RICO-3288 MIPI DSI

Edit by Jason Wang

MIPI DSI

1. Introduction¶

The Display Serial Interface (DSI) is part of a group of communication protocols defined by the MIPI Alliance. The MIPI Controller is a digital core that implements all protocol functions defined in the MIPI DSI Specification. The MIPI Controller provides an interface between the system and the MIPI D-PHY, allowing the communication with a DSI-compliant display. The MIPI Controller supports one to four lanes for data transmission with MIPI D-PHY.

The MIPI Controller supports the following features:

- Compliant with MIPI Alliance standards
- Support the DPI interface color coding mappings into 24-bit Interface
  - 16 bits per pixel, configurations 1,2,and 3
  - 18 bits per pixel, configurations 1 and 2
  - 24 bits per pixel
- Programmable polarity of all DPI interface signals
- Extended resolutions beyond the DPI standard maximum resolution of 800x480 pixels:
  - Up to 2047 vertical active lines
  - Up to 63 vertical back porch lines
  - Up to 63 vertical front porch lines
  - Maximum resolution is limited by available DSI Physical link bandwidth which
- depends on the number of lanes and maximum speed per lane
- All commands defined in MIPI Alliance Specification for Display Command Set (DCS)
- Interface with MIPI D-PHY following PHY Protocol Interface (PPI), as defined in MIPI
- Alliance Specification for D-PHY
- Up to four D-PHY Data Lanes
- Bidirectional communication and escape mode support through data lane 0
- Transmission of all generic commands
- ECC and Checksum capabilities
- End of Transmission Packet(EOTp)
- Ultra Low-Power mode

2. How to Use¶

Kernel configuration

Add the following configuration to Kernel

```
CONFIG_LCD_MIPI=y
CONFIG_MIPI_DSI=y
CONFIG_RK32_MIPI_DSI=y
```

LCD pins configuration¶

Modify the child node “power_ctr” of “lcdc0” in path arch/arm/boot/dts/lcd-kd080d24-39th-mipi.dtsi. There are power enable pin “mipi_lcd_en”, reset pin “mipi_lcd_rst”. You can modify those configuration according to the LCD. For example:

```
disp_mipi_power_ctr: mipi_power_ctr {
        compatible = "rockchip,mipi_power_ctr";

        mipi_lcd_en: mipi_lcd_en {
            compatible      = "rockchip,lcd_en";
            rockchip,gpios  = <&gpio7 GPIO_A3 GPIO_ACTIVE_HIGH>;
            rockchip,delay  = <20>;
        };

    mipi_lcd_rst: mipi_lcd_rst {
            compatible      = "rockchip,lcd_rst";
            rockchip,gpios  = <&gpio7 GPIO_A4 GPIO_ACTIVE_HIGH>;
            rockchip,delay  = <20>;
        };
    };
```

Configure information of MIPI¶

configure information of MIPI, for example:

```
disp_mipi_init: mipi_dsi_init {
        compatible              = "rockchip,mipi_dsi_init";
        rockchip,screen_init        = <1>;
        rockchip,dsi_lane           = <4>;
        rockchip,dsi_hs_clk         = <600>;
        rockchip,mipi_dsi_num       = <1>;
};
```

Prorerty

- rockchip,screen_init: the value of 0 means without special instruction while initial. The value of 1 means need the special instruction to initial the LCD display.
- rockchip,dsi_lane: the number of lane.
- rockchip,dsi_hs_clk: the value of hsclk.
- rockchip,mipi_dsi_num：the number of DSI interface, single channel is 1, dual channel is 2.

The configuration of initial instructions¶

- It will need the special instructions to initial the When the value of “rockchip,screen_init” is set to 1.
- rockchip,cmd_debug: Set this value to 1 and you will get the debug information from the debug serial.
- rockchip,on-cmdsXX: the information of each instruction.
- rockchip,cmd_type：the mode for data transfer. PDT mode or HSDT mode.
- rockchip,dsi_id：The DSI interface for instructions transfer. If this value is set to 0 means using the DSI0 (it’s the left side of the display while dual lane MIPI display) for instruction transfer.If this value is set to1 means using the DSI1(it’s the right side of the display while dual lane MIPI display) for instruction transfer. This value is set to 2 means using both the DSI0 and DSI1 for instruction transfer.
- rockchip,cmd：the sequence of the instructions. The first Byte is the DSI data type, the second Byte is the RGE. The rest are the instructions.
- rockchip,cmd_delay：the delay after sending instructions. The unit is ms.

```
disp_mipi_init_cmds: screen-on-cmds {
        compatible              = "rockchip,screen-on-cmds";
        rockchip,cmd_debug          = <1>;

        rockchip,on-cmds1 {
            compatible          = "rockchip,on-cmds";
            rockchip,cmd_type = <LPDT>;
            rockchip,dsi_id             = <0>;
            rockchip,cmd        = <0x39 0xB9 0xff 0x83 0x94>;
            rockchip,cmd_delay      = <0>;
        };

        rockchip,on-cmds2 {
            compatible          = "rockchip,on-cmds";
            rockchip,cmd_type = <LPDT>;
            rockchip,dsi_id             = <0>;
            rockchip,cmd        = <0x39 0xd9 0x00 0x8b 0x02 0x07>;
            rockchip,cmd_delay      = <0>;
        };
……
```

The configuration of display sequence¶

Configure the time sequence in disp_timings node.

- screen-type: Display type. If the MIPI display use single lane, the display type is SCREEN_MIPI. If the MIPI display use dual lanes, the display type is SCREEN_DUAL_MIPI,SCREEN_MIPI.
- lvds-format: Independent options.
- out-face: color configuration, it can be OUT-P888(24 bits) or OUT_P666(18 bits)
- clock-frequency: clock of the LCD, measure by Hz.

```
disp_timings: display-timings {
        compatible              = "rockchip,display-timings";
        native-mode             = <&timing0>;
        timing0: timing0 {
            screen-type         = <SCREEN_MIPI>;
            lvds-format         = <LVDS_8BIT_2>;
            out-face            = <OUT_P888>;
            clock-frequency     = <75000000>;
```

```
                hactive             = <800>;
                vactive             = <1280>;

                hback-porch         = <64>;
                hfront-porch        = <70>;
                vback-porch         = <16>;
                vfront-porch        = <9>;
                hsync-len           = <20>;
                vsync-len           = <2>;

                hsync-active        = <0>;
                vsync-active        = <0>;
                de-active           = <0>;
                pixelclk-active     = <0>;
                swap-rb             = <0>;
                swap-rg             = <0>;
                swap-gb             = <0>;
        };
```

Dsihost configuration

If you using single MIPI (RICO-3288 only support single MIPI), enable the dsihost0, for example:

```
dsihost0: mipi@ff960000{
        compatible = "rockchip,rk32-dsi";
        rockchip,prop = <0>;
        reg = <0xff960000 0x4000>;
        interrupts = <GIC_SPI 19 IRQ_TYPE_LEVEL_HIGH>;
        clocks = <&clk_gates5 15>, <&clk_gates16 4> , <&pd_mipidsi>;
        clock-names = "clk_mipi_24m", "pclk_mipi_dsi", "pd_mipi_dsi";
        status = "okay";
    };
```