

Gigabyte C621 Aorus Xtreme ALC1220VB Problem

Beitrag von „kingcyk“ vom 7. Januar 2020, 10:14

Hi there,

Recently I got a w3175x with Gigabyte C621 Aorus Xtreme. First I use clover and everything seems ok without the audio. I can't get the AppleHDA loaded with AppleALC while there shows no info about audio in Hackintool. Then I tried OC but nothing changed I think.

I do use a ssdt to rename CAVS to HDEF and inject some info but nothing happened. I hope someone can help?



Beitrag von „kingcyk“ vom 9. Januar 2020, 01:08

AppleALC debug below:

Code

1. 2020-01-07 17:19:47.664946+0800 localhost kernel[0]: (kernel) AppleALC: init @ (DBG) AppleALC bootstrap DBG-144-2019-12-02
2. 2020-01-07 17:20:08.666913+0800 localhost kernel[0]: (kernel) AppleALC: iokit @ (DBG) getOSData vendor-id has 8086 value
3. 2020-01-07 17:20:08.667048+0800 localhost kernel[0]: (kernel) AppleALC: iokit @ (DBG) getOSData device-id has A1F0 value
4. 2020-01-07 17:20:08.687534+0800 localhost kernel[0]: (kernel) AppleALC: iokit @ (DBG) getOSData revision-id has 9 value
5. 2020-01-07 17:20:08.687851+0800 localhost kernel[0]: (kernel) AppleALC: iokit @ (DBG) getOSData alc-layout-id has B value
6. 2020-01-07 17:20:08.688172+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) found 2 audio controllers
7. 2020-01-07 17:20:08.708227+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) validating 0 controller 1002:AB38:0
8. 2020-01-07 17:20:08.708546+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 0 mod 1002:AAC8
9. 2020-01-07 17:20:08.708865+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 1 mod 1002:AAE8
10. 2020-01-07 17:20:08.709179+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 2 mod 1002:AB08
11. 2020-01-07 17:20:08.709802+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 3 mod 1022:1457
12. 2020-01-07 17:20:08.710117+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 4 mod 1022:1487
13. 2020-01-07 17:20:08.997168+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 5 mod 1022:15E3
14. 2020-01-07 17:20:09.064317+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 6 mod 8086:C0C
15. 2020-01-07 17:20:09.109197+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 7 mod 8086:F04
16. 2020-01-07 17:20:09.154417+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 8 mod 8086:8CA0
17. 2020-01-07 17:20:09.176821+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 9 mod 8086:8D20
18. 2020-01-07 17:20:09.198502+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 10 mod 8086:8D21

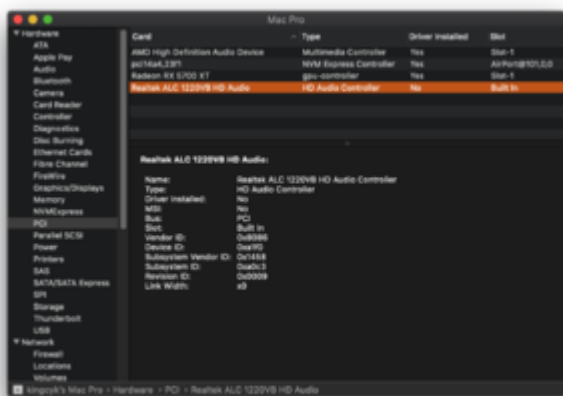
19. 2020-01-07 17:20:09.220368+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 11 mod 8086:9DC8
20. 2020-01-07 17:20:09.262642+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 12 mod 8086:9D71
21. 2020-01-07 17:20:09.283579+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 13 mod 8086:A171
22. 2020-01-07 17:20:09.305491+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 14 mod 8086:A2F0
23. 2020-01-07 17:20:09.326037+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 15 mod 8086:A348
24. 2020-01-07 17:20:09.346422+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 16 mod 10DE:EOF
25. 2020-01-07 17:20:09.366395+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 17 mod 10DE:FB0
26. 2020-01-07 17:20:09.386381+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 18 mod 10DE:FB8
27. 2020-01-07 17:20:09.405856+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 19 mod 10DE:FB9
28. 2020-01-07 17:20:09.424849+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 20 mod 10DE:FBA
29. 2020-01-07 17:20:09.443352+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 21 mod 10DE:FBB
30. 2020-01-07 17:20:09.461586+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 22 mod 10DE:FBC
31. 2020-01-07 17:20:09.479395+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 23 mod 10DE:10EF
32. 2020-01-07 17:20:09.496745+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 24 mod 10DE:10F0
33. 2020-01-07 17:20:09.513707+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 25 mod 10DE:10F1
34. 2020-01-07 17:20:09.530235+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) validating 1 controller 8086:A1F0:9
35. 2020-01-07 17:20:09.546757+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 0 mod 1002:AAC8
36. 2020-01-07 17:20:09.562798+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 1 mod 1002:AAE8
37. 2020-01-07 17:20:09.578417+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 2 mod 1002:AB08
38. 2020-01-07 17:20:09.593627+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 3 mod 1022:1457

39. 2020-01-07 17:20:09.608917+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 4 mod 1022:1487
40. 2020-01-07 17:20:09.624148+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 5 mod 1022:15E3
41. 2020-01-07 17:20:09.639023+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 6 mod 8086:C0C
42. 2020-01-07 17:20:09.653750+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 7 mod 8086:F04
43. 2020-01-07 17:20:09.682246+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 8 mod 8086:8CA0
44. 2020-01-07 17:20:09.696493+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 9 mod 8086:8D20
45. 2020-01-07 17:20:09.710687+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 10 mod 8086:8D21
46. 2020-01-07 17:20:09.724995+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 11 mod 8086:9DC8
47. 2020-01-07 17:20:09.739298+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 12 mod 8086:9D71
48. 2020-01-07 17:20:09.753650+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 13 mod 8086:A171
49. 2020-01-07 17:20:09.767942+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 14 mod 8086:A2F0
50. 2020-01-07 17:20:09.781930+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 15 mod 8086:A348
51. 2020-01-07 17:20:09.795572+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 16 mod 10DE:EOF
52. 2020-01-07 17:20:09.809150+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 17 mod 10DE:FB0
53. 2020-01-07 17:20:09.822733+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 18 mod 10DE:FB8
54. 2020-01-07 17:20:09.836324+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 19 mod 10DE:FB9
55. 2020-01-07 17:20:09.850146+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 20 mod 10DE:FBA
56. 2020-01-07 17:20:09.863902+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 21 mod 10DE:FBB
57. 2020-01-07 17:20:09.877628+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 22 mod 10DE:FBC
58. 2020-01-07 17:20:09.891328+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 23 mod 10DE:10EF

- 59. 2020-01-07 17:20:09.905063+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 24 mod 10DE:10F0
- 60. 2020-01-07 17:20:09.918773+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) comparing to 25 mod 10DE:10F1
- 61. 2020-01-07 17:20:09.932491+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) missing ControllerModInfo for 0 controller
- 62. 2020-01-07 17:20:09.946906+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) missing ControllerModInfo for 1 controller
- 63. 2020-01-07 17:20:09.962906+0800 localhost kernel[0]: (kernel) AppleALC: alc @ (DBG) AppleGFXHDA probe for HDAU

Alles anzeigen

And in PCI:



Beitrag von „DSM2“ vom 9. Januar 2020, 01:24

You will find the "problem" in the name of the Audio Chip itself.

ALC1220-VB2

It's unknown in AppleALC and that's the problem...

Beitrag von „kingcyk“ vom 9. Januar 2020, 01:35

Edit by DSM2 :

No full quotations please!

It is completely sufficient to mark a user with @usernamehere.

wow dsm it's you.

So you mean that maybe I can do the codec in Linux, add to resources in AppleALC and compile?

do you also have this problem? lol.

BTW, I had already do this.

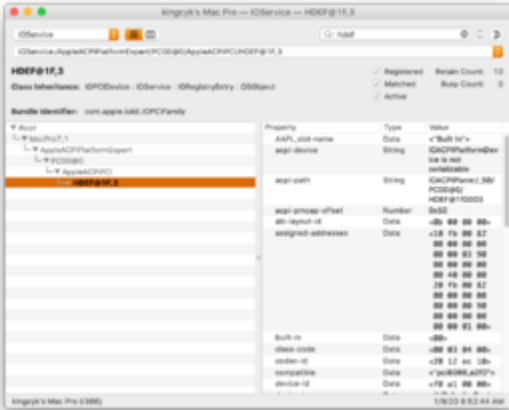
Codec in linux:

Code

1. Codec: Realtek ALC1220
2. Address: 0
3. AFG Function Id: 0x1 (unsol 1)
4. Vendor Id: 0x10ec1220
5. Subsystem Id: 0x1458a0c3
6. Revision Id: 0x100101

And I checked the Info.plist in AppleALC/ALC1220, I bet that's the same.

Maybe my problem is how to get AppleHDA loaded.



nothing below HDEF. PC00 and PCI0(rename) got the same.

Beitrag von „DSM2“ vom 9. Januar 2020, 09:18

To be honest with you: At the time I bought my motherboard, Linux not even recognized the codec used.

So I could not dump anything. I tried a lot at the time back then but I only could fix Audio onboard with VoodooHDA testwise in macOS and afterwards disabled it completely, because I never use Onboard Audio.

Since then I didnt spend any time to check it again because I use a professional Audio Interface from the recording studio for the sound output.

Yes it's the 1220 Chip but the nodes are different compared to the known ones.

You can try ID 16 but I don't think that this will work.

You have to do a full dump and not only the information which codec is used and integrate it in AppleALC.

I don't have time and any use from it, so I don't plan to spend time on this...

There are a lot more useful things I am working on at the moment for the community.

Beitrag von „kingcyk“ vom 9. Januar 2020, 16:38

OK, then I'll try to do a complete codec. And I would like to paste it here if anyone want to help.

Besides, can you share your oc efi? I got some problem in OC it freeze at AppleKeyStore: failed. I get it work first but not now, maybe problems in SSDT, I use clover at now but it has issue with the power, I can't get the power down when I click shutdown.

[codec#0.zip](#)

and ACPI tables:

[tables.zip](#)

Beitrag von „DSM2“ vom 9. Januar 2020, 17:20

No offense but the days of distributing my EFI's are over. Often they were used somewhere where they had nothing to do there.

In the worst case the user changed them and added some kind of garbage, which has no place there and therefore put me in a negative light or were even made money with my work by selling them.

That's why I don't do that anymore...

[MacPeet](#) can maybe help you with the codec.

You can send me your EFI via PN and I can have a look on your problem but I don't share anything... Specially not my configuration.

Beitrag von „MacPeet“ vom 9. Januar 2020, 18:52

[Zitat von DSM2](#)

You will find the "problem" in the name of the Audio Chip itself.

ALC1220-VB2

It's unknown in AppleALC and that's the problem...

Das VB2 hinten dran ist völlig irrelevant. Auch die Angabe der Revision ist unwichtig (wird oft gern in AppleALC gesetzt, aber es läuft auch ohne diese Angabe).

Sein Device ist ganz klar ein 10ec1220 laut seinem CodecDump.

Für 1220 gibt es bereits viele ID's (layout 1, 2, 3, 5, 7, 11, 13, 15, 16, 21, 27, 28, 29, 34)

Entweder stimmt generell was nicht in seiner Injektion der ID (alcID=xx, FixHDA im Clover, AppleHDA ist nicht real, etc.) oder es ist einer der ganz neuen Chipset's.

400 Series PCH HD Audio taucht erst im nächsten Release auf, ist noch nicht mal im aktuellen SourceCode der AppleALC drin.

Hierbei ist das Device im ioreg als HDEF1F,3 zu sehen. Bei allen normalen, gängigen Devices läuft HDEF immer auf HDEF@1B auf.

Es gibt bereits einige Devices, die als HDEF@1F mit AppleALC laufen, kommt aber halt auf den Chipsatz an.

Bei den meisten Usern gehe ich aber generell von eigenen Anwenderfehlern aus, was sich oft gezeigt hat.

Falscher Inject mit den neuen OSX-Versionen (alcid), bzw. fehlender FixHDA im Clover, keine real-AppleHDA in S/L/E durch zahlreiche Fehlversuche, Voodoo ist noch aktiv, manche Rechner brauchen HDAS to HDEF im Clover (ohne gepatchte DSDT), etc..

Ich habe darauf auch manchmal keinen Bock, wenn das Verständnis zum Umsetzen fehlt. Ich schaue mir seinen Codec-Dump demnächst mal an, aber ich glaube nicht, dass der so unterschiedlich ist zu den vielen ID's in der AppleALC.

Fazit: Wenn ich ALLWISSEND wäre, dann bräuchte ich nicht mehr arbeiten gehen, grins. Wäre toll, aber ist leider nicht so.

Beitrag von „apfelnico“ vom 9. Januar 2020, 19:05

Hier mal eine bereinigte DSDT.aml. Beim Öffnen mit MaciASL sind wieder Fehler drin. Einfach auf die entsprechenden Zeilen gehen und die "(Return (Zero))" rausnehmen. Oder die DSDT.dsl für weiteres Bearbeiten nutzen.

Beitrag von „DSM2“ vom 9. Januar 2020, 19:13

Viele IDs gibt es ja, funktionierten alle nur nicht und das lag nicht an einer Fehlkonfiguration, keine IRQ Geschichten oder sonstiges. Bei mir ist das ganze aber auch schon ewig her...

Selbst die damals aktuellste Linux Version konnte keinen dump machen. Ich kann aber auch eh nichts mit anfangen, im Bios deaktiviert und Tonstudio interface angeschlossen. [MacPeet](#)

Beitrag von „kingcyk“ vom 10. Januar 2020, 01:31

@[apfelnico](#) It seems a simply rename from CAVS to HDEF? I've tried in clover rename but can't get it work. Besides, applealc's wiki shows better not do rename I think.

@[MacPeet](#) to be honest I tried these id twice. all of them can't work. and I tried two ways suggested by applealc wiki 1. boot arg alcid=xx and 2. device property or ssdt layout-id like 0B000000. can't work.

Alright, I solved this by spoof the device to 8086:a170. weird.



I tried 8086:a2f0 first but nothing happed.

Beitrag von „DSM2“ vom 13. Januar 2020, 20:01

You should check your private messages [kingcyk](#)

I've attached something for you!