DSI device tree configuration

Contents

DSI device tree configuration

Stable: 07.10.2019 - 09:13 / Revision: 07.10.2019 - 09:12

Contents

1 Article purpose

This article explains how to configure the *DSI* [1] **when the peripheral is assigned to the Linux®OS**.

The configuration is performed using the **device tree mechanism** [2].

The Device tree provides a hardware description of the DSI [1] used by the STM32 *DSI Linux driver*.

2 DT bindings documentation

The DSI is represented by the STM32 DSI device tree bindings [3].

3 DT configuration

This hardware description is a combination of the **STM32 microprocessor** device tree files (*.dtsi* extension) and **board** device tree files (*.dts* extension). See the Device tree for an explanation of the device tree file split.

STM32CubeMX can be used to generate the board device tree. Refer to How to configure the DT using STM32CubeMX for more details.

3.1 DT configuration (STM32 level)

The DSI device tree node is declared in stm32mp157c.dtsi [4]. The declaration (shown below) defines the hardware registers base address, the clocks and the reset.

```
                dsi: dsi@5a000000 {
                        compatible = "st,stm32-dsi";
                        reg = <0x5a000000 0x800>;
                        clocks = <&rcc DSI_K>, <&clk_hse>, <&rcc DSI_PX>;
                        clock-names = "pclk", "ref", "px_clk";
                        resets = <&rcc DSI_R>;
                        reset-names = "apb";
                        status = "disabled";
                };
```

This device tree part is related to STM32 microprocessors. It must be kept as is, without being modified by the end-user.

3.2 DT configuration (board level)

The DSI device tree related to a particular board may have the following nodes, depending on the board hardware:

- **dsi** node: containing the in/out port descriptions and a **panel** sub-node.
- **ltdc** node: containing the in/out port description related to the dsi node.
- **panel_backlight** node: related to the panel node.

A full example of the STM32MP157 Evaluation board device tree is available in stm32mp157c-ev1.dts [5].

```
&dsi {
        #address-cells = <1>;
        #size-cells = <0>;
        status = "okay";

        ports {
                #address-cells = <1>;
                #size-cells = <0>;

                port@0 {
                        reg = <0>;
                        dsi_in: endpoint {
                                remote-endpoint = <&ltdc_ep0_out>;
                        };
                };

                port@1 {
                        reg = <1>;
                        dsi_out: endpoint {
                                remote-endpoint = <&panel_in>;
                        };
                };
        };

        panel@0 {
                compatible = "raydium,rm68200";
                reg = <0>;
                reset-gpios = <&gpiof 15 GPIO_ACTIVE_LOW>;
                backlight = <&panel_backlight>;
                status = "okay";

                port {
                        panel_in: endpoint {
                                remote-endpoint = <&dsi_out>;
                        };
```

```
                };
        };
};

&ltdc {
        status = "okay";

        port {
                #address-cells = <1>;
                #size-cells = <0>;

                ltdc_ep0_out: endpoint@0 {
                        reg = <0>;
                        remote-endpoint = <&dsi_in>;
                };
        };
};

...
        panel_backlight: panel-backlight {
                compatible = "gpio-backlight";
                gpios = <&gpiod 13 GPIO_ACTIVE_LOW>;
                default-on;
                status = "okay";
        };
};
```

4 How to configure the DT using STM32CubeMX

The STM32CubeMX tool can be used to configure the STM32MPU device and get the corresponding platform configuration device tree files.
The STM32CubeMX may not support all the properties described in the above DT bindings documentation paragraph. If so, the tool inserts **user sections** in the generated device tree. These sections can then be edited to add some properties and they are preserved from one generation to another. Refer to STM32CubeMX user manual for further information.

5 References

Please refer to the following links for additional information:

- [1.01.1] DSI internal peripheral
- Device tree
- st,stm32-ltdc.txt Linux kernel bindings (including dsi)
- Linux kernel STM32MP157C device tree (stm32mp157c.dtsi)
- Linux kernel STM32MP157 Evaluation board device tree (stm32mp157c-ev1.dts)

Display Serial Interface (MIPI® Alliance standard)

Operating System

Device Tree

General-Purpose Input/Output (A realization of open ended transmission between devices on an embedded level. These pins available on a processor can be programmed to be used to either accept input or provide output to external devices depending on user desires and applications requirements.)

Permission error

Stable: 04.02.2020 - 07:47 / Revision: 04.02.2020 - 07:34

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer

Permission error

Stable: 25.10.2019 - 09:04 / Revision: 25.10.2019 - 09:04

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer

Permission error

Stable: 31.01.2020 - 13:04 / Revision: 31.01.2020 - 13:02

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer

Permission error

Stable: 16.01.2020 - 10:17 / Revision: 16.01.2020 - 10:16

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer