

Erledigt

"Du hast ja Alles" - hmmm vielleicht, wenn ich einen Laptop habe.

Beitrag von „Brumbaer“ vom 3. Februar 2018, 13:22

Dumdidum

Dumdidum, Dumdidum, Dumdidum, Dumdidum.

Oberwasser, ich habe Oberwasser, denn nun bootet der Miix endlich, wie er soll, wann er soll, Dumdidum.

Ja, ja, ja, jetzt wird wieder in die Hände gespuckt!

Und als nächstes, werden all die Kexte, Patches und Häkchen, die ich in meiner Verzweiflung hinzugefügt hatte, wieder entfernt, Dumdidum.

Hinterher, sehen Config.plist, Other und drivers64UEFI Ordner ganz normal aus und - das System bootet immer noch. Bevor ich es vergesse: Dumdidum.

The sound of music

So jetzt was Einfaches, Sound. Der Miix hat einen ALC298. Das weiß ich von Blick durchs Fenster, aber ich hätte es auch im IORegistryExplorer feststellen können.

Ich bin ein großer AppleALC Fan und er hat mich bisher noch nicht im Stich gelassen. AppleALC braucht Lilu, aber das habe ich schon im Other Ordner.

Nach AppleALC gegoogelt und da wird ein Link zu den "Supported" Codecs angeboten und - Dumdidum - die Layouts 3, 11, 13, 28, 29, 47 und 72 werden angeboten. Wo fängt man am besten an ? Wie wäre es mit dem Anfang. Ruckzucki, Layout Id 3 in der Config gesetzt, Neustarti und Booti - Dumdidum - Systemsteuerung, Ton, sieht gut aus, mal ein Böng und mal ein Bing und mal ein Tschingderassabum.

Plaudertasche

Strg-Leerschritt und nichts. Immer noch Dumdidum, aber eine Oktave tiefer, etwas knurriger, ich bin scheinbar leicht zu verstimmen. Siri angeklickt, "Öffne iTunes", Siri öffnet iTunes. Dumdidum, wieder zurück in normaler Tonlage.

Siri, kann die Systemeinstellungen öffnen, aber nicht ihr eigenes Kontrollfeld, das BT Kontrollfeld hingegen schon. Egal. Die Siri-Öffnen-Taste auf Strg Leerschritt gelegt und - geht doch.

Es kommen Töne raus, es gehen Töne rein, Layout Id 3 muss es sein.

Trari-Trara, die Post ist da

Zufälle gibt's, da bringt am selben der Tag der Postbote doch tatsächlich die DW1560. Es ist dem Postboten gegenüber ungerecht von zufällig zu reden, der ist sehr zuverlässig.

"6 Schrauben, 6 Schrauben, die gehen leicht auf, der Rechner, der Rechner, der öffnet seinen Bauch". Auch nicht besser als Dumdidum, aber noch habe ich Oberwasser.

Ok, Original raus, DW1560 rein. Antennen angeschlossen, Bildschirm hochgehalten, Rechner eingeschaltet und es kommt ein Logo, eigentlich ein Grund zur Freude, wenn es nicht so hässlich wäre, aber nicht hässlich genug um mir die Laune zu verderben. Dumdidum.

Neustart, weder WiFi noch Bluetooth. Keine Überraschung, ist ja kein Standard Mac Gerät. Ich habe mit Absicht eine Lenovo Variante genommen, für den Fall, dass es eine Whitelist gibt.

IORegistry Editor zeigt aber, dass die Teile erkannt werden. Falls man keine Mac kompatible Karte hat, ist der Standardweg für WiFi die Verwendung von FakePCIID und FakePCIID_Broadcom_WIFI und für Bluetooth die Verwendung von BrcmPatchRAM2 und BrcmFirmwareData.

Reinkopiert und siehe da WiFi WiFiet, aber BT BTed nicht.

Laut IORegistryExplorer handelt es sich um einen BCM20702A mit der VendorId 0x489 und der ProductId 0xe07a. Das sind USB Ids, äquivalent zur PCIId.

Die DeviceClass ist 0xFF, das ist seltsam normalerweise ist sie bei BT Geräten 0xE0.

"Kein Problem", denke ich noch, "trägst den Wert in BrcmPatchRAM2 nach". Gesagt, getan und was auch immer ich mache, es geht nicht. BrcmPatchRAM2 und BrcmFirmwareData wieder raus.

Da gibt's doch ne App für, oder en Kext ?

Apples BT Treiber befinden sich als PlugIns im IOBluetoothFamily.kext. PlugIns sind Kexte, die im PlugIns Ordner anderer Kexte abgelegt sind.

- BroadcomBluetoothHostControllerUSBTransport.kext
- BroadcomBluetooth20703USBTransport.kext
- CSRBluetoothHostControllerUSBTransport.kext
- CSRHIDTransitionDriver.kext
- IOBluetoothHostControllerTransport.kext
- IOBluetoothHostControllerUARTTransport.kext
- IOBluetoothHostControllerUSBTransport.kext
- IOBluetoothSerialManager.kext
- IOBluetoothUSBDFU.kext

Die CSR können aus der Liste raus (nicht aus dem Ordner), falscher Hersteller, da waren es nur noch 7.

UART und Serial Manager haben mit speziellen Funktionen zu tun, können auch raus, da waren es nur noch 5.

IOBluetoothHostControllerTransport scheint nichts mit USB zu tun zu haben, also erst mal raus,

da waren es nur noch 4.

- BroadcomBluetooth20703USBTransport.kext
- BroadcomBluetoothHostControllerUSBTransport.kext
- IOBluetoothUSBDFU.kext
- IOBluetoothHostControllerUSBTransport.kext

Mit plist und Tücke

Kexte, die als Gerätetreiber dienen, haben in ihrer Info.plist meist IOKitPersonalities Einträge, in denen die Geräte identifiziert werden, die sie treiben sollen.

Fangen wir mit der Info.plist des BroadcomBluetoothHostControllerUSBTransport.kext an. Tausende Einträge, alle rufen dieselbe Klasse auf und unterscheiden sich durch idProduct und idVendor.

Die Controller basieren alle auf den 2045 und 2046 Serien von Broadcom also laut Google nur BT2.0 und BT 2.1. "Ihrrrr seieieieied rrrraus".

Das nächste Broadcom Kext ist BroadcomBluetooth20703USBTransport.

Die Einträge unterscheiden sich wieder nur in idProduct und idVendor. Der folgende Eintrag ist typisch:



Key	Type	Value
CFBundleIdentifier	String	com.apple.iokit.BroadcomBluetooth20703USBTransport
IOClass	String	BroadcomBluetooth20703USBTransport
IOProviderClass	String	IOUSBHostDevice
LMPLoggingEnabled	Boolean	YES
idProduct	Number	33425
idVendor	Number	1452

Alle verwenden eine Klasse, die BroadcomBluetooth20703USBTransport heißt. Dank Google wissen wir, dass der Chipsatz 20703 BT4.1 unterstützt, der 20702 aber nur BT4.0. Aufeinanderfolgende Teilenummer beim selben Hersteller, da kann man den Treiber schon mal probieren, kann funktionieren, muss aber nicht.

Also lassen wir für das BT Modul diesen Treiber laden. Sowa haben wir schon mal für den USB Treiber gemacht. Der Eintrag wird kopiert und in das USB Kext im Other Ordner eingesetzt. Es muss nicht das USB Kext sein, man kann auch ein neues Kext machen, oder irgendein anderes nehmen, aber ich habe das Kext genau für solche Anpassungen eingeplant also hinein. idProduct und idVendor müssen noch angepasst werden.

▼ IOCPersonalities	Dictionary	(3 items)
▼ BT	Dictionary	(7 items)
CFBundleIdentifier	String	com.apple.iokit.BroadcomBluetooth20703USBTransport
IOClass	String	BroadcomBluetooth20703USBTransport
IOProviderClass	String	IOUSBHostDevice
LMPLoggingEnabled	Boolean	NO
idProduct	Number	57466
idVendor	Number	1161
IOProbeScore	Number	2000
► Air7,1	Dictionary	(7 items)

BrcmPatchRAM2 und BrcmFirmwareData werden nicht gebraucht. Neustart und WiFi und BT. Lauft, lauft bei mir.

PS.

Dumdidum