

In Arbeit

Monterey Intel HD 530 DVI

Beitrag von „luca00711“ vom 12. Mai 2022, 12:45

Guten Tag,

ich habe mein neues System aufgesetzt, macOS 12.3.1.

Ich habe hierbei auf die IGPU (Intel HD Graphics 530) von meiner CPU (Intel I5-6600K Skylake) gesetzt, weil ich nur eine Pascal Nvidia Karte besitze.

Es läuft bisher auch alles super. Außer eine Sache, und zwar funktioniert nur mein HDMI Port, ich würde aber gerne meinen zweiten Bildschirm an den DVI Port anschließen.

Ich habe es durch Framebuffer patching versucht (s. Anhang Bilder).

Im Anhang auch noch meine config.plist.

Ich bedanke mich schonmal im Voraus für eure Bemühungen.

Liebe Grüße

Beitrag von „ozw00d“ vom 13. Mai 2022, 14:15

Code

1. <dict>
2. <key>Comment</key>
3. <string>10.11-SKL-1912000-4_displays</string>
4. <key>Find</key>
5. <data>

6. AQMDAw==
7. </data>
8. <key>Name</key>
9. <string>AppleIntelSKLGraphicsFramebuffer</string>
10. <key>Replace</key>
11. <data>
12. AQMEAw==
13. </data>
14. </dict>

Alles anzeigen

so war es mit kext2patch in clover. wie das in oc geht keine Ahnung.

Beitrag von „luca00711“ vom 13. Mai 2022, 16:57

[Zitat von ozw00d](#)

Code

1. <dict>
2. <key>Comment</key>
3. <string>10.11-SKL-1912000-4_displays</string>
4. <key>Find</key>
5. <data>
6. AQMDAw==
7. </data>
8. <key>Name</key>
9. <string>AppleIntelSKLGraphicsFramebuffer</string>
10. <key>Replace</key>
11. <data>
12. AQMEAw==
13. </data>
14. </dict>

Alles anzeigen

so war es mit kext2patch in clover. wie das in oc geht keine Ahnung.

Trotzdem danke 😊

Beitrag von „Hecatomb“ vom 13. Mai 2022, 17:11

Werte übernommen? Geht's nun?

Beitrag von „luca00711“ vom 13. Mai 2022, 17:49

[Zitat von Hecatomb](#)

Werte übernommen? Geht's nun?

Leider nicht.

Beitrag von „Hecatomb“ vom 13. Mai 2022, 18:59

vielleicht bringt dir dieser boot arg ja was.

- `-raddvi` to enable DVI transmitter correction (required for 290X, 370, etc.).

oder schau mal was es sonst so noch gibt :

<https://github.com/acidanthera/WhateverGreen>

ansonsten lese mal dort genauer nach:

<https://dortania.github.io/Ope...l-patching/connector.html>

VERSUCHS ABER MAL MIT DER CONFIG IM ANHANG. benenne sie aber zu config.plis vorher um.

Beitrag von „luca00711“ vom 13. Mai 2022, 19:36

Zitat von Hecatomb

vielleicht bringt dir dieser boot arg ja was.

- `-raddvi` to enable DVI transmitter correction (required for 290X, 370, etc.).

oder schau mal was es sonst so noch gibt :

<https://github.com/acidanthera/WhateverGreen>

ansonsten lese mal dort genauer nach:

<https://dortania.github.io/Ope...l-patching/connector.html>

VERSUCHS ABER MAL MIT DER CONFIG IM ANHANG. benenne sie aber zu config.plist vorher um.

Alles anzeigen

Klappt leider immer noch nicht, noch mit dem boot arg, mit deiner config.plist, noch mit deiner config.plist und dem boot arg.

Darf ich fragen, was du an der config.plist geändert hast?

Beitrag von „Hecatomb“ vom 13. Mai 2022, 21:14

framebuffer-con3-type auf 04000000 gesetzt

Patching Connector Types

- Images and info based off [CorpNewt's Vanilla Guide](#)

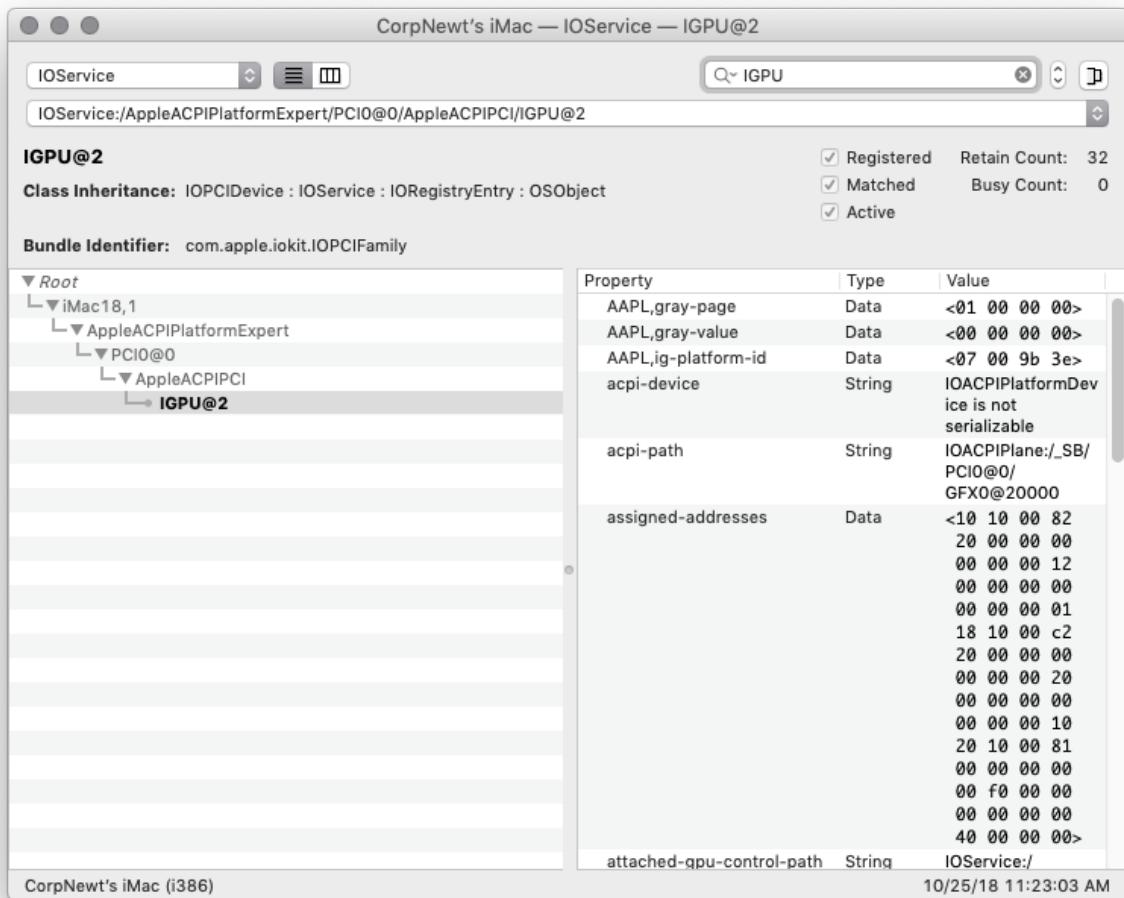
[\(opens new window\)](#)

This section is mainly relevant for users who either get black screen or incorrect color output on their displays(usually HDMI ports). This is due to Apple forcing display types onto your hardware,. To work around it, we'll patch Apple's connector types to properly respect our hardware.

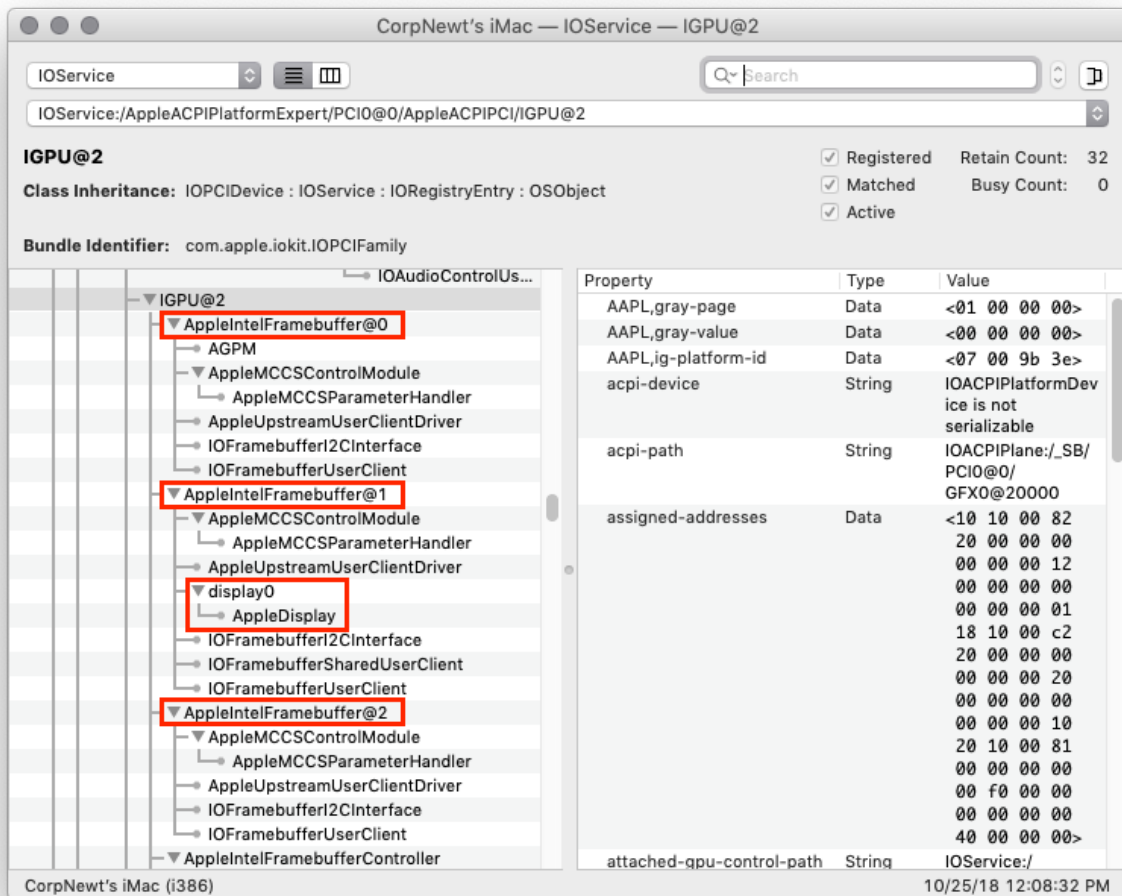
For this example, let's take a UHD 630 system with an HDMI display attached. The machine has already been correctly setup however there's a Pink/Purple tint on the HDMI display.

Grab a copy of [IOReg](#)

[\(opens new window\)](#) and search for the `iGPU` entry:



Next, clear out the entry so we can see the children of the iGPU device:



As we can see in the above screenshot, we have a few framebuffer entries listed. These are all display personalities present in the framebuffer personality, and all have their own settings.

For us, we care about the entries that have a `display0` child, as this is what's driving a physical display. In this example, we can see it's `AppleIntelFramebuffer@1`. When we select it, you'll see in the left pane it has the property `connector-type` with the value `<00 04 00 00>`. And when we look to the below list:

Code

1. `<02 00 00 00>` LVDS and eDP - Laptop displays
2. `<10 00 00 00>` VGA - Unsupported in 10.8 and newer
3. `<00 04 00 00>` DisplayPort - USB-C display-out are DP internally
4. `<01 00 00 00>` DUMMY - Used when there is no physical port
5. `<00 08 00 00>` HDMI

6. <80 00 00 00> S-Video
7. <04 00 00 00> DVI (Dual Link)
8. <00 02 00 00> DVI (Single Link)

- Note: VGA on Skylake and newer are DisplayPorts internally and so are supported by macOS. Please use the DisplayPort connector for these systems.

Looking closer, we see that the HDMI port was actually listed as a DisplayPort. This is where WhateverGreen's patching mechanisms come into play.

Since the incorrect port was located at `AppleIntelFramebuffer@1`, this is port 1. Next we'll to enable WhateverGreen's patching mechanism for `con1`, and then set the connector type to HDMI. To do this, we set the following Properties under `DeviceProperties` -> Add -> `PciRoot(0x0)/Pci(0x2,0x0)`:

- `framebuffer-patch-enable` = 01000000
 - Enables WhateverGreen's patching mechanism
- `framebuffer-conX-enable` = 01000000
 - Enables WhateverGreen's patching on conX
- `framebuffer-conX-type` = 00080000
 - Sets the port to HDMI(<00 08 00 00>)

Note: Remember to replace the `conX` in both patches with `con1` to reflect the port that we want fixed, then set the values as listed above.

Key	Type	Value
Root	Dictionary	8 key/value pairs
ACPI	Dictionary	4 key/value pairs
Booter	Dictionary	2 key/value pairs
DeviceProperties	Dictionary	2 key/value pairs
Add	Dictionary	1 key/value pair
PciRoot(0x0)/Pci(0x2,0x0)	Dictionary	4 key/value pairs
AAPL,ig-platform-id	Data	<07009B3E>
framebuffer-patch-enable	Data	<01000000>
framebuffer-con1-enable	Data	<01000000>
framebuffer-con1-type	Data	<00080000>
Delete	Dictionary	0 key/value pairs
Kernel	Dictionary	7 key/value pairs
Misc	Dictionary	6 key/value pairs
NVRAM	Dictionary	6 key/value pairs
PlatformInfo	Dictionary	6 key/value pairs
UEFI	Dictionary	9 key/value pairs

PS: Musst mal bisschen rum probieren.eventuell ist der con3 bei dir ja tot. anderes setz mal nur die Werte wie im guide und schau mal ob dann ein signal auf dvi kommt.

Kannst ja mal diesen wert anstatt den anderen setzen 00 02 00 00

Beitrag von „luca00711“ vom 13. Mai 2022, 23:24

Habe gerade mal im IORegistryExplorer nachgeschaut und mir ist aufgefallen, dass es bei mir gar kein AppleIntelFramebuffer@3 gibt.

Es gibt nur:

AppleIntelFramebuffer@0

AppleIntelFramebuffer@1

AppleIntelFramebuffer@2

Und beim AppleIntelFramebuffer@2 ist das Child Objekt:

AppleDisplay, also mein momentaner HDMI angeschlossener richtig?

Bedeutet der HDMI Port ist con2?

Und heißt das der DVI Port ist gar nicht im System? Bedeutet der wird gar nicht angesteuert?

Zur Aufklärung mein Mainboard hat die Ports:

1x HDMI wo mein Main-Monitor angeschlossen ist

1x VGA

1x DVI

Im Anhang nochmal ein Screenshot aus dem IORegistryExplorer und meine Aktuelle config.plist.

Vielen Dank nochmal für die Hilfsbereitschaft.

Beitrag von „grt“ vom 14. Mai 2022, 08:38

du musst in den deviceproperties den portcount auf 4 setzen, dann taucht der framebuffer @3 auch auf.

ohne gewähr, da ich am handy bin:

```
framebuffer-patch-enable 01000000
```

```
framebuffer-portcount 04000000
```

evtl. noch

```
framebuffer-pipecount 03000000
```

wenn 3 displays gleichzeitig angesprochen werden sollen.

datentyp jeweils data. und ob die bindestriche jeweils richtig sind, bin ich nicht sicher.

Beitrag von „luca00711“ vom 14. Mai 2022, 11:06

Zitat von grt

du musst in den deviceproperties den portcount auf 4 setzen, dann taucht der framebuffer @3 auch auf.

ohne gewähr, da ich am handy bin:

```
framebuffer-patch-enable 01000000
```

```
framebuffer-portcount 04000000
```

evtl. noch

```
framebuffer-pipecount 03000000
```

wenn 3 displays gleichzeitig angesprochen werden sollen.

datentyp jeweils data. und ob die bindestriche jeweils richtig sind, bin ich nicht sicher.

Alles anzeigen

VIELEN VIELEN DANK.

Mit dem framebuffer-portcount 04000000 erschien framebuffer@3, hab dann mal den boot arg agdpmod=pikera hinzugefügt, weil ich gehnt habe es liegt an AGDP.

Habe dann nochmal den con3-type = auf 04000000 (DVI-D) gestellt. Und jetzt läuft der zweite Monitor auf DVI.

Ich bin super glücklich, muss jetzt nochmal bisschen rum probieren, weil ich mit dem boot arg agdpmo=pikera no signal black screen bekomme, sobald der login screen da ist. Muss immer den DVI dann anschließen und dann das HDMI ab und wieder anschließen für Bild.

Vielleicht hat da auch ja jemand noch einen Tipp.

Ich bedanke mich nochmal bei jedem der hier seine Hilfe angeboten hat.

Beitrag von „cobanramo“ vom 14. Mai 2022, 14:30

Du must den pikera bootarg wegnehmen,

Den blackscreen nach der anmeldung bekommst hiermit weg.

- `igfxagdc=0` boot argument (`disable-agdc` device property) to disable AGDC.

Gruss Coban

Beitrag von „luca00711“ vom 16. Mai 2022, 18:22

[Zitat von cobanramo](#)

Du must den pikera bootarg wegnehmen,

Den blackscreen nach der anmeldung bekommst hiermit weg.

- `igfxagdc=0` boot argument (`disable-agdc` device property) to disable AGDC.

Gruss Coban

Wenn ich den boot arg pikera wegnehme und stattdessen `igfxagdc=0` mache, geht der DVI

Port nicht mehr.

Sobald ich den Monitor an den DVI Port stecke wird kurz der Bildschirm schwarz als würde er erkannt werden, aber es kommt dann kein Bild und wird auch nicht in den Einstellungen angezeigt, dass ein anderer Monitor angeschlossen ist.

Hab auch schon: `agdpmod=ignore` und `agdpmod=vit9696` ausprobiert, beides ohne erfolg. Es funktioniert bis jetzt nur bei pikera.

Jemand noch ein Tipp?

Beitrag von „cobanramo“ vom 17. Mai 2022, 01:01

[Zitat von luca00711](#)

`igfxagdc=0` mache, geht der DVI Port nicht mehr.

Dann ist dein FrameBuffer noch nicht sauber am laufen, könntest so machen;

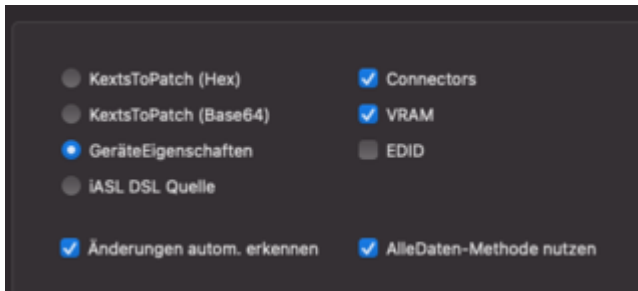
Wirf den Pikera rein, guck dir die FrameBuffers im loreg und trage sie auch korrekt im Config ein.

Danach pikera raus, das mit `agdp bootarg` ist normal eh für den "external GPU" würd ich meinen.

- `agdpmod=vit9696` disables check for `board-id` (or add `agdpmod` property to external GPU).
- `agdpmod=pikera` replaces `board-id` with `board-ix`
- `agdpmod=ignore` disables AGDP patches (`vit9696,pikera` value is implicit default for external GPUs)

Ich hatte damals an meinem HD530 keinen 2. Monitor dran daher kann ich es nicht 100% sagen aber du kannst auch mal den "Alle Daten Methode" austesten.

Damit lief das auch gut.



Das mit Pikera & paar Sec. Black Screen bei der Anmeldung kam irgendwann mit Whatevergreen ca. Version 1.3.8 oder so.

Davor ging das auch ohne probleme, alternativ kannst du auch mal die Versionen davor auch testen.

Gruss Coban

Beitrag von „luca00711“ vom 17. Mai 2022, 21:16

[Zitat von cobanramo](#)

Dann ist dein FrameBuffer noch nicht sauber am laufen, könntest so machen;

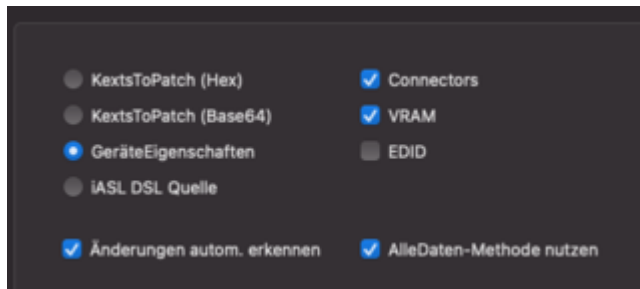
Wirf den Pikera rein, guck dir die FrameBuffers im loreg und trage sie auch korrekt im Config ein.

Danach pikera raus, das mit agdp bootarg ist normal eh für den "external GPU" würd ich meinen.

- `agdped+v11986` disables check for `board-id` (or add `agdped` property to external GPU).
- `agdped+pikera` replaces `board-id` with `board-ix`
- `agdped+ignore` disables AGDP patches (`v11986,pikera` value is implicit default for external GPUs)

Ich hatte damals an meinem HD530 keinen 2. Monitor dran daher kann ich es nicht 100% sagen aber du kannst auch mal den "Alle Daten Methode" austesten.

Damit lief das auch gut.



Das mit Pikera & paar Sec. Black Screen bei der Anmeldung kam irgendwann mit Whatevergreen ca. Version 1.3.8 oder so.

Davor ging das auch ohne probleme, alternativ kannst du auch mal die Versionen davor auch testen.

Gruss Coban

Alles anzeigen

Erstmal vielen Dank für deine Erklärung.

Könntest du mir etwas helfen?

Ich habe folgendes versucht

```
framebuffer-con2-enable 01000000
```

```
framebuffer-con2-type 00080000
```

```
framebuffer-con2-busid 06000000
```

```
framebuffer-con3-enable 01000000
```

```
framebuffer-con3-type 04000000
```

```
framebuffer-con3-busid 07000000
```

Danach hatte ich nur blackscreen auf HDMI.

Und kann es sein, dass framebuffer-portcount auf 03000000 muss und nicht 04000000?

Wenn ich das richtig verstehe hab ich ja nur 3

HDMI, DVI und VGA

In den Anhängen Bilder von den einzelnen Framebuffers

Oder was genau meinst du mit

[Zitat von cobanramo](#)

trage sie auch korrekt im Config ein.

Habe ich das richtig gemacht?

Entschuldigung, wenn ich damit störe.

Beitrag von „cobanramo“ vom 18. Mai 2022, 01:10

con0, con1, con2, con3 = **4 Ports**, framebuffer-portcount 04000000

con0, con1, con2 = **3 Ports**, framebuffer-portcount 03000000

usw.

Ich sehe nicht was du angeschlossen hast was was zeigt oder nicht zeigt, daher ist das für mich rätselraten.

Ich hab jetzt einfach mal die Connectoren von dir Dupliziert, So muss das aussehen;



Die Daten können von con0 & con1 natürlich jetzt falsch sein, das gilt es herauszufinden.

Guck machst dir einfach, lass das jetzt mit Dualmonitor weg, zuerst das mal einzeln lösen; und ignoriere mal den VGA Port.

Der soll ja anscheinend als DisplayPort definiert sein laut dieser Info oder?

- *Note: VGA on Skylake and newer are DisplayPorts internally and so are supported by macOS. Please use the DisplayPort connector for these systems.*

Fange erstmal mit framebuffer-portcount 04000000 an.

Häng einfach an den HDMI Port ein Monitor an, guck dir die Daten im loreg und Hackintool an; und passe im Config.plist den richtigen "conX" mit den richtigen Daten.

Danach hängst du den Monitor ab und schliesst es an den DVI Port an.

Guckst dir die Daten nochmal an und korrigierst dessen "conX" auch im Config.

Richtige Daten & Werte entnimmst du von hier;

Code

1. <02 00 00 00> LVDS and eDP - Laptop displays
2. <10 00 00 00> VGA - Unsupported in 10.8 and newer
3. <00 04 00 00> DisplayPort - USB-C display-out are DP internally
4. <01 00 00 00> DUMMY - Used when there is no physical port
5. <00 08 00 00> HDMI

6. <80 00 00 00> S-Video
7. <04 00 00 00> DVI (Dual Link)
8. <00 02 00 00> DVI (Single Link)

Wenn du deine Ports sauber definiert hast machst du mal neustart und hängst beide Monitor an und guckst was es im Systemeinstellung zeigt.

Wenn das funktioniert musst du einfach mal austesten ob du das auch mit 3 Ports passt;

con0, con1, con2 = **3 Ports**, framebuffer-portcount 03000000

Intel HD Graphics 510-580 (Skylake)

Supported since OS X 10.11.4

SKL framebuffer list:

Framebuffer	Type	Connectors	TOTAL STOLEN Memory
0x19180000	mobile	3	56 MB
0x19180000	mobile	3	56 MB
0x19260000	mobile	3	56 MB
0x19270000	mobile	3	56 MB
0x19180000	mobile	3	56 MB
0x19380000	mobile	3	56 MB
0x19120000	mobile	3	56 MB
0x19220001	desktop	0	1 MB
0x19170001	desktop	0	1 MB
0x19120001	desktop	0	1 MB
0x19220001	desktop	0	1 MB
0x19180002	mobile	3	56 MB
0x19260002	mobile	3	56 MB
0x191E0003	mobile	3	41 MB
0x19260004	mobile	3	35 MB
0x19270004	mobile	3	56 MB
0x19380006	mobile	4	35 MB
0x19180006	mobile	1	35 MB
0x19260007	mobile	3	35 MB

hat nämlich nur 3 Connectoren.

```

ID: 19120000, STOLEN: 34 MB, FBMEM: 31 MB, VRAM: 1638 MB, Flags: 0x0000110F
TOTAL: STOLEN: 56 MB, TOTAL: CLERK: 1 MB (1672884 bytes), MAX STOLEN: 124 MB, MAX OVERALL: 125 MB (129188876 bytes)
Model name: Intel HD Graphics SKL_CNB
Camera: CameraDisabled (0), Res: 1280 Hz, FreqMax: 1280 Hz
Mobile: 1, PipeCount: 3, PortCount: 3, FBMemoryCount: 3
[75] busid: 0x00, pipe: 0, type: 0x00000000, flags: 0x00000000 - ConnectorDummy
[7] busid: 0x05, pipe: 0, type: 0x00000400, flags: 0x0000018F - ConnectorDP
[2] busid: 0x0A, pipe: 10, type: 0x00000400, flags: 0x0000018F - ConnectorDP
19180000 01900000 30000000
01900000 00040000 81910000
02040000 00040000 81910000

```

Das ganze sollte am Schluss ohne den Pikera bootarg laufen, ansonsten gibt es eben diesen Black Screen beim anmelden.

Oder du kannst es dir einfacher machen und zu einem uralten Whatevergreen zurückgehen wo es noch ohne diesen BlackScreen startete.

Versuch es mal ob du mit den Connectoren auf die reihe kriegst.

Gruss Coban

Beitrag von „luca00711“ vom 26. Mai 2022, 12:34

[Zitat von cobanramo](#)

con0, con1, con2, con3 = **4 Ports**, framebuffer-portcount 04000000

con0, con1, con2 = **3 Ports**, framebuffer-portcount 03000000

usw.

Ich sehe nicht was du angeschlossen hast was was zeigt oder nicht zeigt, daher ist das für mich rätselraten.

Ich hab jetzt einfach mal die Connectoren von dir Dupliziert, So muss das aussehen;



Die Daten können von con0 & con1 natürlich jetzt falsch sein, das gilt es herauszufinden.

Guck machst dir einfach, lass das jetzt mit Dualmonitor weg, zuerst das mal einzeln lösen; und ignoriere mal den VGA Port.

Der soll ja anscheinend als DisplayPort definiert sein laut dieser Info oder?

- *Note: VGA on Skylake and newer are DisplayPorts internally and so are supported by macOS. Please use the DisplayPort connector for these systems.*

Fange erstmal mit framebuffer-portcount 04000000 an.

Häng einfach an den HDMI Port ein Monitor an, guck dir die Daten im loreg und Hackintool an;

und passe im Config.plist den richtigen "conX" mit den richtigen Daten.

Danach hängst du den Monitor ab und schliesst es an den DVI Port an.

Guckst dir die Daten nochmal an und korrigierst dessen "conX" auch im Config.

Richtige Daten & Werte entnimmst du von hier;

Code

1. <02 00 00 00> LVDS and eDP - Laptop displays
2. <10 00 00 00> VGA - Unsupported in 10.8 and newer
3. <00 04 00 00> DisplayPort - USB-C display-out are DP internally
4. <01 00 00 00> DUMMY - Used when there is no physical port
5. <00 08 00 00> HDMI
6. <80 00 00 00> S-Video
7. <04 00 00 00> DVI (Dual Link)
8. <00 02 00 00> DVI (Single Link)

Wenn du deine Ports sauber definiert hast machst du mal neustart und hängst beide Monitor an und guckst was es im Systemeinstellung zeigt.

Wenn das funktioniert musst du einfach mal austesten ob du das auch mit 3 Ports passt;

con0, con1, con2 = **3 Ports**, framebuffer-portcount 03000000

HD530 mit ID 0x19120000 hat nämlich nur 3 Connectoren.

Intel HD Graphics 510-580 (Skylake)

Supported since OS X 10.15.4

DKG Framebuffer list:

Framebuffer	Type	Connectors	TOTAL STOLEN Memory
0x1910000	mobile	3	56 MB
0x1910000	mobile	3	56 MB
0x1920000	mobile	3	56 MB
0x1927000	mobile	3	56 MB
0x1930000	mobile	3	56 MB
0x1930000	mobile	3	56 MB
0x1930000	mobile	3	56 MB
0x1930000	mobile	3	56 MB
0x1930000	mobile	3	56 MB
0x1932000	desktop	0	1 MB
0x1937000	desktop	0	1 MB
0x1932000	desktop	0	1 MB
0x1932000	desktop	0	1 MB
0x1940000	mobile	3	56 MB
0x1940000	mobile	3	56 MB
0x1950000	mobile	3	41 MB
0x1920004	mobile	3	35 MB
0x1927004	mobile	3	56 MB
0x1930006	mobile	4	35 MB
0x1930006	mobile	1	39 MB
0x1920007	mobile	3	35 MB

```
ID: 1912000, STOLEN: 34 MB, FBMEM: 21 MB, VMEM: 1636 MB, Flags: 0x000010F
TOTAL STOLEN: 56 MB, TOTAL CLIPCH: 1 MB (7672884 bytes), MAX STOLEN: 124 MB, MAX OVERALL: 125 MB (131628576 bytes)
Model name: Intel HD Graphics SKL_CNB
Camera: CameraDisabled (0), Freq: 1388 Hz, FreqMax: 1388 Hz
Module: 1, PipeCount: 3, PortCount: 3, FBMemoryCount: 3
[2] busID: 0x03, pipe: 0, type: 0x00000000, flags: 0x00000000 - ConnectorDummy
[7] busID: 0x05, pipe: 0, type: 0x00004000, flags: 0x00000187 - ConnectorDP
[3] busID: 0x04, pipe: 10, type: 0x00000400, flags: 0x00000187 - ConnectorDP
FH000000 01000000 20000000
01050000 00040000 01010000
02060000 00040000 01010000
```

Das ganze sollte am Schluss ohne den Pikera bootarg laufen, ansonsten gibt es eben diesen Black Screen beim anmelden.

Oder du kannst es dir einfacher machen und zu einem uralten Whatevergreen zurückgehen wo es noch ohne diesen BlackScreen startete.

Versuch es mal ob du mit den Connectoren auf die reihe kriegst.

Gruss Coban

Alles anzeigen

Es läuft nun ohne Pikera irgendwie ist das alles merkwürdig aber ich habe es geschafft.

busid 0x4 gehörte zu con2 (HDMI)

busid 0x6 gehörte zu con3 (DVI)

Was noch merkwürdiger ist, wenn bei con3 der connector-type auf DVI (Dual Link) ist bekomme ich kein Signal, aber auf connector-type HDMI funktioniert er wunderbar.

Ich frage mich aber wieso das so ist, laut dem registry explorer ist framebuffer@2 = port number 0x6 und framebuffer@3 = port number 0x7.

Oder ist port number im Registry Explorer was anderes als die bus id?

Vielleicht weiß ja jemand wieso das alles so verdreht sein kann.

Versuche sogar jetzt noch irgendwie meinen VGA Port zum laufen zu bekommen, bin mal gespannt, ob das funktioniert.

Das Problem ist jetzt nur, wenn ich will das mein DVI Port funktioniert muss ich vorher den Rechner mit angeschlossenem zweiten Monitor hochfahren, auch wenn er so kein Bild hat. Ich starte ihn dann neu mit nicht angeschlossenem zweiten Monitor, und stecke ihn dann nachdem hochfahren an. So funktioniert er.

Ich bräuchte noch einmal Hilfe dabei, dass ich einfach ganz normal den Rechner hochfahren kann und den zweiten Monitor anschließen kann.

Vielen Dank

Liebe Grüße

Luca

Beitrag von „ozw00d“ vom 27. Mai 2022, 13:29

An nem Mac gibts nur kein DVI soweit ich weiss.
Ein paar Unterschiede dazu sind:

DVI

- Adapters can convert to HDMI.

- Only transmits video.
- Max data rate of 9.9 Gbit/sec.
- Capable of up to 3840x2400 at 30 Hz

HDMI

- Supported by more devices.
- Transmits video and audio.
- Max data rate of 42.6 Gbit/sec.
- Capable of up to 8k at 120 Hz.

Habe noch keinen Mac neuerer Generation mit DVI gesehen.

Bei VGA wirst du scheitern da VGA nicht unterstützt wird.

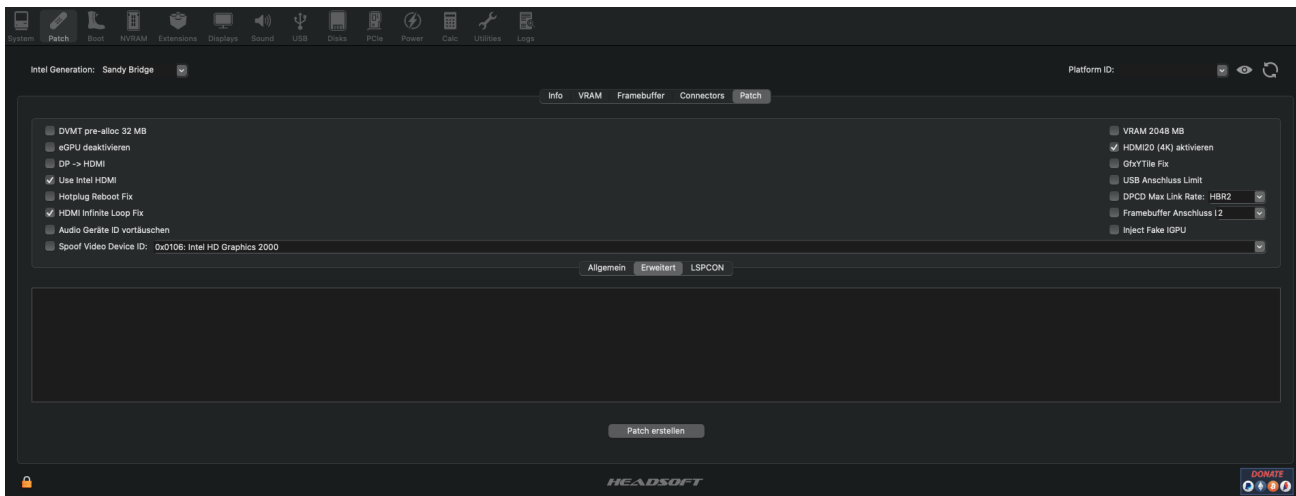
Beitrag von „hackmac004“ vom 27. Mai 2022, 14:50

Du kannst es mal mit dem bootarg igfxonln=1 versuchen, evtl. brauchst du den DVI dann nicht an und abstecken.

Beitrag von „Hecatomb“ vom 27. Mai 2022, 16:20

Hast du mal im Hackintool die Infos angeschaut? Eventuell steht da ja was nützliches dabei. Bekommst du angezeigt wenn du mit etwas Geduld über manche Sachen gehst.

Kannst ja auch mal die die EDID vom Monitor mit in den patch packen...



se Intel HDMI

If TRUE, hda-gfx=onboard-1 will be injected into the GFX0 and HDEF devices. Also, if an ATI or Nvidia HDMI device is present, they'll be assigned to onboard-2. If FALSE, then ATI or Nvidia devices will get onboard-1 as well as the HDAU device if present.

s 2000

EDID

Watch out, this is really messy (see `AppleIntelFramebufferController::MapFBToPort`).

I am not fully sure why this exists, and recommend setting index to array index (i.e. the sequential number from 0).

The only accepted values are 0, 1, 2, 3, and -1 (0xFF). When index is equal to array index the logic is simple:

Port with index 0 is always considered built-in (of LVDS type) regardless of any other values.

Ports with indexes 1~3 are checked against type, HDMI will allow the use of digital audio, otherwise DP is assumed.

Port with index 0xFF is ignored and skipped.

When index != array index port type will be read from `connector[index].type`.

Say, we have 2 active ports:

0 - [1] busID 4 type LVDS

1 - [2] busID 5 type DP

2 - [3] busID 6 type HDMI

3 - [-1] busID 0 type Dummy

This will result in 2 framebuffer which types will be shifted:

0 - busID 4 type DP

1 - busID 5 type HDMI

In fact busID values are also read as `connector[index].busID`, but are later mapped back via

`AppleIntelFramebufferController::getGMBusIDfromPort` by looking up a connector with the specified index.

The lookup will stop as soon as a special marker connector (-1) is found. To illustrate, if we have 2 active ports:

0 - [1] busID 4 type LVDS

1 - [2] busID 5 type DP

2 - [-1] busID 6 type HDMI

3 - [-1] busID 0 type Dummy

The result will be 2 framebuffer which types and the second busID will be shifted:

0 - busID 4 type DP

1 - busID 6 type HDMI

It is also used for port-number calculation.

- LVDS displays (more precisely, displays with `CNConnectorAlwaysConnected` flag set) get port-number 0.

- Other displays go through index - port-number mapping: 1 - 5, 2 - 6, 3 - 7, or fallback to 0.

HDMI Infinite Loop Fix



Aut



Sp

Fix the infinite loop when the graphics driver tries to establish a HDMI connection with a higher pixel clock rate, for example connecting to a 2K/4K display with HDMI 1.4, otherwise the system just hangs (and your builtin laptop display remains black) when you plug in the HDMI cable.

- For those who want to have "limited" 2K/4K experience (i.e. 2K@59Hz or 4K@30Hz) with their HDMI 1.4 port, you might find this fix helpful.

- For those who have a laptop or PC with HDMI 2.0 routed to IGPU and have HDMI output issues, please note that this fix is now succeeded by the LSPCON driver solution, and it is still recommended to enable the LSPCON driver support to have full HDMI 2.0 experience.

(You might still need this fix temporarily to figure out the connector index of your HDMI port.)

Convert Display Port connector types to HDMI

otplug Reboot Fix

Proven by AppleIntelFramebufferController::MapFBToPort, by a call to AppleIntelFramebufferController::getGMBusIDfromPort.

This is GMBUS (Graphic Management Bus) ID described in https://01.org/sites/default/files/documentation/intel-gfx-prm-osrc-hsw-display_0.pdf.

The use could be found in Intel Linux Graphics Driver source code:
https://github.com/torvalds/linux/blob/6481d5ed076e69db83ca75e751ad492a6fb669a7/drivers/gpu/drm/i915/intel_i2c.c#L43
https://github.com/torvalds/linux/blob/605dc7761d2701f73c17183649de0e3044609817/drivers/gpu/drm/i915/i915_reg.h#L3053

However, it should be noted that Apple identifiers are slightly different from Linux driver. In Linux 0 means disabled, however, for Apple it has some special meaning and is used for internal display.

Other than that the values are the same:

- GMBUS_PIN_DPC (4) HDMIC
- GMBUS_PIN_DPB (5) SDVO, HDMIB
- GMBUS_PIN_DPD (6) HDMID
- GMBUS_PIN_VGADDC (2) VGA until Broadwell inclusive.

So basically you could use 4, 5, 6 for arbitrary HDMI or DisplayPort displays.

Since 5 supports SDVO (https://en.wikipedia.org/wiki/Serial_Digital_Video_Out), it may also be used to support DVI displays.

Starting with Skylake VGA works via SDVO too (instead of a dedicated GMBUS_PIN_VGADDC id).