

Erledigt

USB2 Devices -> EHCI Routing zu AppleUSBECI

Beitrag von „JoeHidden“ vom 26. Januar 2017, 14:54

MacPro 6,1

Ich hab gestern Abend früh Schluss gemacht und teste heute Abend weiter.

@Kuckuck Codeword "Der ESEL ist offline" 😊 Leider auch keine signifikanten Änderungen erkennbar.

In der Info.plist der FakePCIID_XHCIMux finden sich die RM,pr- Einträge.

▼ XHCIMux 8 and 9-series	Dictionary	↕ 7 Schlüssel/Wert-Paare
CFBundleIdentifier	String	↕ org.rehabman.driver.FakePCIID.XHCIMux
▼ FakeProperties	Dictionary	↕ 6 Schlüssel/Wert-Paare
RM,pr2-block	Daten	↕ 1 Bytes: 00
RM,pr2-chipset-mask	Daten	↕ 4 Bytes: FF3F0000
RM,pr2-force	Daten	↕ 4 Bytes: 00000000
RM,pr2-honor-pr2m	Daten	↕ 1 Bytes: 01
RM,pr2-init	Daten	↕ 1 Bytes: 01
RM,pr2m-block	Daten	↕ 1 Bytes: 01
IOClass	String	↕ FakePCIID_XHCIMux
IOMatchCategory	String	↕ FakePCIID_XHCIMux
IOPCIPrimaryMatch	String	↕ 0x9c318086 0x9cb18086 0x8c318086 0x8cb18086 0x8d318086
IOProbeScore	Zahl	↕ 9.001
IOProviderClass	String	↕ IOPCIDevice

Wirklich viel habe ich dazu nicht gefunden aber Rehabman schreibt dazu folgendes:

Zitat

Configuration properties and their defaults: RM,pr2-force <00 00 00 00>. By default forces all XHCI ports to route USB2 devices to EHCI. RM,pr2-init <01>. Will write RM,pr2-force value at startup if non-zero. RM,pr2-block <01>. Will block writes to XUSB2PR if non-zero. RM,pr2m-block <01>. No evidence that OS X drivers attempt to

write XUSB2PRM (offset 0xD4), but since this kext relies on a valid value here (as provided by the BIOS), writes to it are blocked if non-zero. RM,pr2-honor-pr2m <01>: Changes to XUSB2PR will be masked by XUSB2PRM if this is non-zero. RM,pr2-chipset-mask: Writes to XUSB2PR are masked by this value. This is defined by the chipset documentation. Default value depends on chipset. Refer to Intel 7/8/9-series chipset data sheet for more info.

So wie ich das verstehe wäre es doch damit möglich per RM,pr2-force nur die reinen USB 2.0 Ports an den EHCI zu leiten, oder? Das würde ja schon ausreichen. Ein USB 2.0 Device am USB 3.0 (Dongles etc.) dürfte gern auch XHC bearbeitet werden. Bisher erschließt sich mir die Maskierung aber überhaupt nicht - bzw. die Nummerierung auf deren Basis dann maskiert wird. Wirst Du da schlau draus?

--- Themensprung / Warnung 😊 ---

Auf die Gefahr Dir jetzt vollkommen auf die Nerven zu gehen... aber ich möchte das wirklich verstehen. Ich habe mir die DSDT nochmal sehr genau angesehen. Speziell das Thema korrekte _UPC Objekte. Beim XHC stimmen die Angaben von DSDT und IOReg überein, also Portbezeichnungen / Anzahl / Adressen. Bei EH01 und EH02 nicht. Bei EH01 (Adresse 8d26) fehlt der HUB und es fehlen alle Ports, selbst mit USBInjectAll ohne Filter. Bei EH02 werden die sechs Ports korrekt übernommen, aber die Nummern der Ports werden um zehn inkrementiert? Es taucht dem entsprechend auch nur ein USB2.0 Bus im USB Gerätebaum auf. Ich finde das nach wie vor sehr seltsam.

