

skurriles benehmen vom dell E5470 mit externem hdmi-monitor

Beitrag von „grt“ vom 15. Februar 2021, 14:29

moin allerseits..

es geht diesmal um den dell E5470 in meiner signatur.

kurz die eckdaten:

i3 6100 mit HD520

opencore 0.65 / 10.14.6

funktioniert prächtig bis auf den externen monitor am hdmi-anschluss.

wenn ich dem frisch hochgefahrenen läptopp einen monitor per hdmi anstöple, wird dieser ordnungsgemäss erkannt, alles gut bis hierher. was mir als einziges aufgefallen ist, ist ein leichtes flimmern des internen monitors, sobald der externe dazukommt.

ziehe ich den monitor aber wieder ab, dann tritt der rechner (bzw. die grafik) in generalstreik, der interne monitor wird schwarz, allerdings reagiert der klapptopf noch auf tastatureingaben (warntöne), ist im netzwerk erreichbar (ping) und auch die dateifreigabe funktioniert noch. bildschirmfreigabe friert aber ein.

jetzt kommts aber....

wenn ich nämlich den klapptopf mit angeschlossenem externen monitor zuklappe - (manchmal springt dann das bild auf den externen monitor rüber, manschmal schläft der kollege einfach ein..) - und wieder aufklappe, ist der spuk vorbei. ich kann den externen ab sofort abziehen und wieder anstecken sooft ich dazu lust hab, der läppi kriegts mit, und reagiert entsprechend: kurzes schwarzbild, dann macht er je nach dem mit einem oder 2 monitoren ganz brav weiter. auch weitere sleep/wake-zyklen ändern an dem verhalten nichts mehr. das leichte flimmern ist ebenfalls weg.

erst, wenn der läptopp neugestartet oder runtergefahren und wiedergestartet wird, geht das spielchen von vorne los. der externe monitor darf erst dann ungestraft abgezogen werden, wenn der läptop einmal mit angestecktem monitor zu- und aufgeklappt wurde.

getestet hab ich das mit mittlerweile 3 unterschiedlichen monitoren und kabeln. immer dasselbe verhalten....

kexte sind weitgehend aktuell, die config ist relativ sparsam.

mich erinnert das ganze ein wenig an das verhalten des T440s mit der HD4400 - der präsentierte sich mit kreativem pixelsalat, bis man ihn kurz schlafenlegte und wieder aufweckte. die lösung bei dem war der csm-support, der enabled sein musste im bios. hätte ich beim dell längst mal getestet ob das bei dem auch helfen könnte, sowas gibts aber in dessen bios nicht.

was könnte das noch sein?

Beitrag von „Bob_1“ vom 15. Februar 2021, 18:15

wo ich dell hatte, hat dieses kext im Anhang geholfen und habe es aus den Forum osxlatitude gezogen, du kannst es bei dir testen !

[grt](#)

Beitrag von „grt“ vom 15. Februar 2021, 19:17

hattest du ein vergleichbares problem?

Beitrag von „Bob_1“ vom 16. Februar 2021, 08:09

ja und habe viele darüber gelesen und Framebuffer Patches versuchst ohne erfolg

Vielleicht bekommst du es hin ohne kext 😊

Beitrag von „grt“ vom 16. Februar 2021, 10:17

mich würde ja auch die ursache des spielchens interessieren. prinzipiell kann der läptopp ja mit dem hdmi anschluss umgehen..

irgendwas ändert sich nach dem sleep/zugeklappt werden an der config der grafik, was den normalen betrieb mit dualmonitor dann möglich macht.

vga funktioniert nebenbei bemerkt einwandfrei, an- abstecken ohne probleme.

was mir gerade beim schreiben einfällt... bisher bin ich von sleep (ruhezustand) als ursache für das sich einkriegen ausgegangen. es könnte aber auch das zuklappen an sich sein. und das hängt am "pnlf-device", was per ssdt implementiert wird.

kopfkraz muss ich mir mal genauer angucken..

Beitrag von „grt“ vom 18. Februar 2021, 11:35

Bob_1 ich hab mir mal deine kext angeguckt. damit wird die edid und mögliche auflösungen eines displays injected - das sollten dann doch die daten des verbauten displays sein, oder?

inzwischen hab ich das verhalten des displays noch ein wenig mehr eingrenzen können: wenn ich den deckel nach einem start/neustart kurz zu- und wieder aufklappe - es reicht wirklich kurz, hauptsache, der monitor geht kurz mal aus, einschlafen muss der klapptopf nicht - gibt es keine probleme mehr mit anschliessen und abziehen eines hdmi-monitors.

das sollte man doch eigentlich auch per script und startobjekt fürs erste fixen können, oder?

gibts da schon was, oder kennt jemand den "zu- und aufklapp-befehl" für ein script?

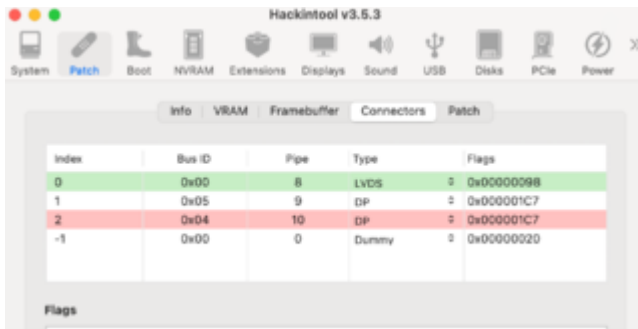
Beitrag von „Bob_1“ vom 18. Februar 2021, 12:08

Kext ist nicht mein Werk 😊, daher ist sie nicht meine kext.

Wenn das dir hilft, kann ich dir mit Hackintool auf mein echten MacBook Pro 2016 anschauen, was wird unter Connectors gezeigt, naja der Prozessor unterstützt nur 3 Ports genauso wie bei MacBook dann kannst du diese Patches auf dein Hackbook anwenden und testen

[grt](#)

Update



IOReg zeigt auch

Framebuffer@0

connector-type |Data|02000000

port-number |Number|0x0

Framebuffer@1

connector-type |Data|00040000

port-number |Number|0x5

Framebuffer@2

connector-type |Data|00040000

port-number |Number|0x6

Beitrag von „grt“ vom 18. Februar 2021, 15:04

die patches hatte ich schon...

anschlüsse anpassen hat nicht geholfen. im moment teste ich das zu&aufklappen per script...
zuklappen klappt ganz gut, aber bisher will alles, was "aufklappt" irgendwelche berechtigungen haben..

will heissen, das display beim anmelden, oder sogar schon bei der anzeige des loginwindows abzuschalten (launchDaemon bzw. -Agent) krieg ich hin, aber wecken nicht... grmpfff...

Beitrag von „Bob_1“ vom 18. Februar 2021, 15:38

du schaffst es, ich drücke dir die Daumen 👍

Beitrag von „grt“ vom 20. Februar 2021, 14:02

fast....

`pmset displaysleepnow` schaltet das display ab, ohne passworтеingabe.

`sleep 5` stoppt das script (hier für 5 sekunden) zwischen den beiden befehlen.

`caffeinate -u -t 5` "behauptet", dass sich irgendein user an dem rechner herumtreibt, und schaltet das display wieder ein, der parameter `-t "ziffer"` verhindert, dass das `caffeinate` den armen rechner unbegrenzt wachhält.

`#!/bin/sh` vorne dran, `exit 0` ans ende, datei mit `chmod a+x /pfad/zur/datei` ausführbar machen.

klappt prächtig.

wenn man jetzt eine plist schreibt:

XML

1. `<?xml version="1.0" encoding="UTF-8"?>`
2. `<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">`
3. `<plist version="1.0">`
4. `<dict>`
5. `<key>Label</key>`
6. `<string>com.user.myscript</string>`
7. `<key>ProgramArguments</key>`
8. `<array><string>/Users/Shared/myscript.sh</string></array>`

9. <key>RunAtLoad</key>
10. <true/>
11. <key>KeepAlive</key>
12. <false/>
13. </dict>
14. </plist>

Alles anzeigen

und die nach /Users/myname/Library/LaunchAgents verfrachtet, wird das script nach der anmeldung einmal ausgeführt, und manuelles zu- und aufklappen ist geschichte.

muss allerdings in der library von allen usern herumliegen. daher bin ich gerade am gucken, wie ich das script schon während der anzeige des loginwindow ausführen kann - also statt in die userlibrary ins system -> Library/LaunchDaemons - allerdings passt hier der ausführungszeitpunkt nicht - ich hab noch ein "afplay /pfad/zu/einer/sounddatei.aiff" am ende des scripts, damit ich höre, wann es ausgeführt wird - und das ist aktuell beim apfel mit dem balken kurz vorm anmeldefenster.

ich bleib aber dran...

EDITH: wenn man die plist statt in /Library/LaunchDaemons nach /Library/LaunchAgents kopiert, und dann den besitzer auf root:wheel (sudo chown root:wheel ...) ändert, wird das script bei jeder anmeldung jedes users ausgeführt. erstmal kann ich damit leben...