

# OpenCore Bootloader

**Beitrag von „apfelnico“ vom 8. Mai 2020, 13:24**

muster48

Ich meine keine "spezielle" SSDT. Gibt es sicher. Ich meine, die Methode "GPRW", die in der DSDT vorhanden ist, umzubenennen. Somit ist die nach wie vor vorhanden, nur zielen Aufrufe via "\_PRW" Methode auf die GPRW nun ins leere. Eine \_PRW-Methode muss nicht zwangsläufig GPRW aufrufen wollen, es geht auch ohne. Aber wenn es der Fall ist, dann würde dieses Konstrukt mit der Umbenennung unterbrochen. Und nun kommt ein nächster Schnitt. Man schreibt eine "eigene" GPRW Methode in eine SSDT. Diese wird dann sofort als \_DIE\_GPRW von den ebnutzenden Routinen erkannt und genutzt. Nun kann man damit die Aufrufe wie man möchte manipulieren und das neue Ergebnis an die umbenannte, aber dennoch weiterhin aktive "EX-GPRW" in der DSDT schicken.

So funktioniert es in zwei Schritten: original GPRW in DSDT umbenennen, zum Beispiel XPRW oder GOGO oder NICO oder was auch immer (nicht vorher vorhanden ist), und eine SSDT in der GPRW definiert wird.

Für weitere Geräte können ja eigene \_PRW definiert werden.

Das Problem an der Sache ist ja, dass oft \_GPE (\_Lxx Level triggered GPE) für mehrere Geräte gelten, nicht individuell je Gerät.

Das kann man alles auseinander popeln, ich habe wie gesagt den Holzhammer genommen und die \_PRW direkt deaktiviert. Global alle. durch Umbenennung.

Weitere Idee:

Sicher kann man auch explizit nur die \_PRW von XHCI und ETH0 (oder wie auch immer konkret diese Geräte heißen (USB-PCH und Ethernet-PCH) umbenennen. Einfach die DSDT in einen Hex-Editor werfen und bei beiden Devices die \_PRW finden und bissel drumrum nehmen, so dass es individuell wird. Dann könnte man diesen gewonnenen String per Bootloader suchen lassen und ersetzen durch einen, in dem das "\_" durch "X" ersetzt wird.