fw_devlink

LPC 2021: Remaining issues to resolve and future improvements

Saravana Kannan

android

fw_devlink: upstream status

Status:

- fw_devlink=on by default
- Enforces probe and suspend/resume ordering
- Supports 23 common DT properties/bindings

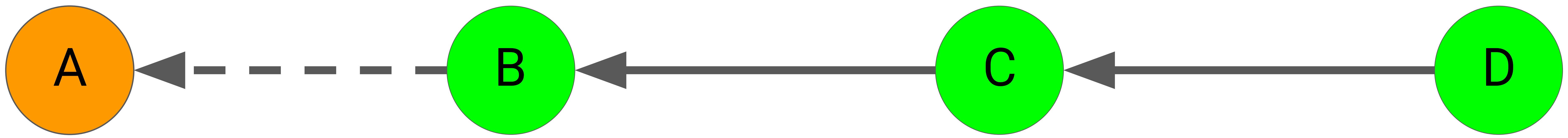
55+ patches merged upstream since LPC 2020

- Automatically handles cyclic dependencies
- Correctly handles early devices
- Better handling of missing drivers
- Support for optional dependencies
- Added support for new properties
- IRQ drivers can now be platform drivers and modules!

android

Deferred probe timeout + fw_devlink=on improvements

Dependency graph



Legend:

- 🟠 Device without driver registered.
- 🟢 Device with driver registered before deferred probe timeout.
- ← Consumer is pointing to mandatory supplier
- ⇠ Consumer is pointing to optional supplier

Prevents unnecessary probe failures after deferred probe time out by relaxing (doesn't block probe) only device links to suppliers without drivers.

fw_devlink=off/permissive:

- B will probe successfully. But,
- C and D could fail to probe if they are attempted before B.

fw_devlink=on

- Probe of C and D will never be attempted before B.
- They'll always probe successfully after B probes without A.

fw_devlink DT support

Currently supports the following 23 devicetree bindings:

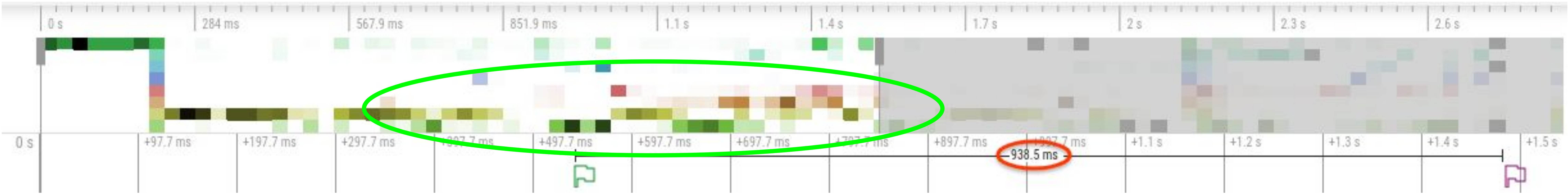
- clocks
- interconnects
- iommus
- iommu-map
- mboxes
- io-channel
- interrupt-parent
- dmas
- power-domains
- hwlocks
- extcon
- interrupts-extended
- nvmem-cells
- phys
- wakeup-parent
- pinctrl-*
- -supply (regulators)
- -gpio and -gpios (gpios)
- remote-endpoint
- pwms
- resets
- leds
- backlight

Discussion slides

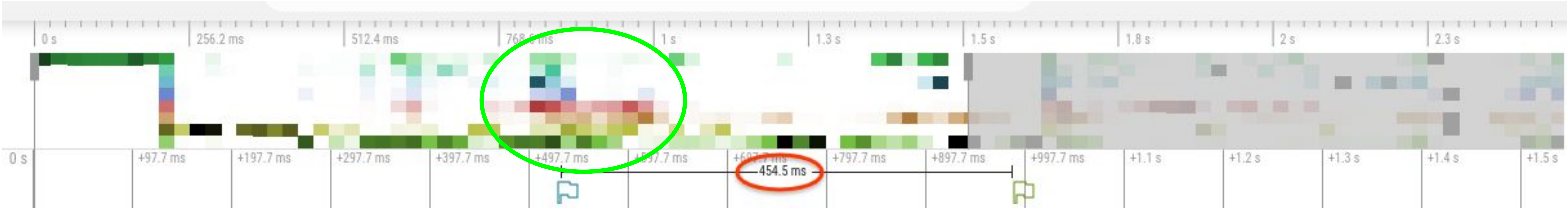
android

Should we enable fully asynchronous probing?

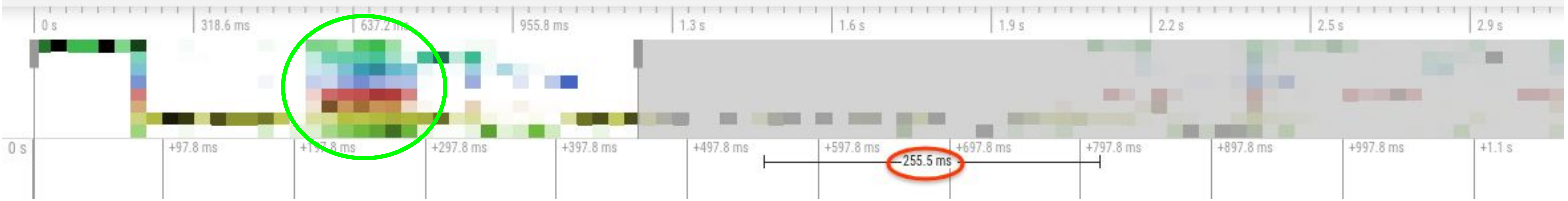
Synchronous probing (~900ms)



Asynchronous probing (~450ms)



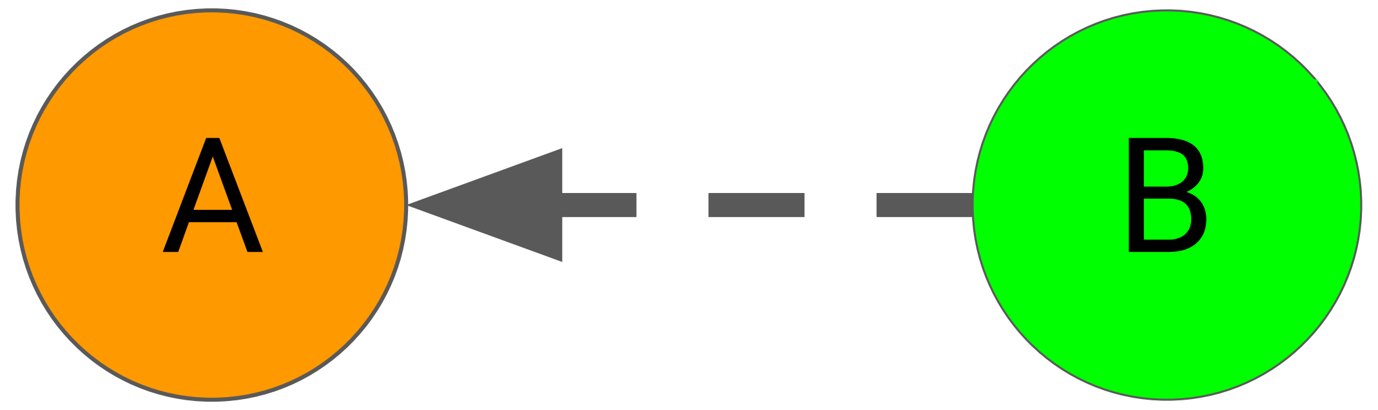
*Parallel module loading (~250ms) - Doesn’t help static kernels, needs userspace support



* - Not fully functional yet.

android

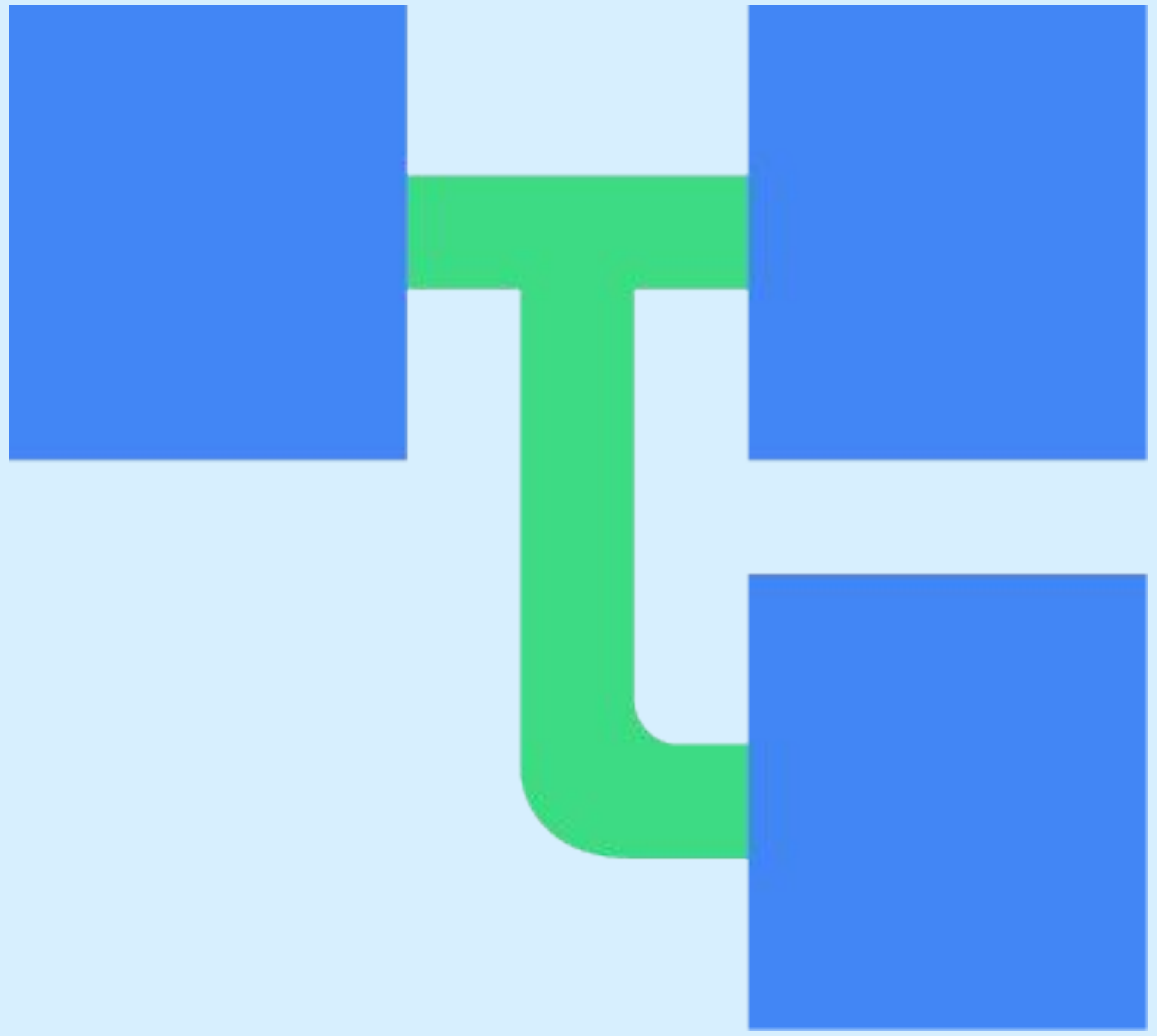
Booting with missing drivers



- A doesn’t have a driver in upstream kernel
- B lists A as a supplier in DT
- Driver of B doesn’t even try to “get” A.

Does B probe?	fw_devlink=on	fw_devlink=permissive
!CONFIG_MODULES	✔	✔
CONFIG_MODULES	✘	✔

- Ask platform to use deferred_probe_timeout=1?
- Add new fw_devlink.timeout and default to 1?
- Set timer for 5s and extend for each module load?
- Other ideas?



android

General discussion topics

Is fw_devlink=on helpful for your use case?

Is fw_devlink=on breaking your use case?

Parent devices depending on their child devices.

Any DT properties we are missing?

How can we improve debugging friendliness?

Any concerns with topologically sorted probing of devices?

Does anyone want to add ACPI support?

Thank you!

android