

Dein PMCR Device Pfad zeigt woanders hin als dein CPU Pfad...

```
20 */
21 DefinitionBlock ( "", "SSDT", 1, "APPLE ", "freqdata", 0x00000001)
22 {
23     External ( _SB_, PR00, DeviceObj ) // (from opcode)
24
25     Method ( DTGP, 5, NotSerialized)
26     {
27         If ( LEqual ( Arg0, ToUUID ( "a8b5b7c6-1318-441c-b0c9-fe695eaf949b" )))
28         {
29             If ( LEqual ( Arg1, One) )
30             {
31                 If ( LEqual ( Arg2, Zero) )
32                 {
33                     Store ( Buffer ( One)
34                         {
35                             0x03
36                         }, Arg4)
37                     Return ( One)
38                 }
39                 If ( LEqual ( Arg2, One) )
40                 {
41                     Return ( One)
42                 }
43             }
44         }
45     }

```

Vermutlich auch der grund warum dein Bios zerreisst.

Erst recht wenn du die LauncherOption & LauncherPath falsch konfiguriert hast. Der versucht eben in die NVram (PMCR) zu schreiben.



4. **LauncherOption**
Type: plist string
Fallback: Disabled
Description: Register the **launcher** option in the firmware preferences for persistence.
Valid values:
• Disabled — do nothing.
• Full — create or update the top priority boot option in UEFI variable storage at bootloader startup.
— For this option to work, `RequestBootHolding` is required to be enabled.
• Short — create a short boot option instead of a complete one.
— This variant is useful for some older types of firmware, typically from Insyde, that are unable to manage full device paths.
• System — create no boot option but assume specified custom option is blessed.
— This variant is useful when relying on `ForceHostSignature` quick and OpenCore **launcher** path management happens through `bless` utilities without involving OpenCore.

This option allows integration with third-party operating system installation and upgrades (which may overwrite the `EFI\BOOT\BOOTx64.efi` file). The `BOOTx64.efi` file is no longer used for bootstrapping OpenCore if a custom option is created. The custom path used for bootstrapping can be specified by using the **launcherPath** option.

Note 1: Some types of firmware may have NVRAM implementation flaws, no boot option support, or other incompatibilities. While unlikely, the use of this option may result in boot failures and should only be used exclusively on boards known to be compatible. Refer to `acidanthers/logoswiker#1222` for some known issues affecting Haseel and other boards.

Note 2: While NVRAM resets executed from OpenCore would not typically erase the boot option created in `bootstrapping`, executing NVRAM resets prior to loading OpenCore will erase the boot option. Therefore, for significant implementation updates, such as was the case with OpenCore 0.6.4, an NVRAM reset should be executed with `bootstrapping` disabled, after which it can be re-enabled.

5. **LauncherPath**
Type: plist string
Fallback: Default
Description: Launch path for the **launcherOption** property.
Default: points to `OpenCore.efi`. User specified paths, e.g. `EFI\Users\launcher.efi`, can be used to provide custom loaders, which are supposed to load `OpenCore.efi` themselves.

Gruß Coban