Displayauflösungen für Macmini 7,1 late 2014 Hackintosh fehlen

Beitrag von „Hecatomb“ vom 27. Mai 2022, 12:19

https://github.com/acidanthera.../Manual/FAQ.IntelHD.en.md

Fix the infinite loop on establishing Intel HDMI connections with a higher pixel clock rate on Skylake, Kaby Lake and Coffee Lake platforms

Add the `enable-hdmi-dividers-fix` property to `IGPU` or use the `-igfxhdmidivs` boot argument instead to 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.

Notes

- 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, see the LSPCON section below.*

LSPCON driver support to enable DisplayPort to HDMI 2.0 output on IGPU

Recent laptops (KBL/CFL) are typically equipped with a HDMI 2.0 port. This port could be either routed to IGPU or DGPU, and you can have a confirmation on Windows 10. Intel (U)HD Graphics, however, does not provide native HDMI 2.0 output, so in order to solve this issue OEMs add an additional hardware named LSPCON on the motherboard to convert DisplayPort into HDMI 2.0.

LSPCON works in either Level Shifter (LS) or Protocol Converter (PCON) mode. When the adapter works in LS mode, it is capable of producing HDMI 1.4 signals from DisplayPort, while in PCON mode, it could provide HDMI 2.0 output. Some onboard LSPCON adapters (e.g. the one on Dell XPS 15 9570) have been configured in the firmware to work in LS mode by default,

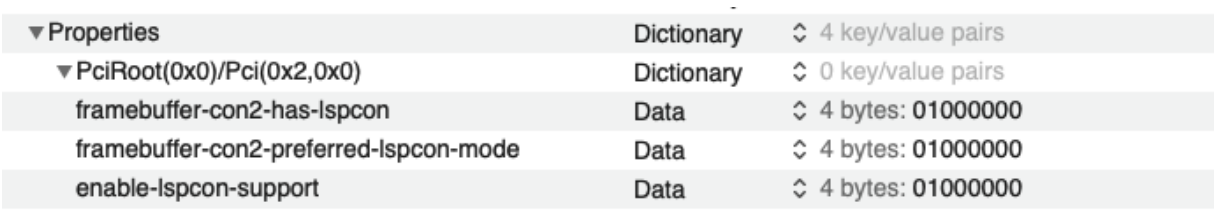
resulting a black screen on handling HDMI 2.0 connections.

Starting from version 1.3.0, *WEG* now provides driver support for the onboard LSPCON by automatically configuring the adapter to run in PCON mode on new HDMI connections, and hence solves the black screen issue on some platforms.

- LSPCON driver is only applicable for laptops and PCs with HDMI 2.0 routed to IGPU.
- LSPCON driver is necessary for all newer platforms unless the new IGPU starts to provide native HDMI 2.0 output.
- Supported Intel Platform: SKL, KBL, CFL and later. SKL: Intel NUC Skull Canyon; Iris Pro 580 + HDMI 2.0 with Parade PS175 LSPCON. CFL: Some laptops, e.g. Dell XPS 15 9570, are equipped with HDMI 2.0 and Parade PS175 LSPCON.
- If you have confirmed that your HDMI 2.0 is routed to IGPU and is working properly right now, you don't need to enable this driver, because your onboard LSPCON might already be configured in the firmware to work in PCON mode.

Instructions

- Add the `enable-lspcon-support` property to `IGPU` to enable the driver, or use the boot-arg `-igfxlspcon` instead.
- Next, you need to know the corresponding connector index (one of 0,1,2,3) of your HDMI port. You could find it under IGPU in IORegistryExplorer (i.e. AppleIntelFramebuffer@0/1/2/3). *If you only have a 2K/4K HDMI monitor, you might need temporarily to enable the* ***infinite loop fix*** *before connecting a HDMI monitor to your build, otherwise the system just hangs, so you won't be able to run the IORegistryExplorer and find the connector index.*
- Add the `framebuffer-conX-has-lspcon` property to `IGPU` to inform the driver which connector has an onboard LSPCON adapter. Replace `X` with the index you have found in the previous step. The value must be of type `Data` and should be one of `01000000` (True) and `00000000` (False).
- (*Optional*) Add the `framebuffer-conX-preferred-lspcon-mode` property to `IGPU` to specify a mode for your onboard LSPCON adapter. The value must be of type `Data` and should be one of `01000000` (PCON, DP to HDMI 2.0) and `00000000` (LS, DP to HDMI 1.4). Any other invalid values are treated as PCON mode. If this property is not specified, the driver assumes that PCON mode is preferred.



Spoiler: Debugging

Support all possible Core Display Clock (CDCLK) frequencies on ICL platforms

Add the `enable-cdclk-frequency-fix` property to `IGPU` or use the `-igfxcdc` boot argument instead to support all valid Core Display Clock (CDCLK) frequencies on ICL platforms, otherwise a kernel panic would happen due to an unsupported CD clock decimal frequency.

Core Display Clock (CDCLK) is one of the primary clocks used by the display engine to do its work. Apple's graphics driver expects that the firmware has already set the clock frequency to either 652.8 MHz or 648 MHz (the actual value depends on hardware), but quite a few laptops set it to a much lower value, such as 172.8 MHz, and hence you will see a kernel panic message like "Unsupported CD clock decimal frequency 0x158". This patch reprograms the clock to set its frequency to one of supported value, so that this precondition can be satisifed.

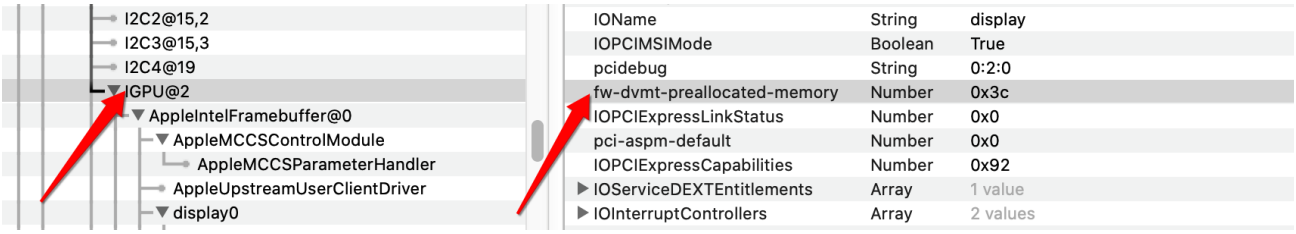
Spoiler: Debugging

Fix the kernel panic caused by an incorrectly calculated amount of DVMT pre-allocated memory on Intel ICL platforms

Add the `enable-dvmt-calc-fix` property to `IGPU` or use the `-igfxdvmt` boot argument instead to fix the calculation of the amount of DVMT pre-allocated memory on ICL platforms, otherwise a kernel panic saying `Unsupported ICL SKU` would happen.

Apple’s graphics driver reads the DVMT value set in the BIOS or the firmware and uses a “magic” formula to calculate the amount of memory in bytes. Unfortunately, the formula only works for a pre-allocated memory size that is a multiple of 32MB. Problem arises as laptops now have DVMT set to 60MB on ICL+ platforms by default, and the framebuffer controller ends up with initializing the stolen memory manager with an incorrect amount of pre-allocated memory. Even though one might be able to modify DVMT settings via `EFI shell` or `RU.EFI`, these methods are not applicable to some laptops, such as Surface Pro 7, that use custom firmware. As such, this patch calculates the correct number of bytes beforehand and patches the driver so that it will initialize the memory manager with proper values and aforementioned kernel panics can be avoided.

Apple has removed the kernel panic if the stolen memory is not enough, but you are encouraged to patch the framebuffer so that it fits into your available amount of stolen memory. Once the patch is enabled, you could find your actual amount of DVMT pre-allocated memory in the property `fw-dvmt-preallocated-memory` under the graphics device. (Only available in DEBUG version) The unit is megabyte, and the size in the example below is 60 MB. (0x3C = 60)



Spoiler: Debugging

Customize the behavior of the backlight smoother to improve your experience

Add the `enable-backlight-smoother` property to `IGPU` or use the `-igfxbls` boot argument instead to make brightness transitions smoother on Intel IVB+ platforms.

The graphics driver adjusts the panel brightness by writing values to related registers. Brightness Smoother (BLS) intercepts these write operations and gradually changes the register value. You may think of the graphics driver changing the brightness like climbing the stairs while BLS works like taking the escalator.

BLS uses a simple algorithm: it reads the register value `SRC` that represents the current brightness level and calculates the distance `D` to the register value `DST` requested by the graphics driver. It then moves toward the target value in `N` steps, each of which takes `T` milliseconds. By default, `N` is 35 and `T` is 7, but you may change their values by adding the properties `backlight-smoother-steps` and `backlight-smoother-interval`. It is recommended to keep `T` less than 10 milliseconds and the total amount of time `N * T` less than 350 milliseconds.

Besides, you may use the property `backlight-smoother-threshold` to ask BLS to skip the smoother process if the distance `D` falls below the threshold. In other words, BLS will write `DST` to the register directly. The default threshold value is 0.

If you want to prevent the built-in display from going black at the lowest brightness level, you may use the property `backlight-smoother-lowerbound` to specify the minimum register value that corresponds to the new, lowest brightness level. Similarly, `backlight-smoother-upperbound` can be used to specify the maximum value instead. See the example below. If these two properties are not present, BLS uses the default range [0, 2^32-1].

Example: Configure the smoother for a Haswell-based laptop with Intel HD Graphics 4600Example: Configure the smoother for a Coffee Lake-based laptop with Intel UHD Graphics 630