

Kext as Kext can oder USB 3.0 ohne USBInjectAll

Beitrag von „Brumbaer“ vom 25. August 2017, 21:51

1) Ja

2) Wenn USB 3.0 läuft steckst du einen USB 3.0 Gerät an die Buchse und schaust an welchem SS Port das Gerät erscheint. Dann machst du es ab und steckst ein USB 2.0 Gerät an und schaust an welchem HS Port das Gerät erscheint.

3) Siehe doch mal zu dass du den richtigen Treiber installierst. Dann stecke ein USB 2.0 Gerät nacheinander in die Buchsen die du zu verwenden gedenkst. Schau bei welchem HS Port das Gerät erscheint. Dann hast du eine Liste aller verwendeten HS Ports. Dann machst du das gleiche mit einem USB 3.0 Gerät und schreibst dir die SS Ports auf.

4) Das Feld port-count enthält keinen port-count. Das Feld ist verwirrend benannt. Port-count enthält die höchste Port-Id. Die entspricht häufig der Anzahl Ports, aber nicht immer.

Der Gedanke hinter port-count ist wohl, dass man gerne ein Array mit einem Eintrag für jeden Port hätte. Damit man schnell auf den Eintrag für ein Port zugreifen kann verwendet man die Port-Id als Index für das Array. Das Array, muss dann nicht so viele Einträge haben, wie man Ports benutzt, sondern wie hoch die Port-Id sein kann. Port-Count ist also nicht die Anzahl der Ports, sondern die Anzahl der Einträge in dem Feld für Ports, und die entspricht der höchsten Port Id. Da der verwendete Controller als höchste Port-Id 0x1A hat, ist man mit der 0x1A auf der sicheren Seite, aber "korrekter" wäre in deinem Beispiel natürlich 0x15 (falls SSP5) das höchste verwendete Port ist.

5) X299 und Z270 verwenden den gleichen XHCI Controller, deshalb funktioniert die Treiber Anpassung für mein System auch für dein System. Wie in meinem zweiten Post steht findet man die Id in lesbarer Form im "name" Feld des ersten XHCI Eintrages in IORegistryExplorer. da steht pci8086,a2af. Das entspricht dem IOPrimaryMatch 0xa2af8086.

Was an deinem Kext schiefgeht, kann ich nicht sagen ohne es zu sehen.