Erledigt

[TOOL] „Screen Wakener“ – „Schwarzen Monitor“ an iGPU aktivieren, sofern er in den Systemeinstellungen erkannt wird

Beitrag von „mabam“ vom 20. Mai 2019, 14:39

„Screen Wakener“ ist ein Workaround für Bildschirme, die an einer iGPU hängen und (manchmal oder immer) nach dem Booten schwarz bleiben. Aber nur, wenn der Bildschirm in „Systemeinstellungen“ ▸ „Monitore“ erkannt wird (und z. B. aktiviert werden kann, indem man dort die Auflösung o. Ä. ändert – kann man natürlich nur über einen zweiten, funktionierenden Bildschirm überprüfen).

Wenn das bei dir nicht der Fall ist, empfehle ich den „General Framebuffer Patching Guide (HDMI Black Screen Problem)“ (darf ich hier leider nicht verlinken, also googeln). Die Anleitung dort ist sehr klar geschrieben, wie ich finde, und half mir nach langer Suche, das Problem mit meinem schwarzen Monitor zum lösen (bevor er in den Systemeinstellungen sichtbar war).

„Screen Wakener“ aktiviert den schwarzen Bildschirm vor dem Einloggen (oder danach, falls in den Systemeinstellungen „automatische Anmeldung“ aktiviert ist).

„Screen Wakener“ funktioniert nur mit einer iGPU und für nur einen „Schwarzen Monitor“ gleichzeitig. Wenn deiner an eine eGPU angeschlossen ist und/oder du mehr als einen „Schwarzen Monitor“ hast, müssen Shell Script und Einstellungen manuell editiert werden.

Mehr Infos und Download unter https://github.com/mabam/ScreenWakener bzw.

https://github.com/mabam/ScreenWakener/releases.

Ich empfehle, vor der Nutzung von „Screen Wakener“ die EDID zu patchen. Dazu gibt es verschiedene Anleitungen und Skripte. Einfach nach „EDID patch“ oder „EDID override“ googeln.

Wenn deine Systemsprache deutsch ist, ist „Screen Wakener“ das auch. Ansonsten englisch.

Beitrag von „herrfelix“ vom 20. Mai 2019, 15:59

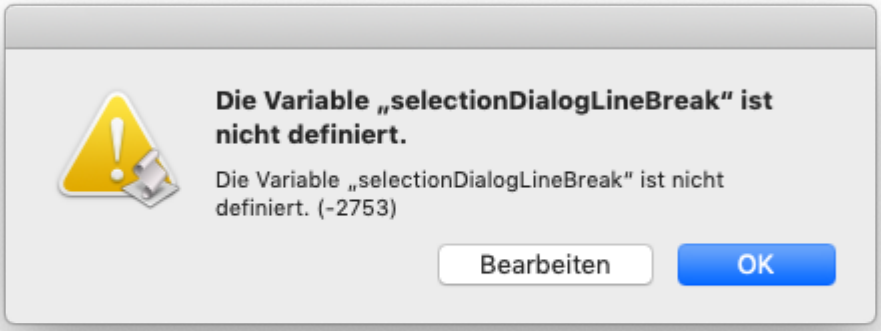
Das schau ich mir gleich mal an, denn ich habe das Problem an meinem Monitor über

DisplayPort auch.

Du solltest aber noch einmal den Link anpassen, im Moment ist der Punkt am Ende des Links mit verlinkt.

EDIT:

Ich bekomme die folgende Fehlermeldung:



Beitrag von „mabam“ vom 20. Mai 2019, 16:57

Danke für beide Hinweise!

Den Link habe ich korrigiert. Die Fehlermeldung schaue ich mir mal an. Selbst bekomme ich die nicht.

Beitrag von „herrfelix“ vom 20. Mai 2019, 17:24

Die Fehlermeldung kommt direkt hiernach:

Beitrag von „mabam“ vom 20. Mai 2019, 20:41

Meine erste Vermutung war, dass ein Fehler in der Logik ist, wenn keine EDID-Override-Datei existiert (oder sich darin kein Monitor-Name befindet). Aber wenn ich die bei mir lösche, läuft Screen Wakener trotzdem ohne Probleme.

Ich habe den entsprechenden Teil aus dem Script rauskopiert und ein paar Variablen darin durch statische Werte ersetzt, damit er für sich lauffähig ist. Siehe anbei.

Kannst du ihn mal bei dir laufen lassen und mir das Ergebnis mitteilen?

EDIT:

Moment: Laut deinem Profil hast du eine RX 570 Nitro+ 4GB. Nutzt du auch die iGPU? Und betreibst du den schwarzen Bildschirm an der Karte oder der iGPU?

Ich habe nur eine iGPU und kann den Script daher nicht auf die Systeminformationen einer eGPU bzgl. der Software Connector abstimmen. Wenn dein Bildschirm an der eGPU hängt,

würde das die Fehlermeldung erklären. In dem Fall bleibt mir nur, den Infos zu Screen Wakener hinzuzufügen, dass es nur für eine iGPU geeignet ist.

Den Launch Agent und Shell Script zur automatischen Aktivierung deines Monitors müsstest du aber vermutlich trotzdem nutzen können. Nur musst du ihn halt einmalig manuell konfigurieren. Das sollte aber nicht schwierig sein.

Beitrag von „herrfelix“ vom 20. Mai 2019, 21:48

Das Teil-Script läuft. Die Ausgabe lautet:

{"1: Hersteller-ID: 1234 • Produkt-ID: 5678 • (AppleIntelFramebuffer@0)"}

Ich nutze nur die RX 570 – die iGPU ist deaktiviert. Dort über DisplayPort hängt mein Monitor dran. Die EDID ist gepatched.

Beitrag von „mabam“ vom 20. Mai 2019, 21:59

Okay, das bestätigt, dass es mit einer eGPU nicht geht. Der AppleScript lässt sich natürlich entsprechend ändern, aber das müsste dann wer auf einem Hack mit eGPU (und idealerweise auch aktivierter iGPU) machen.

Wenn du den Launch Agent manuell mit dem Shell Script betreiben willst, helfe ich dir gerne. Sollte eigentlich eine Kleinigkeit sein. Screen Wakener muss aber an Ort und Stelle bleiben, da sich sowohl der Shell Script als auch die zugehörigen Einstellungen in „Screen Wakener.app“ befinden und der Launch Agent diese Dateien genau da erwartet.

Also wenn du das möchtest, dann gib in Terminal das Folgende ein und poste hier die Ausgabe:

```
/Applications/Screen\ Wakener.app/Contents/Resources/displayplacer list
```

Beitrag von „herrfelix“ vom 20. Mai 2019, 22:18

Sehr gerne! Hier ist die Ausgabe aus dem Terminal.

Code

1. Persistent screen id: 64DDC545-AAAA-A44B-FF5C-8C4FBA9400C2
2. Contextual screen id: 459095581
3. Type: 34 inch external screen
4. Resolution: 3440x1440
5. Hertz: 60
6. Color Depth: 8
7. Scaling:off
8. Origin: (0,0) - main display
9. Rotation: 0
10. Resolutions for rotation 0:
11. mode 0: res:3440x1440 hz:60 color_depth:4
12. mode 1: res:3440x1440 hz:60 color_depth:8 <-- current mode
13. mode 2: res:3440x1440 hz:50 color_depth:4
14. mode 3: res:3440x1440 hz:50 color_depth:8
15. mode 4: res:1024x768 hz:60 color_depth:4
16. mode 5: res:1024x768 hz:60 color_depth:8
17. mode 6: res:800x600 hz:60 color_depth:4
18. mode 7: res:800x600 hz:60 color_depth:8
19. mode 8: res:640x480 hz:60 color_depth:4
20. mode 9: res:640x480 hz:60 color_depth:8
21. mode 10: res:1280x1024 hz:60 color_depth:4
22. mode 11: res:1280x1024 hz:60 color_depth:8
23. mode 12: res:1920x1080 hz:60 color_depth:4
24. mode 13: res:1920x1080 hz:60 color_depth:8
25. mode 14: res:848x480 hz:60 color_depth:4
26. mode 15: res:848x480 hz:60 color_depth:8
27. mode 16: res:1280x960 hz:60 color_depth:4
28. mode 17: res:1280x960 hz:60 color_depth:8
29. mode 18: res:1360x768 hz:60 color_depth:4
30. mode 19: res:1360x768 hz:60 color_depth:8
31. mode 20: res:1600x1200 hz:60 color_depth:4
32. mode 21: res:1600x1200 hz:60 color_depth:8

33. mode 22: res:1920x1440 hz:60 color_depth:4
34. mode 23: res:1920x1440 hz:60 color_depth:8
35. mode 24: res:1720x720 hz:60 color_depth:4
36. mode 25: res:1720x720 hz:60 color_depth:8
37. mode 26: res:1280x720 hz:60 color_depth:4
38. mode 27: res:1280x720 hz:60 color_depth:8
39. mode 28: res:1280x536 hz:60 color_depth:4
40. mode 29: res:1280x536 hz:60 color_depth:8
41. mode 30: res:1344x1008 hz:60 color_depth:4
42. mode 31: res:1344x1008 hz:60 color_depth:8
43. mode 32: res:1344x562 hz:60 color_depth:4
44. mode 33: res:1344x562 hz:60 color_depth:8
45. mode 34: res:1440x900 hz:60 color_depth:4
46. mode 35: res:1440x900 hz:60 color_depth:8
47. mode 36: res:1680x1050 hz:60 color_depth:4
48. mode 37: res:1680x1050 hz:60 color_depth:8
49. mode 38: res:1600x670 hz:60 color_depth:4
50. mode 39: res:1600x670 hz:60 color_depth:8
51. mode 40: res:1920x1200 hz:60 color_depth:4
52. mode 41: res:1920x1200 hz:60 color_depth:8
53. mode 42: res:1920x804 hz:60 color_depth:4
54. mode 43: res:1920x804 hz:60 color_depth:8
55. mode 44: res:2048x858 hz:60 color_depth:4
56. mode 45: res:2048x858 hz:60 color_depth:8
57. mode 46: res:2560x1440 hz:60 color_depth:4
58. mode 47: res:2560x1440 hz:60 color_depth:8
59. mode 48: res:2560x1600 hz:60 color_depth:4
60. mode 49: res:2560x1600 hz:60 color_depth:8
61. mode 50: res:3440x1934 hz:60 color_depth:4
62. mode 51: res:3440x1934 hz:60 color_depth:8
63. mode 52: res:1720x720 hz:60 color_depth:4 scaling:on
64. mode 53: res:1720x720 hz:60 color_depth:8 scaling:on
65. mode 54: res:1720x720 hz:50 color_depth:4 scaling:on
66. mode 55: res:1720x720 hz:50 color_depth:8 scaling:on
67. mode 56: res:960x540 hz:60 color_depth:4 scaling:on
68. mode 57: res:960x540 hz:60 color_depth:8 scaling:on
69. mode 58: res:800x600 hz:60 color_depth:4 scaling:on
70. mode 59: res:800x600 hz:60 color_depth:8 scaling:on
71. mode 60: res:960x720 hz:60 color_depth:4 scaling:on
72. mode 61: res:960x720 hz:60 color_depth:8 scaling:on
73. mode 62: res:720x450 hz:60 color_depth:4 scaling:on

74. mode 63: res:720x450 hz:60 color_depth:8 scaling:on
75. mode 64: res:840x525 hz:60 color_depth:4 scaling:on
76. mode 65: res:840x525 hz:60 color_depth:8 scaling:on
77. mode 66: res:960x600 hz:60 color_depth:4 scaling:on
78. mode 67: res:960x600 hz:60 color_depth:8 scaling:on
79. mode 68: res:1280x720 hz:60 color_depth:4 scaling:on
80. mode 69: res:1280x720 hz:60 color_depth:8 scaling:on
81. mode 70: res:1280x800 hz:60 color_depth:4 scaling:on
82. mode 71: res:1280x800 hz:60 color_depth:8 scaling:on
83. mode 72: res:1720x967 hz:60 color_depth:4 scaling:on
84. mode 73: res:1720x967 hz:60 color_depth:8 scaling:on
85.
86. Execute the command below to set your screens to the current arrangement:
87.
88. displayplacer "id:64DDC545-AAAA-A44B-FF5C-8C4FBA9400C2 res:3440x1440 hz:60 color_depth:8 scaling:off origin:(0,0) degree:0"

Alles anzeigen

Beitrag von „mabam“ vom 20. Mai 2019, 22:54

Nachdem du außer deinem „schwarzen Monitor“ offensichtlich keinen anderen angeschlossen hast, ist in obiger Ausgabe auch nur einer zu finden. Das erleichtert die Auswahl 🙂

Wir brauchen die Contextual screen id, also „459095581“.

Der aktuelle Modus ist „1“. Ich nehme an, dass das der Modus ist, auf den der Bildschirm auch nach dem Aktivieren wieder eingestellt werden soll?

Dann nehmen wir als temporären Modus (denn wenn wir ihn nicht ändern, bleibt der Monitor schwarz) am besten „3“, würde ich vorschlagen. Denn zum Aktivieren muss die Auflösung und/oder Bildwiederholrate geändert werden. Über nur die Farbtiefe geht das nicht. Und über die Bildwiederholrate ist es unauffälliger 🙂

Wechsle in Terminal bitte nach ...:

```
cd /Applications/Screen\ Wakener.app/Contents/Resources/
```

... und erstelle die Einstellungs-Datei:

```
echo -e "DisplayID=459095581\ncurrentMode=1\ntempMode=3" > ./ScreenWakenerPrefs
```

Und dann muss der Launch Agent in den entsprechenden Ordner kopiert werden ...:

```
sudo cp ./org.mabam.ScreenWakener.plist /Library/LaunchAgents/
```

... und root als Eigentümer bekommen:

```
sudo chown root:wheel /Library/LaunchAgents/org.mabam.ScreenWakener.plist
```

Wenn es zwischen iGPU und eGPU nicht noch irgendwelche entscheidenden Unterschiede gibt, sollte der Bildschirm nach einem Neustart eigentlich aktiviert werden. Also im Login Window oder, wenn du Auto-Login eingestellt hast, nach dem automatischen Einloggen.

Bin gespannt!

Poste nach dem Neustart doch mal das Log:

```
cat /Applications/Screen\ Wakener.app/Contents/Resources/ScreenWakenerLog.txt
```

Beitrag von „herrfelix“ vom 21. Mai 2019, 09:08

Super, ich danke dir! Nach dem Neustart sagt mir das Log, dass der Wakender zumindest läuft.

Tue May 21 09:06:34 CEST 2019

Contextual screen id: 459095581

mode 1: res:3440x1440 hz:60 color_depth:8 <-- current mode

Ich teste es dann im laufe des Tages mal mit dem Sleep und melde mich wieder.

Beitrag von „mabam“ vom 21. Mai 2019, 09:28

Du scheinst Auto-Login eingestellt zu haben. Das sieht soweit schonmal gut aus. D. h., der Monitor blieb jetzt nach dem Neustart nicht schwarz? Oder hast du das nur manchmal?

Bleibt er bei dir auch nach dem Sleep schwarz? Das hatte ich selbst nie. ~~Dann muss im Launch Agent was geändert werden.~~ Schaue ich mir später an.

EDIT (jetzt etwas ausführlicher):

Nicht am Launch Agent muss dann was geändert werden, sondern das ließe sich über SleepWatcher bewerkstelligen.

Der Eintrag im Log sagt übrigens nicht unbedingt aus, dass die Aktivierung auch funktioniert hat, sondern bedeutet normalerweise einfach, dass Screen Wakener den entsprechenden Befehl ausgelöst hat. Auf meinem Rechner hat das dann auch zur Folge, dass der Bildschirm aktiviert wurde. Auf anderen Rechnern eigentlich auch, aber das kann ich nicht mit Bestimmtheit sagen. Wenn dein Monitor nicht immer schwarz bleibt, sondern nur manchmal, kannst du es nur herausfinden, indem du Screen Wakener eine zeitlang verwendest. So habe ich das auch gemacht.

Liege ich richtig in der oben geäußerten Annahme, dass du Auto-Login aktiviert hast? Das würde sich dann nämlich mit den an meinem Rechner gemachten Beobachtungen decken (sonst wäre im Log eine Kleinigkeit anders).

Sorry für die Fragen, aber der eine Rechner ist nicht der andere und ich brauche zuerst bestimmte Erfahrungswerte mit dem Tool auf anderen Systemen, bevor ich manche Aussagen mit größerer Bestimmtheit treffen kann.

Wenn du möchtest, können wir SleepWatcher gerne angehen. Der Skript, den SleepWatcher dann auslösen muss, wird entweder sehr ähnlich oder sogar derselbe sein wie der, den der jetzige Launch Agent auslöst (SleepWatcher hat für einen Skript mit anderer Funktion bei mir gut funktioniert, sollte also auch hierfür kein Problem sein). Bevor wir das angehen, würde ich allerdings zuerst sichergehen wollen, dass der manuell konfigurierte Screen Wakener bei dir sauber läuft. Daher die Fragen.

EDIT 2:

Vielleicht noch eine Anmerkung zur Verdeutlichung: „Login Screen“ ist offiziell nur der Anmeldebildschirm nach dem *Booten* des Rechners. Auch, wenn man einen gleichen oder ähnlichen auch nach dem Aufwachen aus Sleep bekommt (sofern man den in den Einstellungen nicht abgeschaltet hat), nennt Apple ihn offiziell *nicht* Login Screen. Für die Definition, wann ein Launch Agent auslösen soll, macht das einen entscheidenden Unterschied. Meines Wissens kann man einen Launch Agent nicht so einstellen, dass er nach dem Aufwachen aus Sleep auslöst. Darum habe ich SleepWatcher ins Spiel gebracht.

Beitrag von „user232“ vom 21. Mai 2019, 12:31

Ich habe hier ein ZBOX das mir mit unterschiedlichsten Konfigurationen den DUAL-Screen verweigert. Solltest du dich mal mittels TV austoben wollen, dann schreib mir bitte 🙂

Beitrag von „mabam“ vom 21. Mai 2019, 23:17

> Zitat von user232
>
> Ich habe hier ein ZBOX das mir mit unterschiedlichsten Konfigurationen den DUAL-Screen verweigert. Solltest du dich mal mittels TV austoben wollen, dann schreib mir bitte 🙂

Da muss ich leider passen, weil ich mich da nicht auskenne – tut mir leid.

Für Screen Wakener habe ich auch nur 1+1 zusammengezählt:

Nachdem die iGPU mithilfe des eingangs erwähnten „General Framebuffer Patching Guide“ dann richtig lief, habe ich festgestellt, dass der manchmal schwarze Bildschirm sich aktivieren lässt, indem ich seine Konfiguration in den Systemeinstellungen ändere.

Auf meiner Suche nach einem Tool, das sich per Shell steuern lässt und somit über einen Skript automatisieren (statt immer manuell in die Systemeinstellungen gehen zu müssen), fand ich displayplacer. Um den herum habe ich dann den AppleScript, Shell Skript und Launch Agent gebaut, um die nötigen Informationen zum Bildschirm aus dem System zu ziehen und displayplacer mit dem entsprechenden Befehl zu füttern. Das ist im Prinzip alles.

Tiefergehende Kenntnisse zur Materie habe ich auch nicht, nur ein paar Erfahrungswerte, die sich auf das beziehen, was Screen Wakener tut.

Beitrag von „user232“ vom 22. Mai 2019, 12:22

okay, schade um die Box, wäre echt ein super Teil, aber ohne zuverlässigen Dualscreen als Hack für mich unbrauchbar.

Beitrag von „mabam“ vom 23. Mai 2019, 23:59

> Zitat von herrfelix
>
> Das Teil-Script läuft. Die Ausgabe lautet:
>
> {"1: Hersteller-ID: 1234 • Produkt-ID: 5678 • (AppleIntelFramebuffer@0)"}

> Ich nutze nur die RX 570 – die iGPU ist deaktiviert. Dort über DisplayPort hängt mein Monitor dran. Die EDID ist gepatched.

herrfelix :

Ich probiere jetzt doch, Screen Wakener auch für eGPUs und nicht nur iGPUs zum Laufen zu bringen. Ist nämlich glaube ich einfacher, als ich zuerst dachte.

Wärst du bereit, den angehängten Teil-Skript mal bei dir laufen zu lassen und das Ergebnis hier zu posten? Das würde mir sehr helfen.

Alles was der Skript tut ist, sich Infos zum Software Connector aus der IO Registry zu ziehen.

@alle:

Dieser Skript sollte eigentlich auf jedem Mac/Hack laufen. Wenn ihn noch andere hier im Forum testen möchten – gerne auch auf Rechnern mit mehreren Monitoren die am besten noch an sowohl iGPU als eGPU angeschlossen sind – wäre das ebenso hilfreich für mich.

Beitrag von „herrfelix“ vom 25. Mai 2019, 11:43

Gerade habe ich keine Zeit zum testen, werde mich aber die Tage dazu melden!

Beitrag von „mabam“ vom 25. Mai 2019, 23:16

Beitrag von „herrfelix“ vom 27. Mai 2019, 13:27

Hey, also. Ich hab tatsächlich zu Beginn zu unaufmerksam gelesen. Ich habe das Problem des schwarzen Bildschirms NUR nach dem Sleep – auch nicht immer, aber vor allem nach längeren Sleeps (über Nacht etc.). Vielleicht sollten wir also noch einmal kurz klären, ob dein Script dafür überhaupt taugt.

Auto Login habe ich aktiviert.

Das letzte Teilscript, dass du angehängt hast, gibt dieses aus:

{"ATY,AMD,RadeonFramebuffer@1"}

Ich teste jetzt mal den SleepWatcher.

Beitrag von „mabam“ vom 27. Mai 2019, 14:25

Solange du Auto-Login aktiviert hast, kannst du zum Testen mit SleepWatcher meinen Skript in /Applications/Screen\ Wakener.app/Contents/Resources/org.mabam.ScreenWakener.sh verwenden.

Wenn es so läuft, wie es soll, ist es aber „sauberer“, ein paar Sachen aus dem Skript rauszuschmeißen. Allerdings verweist der Skript auch auf

`/Applications/Screen\ Wakener.app/Contents/Resources/ScreenWakenerLogSwitch`,

`/Applications/Screen\ Wakener.app/Contents/Resources/ScreenWakenerPrefs` und

`/Applications/Screen\ Wakener.app/Contents/Resources/displayplacer`.

Die Befehle zum Loggen kannst du dann rausschmeißen, wodurch die erste Zeile überflüssig wird. Wenn du die Preferences fest in den Skript einbaust, ist die zweite Zeile auch überflüssig. Ohne `displayplacer` in der dritten Zeile geht aber gar nichts. Den könntest du aber auch nach /usr/local/bin kopieren und `root:wheel` als Owner setzen, damit auch die dritte Zeile unnötig

wird und du ScreenWakener.app komplett löschen kannst.

Wenn dein Test mit SleepWatcher erfolgreich ist und du brauchst Hilfe, um den Skript für dauerhaften Betrieb „sauber“ einzustellen, dann gib bescheid.

Danke für die Ausgabe des Skripts. Wenn ich diese Änderung einbaue, sollte auch die automatische Konfiguration von Screen Wakener auf deinem Rechner laufen. Ich habe aber von jemand anderem die Rückkopplung, dass dieser Teil-Skript auf seinem Rechner nicht läuft. Das muss ich mir also nochmal anschauen.

Um Screen Wakener auch zum Aktivieren nach Sleep verwenden zu können, könnte ich SleepWatcher darin einbauen und in einem der Dialoge eine entsprechende Option anbieten. Weiß noch nicht, ob ich das mache. Erstmal möchte ich schauen, dass er auf allen Macs/Hacks lauffähig ist.

Beitrag von „mabam“ vom 27. November 2022, 00:17

Ich habe kleine Bugs behoben, die mit Logging und Deinstallation von Screen Wakener zu tun haben.

Da ich nur eine iGPU habe, habe ich die Funktionalität nicht mehr auf andere GPUs erweitert.

Habe das Tool unter High Sierra und Catalina benutzt und jetzt tut es unter Ventura seinen Dienst. Bei Bedarf ist es nach wie vor unter dem Link im ersten Post zu haben.

Beitrag von „Sascha_77“ vom 27. November 2022, 10:08

Bzgl. SleepWatcher hatte ich mal eine Gui gebaut

https://www.sl-soft.de/sleep-o-mat/

Beitrag von „mabam“ vom 27. November 2022, 11:32

Cool!

sleepwatcher habe ich hier im Einsatz: Wacom Intuos S funktioniert nicht nach dem Aufwachen

Aber dein Tool muss ich mir merken.