

[Anleitung] iGPU - Framebuffer patching & Blackscreen Problem beheben für Clover und OpenCore

Beitrag von „hackmac004“ vom 11. November 2020, 21:45

Dabei geht es darum die iGPU mit voller Grafikleistung nutzbar zu machen und endlich keinen Blackscreen mehr zu erhalten.

Da einige, die zum ersten Mal einen Hackintosh installieren, nur eine iGPU zur Verfügung haben und damit gar kein Bild bekommen, erkläre ich zuerst, wie man überhaupt erstmal ein Bild bekommt ohne Grafikleistung (vesa mode).

In den weiteren Punkten sind verschiedene Möglichkeiten beschrieben wie du deine iGPU mit voller Leistung zum laufen bekommen kannst. Dabei wird auf ein möglichst großes Spektrum an Systemen eingegangen.

Grundvoraussetzungen

Kexte:

- aktuelle Version von **Lilu.kext** und **Whatevergreen.kext**
- <https://www.sl-soft.de/kext-updater/>

Software:

- Hackintool - <https://github.com/headkaze/Hackintool/releases>
- Clover Configurator
- TextEdit, ProperTree, Xcode oder PlistEdit Pro

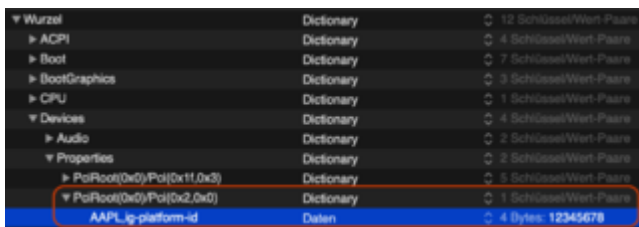
1. Vesa mode für Clover und OpenCore

Um mit Clover den vesa mode zu erhalten reicht es aus in der config.plist mit Clover

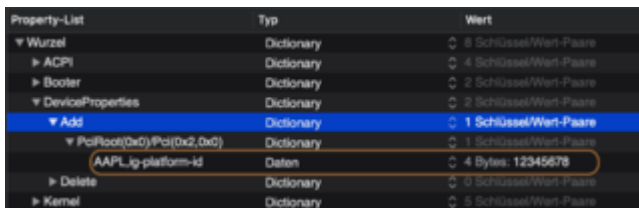
Configurator unter dem Punkt Graphics irgendeinen 8-stelligen Zahlenwert bei ig-platform-id einzutragen und einen Haken bei Inject Intel zu setzen.



oder hier in die config eintragen



Bei OpenCore setzt du irgendeinen 8-stelligen Zahlenwert genau an diese Stelle.



Nun bekommst du ein Bild und kannst macOS installieren, vorausgesetzt alles andere stimmt in deiner EFI 😊

Windows Nutzer, die noch kein macOS zur Verfügung haben, können diese Einstellungen mit ProperTree vornehmen.

Du wirst bemerken, dass das Dock nicht transparent ist, Glitches vorkommen und das der GPU Speicher bei etwas um die 8 MB liegt. Um volle Grafikleistung zu erhalten müssen die richtigen Framebuffer (FB) gepatcht und bei den DeviceProperties eingetragen werden. War das

patchen erfolgreich, treten diese Phänomene nicht mehr auf.

Die gepatchten FB sind bei jedem mainboard verschieden. Das heißt nur weil jemand die gleiche iGPU hat kannst du nicht die funktionierenden FB übernehmen. Nur wenn es das gleiche mainboard ist und die gleiche device-id der iGPU.

Für alles Weitere solltest du im Vorfeld folgendes tun:

Kopiere den EFI Ordner, mit dem du installieren konntest vom Stick in die EFI Partition deiner Festplatte. Boote von der Festplatte und schau ob alles soweit läuft. Funktioniert das, lege den Stick bei Seite und nehme nun alle Änderungen an der config.plist vor die sich auf deiner Festplatte befindet. Gibt es Fehlversuche, kommst du mit dem Stick wieder zurück in dein System.

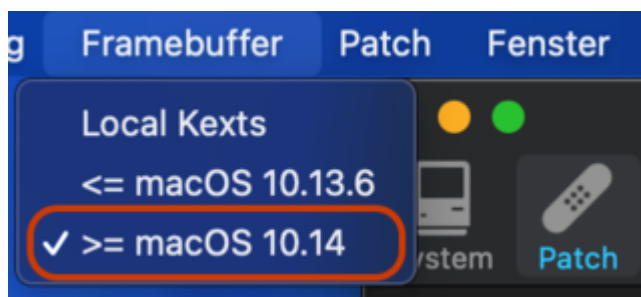
Außerdem musst du bei Clover alle Schritte, die du gemacht hast, um den vesa-mode zu erhalten, in der config auf der Festplatte rückgängig machen!

Das ig-platform-id Feld leeren und den Intel Inject Haken entfernen.

Bei OC musst du nichts machen, da die ig-platform-id in den nächsten Schritten erneuert wird.

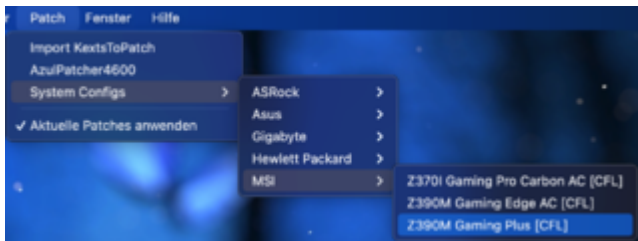
2. Voreingestellte Patches im Hackintool

Öffne das Hackintool und schau zuerst nach, dass diese Optionen, entsprechend deiner macOS Version, angewählt sind:



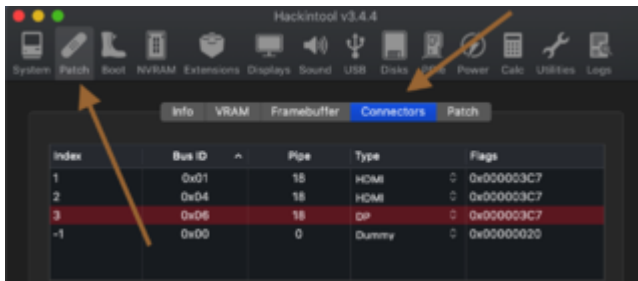


Nun kannst du hier schauen ob deine FB patch im Hackintool schon hinterlegt ist.

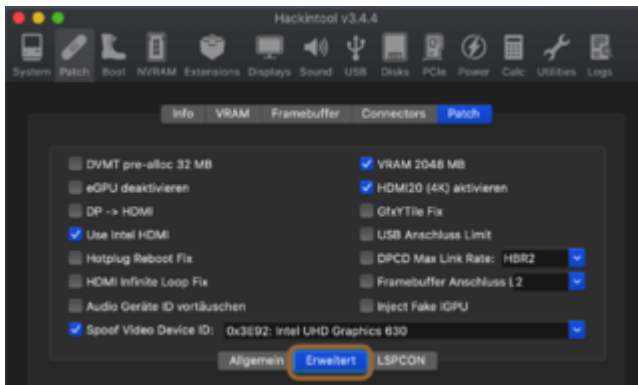
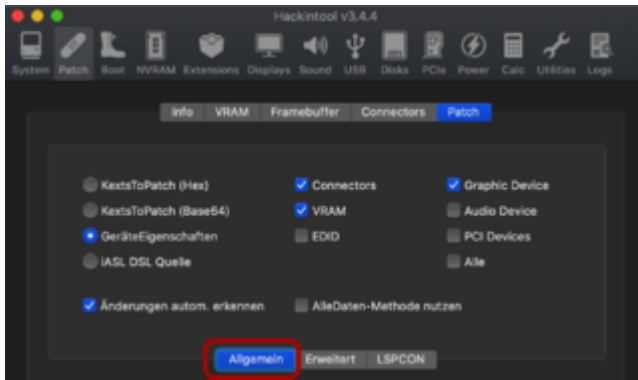


Hast du Glück und dein board ist enthalten, werden dir an dieser Stelle die FB angezeigt.

Dies dient zunächst als gute Grundlage.



Nun gehst du auf den Reiter Patch, wo es noch 3 Fenster für Einstellungen gibt. Geh alles durch und mit den Einstellungen, die in den Bildern zu sehen sind, sollte es in den meisten Fällen funktionieren.

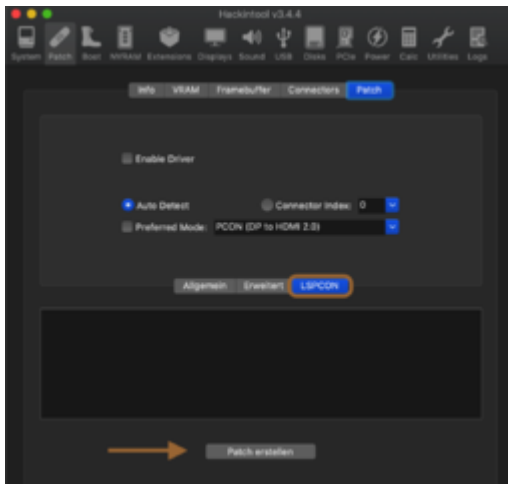


Mit Spoof Video Device ID wählst du deine passende device-id aus. Du findest sie auf der Intel Ark Seite. Für meinen Fall (i5 8400) hier

<https://ark.intel.com/content/...cache-up-to-4-00-ghz.html>

an dieser Stelle.

Prozessorgrafik	
Prozessorgrafik 1	Intel® UHD-Grafik 630
Grundtaktfrequenz der Grafik	350 MHz
Max. dynamische Grafikfrequenz	1,05 GHz
Max. Videospeicher der Grafik	64 GB
4K-Unterstützung	Yes, at 60Hz
Max. Auflösung (iCMI 1.4)	4096x2304@24Hz
Max. Auflösung (DP)	4096x2304@60Hz
Max. Auflösung (DP – Integrierter Flachbildschirm)	4096x2304@60Hz
Unterstützung für DirectX®	12
OpenGL® Unterstützung	4.5
Intel® Quick-Sync-Video	Ja
Intel InTru 3D-Technik	Ja
Intel® Clear-Video-HD-Technik	Ja
Intel® Clear-Video-Technik	Ja
Anzahl der unterstützten Bildschirme 1	3
Geräteerkennung	0x3E32



Überprüfe nochmals alle Einstellungen sorgfältig, weil durch bestimmte Modifikationen sich andere Werte verändern können. Anschließend klickst du auf Patch erstellen.

3. Einfügen des Patches in die config.plist

Monte vorher genau die EFI Partition, in welche der Patch eingefügt werden soll.

Dann gehst du im Hackintool an diese Stellen.



Nun speicherst du die config und startest neu.

4. Patch mittels Whatevergreen (WEG)

Meines Erachtens hat sich WEG soweit entwickelt, dass allein durch die folgenden Angaben in den DeviceProperties die richtigen Ports „automatisch“ beim boot gefunden werden können.

Dies ist ein Beispiel für Coffee Lake i5 8400. Du musst die platform-id und die device-id entsprechend deinem System noch anpassen.

Bei OC

Property-List	Typ	Wert
▼ Wurzel	Dictionary	8 Schlüssel-Wert-Paare
▶ ACPI	Dictionary	4 Schlüssel-Wert-Paare
▶ Booter	Dictionary	2 Schlüssel-Wert-Paare
▼ DeviceProperties	Dictionary	2 Schlüssel-Wert-Paare
▼ Add	Dictionary	1 Schlüssel-Wert-Paare
▼ PciRoot(0x0)/Pci(0x2,0x0)	Dictionary	3 Schlüssel-Wert-Paare
AAPL_ig-platform-id	Daten	4 Bytes: 07009B3E
device-id	Daten	4 Bytes: 923E0000
framebuffer-patch-enable	Daten	4 Bytes: 01000000
▶ Delete	Dictionary	0 Schlüssel-Wert-Paare

In Textform sehen die Einstellungen so aus:

Code

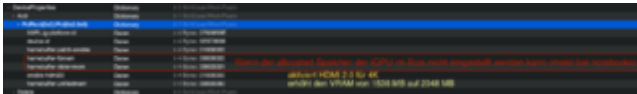
```
1. </dict>
2. <key>DeviceProperties</key>
3. <dict>
4. <key>Add</key>
5. <dict>
6. <key>PciRoot(0x0)/Pci(0x2,0x0)</key>
7. <dict>
8. <key>AAPL,ig-platform-id</key>
9. <data>
10. BwCbPg==
11. </data>
12. <key>device-id</key>
13. <data>
14. kj4AAA==
15. </data>
16. <key>framebuffer-patch-enable</key>
17. <data>
18. AQAAAA==
19. </data>
20. </dict>
21. </dict>
22. <key>Delete</key>
23. <dict/>
24. </dict>
```

Alles anzeigen

Für Clover:



Unter bestimmten Voraussetzungen können bzw. müssen diese zusätzlichen Einträge noch genutzt werden.



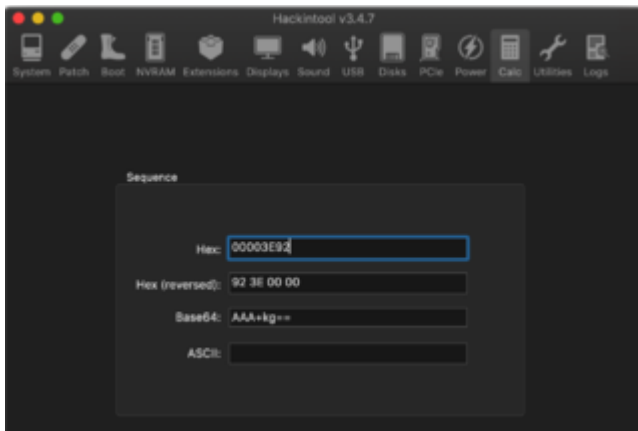
Falls du Clover verwendest, kannst du sie auch so in den CloverConfigurator schreiben.

Wie du deine richtige device id findest ist unter Punkt 2 erklärt.

Ist diese z.B. als 0x3E92 angegeben, bedeutet das in Hex: 00 00 3E 92

Dies muss jedoch als Hex reversed eingetragen werden was dann 92 3E 00 00 ist.

Diese Umkehrung lässt sich ganz leicht mit dem Hackintool Calculator errechnen.



Die ig-platform-id muss der der verwendeten CPU Architektur entsprechen und auch im Hex reversed Format eingetragen werden.

Beispielhaft für Coffee Lake Desktop ig-platform-id

3E 9B 00 07 : Hex

07 00 9B 3E : Hex reversed

Im nächsten Abschnitt sind einige Empfehlungen für die entsprechende platform-IDs zu finden.

Tip: Ein Displayport lässt sich mit dieser Methode eher lauffähig bekommen. HDMI ports auch, aber die Wahrscheinlichkeit ist etwas geringer.

Anmerkung: Bei HDMI kommt es vor, dass das Kabel nach dem boot ab- und wieder angesteckt werden muss, damit es Bild gibt.

Das bootarg `igfxonln=1` kann in diesem Fall helfen, sodass per HDMI nach dem boot und dem sleep gleich das Bild erscheint.

Im Clover Configurator bleiben alle Felder in der Sektion *Graphics* und in der Sektion *Devices /Fake ID* leer. Es sollen auch keinerlei Haken gesetzt werden. Dies erledigt alles Whatevergreen.

In manchen Fällen hat jedoch ein Haken bei Inject Intel geholfen. Dies wäre also optional möglich.

5. Multi-Monitor-Betrieb

Falls du mehrere Monitore an der iGPU betreiben möchtest und es kommt zu Fehldarstellungen, dann können dir diese 2 bootargumente helfen.

```
igfxagdc=0 & igfxonln=1
```

(credits für den Fund [KarlDieter](#))

6. Framebuffer herausfinden

Sind deine FB nicht hinterlegt, du findest sie nirgends im Internet oder die WEG Methode greift nicht, bleibt dir leider nichts weiter übrig als alle erdenklichen Kombinationen auszuprobieren.

Als erstes musst du die passende Intel Generation und Platform ID für deine Hardware auswählen.



Platform-id Übersicht für Desktop und Laptop für verschiedene CPU - Architekturen

Code

1. Gen 2: Sandy Bridge (Intel HD Graphics 2000/3000)
2. - Support started with OS X 10.7.x and ended with macOS 10.13.6
3. - Metal support is not available
4. - AAPL,snb-platform-id (desktop): 0x00030010 (default)
5. - AAPL,snb-platform-id (laptop): 0x00010000 (default)
- 6.
- 7.
8. Gen 3: Ivy Bridge (Intel HD Graphics 2500/4000)
9. - Support started with OS X 10.8.x
10. - AAPL,ig-platform-id (desktop): 0x0166000A (default), 0x01620005
11. - AAPL,ig-platform-id (laptop): 0x01660003 (default), 0x01660009, 0x01660004
- 12.
13. Gen 4: Haswell (Intel HD Graphics 4200-5200)
14. - Support started with OS X 10.9.x
15. - AAPL,ig-platform-id (desktop): 0x0D220003 (default)
16. - AAPL,ig-platform-id (laptop): 0x0A160000 (default), 0x0A260005 (recommended)
0x0A260006 (recommended)
- 17.
18. Gen 5: Broadwell (Intel HD Graphics 5300-6300)
19. - Support started with OS X 10.10.2
20. - AAPL,ig-platform-id (desktop): 0x16220007 (default)
21. - AAPL,ig-platform-id (laptop): 0x16260006 (default)
- 22.
23. Gen 6: Skylake (Intel HD Graphics 510-580)
24. - Support started with OS X 10.11.4
25. - AAPL,ig-platform-id (desktop): 0x19120000 (default)
26. - AAPL,ig-platform-id (laptop): 0x19160000 (default)
- 27.
28. Gen 7: Kaby Lake (Intel HD Graphics 610-650)
29. - Support started with macOS 10.12.6
30. - AAPL,ig-platform-id (desktop): 0x59160000 (default), 0x59120000 (recommended)
31. - AAPL,ig-platform-id (laptop): 0x591B0000 (default)
- 32.
33. Gen 8: Coffee Lake & Comet Lake (Intel UHD Graphics 630)
34. - Support started with macOS 10.13.6 (17G2112) / 10.14 beta 4 (18A336e)
35. - AAPL,ig-platform-id (desktop): 0x3EA50000 (default), 0x3E9B0007 (recommended)
36. AAPL,ig-platform-id (laptop): 0x3EA50009 (default)

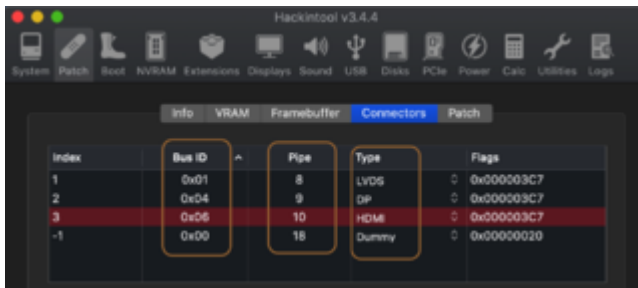
Alles anzeigen

Weiterführende Informationen findest du hier:

<https://github.com/acidanthera.../Manual/FAQ.IntelHD.en.md>

Die richtigen FB ergeben sich aus einer Kombination von Index (1-3), BusID (1-6), Pipe (8, 9, 10, 18) und Typ (HDMI, DVI, DP, LVDS).

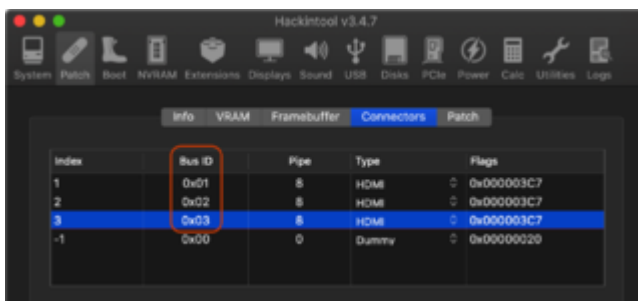
Für laptops wählst du als Index=0 und Type=LVDS damit das interne Display läuft.



Konzentrier dich erstmal auf nur einen Port (Type), den du vorzugsweise verwenden willst.

Beginne erstmal nur mit einer Pipe (8, 9, 10 oder 18)

Die erste Einstellung soll dann, anhand eines HDMI Beispiels, so aussehen. Du kannst natürlich auch den DP oder DVI wählen. Index wird immer 1-3 verwendet, wodurch du Überschneidungen von Index und BusID vermeidest.



Nun machst du teilweise Schritt 2 (patch erstellen) und 3 (patch in die config einfügen) von weiter oben. Speichern --> Neustart.

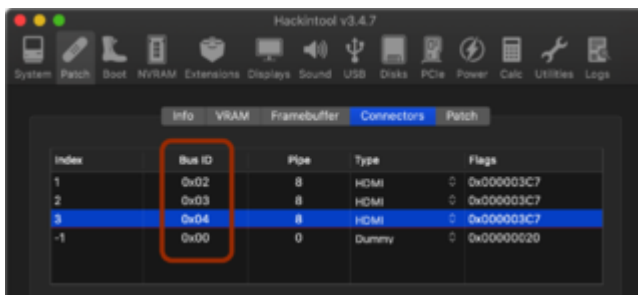
Schlägt ein Versuch fehl, kommst du per Stick wieder ins System und kannst weitere

Veränderungen vornehmen.

Nun werden nur noch die BusIDs geändert nach diesem Schema, wobei jede Spalte für eine config.plist steht.

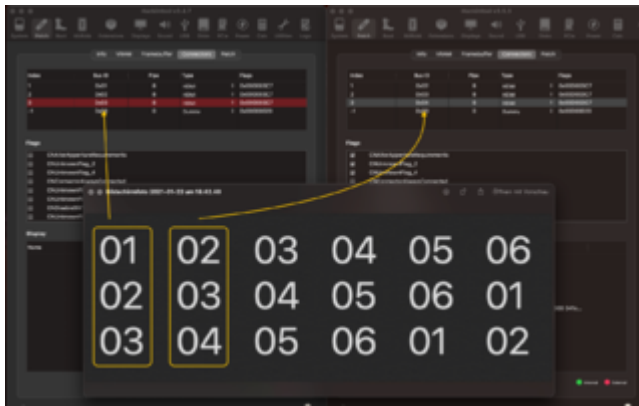
```
01 02 03 04 05 06
02 03 04 05 06 01
03 04 05 06 01 02
```

Die nächste config würde dann das beinhalten.



....und so weiter.....

Zum besserem Verständnis: So ist das gemeint



Wenn die 6 Möglichkeiten für Pipe 8 durchgespielt sind, wiederhole das gleiche, nur setzt du die Pipe auf 9.

Den nächsten Durchgang dann mit Pipe 10 und den nächsten mit Pipe 18.

Leider gibt es sehr viele Kombinationsmöglichkeiten und hoffentlich ist die zündende dabei.

Viel Erfolg! 😊

Das Thema FB patching der iGPU hat sich mir immer als eines dargestellt, wo immer wieder ungeklärte Fragen und Phänomene auftreten. Dieser Thread kann gerne dazu genutzt werden um solche zu stellen und zu klären. Verbesserungsvorschläge und nützliche Ergänzungen sind natürlich auch erwünscht.

Beitrag von „AkimoA“ vom 12. November 2020, 11:12

Endlich macht es mal jemand Daumen hoch dafür 😊

Was mir aufgefallen ist, du hast in deinem Beispiel den "framebuffer-stolenmem" Parameter verwendet was evtl bei den meisten für Verwirrung sorgt da man diesen nur braucht wenn man im Bios keine Option an den Speichereinstellungen hat zu schrauben .

Das kommt bei Coffee Lake Desktop recht selten bis garnicht vor und wird oft nur bei mobilen und vermehrt bei älteren iGPUs verwendet.

Wenn es dir bei stolenmem darum geht den VRAM zu erhöhen was nicht empfohlen ist wäre das framebuffer-unifiedmem.

Beitrag von „5T33Z0“ vom 12. November 2020, 11:57

Super Guide. Der fehlte noch. CTRL+D 😊

Zwei Fragen hätte ich noch:

1. Wenn ich mit Hackintool Sachen ausprobieren, wirkt sich das nur auf die plist aus die ich damit bastle aus oder verändert es auch intern etwas an der laufenden Konfiguration, sodass ich die kaputt machen kann?
2. Das mit dem definieren der Anschlüsse habe ich noch nicht verstanden Ich verstehe auch die Begriffe nicht - LVDS = extern oder intern? Ich habe zum Beispiel ein Lenovo T530 IvyBridge Laptop mit nem Dock und verwende

AAPL,ig-platform-id 04006601, mit 03006601 bleibt das externe Display aus.

Das Bild wird durchgeschleust zum Dock. Das Dock hat sowohl einen DVI als auch einen DisplayPort. Der DVI Port funktioniert. Der Displayport funktioniert nicht. Der Mini-DP am Laptop selbst funktioniert jedoch. Wie findet man sowas raus?

Danke!

Beitrag von „hackmac004“ vom 12. November 2020, 12:02

[AkimoA](#) Danke für dein feedback und deinen Hinweis. Stimmt, wenn man den Speicher im Bios einstellen kann, braucht es `framebuffer-stolenmem` und auch `framebuffer-fbmem` nicht. Dann würde ja `framebuffer-patch-enable` erstmal ausreichen.

Ich werde das noch ändern mit den entsprechenden Hinweisen.

`framebuffer-unifiedmem` habe ich bei mir im Einsatz um den VRAM auf 2048MB zu erhöhen. Bisher hab ich keinerlei Nachteile dadurch merken können. Richtig, auf der WEG Seite wird das nicht empfohlen. Wäre interessant zu erfahren wieso es nicht empfohlen ist.

5T33Z0 Danke für's feedback 😊

1. Wenn du im hackintool Einstellungen unter Patch veränderst, dann werden diese erst im System aktiv nachdem sie in deine config eingefügt wurden nach einem Neustart.

2. LVDS ist für interne Displays. Ich werde die Formulierung an der Stelle noch präzisieren.

Da ich keinen Laptop habe zum testen, kann ich leider auch nicht sagen, wie dort die BusID und die Pipe eingestellt werden müssen. Soweit ich erfahren konnte, läuft das interne, wenn `Index=0` ist und `Typ=LVDS` ist. Kannst du das vielleicht bei dir mal testen mit unterschiedlichen BusIDs und Pipes?

Dann ist die `AAPL,ig-platform-id 04006601` bei dir der richtige FB. Es stehen mehrere zur Auswahl und man muss auch dort den richtigen erwischen, wobei man sich erst an die empfohlenen halten sollte und wenn der nicht läuft dann die anderen ausprobieren.

Wenn du im Hackintool auf Patch--> Connectors geht, werden dir die funktionierenden Ports rot hinterlegt, sofern angeschlossen. Wenn noch ein Index frei ist, kannst du da mit versch. Einstellungen versuchen deinen DP am Dock zu aktivieren.

Beitrag von „AkimoA“ vom 12. November 2020, 12:46

[Zitat von hackmac004](#)

framebuffer-unifiedmem habe ich bei mir im Einsatz um den VRAM auf 2048MB zu erhöhen. Bisher hab ich keinerlei Nachteile dadurch merken können. Richtig, auf der WEG Seite wird das nicht empfohlen. Wäre interessant zu erfahren wieso es nicht empfohlen ist.

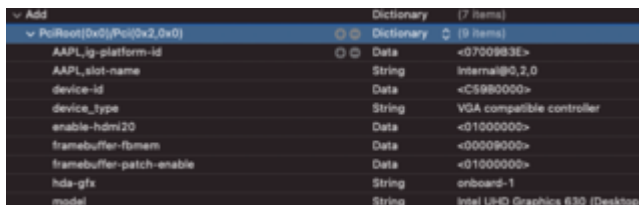
Ich persönlich bau das bei anderen auch mit ein wenn sie grundsätzlich genug RAM verbaut haben da es ja davon abgeht und sehe es gleich, aber eigtl. geht es bei der Sache nicht darum mehr VRAM für die Interne GPU zu haben sondern generell diese zu unterstützen bei 4K Auflösung soweit ich mich erinnere.

Beitrag von „SchmockLord“ vom 12. November 2020, 13:03

Hi, ich versuch grad jemanden auf Github zu helfen, der Probleme mit Audio über HDMI/DP hat:
<https://github.com/SchmockLord...e-Z490-Vision-D/issues/92>

Ich kann es nur schwer testen, weil ich selber das Vision D nicht mehr laufen hab und ich werd meinen miniITX jetzt nicht deswegen zerlegen.

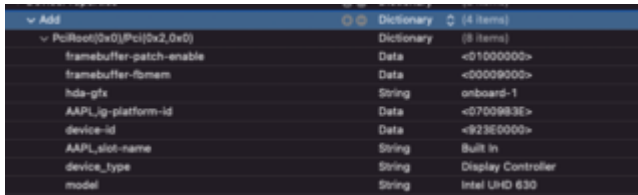
Wenn ich die Settings auf meinem AsRock Z490 Phantom Gaming ITX/TB3 benutze, funktioniert alles: Video + Audio über HDMI und DP. Bei ihm aber HDMI nicht mehr, DP ja. Audio aber nicht.



Key	Type	Value
AAPLig-platform-id	Data	<0700983E>
AAPLslot-name	String	Internal@0,2,0
device-id	Data	<C5980000>
device_type	String	VGA compatible controller
enable-hdmi20	Data	<01000000>
framebuffer-fbmem	Data	<00009900>
framebuffer-patch-enable	Data	<01000000>
hda-gfx	String	onboard-1
model	String	Intel UHD Graphics 630 (Desktop)

Wenn er die folgende Config benutzt, funktioniert wiederum Video über DP Und HDMI. Aber

immer noch kein Audio. Hauptsächlich die Device-ID ist anders, das ist die aus dem Dortania Guide. Die oben hab ich von Intel und funzt bei mir prima.



Key	Type	Value
framebuffer-patch-enable	Data	<01000000>
framebuffer-fbmem	Data	<00000000>
hda-gfx	String	onboard-1
AAPL_lg-platform-id	Data	<0700983E>
device-id	Data	<923E0000>
AAPL_slot-name	String	Built In
device_type	String	Display Controller
model	String	Intel UHD 630

Für mich sieht es so aus, als würde er beim Vision D explizite Framebuffer Settings brauchen, ich beim meinem AsRock Z490 Phantom Gaming ITX/TB3 aber nicht.

Hab mit den Settings oben während ich hier schreibe am Mainboard DP und HDMI an meinem Samsung CRG9 angeschlossen und höre Musik über den Monitor.

Hast du ne Idee?

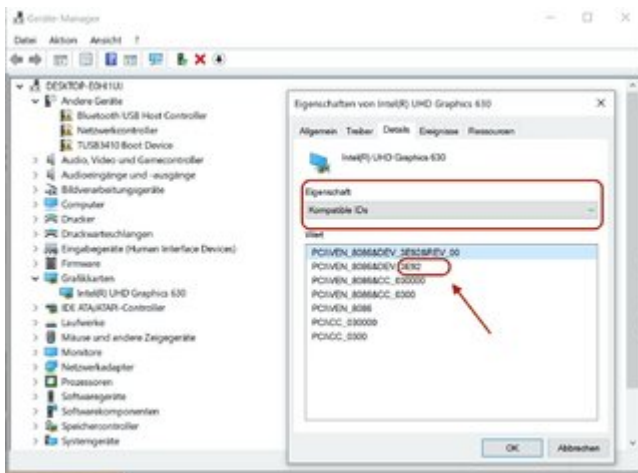
Danke übrigens für den Guide, der kam genau zur richtigen Zeit 😊

Beitrag von „hackmac004“ vom 12. November 2020, 13:59

[SchmockLord](#) Gerne 😊

Interessant, dass HDMI & DP mit einer anderen device id bei ihm funktioniert, obwohl intel eine andere angibt.

Man kann die device id übrigens auch unter Win an dieser Stelle auslesen, um wirklich sicher zu gehen.



Da bei mir Audio via DP und HDMI immer funktioniert haben, kann ich da leider nicht viel helfen.

Ich hätte getippt, dass die layout-id noch richtig eingetragen werden muss, aber selbst ohne layout-id läuft bei mir der Sound über den Monitor.

Beitrag von „KarlDieter“ vom 17. November 2020, 13:26

Vielen Dank für den Guide! Habe es für mein i5-4670k mit HD4600 befolgt und es funktioniert soweit auch mit einem Monitor.

Was allerdings leider nicht funktioniert ist Dual-Monitor. DVI oder HDMI einzeln funktionieren, beide zusammen aber nicht. Kann man da noch etwas an den Ports Feintunen und das ggf. sogar in deinem Guide ergänzen?

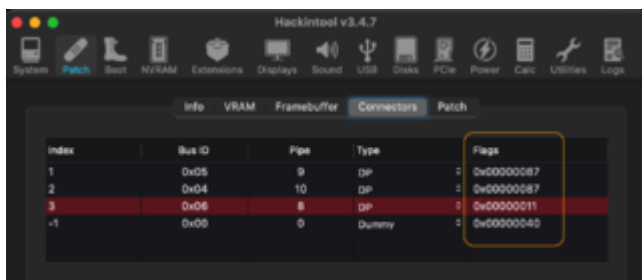
Beitrag von „hackmac004“ vom 17. November 2020, 14:19

Freut mich, dass es soweit geholfen hat. Wenn wir die Lösung finden, füge ich das gern hinzu.

Was mir immer noch ein Rätsel ist, sind die Flags. Die spielen auch eine Rolle und ich hab mich

bisher immer darauf verlassen, was das Hackintool dort einträgt.

Wenn ich nun Haswell und als platformid 0D220003 auswähle, zeigt er mir diese Flags an.



Du könntest probieren an dieser Stelle noch ein bisschen zu experimentieren.

Kannst du deine config ohne SN auch mal hochladen?

Beitrag von „KarlDieter“ vom 17. November 2020, 15:36

Sehr gerne, Config ist anbei.

Hatte in meinen Screenshots oben eine andere platformid getestet. Mit 0D220003 und den Flags aus deinem Screenshot macht es aber leider keinen Unterschied. Werde mal ein bisschen mit den Flags rumspielen, aber dass ich durch Zufall die richtige Kombination aus "CNUnknownFlags" treffe ist ja ungefähr so wahrscheinlich wie ein Lottogewinn 😄

Beitrag von „hackmac004“ vom 17. November 2020, 16:24

Als erstes würde ich es ohne das bootarg igfxonln=1 testen.

Da es bei dir soweit "nur" mit framebuffer-patch-enable funktioniert, würde ich vorschlagen, dass du es mit Schritt 5 noch probierst, wodurch die Ports deklariert in der config drin stehen.

Das sieht bei mir dann so aus also Beispiel.

PciRoot(0x0)/Pci(0x2,0x0)	Dictionary	↕ 12 Schlüssel-Wert-Paare
AAPL_ig-platform-id	Daten	↕ 4 Bytes: 07009B3E
AAPL_slot-name	String	↕ Internal@0,2,0
device-id	Daten	↕ 4 Bytes: 923E0000
device_type	String	↕ VGA compatible controller
enable-hdmi20	Daten	↕ 4 Bytes: 01000000
framebuffer-con0-busid	Daten	↕ 4 Bytes: 01000000
framebuffer-con0-enable	Daten	↕ 4 Bytes: 01000000
framebuffer-con0-type	Daten	↕ 4 Bytes: 00080000
framebuffer-patch-enable	Daten	↕ 4 Bytes: 01000000
framebuffer-unifiedmem	Daten	↕ 4 Bytes: 000800BE
hda-gfx	String	↕ onboard-1
model	String	↕ Intel UHD Graphics 630 (Desktop)

Ich gehe davon aus, dass dir im Hackintool die richtigen BusIDs etc. angezeigt werden, wenn sie rot hinterlegt sind allein durch die framebuffer-patch-enable funktion. Vondaher musst du da nicht groß suchen welche BusIDs es sind auf welchem Index.

Ich hab aber auch schon erlebt, dass das auch nicht funktioniert. Allgemein kann ich sagen, das die iGPU in Hinsicht mulitmonitoring nicht die zuverlässigste ist. Bei manchen klappt's, bei manchen leider nicht.

Beitrag von „KarlDieter“ vom 17. November 2020, 17:15

Bringt leider beides auch kein Erfolg...ich vermute es funktioniert bei mir einfach nicht mehr..

Kommando zurück! Mit Bootflag "igfxagdc=0" habe ich endlich Dual Monit😊 Allerdings ist der zweite Monitor komischerweise ein Stück abgeschnitten (siehe Anhang)...mal sehen ob der letzte Feinschliff das auch noch behebt...

Beitrag von „hackmac004“ vom 17. November 2020, 17:41

Du bist auf dem richtigen Weg😊 Sag gern Bescheid ob und wie du es geschafft hast, wenn es richtig läuft. Dann kommt das mit in die Anleitung.

Beitrag von „KarlDieter“ vom 18. November 2020, 12:30

Nach gefühlt hundert Neustarts habe ich jetzt endlich eine funktionierende Dual-Monitor Konfiguration gefunden...scheinbar bekommt Clover das mit den beiden Bootflags "igfxonln=1" und "igfxagdc=0" und der gesetzten Device ID ohne weitere Patches am besten hin. Anschließend ist wichtig, dass mein WQHD-Monitor über DVI angeschlossen wird und mein HD-Monitor über HDMI. Andersrum läuft der WQHD nicht in voller Auflösung.

Hoffe das bleibt jetzt auch updatefähig...

Beitrag von „GerhardM“ vom 23. November 2020, 10:52

[hackmac004](#) Danke für die Anleitung

Ich möchte hier noch meine Erfahrungen ergänzen. Bei meinem Hackintosh muss ich als iGPU Kennung die 0x3E9B angeben, welche eigentlich zu einer Mobilversion der i7 CPU der 9. Generation gehört (war ein Tip von [hackmac004](#)). Die Kennung für meinen verbauten i7 9700k ist eigentlich 0x3E98.

Unter Catalina (macOS 10.15.7) läuft bei mir damit die iGPU perfekt, und auch Sidecar funktioniert, allerdings bisher nicht wireless, sondern nur mit Kabelverbindung zu meinem iPad pro 10.5.

Gruß

Gerhard

Edit: doch, Sidecar klappt auch wireless

Edit 2: auch das Entsperren mit der Apple Watch klappt (habe eine Serie 0, also "uralt"), jedoch muss man nach Entsperren das Monitorkabel (bei mir HDMI), das am iGPU Port hängt, abziehen und wieder anstecken

Beitrag von „revunix“ vom 4. Dezember 2020, 18:09

Nabend,

könnte mir vielleicht mal jemand helfen? Ich habe jetzt schon ein paar Sachen ausprobiert aber irgendwie funktioniert das nicht.

Habe das ganze jetzt mal so gemacht:

Code

1. `<key>PciRoot(0x0)/Pci(0x2,0x0)</key>`
2. `<dict>`
3. `<key>AAPL,ig-platform-id</key>`
4. `<data>`
5. `AwCSPg==`
6. `</data>`
7. `<key>AAPL,slot-name</key>`
8. `<string>Internal@0,2,0</string>`
9. `<key>device_type</key>`
10. `<string>Display controller</string>`
11. `<key>enable-hdmi20</key>`
12. `<data>`
13. `AQAAAA==`
14. `</data>`
15. `<key>framebuffer-patch-enable</key>`
16. `<data>`
17. `AQAAAA==`


```
18. </data>
19. <key>framebuffer-unifiedmem</key>
20. <data>
21. AAAAgA==
22. </data>
23. <key>hda-gfx</key>
24. <string>onboard-1</string>
25. <key>model</key>
26. <string>Intel UHD Graphics 630 (Desktop)</string>
27. </dict>
```

Alles anzeigen

Allerdings bekomme ich dann eine KP und ein grünes Bild, und er bleibt beim Boot hängen.

Im UEFI habe ich den [DVMT](#) mal auf 32, 128 und 1024MB ausprobiert. Die eGPU habe ich im UEFI ebenfalls deaktiviert und auf iGPU gestellt. Muss ich da sonst noch etwas beachten?!

Ich verzweifle gerade wirklich sehr.

Es geht um die Hardware in meiner Signatur.

Danke im voraus & Grüße,

alex

Beitrag von „hackmac004“ vom 4. Dezember 2020, 18:57

[revunix](#) Für CL desktops hat sich diese platform id bewährt: 3E 9B 00 07

Das hinzufügen der device id ebenso.

Du kannst es mal mit diesen Einstellungen versuchen.



DeviceProperties	Dictionary	2 Schlüssel/Wert-Paare
Add	Dictionary	1 Schlüssel/Wert-Paare
PciRoot(0x0)/Pci(0x2,0x0)	Dictionary	4 Schlüssel/Wert-Paare
AAPL,ig-platform-id	Daten	4 Bytes: 07009B3E
device-id	Daten	4 Bytes: 923E0000
framebuffer-patch-enable	Daten	4 Bytes: 01000000
framebuffer-unifiedmem	Daten	4 Bytes: 00000080

Beitrag von „revunix“ vom 4. Dezember 2020, 20:06

[hackmac004](#) Damit bekomme ich kein Bild.

Beitrag von „hackmac004“ vom 4. Dezember 2020, 20:33

Hast du den [DVMT](#) mal auf 64 MB eingestellt?

Mit der Methode, einfach nur den patch zu enablen, funktioniert es nicht immer. Bei mir läuft damit auch kein HDMI. Dazu musste ich die richtige pipe, Index und BusID einstellen. Wie das geht ist unter Punkt 6 erklärt. Dafür würde ich die Platform id und device id nutzen, die oben im Bild zu sehen sind.

Warum möchtest du die iGPU eigentlich nutzen, obwohl du eine RX 580 hast?

Beitrag von „revunix“ vom 4. Dezember 2020, 21:51

Ich habe den [DVMT](#) auf 64 gestellt. Mein Drecks Mainboard hat nur HDMI und DVI. Und warum ich das nutzen will ist der Punkt, das ich Performance Probleme habe wenn ich etwas an den Monitoren umstelle oder in den Monitor-Sleep gehe. Ich wollte einfach nur testen ob es an der

GPU liegt. Aber ich habe mir jetzt mal ein **Gigabyte Z390 M** bestellt und werde morgen einfach mal alles umbauen. Wollte halt das umbauen erstmal vermeiden aber das scheint ja nicht richtig zu funktionieren mit der iGPU ohne mehrere Stunden zu verschwenden.

Und zu meinem eigentlichen Problem gibt es sonst keine Lösung. Offensichtlich hat das niemand.

Mit diesen Daten habe ich zwar Bild, aber nur 14MB RAM & auch nur wenn ich die RX580 ausbaue.

Code

1. `<key>PciRoot(0x0)/Pci(0x2,0x0)</key>`
2. `<dict>`
3. `<key>AAPL,ig-platform-id</key>`
4. `<data>`
5. `pT4AAA==`
6. `</data>`
7. `<key>AAPL,slot-name</key>`
8. `<string>Internal@0,2,0</string>`
9. `<key>framebuffer-stolenmem</key>`
10. `<data>`
11. `AAAwAQ==`
12. `</data>`
13. `</dict>`

Alles anzeigen

Beitrag von „ricer“ vom 4. Januar 2021, 19:47

Ich hatte gehofft mit der Anleitung vielleicht mein DP-Audio Problem zu lösen.

Hab also mal angefangen die Framebuffer für mein Board (Asus TUF Gaming Z490-Plus) herauszufinden.

Es ist eigentlich völlig egal was ich hier als Bus ID oder Pipe einstelle, Bild über Display-Port

hab ich immer. Auch ganz ohne besondere Framebuffer-Patches läuft es.

Was jedoch nicht geht ist Audio...

Lohnt es sich dennoch alle möglichen Kombinationen durchzugehen, oder mit ich da auf der falschen Fährte?

Beitrag von „g-force“ vom 4. Januar 2021, 22:48

[ricer](#) Hast Du schon unter "Dienstprogramme - Audio-MIDI-Setup" nachgeschaut, ob dein HDMI dort überhaupt auftaucht und (falls ja) als Ausgabe definiert ist?

Beitrag von „ricer“ vom 5. Januar 2021, 07:53

nein, wird dort nicht angezeigt.

kurios: wenn ich die EDID patche (zusätzlicher Eintrag "AAPL00,override-no-connect") werden mir plötzlich 2 Displays angezeigt und HDMI-Sound funktioniert.

Allerdings die gewünschten Auflösungen nicht mehr...

Beitrag von „5T33Z0“ vom 5. Januar 2021, 09:04

Ich wollte wissen, was AGDC bedeutet. Bei der Such bin ich dann auf der Seite von Whatevergreen gelandet. Da stand aber auch nix dazu. Allerdings habe ich gesehen, dass man den VESA Mode auch einfach via Boot-Argument aktivieren kann:

-igfxvesa für Intel On-Board Grafik

-radvesa für ATI/AMD Grafikkarten

Mit Clover ist es sogar noch einfacher: es gibt im Hilfe Menü eine Eintrag, um den Rechner im VESA Modus zu starten.

Beitrag von „cobanramo“ vom 5. Januar 2021, 11:38

AGDC = AppleGraphicsDeviceControl

Ein Plugin der AGDP (AppleGraphicsDevicePolicy.kext) und somit zuständig für die Ansteuerung der Grafikkarte unter macOS.

(Powerstates, Kombination aus GPU und iGPU abhängig vom Smbios)

Regelt eben das Verhalten der Grafikkarte unter macOS

Gruß Coban

Beitrag von „easy6“ vom 18. Januar 2021, 18:16

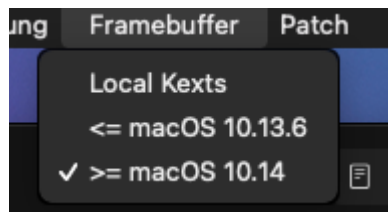
Hallo,

ich bin gerade dabei mich hier einzulesen, habe schon vieles, auch Tipps & Werte aus dem Forum probiert, nicht funktioniert.

Frage, es wird am Anfang darauf hingewiesen....

[Zitat von hackmac004](#)

Öffne das Hackintool und schau zuerst nach, dass diese Optionen, entsprechend deiner macOS Version, angewählt sind:



...ich habe 11.1, das taucht hier garnicht auf, soll ich nun den angehakten Eintag wählen oder geht das für Big Sure nicht?

Beitrag von „hackmac004“ vom 18. Januar 2021, 18:26

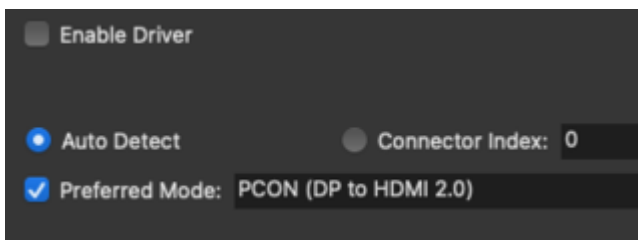
>= heißt größer gleich, also für Mojave und alles was danach kommt.

Beitrag von „easy6“ vom 18. Januar 2021, 18:38

...Ahhh,

ich bin so verwurstelt, sorry , ich muss erstmal was essen 😊

Mein Board hat zwar HDMI 2.0 aber über DP mit Zusatzchip, müsste das hier aktiviert werden?



Beitrag von „clickmikes“ vom 10. März 2021, 10:14

Hallo Leute

Ich bräuchte mal Eure Hilfe bei meinem neuesten Experiment.

MB ist Asus Prime Z490M-Plus mit ner i7-10700 (OpenCore 0.6.7) und ich will das ganze mit 2 Monitoren zum laufen bringen.

Funktioniert auch soweit ganz gut, aber beim Start krieg ich kein richtiges Bild, sondern schwarze Streifen über beide Bildschirme (siehe Foto)

Wenn ich DVI ausstecke und wieder einstecke, ist das Bild OK.

Hat einer der Experten einen tipp, woran das liegen könnte??

Danke schonmal im Voraus 😊

Michael

Beitrag von „hackmac004“ vom 10. März 2021, 11:07

[clickmikes](#) Hast du schon diese beiden bootargs drin `igfxagdc=0 & igfxonln=1?`

Beitrag von „clickmikes“ vom 10. März 2021, 12:22

[Edit:]

Jetzt läuft alles.

Ich hab was gefunden unter diesem Titel:

[GUIDE] General Framebuffer Patching Guide (HDMI

Black Screen Problem)

und auf Seite 144 #1,437 beschreibt er in etwa das was bei mir passiert und über das CNConnectorAlwaysConnected Flag.

Das hab ich dann in meinem Patch (mit Hilfe vom Hackintool) so geändert und was soll ich sagen: Stabiler Grafik output auf beiden Bildschirmen nach Reboot und Systemstart...

Die Flags in conx-alldata sehen bei mir jetzt so aus: CF030000 (statt vorher C7030000)..

[hackmac004](#) Ja, hat leider nix gebracht.

Ich hatte vermutet, dass es vielleicht mit der Reihenfolge meiner Connectors zu tun hat, deswegen hab ich im Patch Con0 und Con2 getauscht. So habe ich im IORegExplorer die beiden Bildschirme auf den ersten beiden Framebuffer Einträgen und der dritte ist leer. ... leider ohne Erfolg.

Con0 ist jetzt DVI, Con1 ist HDMI und Con2 ist DP.

Mir scheint, das ist ein Sync Problem oder hat was mit dem GFX clock zu tun, aber ich hab nix gefunden, was ich probieren könnte...

Beitrag von „GerhardM“ vom 10. März 2021, 19:50

Ich hatte das gleiche Problem, habe es bisher als "Schönheitsfehler" abgetan, aber mit der Änderung des CNConnectorAlwaysConnected Flags ist es jetzt hier auch weg, cool 😊

Ich habe das Flag ebenfalls über das Hackintool gesetzt, dann aber mittels Clover Configurator (Clover ist noch immer mein Favorit 🤪) in die config.plist eingebracht, indem ich unter devices->properties die Properties Keys framebuffer-con0-flags und framebuffer-con2-flags mit jeweils dem Properties Value CF030000 eingefügt habe.

EDIT:

setzt man das Flag wie beschrieben, sind die Monitoranzeigen zwar von Anfang an stabil, aber im Hackintool sind unter Patch->Connectors die Connectoren nicht mehr rot unterlegt 🤔
Klingt seltsam, aber ich habe das mehrfach reproduziert.

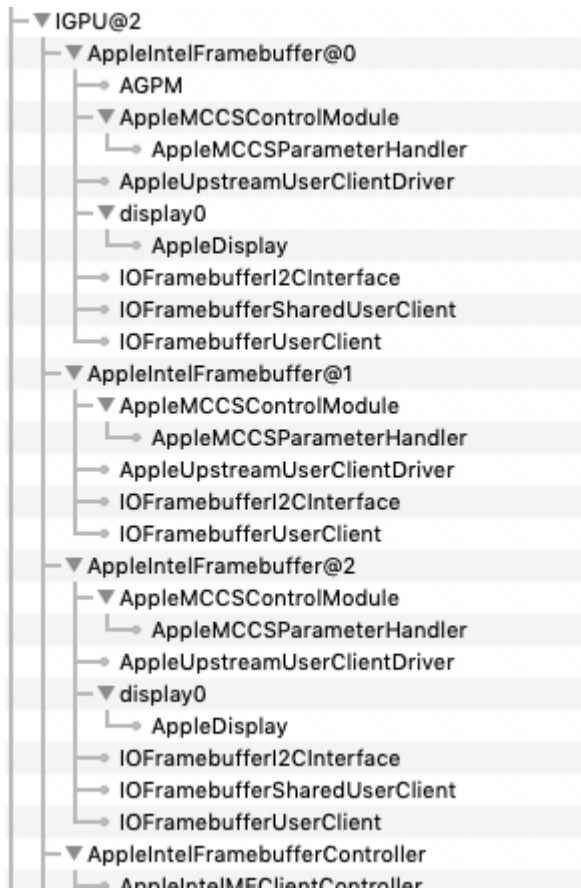
Beitrag von „Keksfamilie“ vom 6. April 2021, 14:47

Hallo zusammen,

habe auch wie meine Vorredner das Problem, dass ich nach dem Start nur einen funktionierenden Monitor habe (und den nicht lange). Habe das mal abgefilmt.

Aus- und Einstecken eines beliebigen Monitors behebt das Problem.

Wie man sehen kann ist das eine Display an Connector 0, das andere an 2:



Und ich habe auch schon probiert mit und ohne CNConnectorAlwaysConnected das Problem zu lösen, ohne Erfolg.

Derzeit sieht das in der OC plist so aus (con 0 ist HDMI, da musste ich pipe und busid ändern, con2 ist DP der direkt funktionierte):

↑ PcRoot(0x0)/Pci(0x2,0x0)	0 Dictionary	16 key/value pairs
AAPL_ig-platform-id	0 Data	<07009B3E>
AAPL_slot-name	0 String	Internal@0,2,0
device-id	0 Data	<9B3E0000>
device_type	0 String	VGA compatible controller
enable-hdmi20	0 Data	<01000000>
framebuffer-con0-enable	0 Data	<01000000>
framebuffer-con0-flags	0 Data	<CF030000>
framebuffer-con0-type	0 Data	<00080000>
framebuffer-con0-busid	0 Data	<01000000>
framebuffer-con0-pipe	0 Data	<08000000>
framebuffer-con2-flags	0 Data	<CF030000>
framebuffer-con2-enable	0 Data	<01000000>
framebuffer-patch-enable	0 Data	<01000000>
framebuffer-unifiedmem	0 Data	<00000080>
hda-gfx	0 String	onboard-1
model	0 String	Intel UHD Graphics 630

Habe auch igfxonln=1 und igfxagdc=0 gesetzt.

Beitrag von „hackmac004“ vom 6. April 2021, 15:39

Laut intel ark Seite ist deine device-id 0x9BC5, die dann so hier C5 9B 00 00 in die config eingetragen wird. Vielleicht hilft's...

Beitrag von „Keksfamilie“ vom 6. April 2021, 16:36

[hackmac004](#) bin mit Mojave unterwegs, da gibts die 0x9BC5 noch nicht.

Beitrag von „hackmac004“ vom 6. April 2021, 18:16

Stimmt bei Comet lake ist Catalina ja, das eigentlich zuerst unterstützte macOS. Hast du die Möglichkeit es mal nebenher zu installieren um zu schauen, ob es dann funktioniert?

Ansonsten les ich raus, das bei dir der DP OOB funktioniert, richtig? Wenn dem so ist, brauchst du dafür eigentlich keine Einträge. Du könntest Con 2 versuchsweise mal rausnehmen.

Eventuell hilft der Eintrag framebuffer-portcount DATA 04000000. Ich weiß nicht genau was der macht, hat aber neulich auch an einer anderen Stelle geholfen.

Beitrag von „Keksfamilie“ vom 6. April 2021, 20:58

Ja genau, DP ging OOTB. Habe da mal die Einträge rausgenommen und danach auch mal den port count patch reingesetzt... Hatte nichts gebracht, verhält sich wie vorher.

Beitrag von „hackmac004“ vom 6. April 2021, 22:22

Ist wirklich schwer zu sagen wie das gelöst werden kann. Die mir bekannten Methoden hast du alle schon verwendet. Ich vermute mal Catalina ist keine Option für dich? Das und evtl. noch verschiedene SMBIOSe ausprobieren fällt mir höchstens noch ein.

Beitrag von „Tom909“ vom 11. Juli 2021, 23:55

Hi, ich versuche mich da was einzulesen und versuche gerade auf einem Laptop den HDMI Port ins Leben zu rufen. Zurzeit nutze ich halt die IG Plattform 04006601 die halt empfohlen wird wenn das Display halt 1600 oder höher als Auflösung hat. Leider wird aber damit quasi nur das interne Display aktiviert und alle andere werden tot geschaltet. Man kann wohl aber selber das zurecht patchen um weitere Ausgänge zu aktivieren. Die Frage ist nur, ich verstehe das was in den tutorials drin steht noch nicht zu 100% wie ich das selber machen soll. Was ich verstanden habe ist, dass man framebuffer portcount die Menge der Ports festlegen kann z.B. und mit patch enable sozusagen die Freiheit bekommt die dann auch zu patchen, aber noch ist mir unklar was die pipe Angabe soll und was ich da am Ende durchgebe damit die 2 anderen Ports anlaufen, mir würde sogar reichen nur den HDMI, denn der andere hier ist nen VGA Anschluss und der muss nicht angesprochen werden. Mit nem anderen framebuffer habe ich probiert aber dann passt die Auflösung nicht und HDMI schaltet den internen ab, bleibt schwarz, also ich wollte nicht mit anderen framebuffers try and error machen. Ich möchte eigentlich nur, dass der HDMI Port einfach dynamisch das Bild erweitert wenn man anschliesst. Wie komme ich da hin 😊 bzw. wie müsste ich das machen. Ist kein Muss, kann auch ohne die externen Ports leben, würde aber gerne das angehen.

Beitrag von „anonymous_writer“ vom 12. Juli 2021, 08:22

Bei Laptops mit zwei Grafikeinheiten ist der HDMI Port meist nicht mit der iGPU verbunden. Die Information um welche Hardware es sich hier handelt fehlt leider.

Beitrag von „Tom909“ vom 12. Juli 2021, 12:45

Es ist die Intel HD4000. Es eine IVY CPU Bridge mit Chipsatz H76 Express drin. Habe dazu ein passendes Topic in Catalina eröffnet, weil ich insgesamt mit fast allen Baustellen durch bin. Was ich jetzt an Feintuning noch machen würde, wäre halt den HDMI Port in Betrieb zu nehmen. Zurzeit ist halt alles Tod bis auf das Display selbst. Ich habe früher schon selbst kexte gepatched um Ports an damaligen Karten freizuschalten oder anzupassen. Sprich grob was da passiert ist mir schon klar, aber die Arbeit mit Whatevergreen in Kombination mit Lilu und den Spielraum der da geschaffen wurde ist für mich noch frisch. Bisher haben die beiden Kexte alles recht dynamisch bei vielen Geräten richtig gemacht, dass ich bisher nicht wirklich hand anlegen musste. Laptops bilden da leider die Ausnahme 😊

Ich würde also gerne den HDMI Port ins Leben rufen. HDMI Audio scheint auch wohl lilu mit applealc hinbekommen zu haben, steht aufjedenfall in der übersicht bei hackintool aufgelistet. Jetzt müsste ich nur noch den HDMI Port halt richtig ansprechen. Dafür müsste ich halt wissen wie ich diesen Port definieren kann per Patches. Ich hab aufjedenfall mal connector auf 3 erweitert und hab auch nun 3 framebuffer in IOEXplore.

Beitrag von „anonymous_writer“ vom 12. Juli 2021, 13:07

Eine HD-4000 hatte ich hier im Einsatz wo auch HDMI funktionierte. Eventuell hilft dir das weiter.

<https://bitbucket.org/anonymou...er/workspace/projects/DLE>

Beitrag von „Tom909“ vom 12. Juli 2021, 13:19

Danke, ich schnuppere mal in deine OC config rein. Ich werde mal den teil mit der Intel HD 4000 mir genauer anschauen mit dem HackBook und abgleichen.

Hab eben deine Config durchgeschaut, du benutzt in deiner Config halt eine andere IG Platform ID. Meine muss ich nehmen weil ich einen Bildschirm betreibe mit 1600*900 intern. Wenn ich die anderen IG IDs nehme habe ich eine verzerrte Auflösung und kann nix machen. Mit anderer ID wird auch der externe Bildschirm angesprochen aber auch verzerrt. Theoretisch müsste

man also wohl per Hand in meinem Fall den HDMI Port selbst zurecht patchen damit der mit meiner Platform ID geht.

Laut dem Guide Zitat"

If you're using 04006601 as your ig-platform-id, you may need to add the following parameters to fix external outputs as otherwise you will only have one output. (Credit to Rehabman)"

Also ich hab das folgendermassen verstanden, erstmal kann ich mit framebuffer-portcount z.B. auf 2 ports oder 3 oder 4 anheben.

Dann kann mit framebuffer-con1-enable 1 den ersten port in dem Fall den 2. selber definieren.

framebuffer-con1-alldata muss ich dann die flags setzen für den port, was sicher einige daten sind, die mir aktuell nicht zu 100% vorliegen und mir aus anderen hdmi ports vorlagen aus anderen framebuffer als info klauen muss, wahrscheinlich über hackintool und diese Daten dann zusammenwerfen und daumen drücken :D

Was ich nur noch nicht ganz verstanden habe ist der Weg zum Patchen weiterer Ports. Ich weiß dass es irgendwie so sein soll:

Type		Explanation
framebuffer-patch-enable	1	enabling the semantic patches in principle (from the WhateverGreen manual)
framebuffer-memorycount	2	03006601 (1 on 04 vs 2 on 03)
framebuffer-pipecount	2	03006601 (3 on 04 vs 2 on 03)
framebuffer-portcount	4	03006601 (1 on 04 vs 4 on 03)
framebuffer-stolenmem	00000004	03006601 Check here

>td >framebuffer-con1-alldata

```
framebuffer-
con1-enable      1 Data 02050000 00040000
                 07040000 03040000
                 00040000 81000000
                 04060000 00040000
                 81000000
```

all data with a connector, either you give all information of that connector (port-based-type-flag) or that port and the ones following it, like in this case. In this case, the ports in 04 are limited to 1:

05030000 02000000 30020000 (which corresponds to port 5, which is LVDS)

However on 03 there are 3 extra ports: 05030000 02000000 30000000 (LVDS, con0, like 04)

02050000 00040000 07040000 (DP, con1)

03040000 00040000 81000000 (DP, con2)


04060000 00040000 81000000 (DP, con3)

Since we changed the number of PortCount to 4 in a platform that has only 1, that means we need to define the 3 others (and we that starting with con1 to the end).

Beitrag von „OSX-Einsteiger“ vom 12. Juli 2021, 15:36

[Zitat von Tom909](#)

.....Meine muss ich nehmen weil ich einen Bildschirm betreibe mit 1600*900 intern.....

Welche **AAPL,ig-platform-id** 6 **device-id** benutzt du denn  dann könnte man mehr sagen 😊 ,

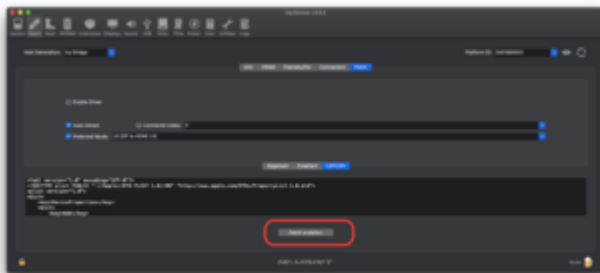
und hast du dich schon mal mit dem Hackintool beschäftigt ?

Ich würde die **Platform ID 01660001** nehmen .

Mit Glück musst du nur die passende **Index** finden und die Bus-ID . Steck denn Monitor Kabel rein wenns Rot leuchtet hast du es



Zu letzt auf **Patch erstellen** Klicken , kann sein das du das mehr mal machen musst bis es greift.



Beitrag von „Tom909“ vom 12. Juli 2021, 15:44

Hi, weiter oben in den Beiträgen habe ich die ID genannt, aber ich fasse mal alles zusammen damit man nicht übersicht verliert:

IG Plattform 04006601 nutze ich zurzeit.

Das wird für notebooks mit ivy und hd4000 empfohlen. Nachteil, alle anderen Ports bis auf der LVDS ist tod, aber man kann wohl die Ports selbst nachpflegen(vorraussetzung, gewusst wie 😊)

Laut dem Guide kann man ja aber selbst anfangen den Patch für mehr funktionalität ja weiter frickeln. Nachteil, man muss wissen wie man den gesamten Port halt ansprechen kann.

Ich kann gerne paar andere IDS probieren und mir dann die Daten rausschreiben lassen wenn der externe Port anläuft. Ich teste mal kurz die anderen Varianten durch und wenn dort HDMI anläuft mir mal versuchen wenn man ein Bild hat mit dem man arbeiten kann, dann auch die Daten rausfischen.

Beitrag von „OSX-Einsteiger“ vom 12. Juli 2021, 15:51

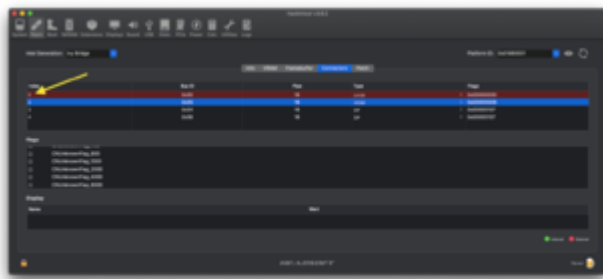
Ich würde die **AAPL,ig-platform-id 01006601** nehmen (ist die Platform-ID 01660001) das ist in meinen Augen die bessere wahl .

Beitrag von „Tom909“ vom 12. Juli 2021, 15:55

Also mit der 01006601 habe ich ein verzerrtes Bild auf dem HDMI ausgang und das interne Display bleibt schwarz 😊

Beitrag von „OSX-Einsteiger“ vom 12. Juli 2021, 16:03

Stell mal Index auf 0 so kenne ich das bei Notebooks , für das Interne Display.



Edit:

Es würde vielleicht helfen wenn du diesen DeviceProperties Posten könntest .

Beitrag von „Tom909“ vom 12. July 2021, 16:13

Hier sind zwei screenshots mit der id 4. Habe die anderen mobile Ids durchprobiert aber kriege damit kein brauchbares bild hin.

So, ich habe die Bildschirmfreigabe konfiguriert, somit kann ich auf dem gerät auch weiterhin Daten aufrufen selbst wenn das Display aktuell nicht richtig angesprochen wird.

Wenn ich den HDMI Stecker stecke, dann ist auch Hackintool rot bei HDMI, sprich die Werte die das stehen könnte ich theoretisch verwenden.

Diese müsste ich dann selber als Patch bei der anderen ID Einpflegen. Port 0x05 wird bei mir

angezeigt. Flags ist 040, aber kann auch 080 ändern, ist aber aber erstmal egal. Interessant wäre nur rauszufinden wie man den HDMI ans laufen bekommt.

Ich glaube ich habe so langsam das fummeln mit den Patches verstanden. Ich habe selber nun weitere 2 Ports angefügt über patch enabler und mal eben ausm netz einen Data string eingefügt und stecke ich nun im betrieb das HDMI Kabel merkt man das intern das display kurz umschaltet von der Funktion aber noch kein bild auf dem HDMI Ausgang. Stecke ich es ab, schaltet auch kurz das display um. Sprich ich habe eine Konnektivität mit ID4 hinbekommen, jetzt heisst es nur den passend so zu Patchen dass ein bild kommt. ansprechen tut wohl der Port jetzt unter Catalina. Ohne diese Patch war keine Reaktion.

So, ich bin immer noch bei dem Problem, dass der halt die Erkennung am Port startet aber kein Bild ausgibt.

Meine eigene Recherche hat folgendes zum Vorschein gebracht:

Der LVDS wird mit

```
05030000 02000000 30020000 INTERNAL PORT
```

korrekt angesprochen

DVI ausgabe bekomme ich mit dem anderen Framebuffer fehlerhaft, bzw der Port erkennt aktuell was mit:

```
02050000 00080000 06000000 DVI AUSGANG BILD
```

das ganze habe ich in meine Config gepackt für die Patches in der Form:

```
<key>AAPL,ig-platform-id</key>
```

```
<data>BABmAQ==</data>
<key>framebuffer-patch-enable</key>
<data>AQAAAA==</data>
<key>framebuffer-fbmem</key>
<data>AACQAA==</data>
<key>framebuffer-stolenmem</key>
<data>AAAwAQ==</data>
<key>framebuffer-unifiedmem</key>
<data>AAAAgA==</data>
<key>framebuffer-portcount</key>
<integer>3</integer>
<key>framebuffer-con2-enable</key>
<data>AQAAAA==</data>
<key>#framebuffer-con2-type</key>
<data>AAgAAA==</data>
<key>framebuffer-con2-alldata</key>
<data>AgUAAAAIAAAGAAAA</data>
```

Jetzt stellt sich mir die Frage was für weitere whatevergreen Parameter noch übergeben muss damit der den Bildschirm auch einbindet, der Port reagiert auf jedenfall nun wenn ich was reinstecke, laptop Bildschirm will umschalten, aber passiert nichts weiter, klemme ich ab, schaltet das display kurz um aber keine ausgabe bisher auf dem externen Display.

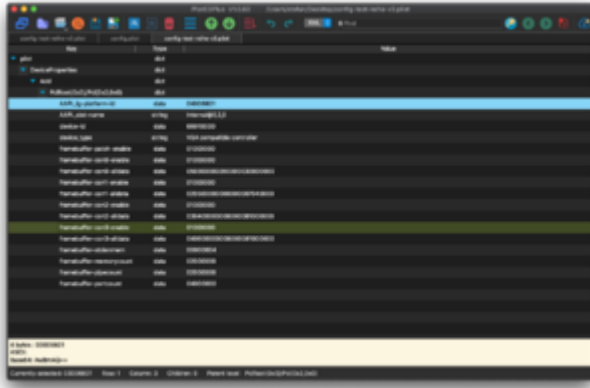
So, habs hinbekommen, bin jetzt den anderen weg gegangen, wenn mein einer framebuffer nicht mag, dann hab ich nun den 01006601 genommen und einfach den internen screen alldata umgeschrieben per Config:

```
<key>AAPL,ig-platform-id</key>
<data>AQBmAQ==</data>
<key>framebuffer-patch-enable</key>
<data>AQAAAA==</data>
<key>framebuffer-fbmem</key>
<data>AACQAA==</data>
<key>framebuffer-stolenmem</key>
<data>AAAwAQ==</data>
<key>framebuffer-unifiedmem</key>
<data>AAAAGAA==</data>
<key>framebuffer-con0-enable</key>
<data>AQAAAA==</data>
<key>framebuffer-con0-alldata</key>
<data>BQMAAAIAAAAwAgAA</data>
```

Damit geht es wie ich es wollte.

Beitrag von „OSX-Einsteiger“ vom 12. Juli 2021, 19:57

Ich habe dir was gebastelt  vielleicht funktioniert das ja 😊



Das ist für dich interessant wie man das entschlüsselt
<https://dortania.github.io/Ope...sid.html#patching-bus-ids>

Beitrag von „Tom909“ vom 12. Juli 2021, 20:09

Wenn Du mich meinst, dann schaue ich mir das gleich genauer an, ich habe das Konzept wie man durch whatever green und übersicht der zu verwendeten Framebuffern spielen kann mit der opencore config datei verstanden und halt erfolgreich zum ende angewendet.

Ich habe auch hdmi audio out getestet am externen bildschirm und das funktioniert auch. Ich habe einfach den con0 auf den framebuffer den ich bisher nutzte einfach gepatched. Damit läuft auch die 1600*600 auflösung und der zweite bildschirm per hdmi wird sauber mit full hd angesprochen und alles hot plug, kannst also im betrieb an und abklemmen. Eigentlich recht simpel wenn man weiß was man patchen muss und welchen framebuffer man kombinieren sollte 😊

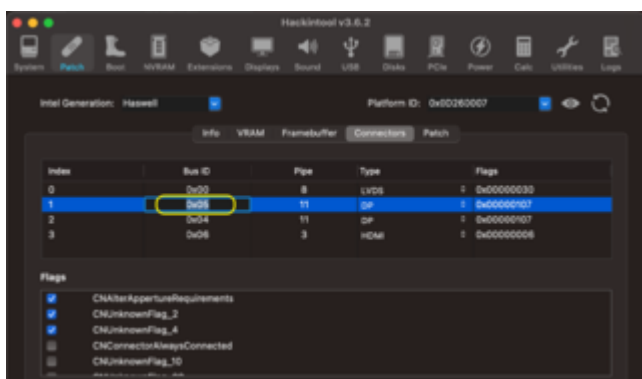
Beitrag von „Max“ vom 12. Juli 2021, 22:01

[hackmac004](#) Was kann ich machen, wenn beim meiner ID z. B: 0700260D sehe und es dort nicht 1,2,3 sondern 0,1,2 gibt, als was soll ich das dann definieren? con-1 oder was?

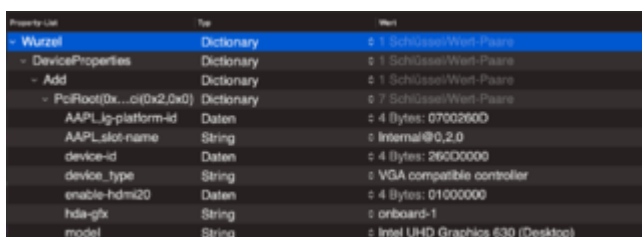
Beitrag von „hackmac004“ vom 12. Juli 2021, 22:39

maxt2007 Kommt darauf an. Was genau willst du erreichen?

Wenn ich die platform id im Hackintool einfach mal auswählen sieht das so aus und wenn ich davon einen patch erstelle, sieht es sehr mager aus.



Du kannst da alles selber, wie in der Markierung zu sehen, verändern.



Da ist noch gar nichts weiter extra definiert. Erst als ich zb. bei Index 1 die Pipe auf 18 änderte sieht der patch besser aus.



Index 0 und busID 0x00 ist für dein internes Display. Index 1-3 kannst du bearbeiten und damit

versuchen deine anderen Outputs flott zu bekommen. Die richtigen Einstellungen kannst du meines Wissens nur durch probieren herausfinden.

Beitrag von „Max“ vom 12. Juli 2021, 22:46

[hackmac004](#) Kannst du mir die config.plist schicken von deinem Bild, was hat der Grafikname von einem Comet Lake gerät bei einer Haswell Properties verloren? Bisher hatte ich mit der 0700260D immer nur ein Blackscreen. Kannst wenn du möchtest denn aktuellen Stand in meinem Thema nachlesen (ThinkPad W540 Opencore)

Beitrag von „hackmac004“ vom 12. Juli 2021, 23:02

Ja klar. Nur zu Info, ich hab die nur beispielhaft erstellt. Wenn du auf exportieren gehst und nichts weiter machts erstellt das Hackintool dir eine config auf dem desktop die nur den patch enthält. Diesen kann man dann auch in seine aktuelle config kopieren.

Das da jetzt Comet Lake steht ist komisch, aber spielt keine große Rolle. Der string Eintrag ist eher kosmetische Natur und kann dabei helfen den Namen der iGPU im Systembericht zu verändern.

Beitrag von „Dec_Bra1n“ vom 13. Juli 2021, 20:44


Is there a Boot Graphics Glitch patch for Big Sur like we have in an older version of macOS?
Code

1. Name: com.apple.iokit.IOGraphicsFamily
2. Find * [HEX]: 01000075 22
3. Replace [HEX]: 010000EB 22
4. Comment: Boot graphics glitch,
5. MatchOS: 10.13.x

Beitrag von „OSX-Einsteiger“ vom 13. Juli 2021, 21:06

[Dec Bra1n](#)

Please be more specific, which hardware is involved and the EFI would also help 😊


From Goolge Translator 

Beitrag von „Dec_Bra1n“ vom 13. Juli 2021, 21:12

I let Asus Tuf Gaming show my signature for full specs. also attach my OC folder

[OC](#)

Beitrag von „OSX-Einsteiger“ vom 13. Juli 2021, 21:14

Der Link geHt nicht 

Beitrag von „Dec_Bra1n“ vom 13. Juli 2021, 21:17

[Zitat von OSX-Einsteiger](#)

The link does not work

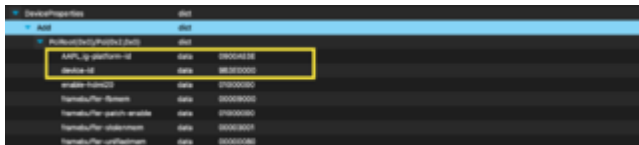


Link aktualisiert

Beitrag von „OSX-Einsteiger“ vom 13. Juli 2021, 21:31

[Dec Bra1n](#)

Das hier sollte die weiter helfen 😊

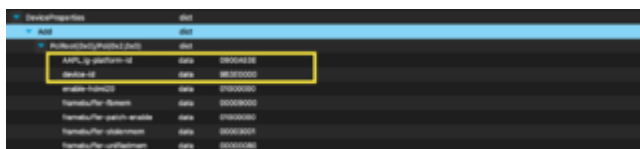


Beitrag von „Dec_Bra1n“ vom 13. Juli 2021, 21:38

[Zitat von OSX-Einsteiger](#)

[Dec Bra1n](#)

Das hier sollte die weiter helfen 😊

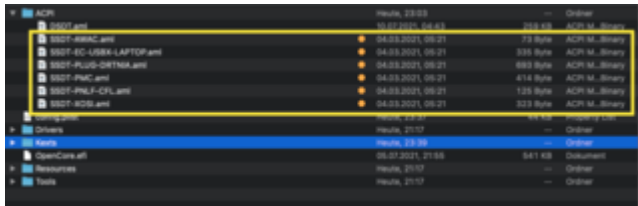


Ich habe das probiert aber es funktioniert nicht

Beitrag von „OSX-Einsteiger“ vom 13. Juli 2021, 23:57

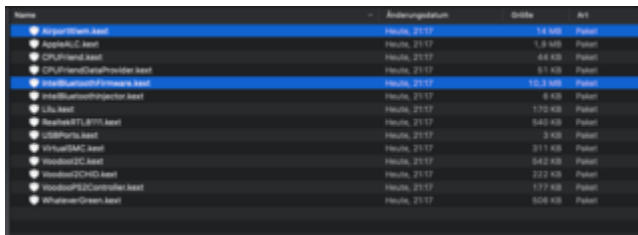
[Dec Bra1n](#)

Das habe Hinzu gefügt so wie es im Guide steht .



<https://dortania.github.io/Openelec-offee-lake-plus.html#acpi>

Aus platz gründen habe ich die Kext gelöscht um denn Ordner hier hochladen zu können 🤪



Beitrag von „Noir0SX“ vom 14. Juli 2021, 00:04

[Zitat von OSX-Einsteiger](#)

Da du die Nightly Texte einsetzt braucht Mann denn boot-args **-lilubetaall**

Gibts schon wieder eine neue macOS Version ?

Beitrag von „OSX-Einsteiger“ vom 14. Juli 2021, 00:14

[NoirOSX](#)

Kommando zurück weil ich die Nightly Kext einsetzte hat natürlich der Kext Updater die nicht

Aktualisiert (weil Aktuell



Beitrag von „Dec_Bra1n“ vom 14. Juli 2021, 00:31

Es ist dasselbe, selbst mit Ihrem OC-Ordner erhalte ich Boot Graphics Glitch. Außerdem verwende ich vollständig gepatchte DSDT, da ich meine GPU nicht über SSDT deaktivieren kann. Ich habe bereits einen Beitrag geschrieben, in dem @Greven mir hilft, die GPU über DSDT zu deaktivieren.

Mit Ihrem OC-Ordner ist meine GPU nur eingeschaltet und Boot Graphics Glitch ist immer noch vorhanden. Aber mit deinem OC-Ordner erhöht sich mein VRAM auf 3GB

Siehe diesen Beitrag

Ein weiteres Problem, das die Intel(R) Iris(TM) Plus Graphics 655 Mine zeigt, ist Comet Lake 630

[dGPU-OFF](#)

Beitrag von „OSX-Einsteiger“ vom 14. Juli 2021, 01:11

Beitrag von „Dec_Bra1n“ vom 15. Juli 2021, 22:06

[Zitat von Tom909](#)

This glitch ist normal, got this on my dell latitude as well. I had issues with this glitch into installer, only thing that fixed it on my computer was setting to efi and csm or legacy op rom support too, but i would not care much about this visuell effect, its not really a problem as a i see, got same effect on my laptop.

Well, I dont have any BIOS setting such as CSM and Legacy i have only 3 major settings such as Fast boot , Secure Boot and VT-D and all are disables but still same Graphics Glitch

Beitrag von „griven“ vom 16. Juli 2021, 00:07

Bei dem glitch den [Dec_Bra1n](#) meint handelt es sich um den weiterverbreiteten verzerrten Apfel in der zweiten Phase des Boots also nachdem der Treiber initialisiert wurde. Dieser Fehler/glitch lässt sich nicht mit einem Framebuffer Patch im Sinne von VRAM, Connectors, PlatformID etc. beheben von daher ist der Weg über die DevProbs der falsche 😊 Was man vielleicht in dem Fall mal testen kann ist das BootArg `gfxrst=1` oder alternativ `gfxrst=4` in Kombination mit `WhateverGreen.kext`.

Beitrag von „Dec_Bra1n“ vom 16. Juli 2021, 00:18

Danke für die Antwort [griven](#)

mit `gfxrst=1` Boot-Argumenten ist das Ergebnis gleich, aber mit `gfxrst=4` Boot-Argument nimmt der Grafikfehler zu

Beitrag von „Tom909“ vom 16. Juli 2021, 01:16

Ansonsten mit -v booten 😄 Ich glaub dann kann das auch weg sein. Eventuell muss man auch nen delay einbauen. Für mich scheint das generell einfach ein delay effekt zu sein beim ansprechen und umschalten auf vesa zu metal. In meinen Augen keine Sache über die man sich grundsätzlich gedanken machen muss. Diesen glitch hatte ich sogar früher mit Nvidia Karten und High Sierra. Wohl ein Fehler der vielleicht mit SMBios und Config eventuell zusammenhängt. Aber eigentlich kein Glitch der einen danach im Betrieb verfolgt.

Beitrag von „Pukka“ vom 12. Januar 2022, 18:41

Hallo Hackintosh Community,

ich bin seit einiger Zeit daran auf meinem SBC MacOS 12.1 zum laufen zu bekommen. Soweit läuft auch alles aber mein zweiter Bildschirm will einfach nicht funktionieren, evtl. kann mir hier jemand weiter helfen.

Mein Aufbau:

Bildschirm 1: direkt per HDMI am Lattepanda Alpha

Bildschirm 2: über ein Typ-C HDMI Adapter am Lattepanda Alpha

Bildschirm 1 funktioniert wunderbar auch wenn er beim Login-Screen kurz aus und wieder an geht, Bildschirm 2 bleibt schwarz. Ich hab mich sowohl auf der OpenCore Seite belesen als auch hier aber alles was ich probiere außer das was ich im Anhang hier mit anhängen funktioniert leider nicht, dann bleibt er beim boot hängen.



(diese Konfiguration wurde bei OpenCore für KabyLake empfohlen)

Index	Bus ID	Pipe	Type	Flags
0	0x00	8	LVDS	0x00000098
1	0x05	9	HDMI	0x000001C7
2	0x04	10	HDMI	0x000001C7
-1	0x00	0	Dummy	0x00000020

Ich wäre über jede Hilfe sehr Dankbar.

OpenCore Version: 0.7.7

WhateverGreen: 1.5.6

Beitrag von „OSX-Einsteiger“ vom 13. Januar 2022, 01:09

[Pukka](#)

Versuch es mal hiermit keine Garantie 😊



Du benötigst für ein Multi Monitor setup beide boot-args **igfxonln=1 igfxagdc=0**

Hier kannst du es nach lesen .

[\[Anleitung\] iGPU - Framebuffer patching & Blackscreen Problem beheben für Clover und OpenCore](#)

Beitrag von „Pukka“ vom 13. Januar 2022, 19:07

[OSX-Einsteiger](#) vielen Dank für deine Hilfe. Ich habs mal in meine config.plist übernommen:

Beitrag von „Pukka“ vom 17. Januar 2022, 13:18

Nach einigem rum probieren habe ich mittlerweile herausgefunden, wenn ich nach dem Login beide Bildschirme ab und wieder anstecke funktionieren beide Monitore wie sie sollen. Hätte evtl. noch jemand eine Idee wie man dieses aus- und wieder anstecken umgehen kann?

Beitrag von „OSX-Einsteiger“ vom 17. Januar 2022, 17:39

[Pukka](#)

Hier hatte ich das geschrieben was dir eigütlich Helfen sollte 😊

[Zitat von OSX-Einsteiger](#)

Du benötigst für ein Multi Monitor setup beide **boot-args igfxonln=1 igfxagdc=0**

Beitrag von „Pukka“ vom 19. Januar 2022, 08:46

[Zitat von Pukka](#)

[OSX-Einsteiger](#) vielen Dank für deine Hilfe. Ich habs mal in meine config.plist übernommen:

