

Erledigt

DSDT für Asus H170M-Plus

Beitrag von „farbweiß“ vom 2. Januar 2018, 00:20

Hallo und ein frohes neues Jahr zusammen,

ich probiere eine DSDT für oben genanntes Mainboard zu erstellen.

Ich orientiere mich dabei an den grandiosen Recourcen von [@al6042](#):

https://www.youtube.com/watch?time_continue=325&v=4quIWFqIKIU

[https://www.hackintosh-forum.d...php/FAQ/?category=20-ACPI](https://www.hackintosh-forum.de...php/FAQ/?category=20-ACPI) (ich glaube das ist von ihm)

Leider komme ich am ende nicht bei einer gut funktionierenden DSDT heraus. Ich habe einige Verständnisfragen zu den Tutorials und würde mich freuen wenn ihr mir weiterhelfen könntet.

Mein System:

Spoiler anzeigen

Ohne DSDT läuft das System ziemlich einwandfrei (sleep, sound, multi-monitor, speedstep, usb (nicht sicher)) mit folgenden Settings:

Config.plist und Kexte:

Spoiler anzeigen

Mein Ziel wäre es die Relevanten Einstellungen der config.plist in die DSDT aufzunehmen, sowie meine Geräte richtig in der DSDT zu definieren.

Mit meinen Versuchen bekomme ich am ende eine bootende DSDT ohne Sound (und vermutlich auch ohne Sleep), allerdings steht mit dieser unter "Systembericht -PCI" keine Tabelle mehr sondern ein einzelner Eintrag zu meiner GPU.

Für das Erstellen der DSDT habe ich mich an der Anleitung "3. Standard DSDT-Patch Verlauf für Desktops Skylake-Chipsätze:" orientiert :

<https://www.hackintosh-forum.d...php/FAQ/?category=20-ACPI>

Vielleicht schreibe ich hier meine Anmerkungen und Fragen zur Anleitung und damit klärt sich schon alles:

1. Extrahiert mit CLOver (f4)

zu Punkt 2 und 3: Ich habe in der DSDT jeweils HDAS-HDEF und HECI-IMEI "find&replaced"

zu Punkt 4

als annähernden Patch habe ich das H97M-E genommen und den Patch folgendermaßen angepasst (realtec Lan raus und "Intel Series 9" mit "Intel Series10" ersetzt:

Code

```
1. # Maintained by: PJALM (help@pjalm.com) for: http://pjalm.com/repos/
2. # These patches are the registered property of PJALM.COM and can not be
3. # redistributed or modified without the written consent of PJALM.COM.
4. # Links to these patches are allowed. All material is protected under the DMCA.
5.
6.
7. # Last Updated : 05/07/2016
8. # Patch Name : H97M-E
9. # Patch Version : 1.01
10.
11.
12. #IASL:Check $BUILD = 20131218
13.
14.
15. #External Fixes
16. into_all all code_regex \\\*\\sExternal\\sreference\\s\\*\\s+ removeall_matched;
17. into definitionblock code_regex ,\\sUnknownObj removeall_matched;
18.
19.
20. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEG0\\)\\s+ removeall_matched;
21. into      definitionblock      code_regex      External\\s\\(_SB_.PCI0.PEG0.PEGP\\)\\s+
    removeall_matched;
22. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEG1\\)\\s+ removeall_matched;
23. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEG2\\)\\s+ removeall_matched;
24. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEGP\\)\\s+ removeall_matched;
```

```

25. into      definitionblock      code_regex      External\s\(_SB_.PCI0.PEGP.GFX0\)\s+
    removeall_matched;
26.
27.
28. # Add the DTGP method
29. into method label DTGP remove_entry;
30. into definitionblock code_regex . insert begin
31. Method (DTGP, 5, NotSerialized)\n
32. {\n
33. If (LEqual (Arg0, Buffer (0x10))\n
34. {\n
35. /* 0000 */ 0xC6, 0xB7, 0xB5, 0xA0, 0x18, 0x13, 0x1C, 0x44,\n
36. /* 0008 */ 0xB0, 0xC9, 0xFE, 0x69, 0x5E, 0xAF, 0x94, 0x9B\n
37. }))\n
38. {\n
39. If (LEqual (Arg1, One))\n
40. {\n
41. If (LEqual (Arg2, Zero))\n
42. {\n
43. Store (Buffer (One) { 0x03 }, Arg4)\n
44. Return (One)\n
45. }\n
46. If (LEqual (Arg2, One))\n
47. {\n
48. Return (One)\n
49. }\n
50. }\n
51. }\n
52. Store (Buffer (One) { 0x00 }, Arg4)\n
53. Return (Zero)\n
54. }
55. end;
56.
57.
58. # Add Darwin to the supported operating systems
59. into method label _INI code_regex (\s+)(If\s\(_OSI\s\("Windows\s2001"\)\))
    replace_matched begin \n
60. If (_OSI ("Darwin"))\n
61. {\n
62. Store (0x2710, OSYS)\n
63. }\n

```

```

64. \n
65. $2
66. end;
67.
68.
69. # Intel 9 Series Fixes
70. into_all all code_regex _T_([0-6]) replaceall_matched begin T_%1 end;
71.
72.
73. into_all method label _INI code_regex INIR\n replace_matched begin Store (Zero, INIR)\n
    end;
74.
75.
76. into device label PCI0 code_regex (\s+Zero){2,} remove_matched;
77. into_all method label ADBG code_regex (?<=Return\s\((MDBG\))\s+Arg0 remove_matched;
78. into_all method label _PS3 code_regex PS3X\n replace_matched begin Store (Zero,
    PS3X)\n end;
79. into_all method label _PS2 code_regex PS2X\n replace_matched begin Store (Zero,
    PS2X)\n end;
80. into_all method label _PS0 code_regex PS0X\n replace_matched begin Store (Zero,
    PS0X)\n end;
81.
82.
83. into_all method label GPEH insert begin Return (Zero) end;
84.
85.
86. into      method      label      _CRS      parent_label      PCI0      code_regex
    DWordField\s\((BUF0,\s\\_SB.PCI0._Y1E._LEN,\sMSLN\) replace_matched begin QWordField
    (BUF0, \\_SB.PCI0._Y1E._LEN, MSLN) end;
87.
88.
89. into_all all code_regex Acquire\s\((MUT0,\s0x0FFF\) replace_matched begin Acquire
    (MUT0, 0xFFFF) end;
90.
91.
92. into method label _HID parent_label _SB.PCI0.LPCB.TPM code_regex (Return\s\((Zero\))
    remove_matched;
93. into method label _HID parent_label _SB.PCI0.LPCB.TPM insert begin Return (Zero) end;
94.
95.
96. into method label _PS0 parent_label NVM0 code_regex (Return\s\((Zero\))
    remove_matched;
97. into method label _PS0 parent_label NVM0 insert begin Return (Zero) end;

```

```

98.
99.
100. into method label _PS0 parent_label XHC code_regex (Return\s\((Zero\)) remove_matched;
101. into method label _PS0 parent_label XHC insert begin Return (Zero) end;
102.
103.
104. into method label _PS0 parent_label RHUB code_regex (Return\s\((Zero\))
    remove_matched;
105. into method label _PS0 parent_label RHUB insert begin Return (Zero) end;
106.
107.
108. into method label _PS3 parent_label NVM0 code_regex (Return\s\((Zero\))
    remove_matched;
109. into method label _PS3 parent_label NVM0 insert begin Return (Zero) end;
110.
111.
112. into method label _PS3 parent_label XHC code_regex (Return\s\((Zero\)) remove_matched;
113. into method label _PS3 parent_label XHC insert begin Return (Zero) end;
114.
115.
116. into method label _DSM parent_label SHUB code_regex (Return\s\((Zero\))
    remove_matched;
117. into method label _DSM parent_label SHUB insert begin Return (Zero) end;
118.
119.
120. into method label RDCA parent_label SHUB code_regex (Return\s\((Zero\))
    remove_matched;
121. into method label RDCA insert begin Return (Zero) end;
122.
123.
124. into method label CNRS parent_label SHUB code_regex (Return\s\((Zero\))
    remove_matched;
125. into method label CNRS insert begin Return (Zero) end;
126.
127.
128. into method label SPL1 parent_label SHUB code_regex (Return\s\((Zero\))
    remove_matched;
129. into method label SPL1 insert begin Return (Zero) end;
130.
131.
132. into method label OE1X parent_label _GPE code_regex (Return\s\((Zero\))
    remove_matched;
133. into method label OE1X parent_label _GPE insert begin Return (Zero) end;
134.

```

```

135.
136. # into method label TINL parent_label _SB.PCI0.RP05 code_regex (Return\s\((Zero\))
    remove_matched;
137. into method label TINL parent_label _SB.PCI0.RP05 insert begin Return (Zero) end;
138.
139.
140. into method label TWAK parent_label _SB.PCI0.RP05 code_regex (Return\s\((Zero\))
    remove_matched;
141. into method label TWAK parent_label _SB.PCI0.RP05 insert begin Return (Zero) end;
142.
143.
144. into method label TPTS parent_label _SB.PCI0.RP05 code_regex (Return\s\((Zero\))
    remove_matched;
145. into method label TPTS parent_label _SB.PCI0.RP05 insert begin Return (Zero) end;
146.
147.
148. into      method      label      SMSR      parent_label      _SB      code_regex
    CreateDWordField\s\((Arg0,\s0x04,\sAEBX)\)\s+ remove_matched;
149. into      method      label      SDSP      parent_label      AMW0      code_regex
    CreateDWordField\s\((Arg0,\sZero,\sACTN)\)\s+ remove_matched;
150. into      method      label      GDSP      parent_label      AMW0      code_regex
    CreateDWordField\s\((Arg0,\sZero,\sACTN)\)\s+ remove_matched;
151. into      method      label      DEVS      parent_label      AMW0      code_regex
    CreateDWordField\s\((Arg0,\s0x04,\sCPAR)\)\s+ remove_matched;
152.
153.
154. # Device injection for the Intel 9 Series SATA
155. into device label SAT0 set_label begin SATA end;
156. into_all all code_regex SAT0 replaceall_matched begin SATA end;
157. into device label SAT1 remove_entry;
158. into_all scope label _SB.PCI0.SAT0.PRT0 set_label begin _SB.PCI0.SATA.PRT0 end;
159. into_all scope label _SB.PCI0.SAT0.PRT1 set_label begin _SB.PCI0.SATA.PRT1 end;
160. into_all scope label _SB.PCI0.SAT0.PRT2 set_label begin _SB.PCI0.SATA.PRT2 end;
161. into_all scope label _SB.PCI0.SAT0.PRT3 set_label begin _SB.PCI0.SATA.PRT3 end;
162. into method label _DSM parent_label SATA remove_entry;
163. into device label SATA insert begin
164. Method (_DSM, 4, NotSerialized)\n
165. {\n
166. Store (Package (0x0A) {\n
167. "AAPL,slot-name", "Built In",\n
168. "name", "Intel AHCI Controller",\n
169. "model", Buffer (0x2D) {"Intel 10 Series Chipset Family SATA Controller"},\n

```

```

170. "device_type", Buffer (0x0F) {"AHCI Controller"},\n
171. "device-id", Buffer (0x04) {0x02,0x1E,0x00,0x00}\n
172. }, Local0)\n
173. DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))\n
174. Return (Local0)\n
175. }
176. end;
177.
178.
179. into method label SDSP parent_label AMW0 code_regex (Create) replace_matched begin
    // %1 end;
180. into method label GDSP parent_label AMW0 code_regex (Create) replace_matched begin
    // %1 end;
181. into      method      label      SMSR      parent_label      _SB      code_regex
    (CreatedWordField\s\(\Arg0,\s0x04,\sAEBX\)) replace_matched begin // %1 end;
182. # Add the missing MCHC device
183. into device label MCHC parent_label PCI0 remove_entry;
184. into device label PCI0 insert begin
185. Device (MCHC)\n
186. {\n
187. Name (_ADR, Zero)\n
188. }
189. end;
190.
191.
192. # Fix the SMBUS to allow for loading of the AppleSMBusController.kext
193. into device label BUS0 parent_label SBUS remove_entry;
194. into device label SBUS insert begin
195. Device (BUS0)\n
196. {\n
197. Name (_CID, "smbus")\n
198. Name (_ADR, Zero)\n
199. Device (DVL0)\n
200. {\n
201. Name (_ADR, 0x57)\n
202. Name (_CID, "diagsvault")\n
203. Method (_DSM, 4, NotSerialized)\n
204. {\n
205. Store (Package (0x02) {\n
206. "address", 0x57
207. }, Local0)\n

```

```

208. DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))\n
209. Return (Local0)\n
210. }\n
211. }\n
212. }
213. end;
214.
215.
216. # Patches the Intel USB3 on Intel 9 Series chipsets
217. into method label _DSM parent_label XHC remove_entry;
218. into device label XHC insert begin
219. Method (_DSM, 4, NotSerialized)\n
220. {\n
221. Store (Package (0x15) {\n
222. "AAPL,slot-name", "Built In",\n
223. "name", "Intel XHCI Controller",\n
224. "model", Buffer (0x37) {"Intel 10 Series Chipset Family USB xHCI Host Controller"},\n
225. "device_type", Buffer (0x0E) {"USB Controller"},\n
226. "AAPL,current-available", 0x0834,\n
227. "AAPL,current-extra", 0x0A8C,\n
228. "AAPL,current-in-sleep", 0x03E8,\n
229. "AAPL,current-extra-in-sleep", 0x0834,\n
230. "AAPL,max-port-current-in-sleep", 0x0A8C,\n
231. "AAPL,device-internal", 0x02,\n
232. Buffer (One) {0x00}\n
233. }, Local0)\n
234. DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))\n
235. Return (Local0)\n
236. }
237. end;

```

Alles anzeigen

Danach musste ich sat0 zu SATA umbenennen

Punkt 5,6,7,8 wie in der Anleitung

Danach habe ich den Patch für Z97X-UD7-TH angepasst um mein Intel Lan einzubauen (hier ist der fehler, oder ?)

Code

1. # Maintained by: PJALM (help@pjalm.com) for: <http://pjalm.com/repos/>


```

2.
3.
4. # These patches are the registered property of PJALM.COM and can not be
5. # redistributed or modified without the written consent of PJALM.COM.
6. # Links to these patches are allowed. All material is protected under the DMCA.
7.
8.
9. # Last Updated : 05/07/2016
10. # Patch Name : Z97X-UD7-TH
11. # Patch Version : 1.0
12.
13.
14. #External Fixes
15. into_all all code_regex \\*\\sExternal\\sreference\\s\\*\\s+ removeall_matched;
16. into definitionblock code_regex ,\\sUnknownObj removeall_matched;
17.
18.
19. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEG0\\)\\s+ removeall_matched;
20. into      definitionblock      code_regex      External\\s\\(_SB_.PCI0.PEG0.PEGP\\)\\s+
    removeall_matched;
21. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEG1\\)\\s+ removeall_matched;
22. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEG2\\)\\s+ removeall_matched;
23. into definitionblock code_regex External\\s\\(_SB_.PCI0.PEGP\\)\\s+ removeall_matched;
24. into      definitionblock      code_regex      External\\s\\(_SB_.PCI0.PEGP.GFX0\\)\\s+
    removeall_matched;
25.
26.
27. # Intel 9 Series Fixes
28. into_all all code_regex _T_([0-6]) replaceall_matched begin T_%1 end;
29.
30.
31. into_all method label _INI code_regex INIR\\n replace_matched begin Store (Zero, INIR)\\n
    end;
32.
33.
34. into device label PCI0 code_regex (\\s+Zero){2,} remove_matched;
35. into_all method label ADBG code_regex (?<=Return\\s\\(MDBG\\))\\s+Arg0 remove_matched;
36. into_all method label _PS3 code_regex PS3X\\n replace_matched begin Store (Zero,
    PS3X)\\n end;
37. into_all method label _PS2 code_regex PS2X\\n replace_matched begin Store (Zero,
    PS2X)\\n end;
38. into_all method label _PS0 code_regex PS0X\\n replace_matched begin Store (Zero,
    PS0X)\\n end;
39.

```

```

40.
41. into_all method label GPEH insert begin Return (Zero) end;
42.
43.
44. into      method      label      _CRS      parent_label      PCI0      code_regex
      DWordField\s\(BUF0,\s\\_SB.PCI0._Y1E._LEN,\sMSLN\) replace_matched begin QWordField
      (BUF0, \\_SB.PCI0._Y1E._LEN, MSLN) end;
45.
46.
47. into_all all code_regex Acquire\s\(MUT0,\s0x0FFF\) replace_matched begin Acquire
      (MUT0, 0xFFFF) end;
48.
49.
50. into method label _HID parent_label _SB.PCI0.LPCB.TPM code_regex (Return\s\((Zero\))
      remove_matched;
51. into method label _HID parent_label _SB.PCI0.LPCB.TPM insert begin Return (Zero) end;
52.
53.
54. into method label _PS0 parent_label NVM0 code_regex (Return\s\((Zero\))
      remove_matched;
55. into method label _PS0 parent_label NVM0 insert begin Return (Zero) end;
56.
57.
58. into method label _PS0 parent_label XHC code_regex (Return\s\((Zero\)) remove_matched;
59. into method label _PS0 parent_label XHC insert begin Return (Zero) end;
60.
61.
62. into method label _PS0 parent_label RHUB code_regex (Return\s\((Zero\))
      remove_matched;
63. into method label _PS0 parent_label RHUB insert begin Return (Zero) end;
64.
65.
66. into method label _PS3 parent_label NVM0 code_regex (Return\s\((Zero\))
      remove_matched;
67. into method label _PS3 parent_label NVM0 insert begin Return (Zero) end;
68.
69.
70. into method label _PS3 parent_label XHC code_regex (Return\s\((Zero\)) remove_matched;
71. into method label _PS3 parent_label XHC insert begin Return (Zero) end;
72.
73.
74. into method label _DSM parent_label SHUB code_regex (Return\s\((Zero\))
      remove_matched;

```

```

75. into method label _DSM parent_label SHUB insert begin Return (Zero) end;
76.
77.
78. into method label RDCA parent_label SHUB code_regex (Return\s\(\Zero\))
    remove_matched;
79. into method label RDCA insert begin Return (Zero) end;
80.
81.
82. into method label CNRS parent_label SHUB code_regex (Return\s\(\Zero\))
    remove_matched;
83. into method label CNRS insert begin Return (Zero) end;
84.
85.
86. into method label SPL1 parent_label SHUB code_regex (Return\s\(\Zero\))
    remove_matched;
87. into method label SPL1 insert begin Return (Zero) end;
88.
89.
90. into method label OE1X parent_label _GPE code_regex (Return\s\(\Zero\))
    remove_matched;
91. into method label OE1X parent_label _GPE insert begin Return (Zero) end;
92.
93.
94. # into method label TIN1 parent_label _SB.PCI0.RP05 code_regex (Return\s\(\Zero\))
    remove_matched;
95. into method label TIN1 parent_label _SB.PCI0.RP05 insert begin Return (Zero) end;
96.
97.
98. into method label TWAK parent_label _SB.PCI0.RP05 code_regex (Return\s\(\Zero\))
    remove_matched;
99. into method label TWAK parent_label _SB.PCI0.RP05 insert begin Return (Zero) end;
100.
101.
102. into method label TPTS parent_label _SB.PCI0.RP05 code_regex (Return\s\(\Zero\))
    remove_matched;
103. into method label TPTS parent_label _SB.PCI0.RP05 insert begin Return (Zero) end;
104.
105.
106. into method label SMSR parent_label _SB code_regex
    CreateDWordField\s\(\Arg0,\s0x04,\sAEBX\)\s+ remove_matched;
107. into method label SDSP parent_label AMW0 code_regex
    CreateDWordField\s\(\Arg0,\sZero,\sACTN\)\s+ remove_matched;

```

```

108. into      method      label      GDSP      parent_label      AMW0      code_regex
      CreateDWordField\s\\(Arg0,\sZero,\sACTN\\)\s+ remove_matched;
109. into      method      label      DEVS      parent_label      AMW0      code_regex
      CreateDWordField\s\\(Arg0,\s0x04,\sCPAR\\)\s+ remove_matched;
110.
111.
112. # Fixes 'Built-in' status for Intel Gigabit Ethernet
113. into method label _DSM parent_label GLAN remove_entry;
114. into device label GLAN insert begin
115. Method (_DSM, 4, NotSerialized)\n
116. {\n
117. Store (Package (0x0C) {\n
118. "AAPL,slot-name", "Built In",\n
119. "name", "Intel Ethernet Controller",\n
120. "model", "Intel I217-V Gigabit Network Controller",\n
121. "device_type", Buffer (0x13) {"Ethernet Controller"},\n
122. "built-in", Buffer (One) {0x01},\n
123. "location", Buffer (0x02) {"1"}\n
124. }, Local0)\n
125. DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))\n
126. Return (Local0)\n
127. }
128. end;

```

Alles anzeigen

Den Realtec Audio eintrag habe ich von den Toleda Patches angepasst (und die layout ID auf 7 gesetzt, damit es die funktionierende Einstellung - audio layout id for AppleALC - aus der config.plist ersetzt)

die anderen Schritte waren wieder wie in der Anleitung.

Ich muss sagen ich kapiere grob, was ich da mache mit der DSDT, ich kapiere aber vor allem, dass ich ohne eure Hilfe aufgeschmissen bin und wäre euch darum sehr dankbar.
Bitte weißt mich darauf hin wenn mein Betrag falsch Formatiert oder ähnliches ist.

[/b]

Beitrag von „al6042“ vom 2. Januar 2018, 16:21

Hallöchen... 😊

Ich habe mir mal deine Daten angeschaut und aus der Original-DSDT und der config.plist ein Päckchen geschnürt, dass dir ggf. helfen könnte...

Die config ist um jegliche, unnötige DSDT-Patches und Fixes bereinigt, da diese ja auch in der gepatchten DSDT untergebracht sind... 😊

Zusätzlich habe ich dir die Kexte meines EFI-Ordners beigelegt, da diese in der Kombi mit den anderen Dateien einwandfrei funktionieren sollten...

Beitrag von „farbweiß“ vom 2. Januar 2018, 17:45

WOW! vielen Dank.

ich musste nur noch den nvda_drv = 1 bootflag setzen und/oder im Clover den "DataHubDxe" ausschalten.

Reiner Zufall, dass ich das probiert habe. Ich konnte die Verbose Ausgabe von Clover nicht schnell genug lesen,
da er ohne zu stoppen direkt neu gestartet ist.

Jetzt läuft alles wie am Schnürchen mit diesem wunderschönen Systembericht:



Name	Type	Vendor	Model
Intel 10 Series Chipset Family SATA Controller	SATA Controller	Intel	10260
Intel 10 Series Chipset Family USB and Thunderbolt Controller	USB Controller	Intel	10260
Intel 10 Series PCI Express Root Port	PCI Express Controller	Intel	10260
Intel 10 Series SATA Controller	SATA Controller	Intel	10260
Intel 10 Series SATA Controller	SATA Controller	Intel	10260
Intel 10 Series SATA Controller	SATA Controller	Intel	10260

Ich werde sicher kein DSDT-Meister, aber vom Prinzip her hatte ich meine DSDT doch richtig

bearbeitet, oder ?

Ich vermute, dass ich ein paar Schritte mehrfach gemacht und zum falschen Zeitpunkt gemacht habe.

Und dass ich nicht weiß was am Ende noch in der config.plist bleiben muss und was die DSDT erledigt.

Beitrag von „al6042“ vom 2. Januar 2018, 17:54

Das sah schon ganz gut aus, ich denke aber, dass der zweite Anlauf mit einem „Komplett“-Patch hier zusammen mit den nicht entfernten config.plist-Einträgen, zu den Schwierigkeiten führte.

was ich dich auch dringend bitten möchte, wäre die Aktualisierung deiner Signatur und ggf. des Profils, damit die neuen Eckdaten einsehbar sind.

zudem wäre der Hinweis auf die installierte OSX-Version dringend wichtig, da z.B. Er „nvda_drv=1“ ab Sierra nicht mehr unterstützt wird.

Beitrag von „farbweiß“ vom 2. Januar 2018, 18:32

jetzt geht es auch ohne nvda_drv und DataHubDxe. 🙄
meine Signatur und mein Profil sind aktualisiert.

nochmals tausend Dank.

markiere ich den Faden als gelöst ? macht ihr das hier so ? edit:(ja macht ihr, ist erledigt)

Beitrag von „farbweiß“ vom 3. Januar 2018, 18:36

Scheinbar läuft doch noch nicht alles so ganz rund.

Ich würde mich über Tips zum Beheben oder zum weiter Untersuchen freuen.

Die Probleme :

-er legt sich zwar schlafen, wacht aber nach kurzer Zeit wieder auf. Oder reagiert schon beim Schlafenlegen auf kleinste Bewegungen der Maus und wacht direkt auf.

Vielleicht liegt es an den nicht/falsch generierten P-States ? Der MSRdumper zeigt nur 2 an, trotz Generate-Pstates/generierter SSDT

-nach dem Schlafen oder Neustart muss ich manchmal USB Geräte ein und ausstecken (es gibt nur Maus und Tastatur im Moment) oder in einen anderen Port stecken.

Vielleicht brauche ich gar keine USB Port Limit fix ? Das Board hat nur insgesamt 13 USB Ports

Beitrag von „al6042“ vom 3. Januar 2018, 21:08

Zum Korrigieren der Schlafstörung durch Maus und Tastatur gibt es eine Methode per DSDT, die bisher noch nicht eingebaut ist.

Der [@kuckkuck](#) kennt da vielleicht noch eine feinere Variante, für die ich aber zu faul bin...

Nein, es liegt nicht an den P-States und Nein, dein Board hat mehr als 15, zum Teil nicht angebundene USB-Ports:

Zitat

Intel® H170 chipset :

6 x USB 2.0/1.1 port(s) (2 at back panel, black, Type-A, 4 at mid-board)

Intel® H170 chipset :

6 x USB 3.1 Gen 1 port(s) (2 at back panel, blue, Type-A, 4 at mid-board)

Intel® H170 chipset :

1 x USB 5Gb/s port(s) (1 at back panel, black, USB Type-CTM, Reversible, Support 3A power output)

Das sind zusammen:

- 6x 2.0 per USB-A ohne Doppel-Belegung für USB1.1,
- 6x 3.1 per USB-A mit Doppel-Belegung für USB2.0 (also 12x)
- 1x 3.1 per USB C

Ergibt $6+12+1 = 19$

Beitrag von „kuckkuck“ vom 3. Januar 2018, 22:20

[Zitat von farbeweiß](#)

er legt sich zwar schlafen, wacht aber nach kurzer Zeit wieder auf. Oder reagiert schon beim Schlafenlegen auf kleinste Bewegungen der Maus und wacht direkt auf.

Und das war früher nicht so?

[Zitat von al6042](#)

Der [kuckkuck](#) kennt da vielleicht noch eine feinere Variante, für die ich aber zu faul bin...

Direkte Variante gibts da nicht, man muss halt gezielt herausfinden was konkret das Problem verursacht und sich dann ein wenig mit dem Gerät auseinandersetzen, bis es läuft. Ansonsten einfach dem PRW Patch rein, dann kann der Hacky aber nicht mehr mit Maus/Tastatur/USB Geräten geweckt werden...

Beitrag von „farbeweiß“ vom 5. Januar 2018, 14:37

Vielen Dank für eure Hinweise.

Der erste Fehler meinerseits war, dass ich den USBInjectAll.kext von [@al6042](#) gar nicht im Clover hatte. Komischerweise hat das hinzufügen aber nichts verändert, vielleicht weil ich "Legacy USB" im Bios an hatte.

Scheinbar funktioniert Sleep/Wake Modus jetzt einwandfrei und die USB Ports nach dem Schlafen/aufwachen auch wieder.

Nochmal kurz zu den P-States (mit Generate P-States) erreiche ich im Gegensatz zur

generierten SSDT nun 4 States (vorher 2). Die SSDT.aml lege ich neben die DSDT in EFI/CLOVER/ACPI/patched und schalte generate p-states aus. Muss ich noch weitere Settings in Clover setzen damit er die SSDT nimmt ?

[Zitat von kuckkuck](#)

Und das war früher nicht so?

Es gibt nicht wirklich ein früher, das hier ist mein erster Build mit dieser Hardware und ich probiere es direkt mit DSDT.

[Zitat von kuckkuck](#)

Direkte Variante gibts da nicht, man muss halt gezielt herausfinden was konkret das Problem verursacht und sich dann ein wenig mit dem Gerät auseinandersetzen, bis es läuft.

Das hieße dann nach dem Aufwachen im kernel.log nachschauen was genau passiert ?

Falls noch weitere USB Probleme auftreten, wäre dann die Anleitung von Rehabman im Englischen Forum, über das man nicht spricht, der nächste Schritt ? Also mit IORegistryExplorer usw. ?

Beitrag von „kuckkuck“ vom 5. Januar 2018, 16:34

Ach DIE Anleitung von Rehabman? Stimmt, der hat ja nur gefühlt 2000 😄

Also wir kommen hier ganz gut ohne den Tomatentony aus und vermeiden aus verschiedenen Gründen gerne die Tools der Website. Anderes Thema. Aber was willst du mit Rehabmans Anleitung erreichen?

Naja USB bei Sleep scheint ja jetzt zu funktionieren, dann passt doch alles 😊
Oder läuft irgendwas noch nicht?

Beitrag von „farbweiß“ vom 5. Januar 2018, 17:20

ich glaube es läuft alles.

Ich wollte nur schonmal überlegen, was ich tun soll, wenn er sich heute Nacht schlafen legt
morgen wieder USB Ports vergisst.

Ich habe bei diesem Rechner hier zum ersten mal einen Hackintosh ohne die Tomaten-Tools zu
nutzen gemacht.

Ich muss sagen Neben den vielen andern Gründen die Tomatenseite nicht zu nutzen,
habe ich jetzt endlich mal das Gefühl zu verstehen was ich überhaupt tue (Clover etc.) und
dass ich Sachen gezielt ändern kann.

Beitrag von „kuckkuck“ vom 5. Januar 2018, 17:24

Das freut mich und genau das ist auch eine der Gründe warum wir von den Tools abraten.
Inzwischen wirst du dir bei möglichen Problemen viel besser selbst helfen können und hast
lange keine so große Abhängigkeit mehr weil ein "ZauberTool" dir "mit nur 3 Klick" OS X auf
den Rechner "gezaubert" hat... 😄

Beitrag von „andreas_55“ vom 5. Januar 2018, 17:40

Schritt 0: Tomatentools

Schritt 1: Hackintosh-Forum

| [Zitat von farbweiß](#)

habe ich jetzt endlich mal das Gefühl zu verstehen was ich überhaupt tue (Clover etc.)
und dass ich Sachen gezielt ändern kann.

Schritt 2: DSDT patchen, Sleep hin bekommen,...

Schritt 3: Bemerkten, dass z.B. nach dem Aufwachen vom Sleep zwar alles funktioniert, aber die kleine Sleep-Leuchtdiode immer noch leuchtet ... und dies fixen 😊

Schritt 4: Es wäre doch mal super cool, wenn man sich seinen eigenen Kext ...?? -->> (nach [@Brumbaer](#) suchen)

...

Schritt 5: Wie komme ich eigentlich an die Betas ran?

...

Viel Spaß im Forum 😄

Beitrag von „griven“ vom 14. Januar 2018, 20:46

Ist es nicht ein schönes Gefühl stück für stück zu begreifen was man da eigentlich tut und eins sei Dir versichert das Grundverständnis für Dinge wie Clover oder darüber welche Extensions wofür benötigt werden und welche vielleicht auch gar nicht ist erst der Anfang. Wie [@andreas 55](#) richtig schreibt die wirklich spannenden Dinge kommen erst noch man darf nur an der Stelle jetzt nicht aufgeben. Es ist zum Beispiel ganz erstaunlich was man mit ein wenig Verständnis über die DSDT Thematik alles anstellen kann 😄