



OpenCore

Reference Manual (0.8.~~8~~.9)

[2023.01.07]

Note 1: This quirk shall not be used to workaround macOS limitation to address BARs over 1 GB. `ResizeAppleGpuBars` should be used instead.

Note 2: While this quirk can increase GPU PCI BAR sizes, this will not work on most firmware as is, because the quirk does not relocate BARs in memory, and they will likely overlap. ~~Contributions to improve this feature are welcome~~In most cases it is best to either update the firmware to the latest version or customise it with a specialised driver like ReBarUEFI.

13. `TscSyncTimeout`

Type: plist integer

Failsafe: 0

Description: Attempts to perform TSC synchronisation with a specified timeout.

The primary purpose of this quirk is to enable early bootstrap TSC synchronisation on some server and laptop models when running a debug XNU kernel. For the debug kernel the TSC needs to be kept in sync across the cores before any kext could kick in rendering all other solutions problematic. The timeout is specified in microseconds and depends on the amount of cores present on the platform, the recommended starting value is 500000.

This is an experimental quirk, which should only be used for the aforementioned problem. In all other cases, the quirk may render the operating system unstable and is not recommended. The recommended solution in the other cases is to install a kernel extension such as `VoodooTSCSync`, `TSCAdjustReset`, or `CpuTscSync` (a more specialised variant of `VoodooTSCSync` for newer laptops).

Note: This quirk cannot replace the kernel extension because it cannot operate in ACPI S3 (sleep wake) mode and because the UEFI firmware only provides very limited multicore support which prevents precise updates of the MSR registers.

14. `UnblockFsConnect`

Type: plist boolean

Failsafe: false

Description: Some types of firmware block partition handles by opening them in `By Driver` mode, resulting in an inability to install File System protocols.

Note: This quirk is useful in cases where unsuccessful drive detection results in an absence of boot entries.

11.19 ReservedMemory Properties

1. Address

Type: plist integer

Failsafe: 0

Description: Start address of the reserved memory region, which should be allocated as reserved effectively marking the memory of this type inaccessible to the operating system.

The addresses written here must be part of the memory map, have a `EfiConventionalMemory` type, and be page-aligned (4 KBs).

Note: Some types of firmware may not allocate memory areas used by S3 (sleep) and S4 (hibernation) code unless CSM is enabled causing wake failures. After comparing the memory maps with CSM disabled and enabled, these areas can be found in the lower memory and can be fixed up by doing the reservation. Refer to the `Sample.plist` file for details.

2. Comment

Type: plist string

Failsafe: Empty

Description: Arbitrary ASCII string used to provide human readable reference for the entry. Whether this value is used is implementation defined.

3. Size

Type: plist integer

Failsafe: 0

Description: Size of the reserved memory region, must be page-aligned (4 KBs).

4. Type

Type: plist string