

Mojave auf Lenovo X250 / Anpassung nach der Installation

Beitrag von „Gymnae“ vom 27. April 2019, 11:15

EDIT: Ich arbeite gerade noch an Audio über Klinke und höherer Lautstärke. Bisher klappt es mit einer anderen SSDT, aber bevor ich das hier veröffentliche teste ich noch weiter.

Nach ein paar Iterationen habe ich meine x250 *fast* so perfekt:

- Bluetooth klappt meistens
- Handover funktioniert
- 1080p Display kann skaliert werden
- 3rd party 1080p Display kann auch in Windows10 PWM geregelt werden
- i5 5300 läuft im idle mit 800 MhZ
- iGPU läuft ebenfalls mit niedrigerem Takt im idle
- Undverolting vom CPU package
- Nur die nötigen USB ports werden injected
- TrackPoint klappt 1A
- SD-Card reader klappt...manchmal
- miniDP mit Audio

Bluetooth

Hier habe ich mich an einen Fred aus diesem Forum gehalten: [Lösungsansatz für Bluetooth-Probleme mit der DW1560 und DW1830](#) - Die dort verlinkte BrcmPatchRAM2__debug.kext hilft bisher am besten, aber eben nicht perfekt. Scheinbar meldet sich die DW1830 Bluetooth Einheit als unterschiedliches device an, je nachdem ob wir mit Batterie- oder Netzbetrieb laufen.

Für den Moment langt es

Handover

Hier musste ich nicht viel vornehmen außer der Platzierung der [AirportBrcmFixup.kext](#)

1080p display Skalierung

FHD im x250 ist nix gut für Brillenträger. Das ist alles winzigklein. Die UI von MacOS beherrscht ja keine freie DPI Skalierung, weil so modern ist sie dann doch nicht.

Es gibt aber Abhilfe: <https://github.com/xzhih/one-key-hidpi>

Dieses einfache Skript aktiviert die Skalierungsoptionen der Retina displays auch für non-retina Geräte.

Ich wechsele aktuell munter zwischen HiDPI und nativ 1080p, da die Batterielaufzeit mit simulierten Retina schon einen merkbaren Knick nach unten erhält. Aber die Augen sind wichtiger.

Alternativ skaliere ich in Chrome alles auf 125% - leider geht das nicht Systemweit.

Anmerkung: So sehr ich Firefox bevorzuge, bei skalierten Auflösungen frisst Firefox deutlich mehr Leistung. Ein Youtube Video z.B. z.B. nutzt einen Kern voll aus und greift nicht auf GPU Beschleunigung zurück. Hier ist Chrome deutlich besser und Safari am besten.

1080p PWM Helligkeitsregelung in Windows

Das x250 hat eine Whitelist, so kann eigentlich nicht jedes Display eingebaut und geregelt werden. Mit einer BIOS Modifikation kann man diese Whitelist aber entfernen. So stehen dann weitere Displayoptionen offen. Ich habe aktuell ein modifiziertes (PCB beschnitten) Sharp LQ125M1JW33 im Einsatz. Dieses Display hat 8bit und fast 110% sRGB Farbraumabdeckung. Nice.

Eine Anleitung für die BIOS Modifikation findet ihr hier:
https://www.reddit.com/r/think...550x250_screen_whitelist/

i5 5300 mit 800MHz idle

Zwei Schritte sind dafür notwendig:

1. MacBookAir7,2 SMBIOS statt das oft empfohlene MacBookPro12,1
2. CPUFriend.kext in Clover's "Other" Ordner
3. <https://github.com/acidanthera...rter/ResourceConverter.sh>

Mit den ResourceConverter erstellt ihr eine `CPUFriendProvider.kext`, die in Clover's "Other"

Ordner gehört. Bei mir hat folgender Befehl funktioniert:

```
./ResourceConverter.sh --kext  
/System/Library/Extensions/IOPlatformPluginFamily.kext/Contents/PlugIns/X86PlatformPlugin.kext/937CB26E2E02BB01.plist
```

iGPU mit niedrigem idle Takt

Wie viele der Schritte hier habe ich diese Info von einem großartigen repository des github user "qwerty". An dieser Stelle kopiere ich seinen Text:

Zitat

Install the included AppleGraphicsPowerManagement_Broadwell.kext into /Library/Extensions. Make sure to chmod, chown it properly and rebuild the kext cache.

Running `ioreg -rlw0 -p IOService -c AGPMController -k Machine | grep GT` after rebooting will tell you if the kext has been installed properly - GT2Floor and GT3Floor should be 6.

Die angesprochene .kext findet ihr in seinem repository: <https://github.com/qwerty12/X250-Hackintosh>

Vermutlich bedeutet dieser Ansatz Arbeit mit jedem Update...

Undervolting vom CPU package

Noch bin ich mir nicht sicher, ob das Voodoo oder effizient ist. Auch ein Tip von "qwerty".

Mit der voltageshift.kext und deren .sh skript an eure möglichen und gewünschten Werte herantasten und dann, der Anleitung für eine dauerhafte Einstellung der Werte folgend, mit undervolting leben:

<https://github.com/sicreative/VoltageShift>

Nur die nötigen USB ports werden injected

Dank [hackintool](#) geht das nun echt einfach. Es erstellt eine .kext, die man in Clover's "Other" folder legt. Dann braucht es keine SSDT oder InjectAll. So könnt ihr auch sicherstellen, dass Webcam und Bluetooth als internal erkannt werden. Hackintool hat eine Anleitung integriert, ich gehe daher nicht weiter darauf ein.

TrackPoint

Das ist auch ganz easy. Mit einer aktuellen Version von VoodooPS2Controller.kext und der [SSDT-Thinkpad_Clickpad.dsl](#) als .aml kompiliert klappt das, quasi, out of the box

SD-Card reader

Achtung, ich habe die Kext in meinem Archiv im Einsatz, aber es ist bekannt das diese .kext zu reboots führen kann, da der Kernel abschmiert. Das passiert zufällig beim entfernen von SD-Karten. Manchmal werden SD-Karten war erkannt, aber "leer" gemounted. Ich würde daher für "business critical" Anwendungen nicht auf den internen SD-card reader vertrauen

miniDP mit Audio

Hier injecte ich die Platform-ID 0x16260006 und setze FixHDA & AddHDMI auf true

Zu guter letzt:

Meine config.plist im Archiv hat keine gültigen Seriennummern, diese müsst ihr euch noch selbst erstellen. Wie das so üblich ist.

Fazit:

Bluetooth nervt noch ein wenig, sonst bin ich happy. Beim screen habe ich zu großzügig am PCB geschnibbelt damit er in das Rückgehäuse vom Deckel passt, so dass ab und an das Bild korrupt wird, aber auch das kriege ich vielleicht noch hin.