

**Erledigt**

## **Anleitung : USB Fix El Capitan 10.11**

**Beitrag von „John Doe“ vom 11. Juli 2015, 13:14**

Nach einigen versuchen die USB-Anschlüsse in El Capitan zu beheben , erkannte ich, dass 10.11 hat einen neuen Mechanismus um die USB-Anschlüsse , an das SMBIOS zu binden. Wenn eure Intel USB-Anschlüsse gut funktionieren (ohne GenericUSBXHCI ), aber ab 10.11 nicht mehr , ist es wahrscheinlich die Ports nicht in SMBIOS definiert sind.

Schritt 1 : Laden der Kexte

Die neue USB Controllerkext in 10.11 ist AppleUSBEHCIPCI.kext und AppleUSBXHCIPCI.kext in der IOUSBHostFamily.kext. Ihr könnt die Ports von eurem aktuellem SMBIOS überprüfen durch öffnen dieser Dateien :

USB 2.0

`/System/Library/Extensions/IOUSBHostFamily.kext/Contents/PlugIns/AppleUSBEHCIPCI.kext/Contents/Info.plist`

USB 2.0 und 3.0

`/System/Library/Extensions/IOUSBHostFamily.kext/Contents/PlugIns/AppleUSBXHCIPCI.kext/Contents/Info.plist`

Das System wird nun standardmäßig AppleUSBEHCIPCI kext zum Gerätenamen EHCI 1 und EHCI 2 hinzufügen. Das lässt sich in der info.plist überprüfen.

▼ iMac13,1-EHC1	Dictionary	↕ 6 Schlüssel/Wert-Paare
CFBundleIdentifier	String	↕ com.apple.driver.AppleUSBMergeNub
IOClass	String	↕ AppleUSBMergeNub
IONameMatch	String	↕ EHC1
IOProviderClass	String	↕ AppleUSBHCIPCI
▼ IOProviderMergeProperties	Dictionary	↕ 3 Schlüssel/Wert-Paare
kUSBmuxEnabled	Boolean	↕ YES
port-count	Daten	↕ 4 Bytes: 01000000
▼ ports	Dictionary	↕ 1 Schlüssel/Wert-Paare
▼ PRT1	Dictionary	↕ 2 Schlüssel/Wert-Paare
UsbConnector	Zahl	↕ 255
port	Daten	↕ 4 Bytes: 01000000
model	String	↕ iMac13,1

Versichert euch das euer USB 2.0 Gerätenamen schon in der kext ist. Man kann IONameMatch ändern so dass es zum USB 2.0 Namen in der DSDT passt oder man kann die USB 2.0 Geräte Namen in der DSDT zu EHCI 1/EHCI 2 (abhängig vom SMBIOS in der info.plist).

Bei USB 3.0 ist es anders. Anstatt des Namen (IOMatch) , AppleUSBXHCIPCI wird ID benutzen um die kext mit USB 3 Geräten in der DSDT zu verknüpfen.

Property-List	Typ	Wert
▼ AppleUSBXHCILPTH1	Dictionary	↕ 6 Schlüssel/Wert-Paare
CFBundleIdentifier	String	↕ com.apple.driver.usb.AppleUSBXHCIPCI
IOClass	String	↕ AppleUSBXHCILPTH
IOPCIPrimaryMatch	String	↕ 0x8c318086
IOPCITunnelCompatible	Boolean	↕ YES
IOProbeScore	Zahl	↕ 1.000
IOProviderClass	String	↕ IOPCIDevice

Selbst wenn euer USB 3.0 Gerät in XHC1 oder XHC umbenannt ist die kext wird trotzdem geladen wenn die Device ID übereinstimmt.

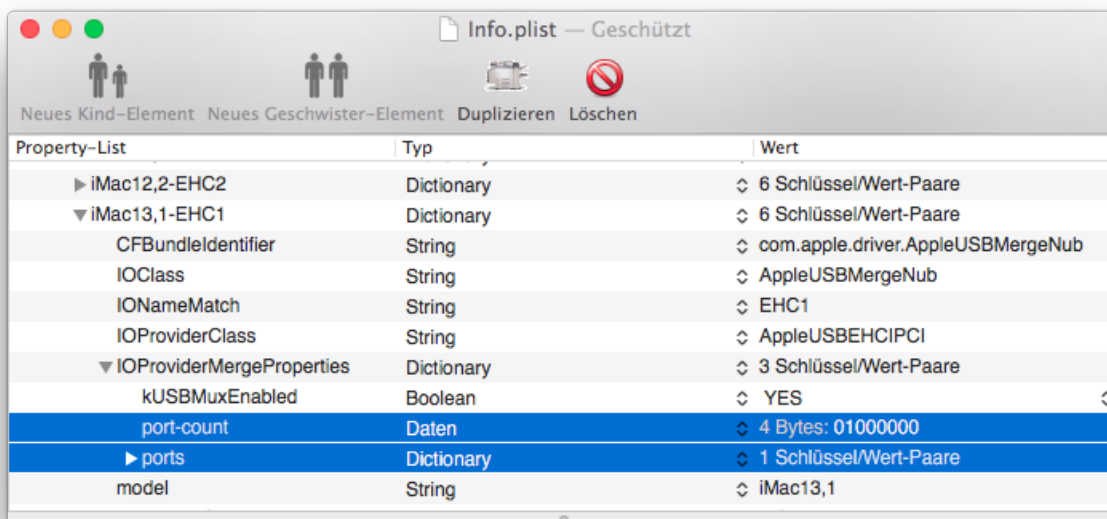
## 2. Port Beschränkung

Selbst wenn die USB kexte geladen sind bedeutet das immer noch nicht das alle USB Port funktionieren , weil Apple eine neue Methode implementiert hat welche Ports für welche Ports für welches SMBIOS zur Verfügung stehen.

Beachtet, dass wenn euer USB 3.0 Gerät nicht den Namen XHC1 hat , ist standardmäßig keine Beschränkung. Wenn nicht alle Ports richtig funktionieren sollte man Portbeschränkung

aktivieren und die Ports manuell hinzufügen.

Zum Beispiel im SMBIOS vom iMac 13,1



Property-List	Typ	Wert
▶ iMac12,2-EHC2	Dictionary	↕ 6 Schlüssel/Wert-Paare
▼ iMac13,1-EHC1	Dictionary	↕ 6 Schlüssel/Wert-Paare
CFBundleIdentifier	String	↕ com.apple.driver.AppleUSBMergeNub
IOClass	String	↕ AppleUSBMergeNub
IONameMatch	String	↕ EHC1
IOProviderClass	String	↕ AppleUSBHCIPCI
▼ IOProviderMergeProperties	Dictionary	↕ 3 Schlüssel/Wert-Paare
kUSBMuxEnabled	Boolean	↕ YES
port-count	Daten	↕ 4 Bytes: 01000000
▶ ports	Dictionary	↕ 1 Schlüssel/Wert-Paare
model	String	↕ iMac13,1

Standardmäßig ist für EHC1 ein Port verfügbar dieser hat die Adresse 1. Beachtet das der Portname nicht wichtig ist . Wenn euer System mehr Ports und Adressen hat werden diese deaktiviert.

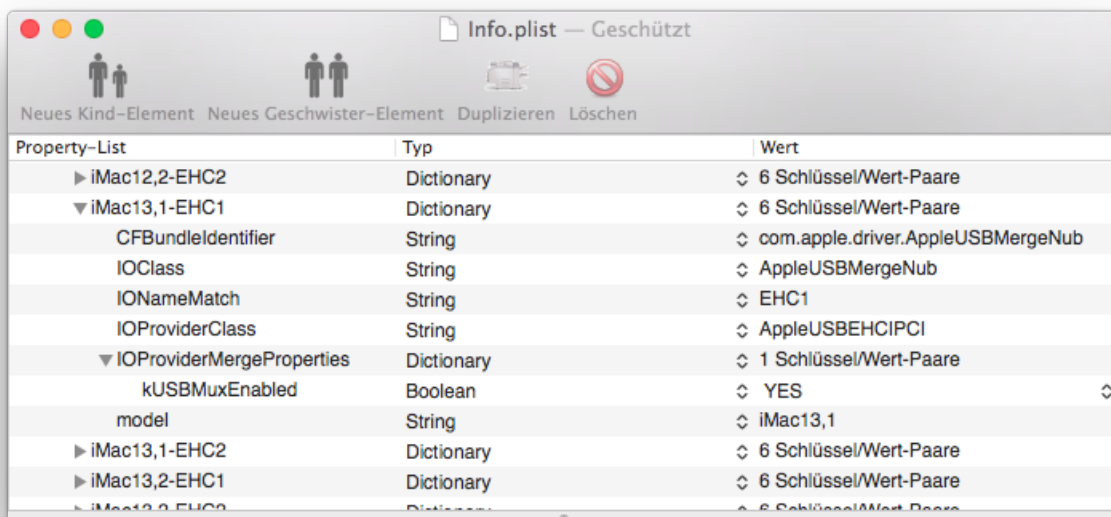
Das gleiche gild auch für XHC1, MacBook Pro 9,2 hat nur die Portadressen 1,2,5,6 zur Verfügung.

Property-List	Typ	Wert
▼ MacBookPro9,2-XHC1	Dictionary	↕ 6 Schlüssel/Wert-Paare
CFBundleIdentifier	String	↕ com.apple.driver.AppleUSBMergeNub
IOClass	String	↕ AppleUSBMergeNub
IONameMatch	String	↕ XHC1
IOProviderClass	String	↕ AppleUSBXHCIPCI
▼ IOProviderMergeProperties	Dictionary	↕ 3 Schlüssel/Wert-Paare
kUSBMuxEnabled	Boolean	↕ YES
port-count	Daten	↕ 4 Bytes: 06000000
▼ ports	Dictionary	↕ 4 Schlüssel/Wert-Paare
▼ PRT1	Dictionary	↕ 2 Schlüssel/Wert-Paare
UsbConnector	Zahl	↕ 3
port	Daten	↕ 4 Bytes: 01000000
▼ PRT2	Dictionary	↕ 2 Schlüssel/Wert-Paare
UsbConnector	Zahl	↕ 3
port	Daten	↕ 4 Bytes: 02000000
▼ PRT5	Dictionary	↕ 2 Schlüssel/Wert-Paare
UsbConnector	Zahl	↕ 3
port	Daten	↕ 4 Bytes: 05000000
▼ PRT6	Dictionary	↕ 2 Schlüssel/Wert-Paare
UsbConnector	Zahl	↕ 3
port	Daten	↕ 4 Bytes: 06000000

Das lässt sich entweder durch entfernen von Portbeschränkung oder hinzufügen beheben.

## 2a. Entfernen von Portbeschränkung

Das ist der einfachste weg : ihr müsst einfach port counts und ports aus der info.plist entfernen das system nimmt sie dann automatisch aus der DSDT.



->

