



Contribution ID: 48

Type: **not specified**

## Powering up "discoverable bus-attached" devices on DT-based platforms

*Monday, November 13, 2023 4:30 PM (45 minutes)*

There is a long standing issue with devices that need any kind of powering up in order for the bus to be able to enumerate them. A device connected to such a bus will only be discovered/enumerated if resources like regulators, gpios and so on, are properly configured beforehand. This is one of the main reasons a major number of DT-based platforms are marking resources like regulators as "always-on", as a hack rather than a solution. The devicetree cannot describe devices connected to buses like PCI, USB and so on, therefore the resources needed to power up a device connected to such buses cannot be tied to any DT node. Tying them to the bus itself is not accurate from the HW point of view as they are only needed if one specific device is ever connected on that bus. There are quite a few examples of this scenario that involve PCIe, USB and MMC devices. MMC has implemented its own solution (pwrseq) a long time ago. For cases like USB onboard hub, there is a provided solution upstream recently, which adds a dedicated devicetree node for the onboard hub for which a platform driver probes and registers the hub which is later sysfs linked by the usb device driver on probe. This approach however does not work for every other device that is connected to a discoverable bus, since there are devices that provide multiple functionalities, like Qualcomm QCA Wifi and Bluetooth chips, which usually use two different buses (PCIe and UART/USB), is powered by an onboard PMIC and use a GPIO for SW reset that is shared between Wifi and Bluetooth. In such a case, in order for the Wifi driver to probe, the device needs to be enumerated first over PCIe, but for that to happen, some configuration needs to be done with respect to regulators (at least). Therefore which driver should handle those resources in such a way that the device itself gets a chance to probe? What subsystem should such a "power sequencer" be part of? How should such a device be represented in devicetree (if possible)? How could a driver for a device from a discoverable bus could get its hands on those resources before probing? Can we have a generic way to solve this problem? This presentation is trying to bring together all these questions and all scenarios, the historically proposed solutions and possibly a proof-of-concept.

<https://lore.kernel.org/all/20211006035407.1147909-1-dmitry.baryshkov@linaro.org/>

<https://lwn.net/Articles/602855/>

<https://www.uwsg.indiana.edu/hypermail/linux/kernel/1406.2/03144.html>

<https://lore.kernel.org/all/20230110172954.v2.1.I75494ebee7027a50235ce4b1e930fa73a578fbe2@changeid/>

**Primary author:** VESA, Abel (Linaro)

**Presenter:** VESA, Abel (Linaro)

**Session Classification:** LPC Refereed Track

**Track Classification:** LPC Refereed Track