig verstanden habe.

Dann bleibt natürlich die Frage offen, ob nicht eigentlich an einem der TSC-Sync-Kexte gebaut werden müsste. Ich kann mir nicht vorstellen, dass das nur bei mir passiert..

```
patches.txt

# Patch string format
# FileGuid SectionType PatchType:FindPatternOrOffset:ReplacePattern
# Please ensure that the latest symbol in patch string is space

# Possible section types:
# PE32 image 10
# Position-independent code 11
# TE Image 12
# DXE Dependency 13
# Version information 14
# User interface string 15
# 16-bit code 16
# Guided freeform 18
# Raw data 19
# PEI Dependency 1B
# SMM Dependency 1C
# Please do not try another section types, it can make the resulting image broken

# Possible patch types:
# P - pattern-based, first parameter is a pattern to find, second - a pattern to replace
# O - offset-based, first parameter is hexadecimal offset, second - a pattern to replace
# Patterns can have . as "any possible value" symbol

#-----
# OSX CPU Power Management patches
# Remove lock from MSR 0xE2 register
#-----

# PowerMgmtDxe | Haswell
F7731B4C-58A2-4DF4-8980-5645D39ECE58 10 P:75080FBAE80F89442430:EB080FBAE80F89442430

# PowerMgmtDxe | Haswell-E
F7731B4C-58A2-4DF4-8980-5645D39ECE58 10 P:0FBA6C24380F:0FBA7424380F

# PowerManagement | Sandy Bridge with ME 8.xx, Ivy Bridge
8C783970-F02A-4A4D-AF09-8797A51EEC8D 10 P:75080FBAE80F89442430:EB080FBAE80F89442430

# PowerManagement | New SB-E/IB-E
8C783970-F02A-4A4D-AF09-8797A51EEC8D 10 P:0FBA6C24380F:0FBA7424380F

# CpuPei | Sandy Bridge with ME 7.xx, old SB-E/IB-E
28B5AFA9-FF33-417B-8497-CB773C2B938F 10 P:800018EB050D0000:000018EB050D0000

# PpmInitialize | Broadwell-E
3FFCAE95-23CF-4967-94F5-16352F68E43B 10 P:0FBA6C24400F:0FBA7424400F

# SiInit | Skylake
299D6F8B-2EC9-4E40-9EC6-DDAA7EBF5FD9 10 P:75080D00800000:EB080D00800000
299D6F8B-2EC9-4E40-9EC6-DDAA7EBF5FD9 12 P:75080D00800000:EB080D00800000

# SiInit | Kaby Lake
299D6F8B-2EC9-4E40-9EC6-DDAA7EBF5FD9 10 P:81E1008000033C1:9090909090909090
299D6F8B-2EC9-4E40-9EC6-DDAA7EBF5FD9 12 P:81E1008000033C1:9090909090909090

# PpmInitialize | Skylake-X
3FFCAE95-23CF-4967-94F5-16352F68E43B 10 P:742CB9E2000000:752CB9E2000000

# CpuInitPei | Skylake-X
01359D99-9446-456D-ADA4-50A711C03ADA 12 P:BE0080000023CE0B:BE0000000023CE0B

# PpmInitialize | 0xE2 & 0x1AA unlock for C422 Chipsets & Some X299 Chipsets
3FFCAE95-23CF-4967-94F5-16352F68E43B 10 P:4889442430...0FBAE8:4889442430...0FBAE0
3FFCAE95-23CF-4967-94F5-16352F68E43B 10 P:4889442450...0FBAE8:4889442450...0FBAE0

# PpmInitialize | 0xE2 & 0x1AA unlock for C620 Chipsets
3FFCAE95-23CF-4967-94F5-16352F68E43B 10 P:480BC2BA00200000:480BC2BA00000000
3FFCAE95-23CF-4967-94F5-16352F68E43B 10 P:480BC2BA00800000:480BC2BA00000000

# AMI APTIO V S3Nvram (Socket 2011-v3 MB: C612, X99)
62DC08AC-A651-4EE9-AF81-EAA9261E9780 10 P:B9080000004184:B93B0000004184

# CpuMpDxe (Socket 2011-v3 MB: Asus-X99)
40BEAB40-CECE-4909-B133-20A413AE19E9 10 P:B9E200000045:B93B00000045
```

Und bei dem Patch verstehe ich wieder mal nur Bahnhof. Aber das war beim ersten auch nicht besser, bis ich aus allen vorhandenen Anleitungen eine zusammengefasst hatte.

Die BIOS-Versionen müssen dann übrigens vor dem Aufflashen umbenannt werden als X99D2.CAP und X299D2.CAP, aber das ist halt Asus.

Was mir dabei gerade auffällt: ich habe ja das BIOS 802 von 2019 auf dem X299, weil der mit den 4-stelligen Varianten anfangs nicht so recht mit macOS laufen wollte. Unterdessen soll vor einem Update auf 3801 ff. jedes Mal eine ME-Firmware geladen werden. Keine Ahnung, wie das ohne Windows gehen soll, ist ne .exe-Datei. Ein aktuelles BIOS, das läuft, hätte ja sicher einige Vorteile..