

# Wann nach OS prüfen bei ACPI Patches?

**Beitrag von „Rexima“ vom 23. Oktober 2020, 16:35**

Ich möchte gerne Dual Boot mit Windows realisieren, jedoch ist das nicht möglich da irgendein Patch ein ACPI Bluescreen verursacht.

Wie findet man am besten raus, welche Patches dafür verantwortlich sind bzw. wann weiß man an welcher Stelle die Abfrage:

Code

1. If (\_OSI ("Darwin"))
2. {
3. Return (ABCDEF)
4. }
5. Else
6. {
7. Return (Zero)
8. }

angebracht ist? Muss das bei jedem Return sein?

---

**Beitrag von „5T33Z0“ vom 23. Oktober 2020, 16:40**

Habe das Problem auch. Wurde hier schon mal diskutiert und Beispiele gezeigt:

[Lenovo T530 ohne DSDT in OpenCore, aber wie?](#)

EDIT: Man muss nach `_DSM` suchen und darüber dann diese OSI Geschichte einfügen und am Ende noch ne mit geschweiften Klammern abschließen jeweils.

Habe es gerade mal bei meiner DSDT durchexerziert – Windows fährt jetzt hoch, aber macOS nicht mehr, doe! 😄

---

### Beitrag von „Rexima“ vom 23. Oktober 2020, 17:33

Ohje 😄

Ja ich hab das Problem, dass bei meinen Dateien nirgends `_DSM` vorhanden ist.

Ich hab auch mal bei allen Return's, die If Abfrage eingebaut, hat aber weiterhin nicht geholfen.

---

### Beitrag von „5T33Z0“ vom 23. Oktober 2020, 17:59

[Rexima](#) Das musst du auch in der DSDT machen, glaube ich. Dump die doch mal mit `SSDTTime` unter Windows. Guck mal in die Sample SSDTs die im Docs Ordner vom OpenCore Download liegen, da sieht man, wie's gemacht wird.

```
SSDT-ALSO.dsl
System DSDT
SSDT-ALSO.dsl

/*
 * Starting with macOS 10.15 Ambient Light Sensor presence is required for backlight functioning.
 * Here we create an Ambient Light Sensor ACPI Device, which can be used by SMCLightSensor kext
 * to report either dummy (when no device is present) or valid values through SMC interface.
 */
DefinitionBlock ("", "SSDT", 2, "ACDT", "ALS0", 0x00000000)
{
    Scope (_SB)
    {
        Device (ALS0)
        {
            Name (_HID, "ACPI0008" /* Ambient Light Sensor Device */) // _HID: Hardware ID
            Name (_CID, "smc-als") // _CID: Compatible ID
            Name (_ALI, 0x012C) // ALI: Ambient Light Illuminance
            Name (_ALR, Package (0x01) // _ALR: Ambient Light Response
            {
                Package (0x02)
                {
                    0x64,
                    0x012C
                }
            })
            Method (_STA, 0, NotSerialized) // _STA: Status
            {
                If (_OSI ("Darwin"))
                {
                    Return (0x0F)
                }
                Else
                {
                    Return (Zero)
                }
            }
        }
    }
}
}
```