

Erledigt

Wie wichtig ist die iGPU im Hacki und gibt es eine iGPU in einem echten Mac?

Beitrag von „griven“ vom 4. Dezember 2018, 23:14

Nun sie werkeln ja auch harmonisch zusammen redbelt wenn eben die Voraussetzungen dafür geschaffen wurden. Es ist eben ein Trugschluss zu denken das SMBIOS hätte nur was mit der Optik zu tun bzw. war das ganz früher mal so aber in der Zwischenzeit hat sich einiges getan



Mal ein konkretes Beispiel:

Ausgehend von Deiner CPU würde man wohl als Modell am ehesten den iMac 18.3 wählen wollen denn das ist ein ziemlich perfektes Abbild von Deinem Setup. Die CPU ist identisch (i7-7700K) und auch die Grafik ist nahezu identisch RX580 vs. Radeon Pro 580 besser geht es SMBIOS technisch gar nicht. Gucken wir dem Bruder Deines Systems mal unter Die Haube bzw. hinter das Display. Wir haben also einen iMac mit der BoardID Mac-BE088AF8C5EB4FA2 das ist gut zu wissen denn damit bewaffnet können wir mal in der AppleGraphicsDevicePolicy gucken was der so können sollte. Der iMac 18.3 wird mit der config3 gelistet und die wiederum besagt folgendes:

Code

1. <key>Config3</key>
2. <dict>
3. <key>GFX0</key>
4. <dict>
5. <key>EDID</key>
6. <dict>
7. <key>index</key>
8. <integer>0</integer>
9. </dict>
10. <key>FeatureControl</key>
11. <integer>12</integer>
12. <key>unload</key>
13. <false/>
14. </dict>
15. <key>IGPU</key>
16. <dict>

17. <key>unload</key>
18. <true/>
19. </dict>
20. </dict>

Alles anzeigen

Wir wissen nun das Gerät hat 2 Grafikeinheiten einmal die dGPU und die iGPU ferner wissen wir Apple erwartet das die iGPU im ACPI als solche zu finden ist und es erwartet wird das die dGPU als GFX0 im System vorhanden ist. Nun ist es bei nicht Apple Geräten oft so das die iGPU als GFX0 definiert ist im ACPI und die dGPU sich unter PEGP oder PEG verbirgt. Genau wissen wir das in Deinem Fall aber erst wenn wir einen Blick in die DSDT gewagt haben. Wichtig ist jedenfalls schon mal das wir diese, von Apple verlangte, Nomenklatur einhalten ansonsten führt das unweigerlich zu Chaos.

Was sagt uns unser Blick in den Kext noch nun er sagt uns das für die iGPU der Key `unload` auf `true` gesetzt wird und das meint nichts anderes als das der Framebuffer für die iGPU nicht geladen bzw. wieder entladen werden soll. So weit so gut denn den Framebuffer (Bildschirmtreiber für die Darstellung von 2D Inhalten auf Bildschirmen) brauchen wir ja auch gar nicht denn das übernimmt ja sowohl im iMac als auch in Deinem Fall die AMD Karte. Unser erstes Arbeitspaket lässt sich also hieraus ableiten:

1. In die DSDT gucken und herausfinden wie die Grafikkarten im ACPI angesprochen werden.
2. Dafür sorgen das dies macOS konform geschieht entweder über Clover oder direktem DSDT Patch.
 - 2a. Wenn über Clover dann darauf achten das man als erstes GFX0 in iGPU umbenennt und erst dann zum Beispiel PEGP in GFX0.
3. Mittels Clover oder DSDT einen Connectorless PlattformID injecten (0x59120003).
4. Dafür Sorge tragen das es ein DEVICE IMEI gibt entweder durch umbenennen des vorhandenen MEI Devices oder durch einfügen eines Fake Devices.

Stellt sich jetzt natürlich die Frage warum der ganze Driss mit dem Apple konformen umbenennen im ACPI und die Antwort darauf ist relativ einfach! MacOS ist nicht Windows und hat für alles und jeden einen Treiber und es ist auch nicht Linux das sich für jede Vendor und DeviceID schon was passendes aussucht sondern macOS erwartet einen Mac und geht daher davon aus die nötigen Informationen zum einen aus dem SMBIOS zu ziehen zum anderen aber auch aus dem ACPI und eben so sicher wie es davon ausgeht dort die entsprechenden Informationen zu finden macht es davon auch abhängig welche Funktionen auf der erkannten

Hardware bereit gestellt werden (AirPlay, H264 en- und decoding, HVEC en- und decoding usw.)

Nachdem wir das erledigt haben sind wir dem Ziel schon ein gutes Stück näher gekommen denn unser nicht ganz echter iMac kennt nur bestenfalls seine iGPU und auch eine (bewusst nicht seine) dGPU und kann zumindest schon mal mit den Informationen über die iGPU arbeiten. Warum schreibe ich eine und nicht seine dGPU auch das ist wieder relativ leicht zu erklären denn der Apple verwendet ja nichts von der Stange sondern die Polaris Karten in den Mac's sind nicht irgendwelche Polaris Karten (haha) sondern es sind Radeon Pro Karten. Der Fachmann staunt und der Laie wundert sich was macht nun eine Brot und Butter Polaris Karte zu einer Radeon Pro und was zum Teufel ist Radeon Pro überhaupt?

Hier kommt wieder der von [CMMChris](#) bereits erwähnte Fakt ins Spiel. Apple ist auf die glorreiche Idee gekommen so genannte "egpu developer kits" zu verkaufen in denen eine AMD RX verbaut war und die es Entwicklern ermöglichen sollte schon mal an der Leistungsfähigkeit der Metal API sowie der kommenden dGPU Generation zu naschen ohne dafür konkret einen Mac in Petto zu haben der das bereits verbaut hatte. Jetzt hat man also eine ganze Horde von Entwicklern die sich die Grafikkiste gekauft haben und wie überzeugt man die nun davon das Dingen wieder in die Ecke zu stellen und einen hochpreisigen Mac zu kaufen ach eigentlich ganz einfach man sparrt sich ein paar wenige, aber essentielle, Funktionen auf die sich nicht über Thunderbold tunneln lassen, lötet die gleiche GPU auf das LogicBoard verpasste dem Dingen aber vorsichtshalber vorher eine andere DeviceID und zack hat man eine Radeon Pro ersonnen. Der Fachmann schüttelt den Kopf und kauft trotzdem denn gibt ja keine Alternative und der Laie wundert sich schon wieder und huldigt Apple oder so ähnlich...

Was sagt uns das nun? Die RX580 ist genau so Pro wie die Radeon Pro 580 man muss macOS das nur verklickern und das geht indem man der die RX entsprechend verkleidet zum Beispiel hiermit: [AMDRadeonPro.kext.zip](#) wobei es sich hierbei um einen alten Bekannten handelt nämlich um den PropertyInjector.kext von [Brumbaer](#). Natürlich kannst Du das Dingen jetzt nicht einfach reinwerfen und hoffen das es geht vielmehr musst Du in der info.plist des Kexts die Device und VendorID unter PCIPrimaryMatch noch an Deine Gegebenheiten anpassen aber ist das einmal getan steht dem fast echten iMac 18.3 nichts mehr im Wege.

Puh nun ist es mehr Text geworden als ich eigentlich schreiben wollte aber vielleicht hilft das ein wenig Licht ins Dunkle zu bringen...