



Arm Solutions at Lightspeed

Making MIPI DSI power states fit Linux kernel

MIPI DSI in a nutshell

- Data is sent in packets
- Video mode (stream of pixels) or Command mode (draw here) peripherals
- Up to 4 virtual channels
- Several lanes (further defined by the PHY)
 - D-PHY: 1 clock lane, up to 4 data lanes, differential pairs
 - C-PHY: up to 3 lanes, embedded clock, 3 wires
- Can be split into several Sub-Links (by distributing lanes)
- Optional bidirectional lane
- RGB data, can support YCbCr, DSC, HDR, etc.
- LP (slow) or HS (fast) transfers

MIPI DSI in a nutshell

- Three bus states:
 - Off
 - Clock on
 - Video on
- Usually one peripheral on a bus
 - Can have more than one

Mapping to DRM

- peripheral::pre_enable
 - host::pre_enable
 - encoder::enable
 - host::enable
 - peripheral::enable
- enable video stream
power on DSI bus, switch to HS mode
send DSI commands

SUN4i: no DSI commands after video stream!
How to shutdown the panel?

Mapping to DRM, try2

- host::mode_set
 - peripheral::pre_enable
- host::pre_enable
- encoder::enable
 - host::enable
 - peripheral::enable
 - peripheral::disable
- host::disable
- encoder::disable
 - host::post_disable
 - peripheral::post_disable

power on DSI bus
send DSI commands
switch to HS mode
enable video stream

shutdown the panel?

disable video stream
shutdown DSI bus???
shutdown the panel?

Special cases (Parade PS8640)

Mapping to DRM, pre_enable_prev_first

- host::mode_set
 - host::pre_enable
 - peripheral::pre_enable
 - encoder::enable
 - host::enable
 - peripheral::enable
 - peripheral::disable
 - host::disable
 - encoder::disable
 - peripheral::post_disable
 - host::post_disable
- power on DSI bus
send DSI commands
enable video stream
switch to HS mode
- disable video stream
shutdown the panel
shutdown DSI bus

Does peripheral use pre_enable_prev_first???

Multiple devices on a bus?

DSI bus callbacks

```
host_pre_enable()
{
    for_each_dsi_device()
        dsi->power_on();

    power_on_bus();

    for_each_dsi_device()
        dsi->lp_commands();
}
host_enable()
{
    for_each_dsi_device()
        dsi->hs_commands();
}
```

Mapping to DRM, DSI bus callbacks

- host::mode_set
- host::pre_enable

- encoder::enable
 - host::enable

- host::disable
- encoder::disable
 - host::post_disable

power on DSI bus
make peripherals send commands
enable video stream
switch to HS mode

disable video stream
make peripherals to shutdown
shutdown DSI bus

DSI bus callbacks, Pro et Contra

- No feature negotiation
- Clean bus state separation
- Multiple panels handled logically!
- Rewrite all hosts
- Rewrite all clients
- Multiple panels enabled simultaneously :-)
- Doesn't use drm_panel funcs anymore
- ULPS?

Mapping to DRM, Runtime PM

- peripheral::pre_enable pm_runtime_get_sync(dsi_host)
send DSI commands
- host::pre_enable
- encoder::enable enable video stream
- host::enable switch to HS mode
- peripheral::enable
- peripheral::disable shutdown the panel
- host::disable pm_runtime_put(dsi_host)
- encoder::disable disable video stream
- host::post_disable
- peripheral::post_disable

Runtime PM, Pro et Contra

- No feature negotiation
- Clean bus state separation
- Rewrite all hosts and clients
- Multiple peripherals case not solved
- ULPS?

Thank You!

Visit linaro.org