

Erledigt

iGPU Encoding mit HD4000 - i5-3570K - "connectorless"

Beitrag von „mitchde“ vom 11. März 2018, 21:33

Die Liste bei der Auswahl ig-id kennt eben diesen ..07 nicht. Man muss das dann per Hand eingeben.

Diese ig-igs headless gibts (keine Bildausgabe!) sind:

```
With i7-3770(K) and iMac13,2 system definition, you can use ig-platform-id "0x01620007"  
With i7-6700(K) and iMac17,1 system definition, you can use ig-platform-id "0x19120001"  
With i7-7700(K) and iMac18,3 system definition, you can use ig-platform-id "0x59120003"  
With i7-8700(K) and iMac18,3 system definition, you can use ig-platform-id "0x59120003"  
These in-platform-ids are "headless" entries meaning that they will not work with a monitor plug
```

Wichtig dabei ist auch dass das MacModell von Haus aus iGPU+GPU hat, also kein MacModell ohne GPU - sprich nur iGPU hat - ist. bei mir **iMac13,2** - das hat **iGPU und GPU** = passt zur i5-3xxx CPU und das Modell hat auch GPU. Wählt man ein Macmodell das nur eine iGPU hat funktioniert das mit dem connectorless soviel ich weiß nicht mehr - selbst wenns System ansonsten läuft.

BSP: wählt man zb. iMac14,1 - was von der CPU durchaus auch gehen kann, hat man ein Modell was nur eine iGPU hat:

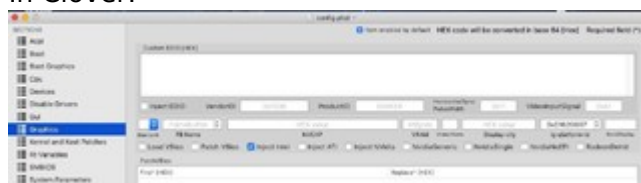
Mac-031B6E74CF7F642A	Dictionary	1 Schlüssel/Wert-Paare
iGPU	Dictionary	2 Schlüssel/Wert-Paare
Mac-06F11F11946D27C5	Dictionary	2 Schlüssel/Wert-Paare

Somit läuft wohl das System, man soll aber diese iGPU Enc Sache so nicht hinbekommen (weils halt aus Apples Systemblick keine GPU hat).

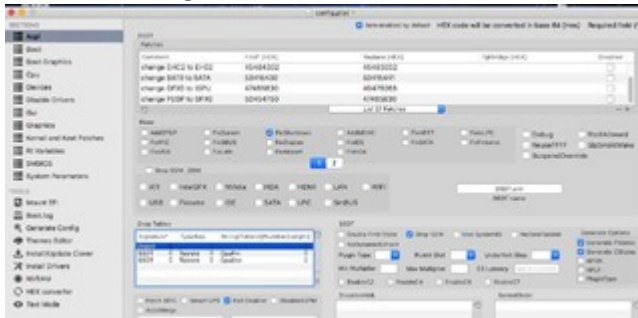
Bei iMac13,2 und auch den anderen oben connectorless Modellen sieht das dann so aus, immer neben iGPU auch ein oder mehrere GPUs.

Mac-FC02E91DDD3F6A4	Dictionary	3 Schlüssel/Wert-Paare
iGPU	Dictionary	2 Schlüssel/Wert-Paare
Vendor10deDevice09e0	Dictionary	4 Schlüssel/Wert-Paare
Vendor10deDevice11a2	Dictionary	4 Schlüssel/Wert-Paare
Vendor10deDevice11a3	Dictionary	2 Schlüssel/Wert-Paare

In Clover:



So sieht Clover für die 2 replace Einträge aus (Dafür gibts Einträge aus Liste). Wichtig dabei ist die Reihenfolge wie auf dem Screenshot. Reihenfolge matters 😊



Shiki (neueste) <https://github.com/vit9696/Shiki...d/2.2.4/2.2.4.RELEASE.zip> .

Code

1. v2.2.4 Added an AppleGVA patch to fix Sandy Bridge accelerator name (shikigva=128)
2. Added /usr/libexec/AirPlayXPCHelper, ffmpeg, Quick Look, and Photos to the renderer whitelist
3. Added Broadwell patches for compatible renderer (discrete GPU hardware acceleration)
4. Implemented GPU detection for compatible renderer, whitelist, and snb name (you can remove shikigva)
5. Fixed invalid Sandy Bridge compatible renderer patches
6. Disabled automatic iTunes crashfix hack for 10.13.4 and higher (you may have to reset DRM once you upgrade)

Bei mir war es so, dass sobald Intelögraphicsfixup auch da war, das ganze **nicht** mehr ging. Also nur LILU+Shiki (**ohne! Intelgraphicsfixup**), sowie Whatevergreen (habe AMD) usw. habe ich als kexte.

Ich glaube Intelgraphicsfixup ist **nur dann** nötig, wenn man die iGPU auch als Bildausgabe Device nutzt! Und scheint daher **connectorless** Modus ohne Bildausgabe und damit ohne dass die iGPU im Systemprofiler auftaucht, zu stören.

Im BIOS habe ich Intel GPU enabled, 64 MB (wobei das wohl bei connectorless egal ist! - soviel ich weiß spielen die xy MB nur eine wichtige Rolle bei ig-id mit Bildausgabe!) , und MAX