# SSD2829

*Advance Information*

## MIPI D-PHY Tx Bridge Chip



**Appendix: IC Revision history of SSD2829 Specification**

| Version | Change Items | Effective Date |
|---|---|---|
| 0.10 | 1st Release | 11-May-16 |
| 0.11 | 1. Section 5.2 Pin Assignment Table (Draft), pins 109 to 128 were amended<br>2. Modified operating temperature from 125oC to 85 oC<br>3. Modified MIPI D-Phy AC and DC Characteristic<br>4. Added back 8-bit MCU mode<br>5. Modified pin assignment table<br>6. Modified MIPI DPHY Lane and Polarity Swap table<br>7. Added description for compressed stream data | 27-May-16 |
| 0.12 | 1. Modified max. PCLK value<br>2. Modified pin assignment table | 16-Aug-16 |
| 0.13 | 1. Modified Table of MIPI packet ID<br>2. Modified AVDD_CDR pin description<br>3. Updated Command table | 26-Sep-16 |
| 1.0 | Updated to Advance Information | 16-Feb-17 |
| 1.1 | 1. Removed 30-bit VPT_EXT<br>2. Change max. of phase difference of channel 0 and 1 from 1 to 2pclk<br>3. Changed data buffer from 5120 to 4128<br>4. Change PLL configuration, from 5MHz<fIN=<100 to 5MHz<fIN=<40MHz<br>5. Modified DC characteristic<br>6. Removed descriptions of TE_IN_1 and TE_OUT_1<br>7. Change PEN range in 0xBA from “501 to 1000” to “501 to 1250”<br>8. Added power on/off sequence<br>9. Remove DSI, lane and polarity swap features | 17-Jul-17 |
| 1.2 | 1. Added “Support MIPI DSI standard version 2.0” to feature list<br>2. Corrected MCU mode type B timing to match with type A<br>3. Added VDDIO 3.3V and VCIP current consumption for DC characteristic<br>4. Reserved VID_COMPRESSED_BYTE_COUNT of 0xC8<br>5. Reserved BYP_BIT_DIV of 0xBB | 27-Sep-17 |

**CONTENTS**

## TABLES

## FIGURES

# 1 GENERAL DESCRIPTION

SSD2829 is a MIPI master bridge chip that converts RGB / MCU interface to MIPI DPHY DSI Output.

For RGB interface, it can support resolution up to WQHD (2560x1600) (native) and UHD (4096x2160) (compressed in/out) format with 60Hz refresh rate.

For MCU interface, it can support resolution up to WQHD (2560x1600) (native) and UHD (4096x2160) (compressed in/out) format with 30Hz refresh rate.

# 2 FEATURES

- Support panel at refresh rate of 60Hz with resolution up to
  - WQHD (2560x1600) (native)
  - UHD (4096 x 2160) (compressed in/out)
- Support MIPI DSI standard version 2.0
- Support MIPI D-PHY standard version 1.1
- Support MIPI DCS standard version 2.0
- Support 2 MIPI D-option DSI engines with throughput up to 10Gbps using 8 D-PHY lanes for each DSI-TX (Each lane is up to 1.25Gbps)
- Support single or dual DSI mode at DSI-TX output
- Support 16, 18, 24-bit per pixel color
- Support MCU interface (DBI version 2.0) up to 24-bits bus width at the input
- Support RGB interface (DPI version 2.0) 48-bits bus width with SDR or DDR pixel clock at the input
- Support serial SPI interface (DBI version 2.0) up to 16-bits at the input
- Support both Video and Command mode
- Support input Left-right or odd-even split in the RGB input
- Support Video BIST generation at the DSI-TX output with different color patterns
- Support Burst or Non-burst video modes
- Each DSI-TX port can control the number of lane independently
- On-chip PLL with variable output frequency
- Power supply: (VDDD and VDDA) 1.2V +/-10%
- IO Power supply: 1.8V +/-10% or 3.3V +/-10%

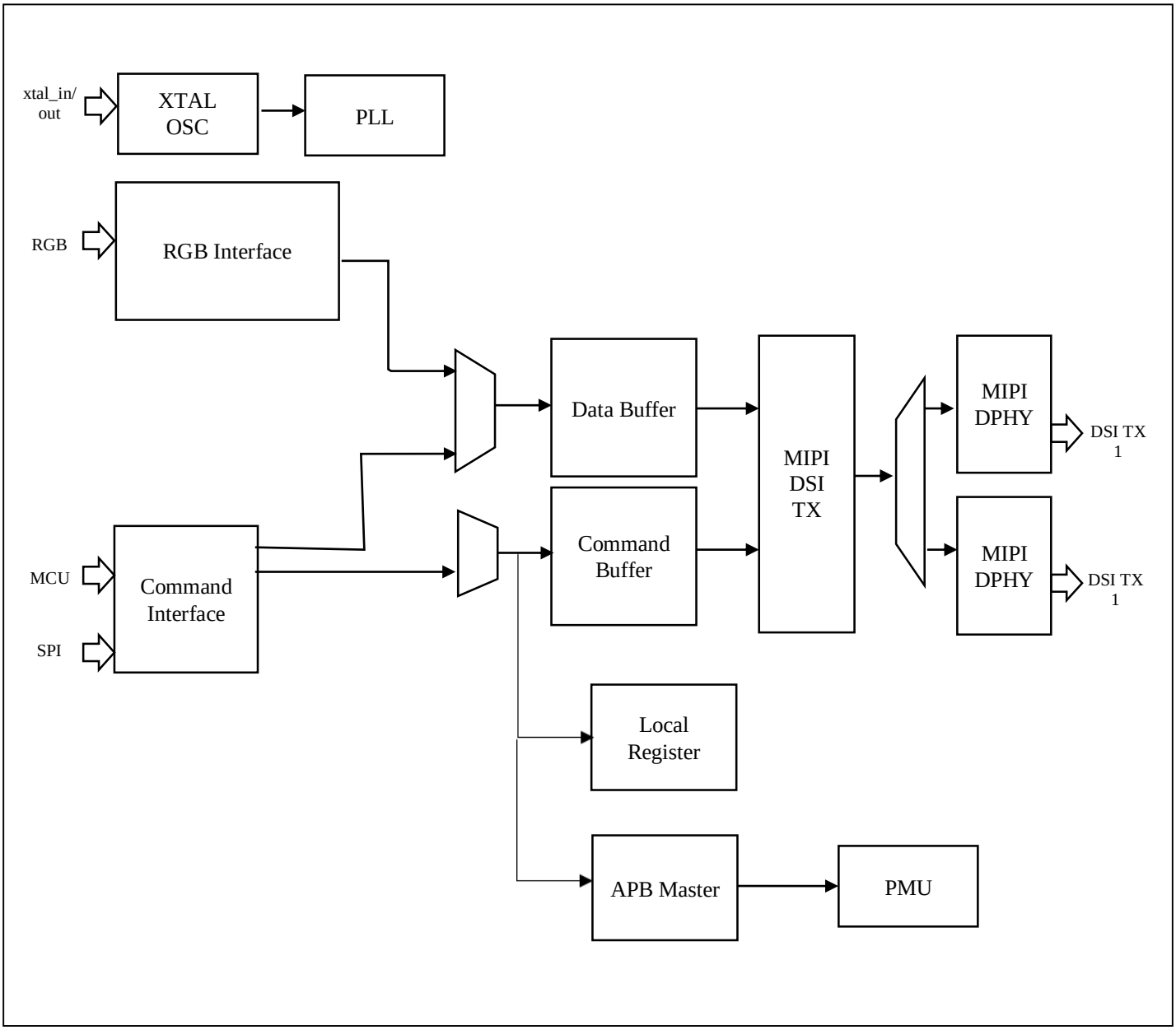
# 3 ORDERING INFORMATION

**Table 3-1: Ordering Information**

| Ordering Part Number | Package Form |
|---|---|
| SSD2829QL9 | LQFP 128L, 14mm x 14mm |

# 4 BLOCK DIAGRAM

**Figure 4-1: SSD2829 Block Diagram**



# 5 FUNCTIONAL DESCRIPTION

## 5.1 RGB Interface

The RGB interface receives parallel video data and routes them to the data buffer. The RGB interface supports 2 pixels per PCLK cycle. The RGB interface input is 48-bit wide and it supports 2-pixels per RGB module using SDR or DDR input pixel clock. The maximum speed for the RGB interface is 160MHz.

## 5.2 Command Interface

The Command interface receives parallel MCU data or SPI data and routes them to the command buffer. The MCU interface supports 8-bit, 16-bit and 24-bit data width. The maximums speed for the MCU interface is 160MHz.

## 5.3 Data Buffer

The data buffer consists of line buffers to store one line worth of video data before packetizing them for MIPI TX transmission. Data for command 0x2C and 0x3C also make use of the data buffer for storage, instead of going to the command buffer.

There are one line buffer per MIPI DSI TX port. For DSI TX0, the line buffer size is 2560 pixels. For DSI TX1, the size is 2064 pixels. Dual DSI TX port can support up to 2 x 2064 = 4128 pixels.

## 5.4 Command Buffer

The command buffer consist of a 4096 bytes deep FIFO to store commands before packetizing them to command packets for MIPI TX transmission. Command 0x2C and 0x3C are excluded in this command buffer. They are routed to use data buffer instead.

There are one command FIFO per MIPI DSI TX port.

## 5.5 MIPI DSI-Tx

MIPI DSI-TX is a dual DSI TX module, up to 8 lanes for D-option.
Each DSI is 1.25Gbps per lane for D-option.

The main features of MIPI DSI-TX transmitter pairs are,

- Dual DSI-TX D-option up to 8-lanes
- 4 lanes for each DSI-TX D-option
- Support up to 2560 pixel/line for 1 DSI
- Up to 1.25Gbps per lane for each D-option lane or 10Gbps for 2 DSI DPHY
- Single or dual DSI mode
- Support 16, 18, 24-bit per pixel
- Support burst or non-burst mode
- Support new commands in DSI-2, such as Execute Queue, Scrambler On/Off, Compressed packets
- Support MIPI Alliance Standard for Display Serial Interface-2, version 1
- Support MIPI Alliance Standard for Display Command Set, version 1.02
- Support MIPI Alliance Standard for D-PHY, version 1.00

The MIPI packets that are supported by MIPI DSI-TX are listed in the table below.

**Table 5-1: DSI-TX Support Format**

| Data Type (Hex) | Data Type (Bin) | Type | Description |
|---|---|---|---|
| 0x01 | 00 0001 | Short | Sync Event V Start |
| 0x11 | 01 0001 | Short | Sync Event V End |
| 0x21 | 10 0001 | Short | Sync Event H Start |
| 0x31 | 11 0001 | Short | Sync Event H End |
| 0x08 | 00 1000 | Short | End of Transmission (EoT) |
| 0x02 | 00 0010 | Short | Color Mode (CM) Off |
| 0x12 | 01 0010 | Short | Color mode (CM) On |
| 0x22 | 10 0010 | Short | Shut Down Peripheral |
| 0x32 | 11 0010 | Short | Turn On Peripheral |
| 0x03 | 00 0011 | Short | Generic Short Write, no parameter |
| 0x13 | 01 0011 | Short | Generic Short Write, 1 parameter |
| 0x23 | 10 0011 | Short | Generic Short Write, 2 parameters |
| 0x04 | 00 0100 | Short | Generic Read, no parameter |
| 0x14 | 01 0100 | Short | Generic Read, 1 parameter |
| 0x24 | 10 0100 | Short | Generic Read, 2 parameters |
| 0x05 | 00 0101 | Short | DCS Short Write, no parameter |
| 0x15 | 01 0101 | Short | DCS Short Write, 1 parameter |
| 0x06 | 00 0110 | Short | DCS Read, no parameter |
| 0x16 | 01 0110 | Short | Execute Queue |
| 0x37 | 11 0111 | Short | Set Maximum Return Packet Size |
| 0x27 | 10 0111 | Short | Scrambling Mode Command |
| 0x09 | 00 1001 | Long | Null Packet |
| 0x19 | 01 1001 | Long | Blanking Packet |
| 0x39 | 11 1001 | Long | DCS Long Write |
| 0x0A | 00 1010 | Long | Picture Parameter Set |
| 0x0B | 00 1011 | Long | Compressed Pixel Stream |
| 0x0E | 00 1110 | Long | Packed Pixel Stream, 16-bit RGB, 5-6-5 Format |
| 0x1E | 01 1110 | Long | Packed Pixel Stream, 18-bit RGB, 6-6-6 Format |
| 0x2E | 10 1110 | Long | Loosely Packed Pixel Stream, 18-bit RGB, 6-6-6 Format |
| 0x3E | 11 1110 | Long | Packed Pixel Stream, 24-bit RGB, 8-8-8 Format |
| 0x02 | 00 0010 | Short | Acknowledge and Error Report |
| 0x11 | 01 0001 | Short | Generic Short Read Response, 1 byte returned |
| 0x12 | 01 0010 | Short | Generic Short Read Response, 2 bytes returned |

| Data Type (Hex) | Data Type (Bin) | Type | Description |
|---|---|---|---|
| 0x1A | 01 1010 | Long | Generic Long Read Response |
| 0x1C | 01 1100 | Long | DCS Long Read Response |
| 0x21 | 10 0001 | Short | DCS Short Read Response, 1 byte returned |
| 0x22 | 10 0010 | Short | DCS Short Read Response, 2 bytes returned |

MIPI DSI Link controller provides MIPI DSI packet assembly and disassembly. During transmission, it will form the DSI packet according to the instruction from the state machine. During reception, it will extract necessary information from the packet and pass to the higher level block. The MIPI DSI Link Controller is also responsible for generating the CRC and ECC for the out-going bit stream. During reception, it will check the correctness of the ECC and CRC field of the incoming stream.

When operated in 2-DSI mode, the MIPI DSI Link Controller is able to split the incoming video into 2 equal portions and send each half of the line to each of the MIPI DSI engines. Each MIPI DSI engine take the half video data and reformat it into RGB 16/18/24-bit packet and send out as 1 packet per line.

A line buffer is used to buffer a single video line from the upstream module and it will regenerate the Video timing with the video settings stored inside the local registers. The output rate from the buffer must be greater than the input rate to prevent data overflow.

MIPI DSI Link controller is also capable of sending DCS/Generic commands to external MIPI DSI drivers via multiple sources.

**Table 5-2 SSD2829 RGB data arrangement**

| | D23 | D22 | D21 | D20 | D19 | D18 | D17 | D16 | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 24bpp | R7 | R6 | R5 | R4 | R3 | R2 | R1 | R0 | G7 | G6 | G5 | G4 | G3 | G2 | G1 | G0 | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
| 18bpp | X | X | X | X | X | X | R5 | R4 | R3 | R2 | R1 | R0 | G5 | G4 | G3 | G2 | G1 | G0 | B5 | B4 | B3 | B2 | B1 | B0 |
| 16bpp | X | X | X | X | X | X | X | X | R4 | R3 | R2 | R1 | R0 | G5 | G4 | G3 | G2 | G1 | G0 | B4 | B3 | B2 | B1 | B0 |

## 5.6 XTAL OSC

This is a crystal oscillator pad. From a circuit point of view, the crystal oscillator I/O cells are not real oscillators, but amplifiers used to generate high quality clock signals. Full range configurable output driving capability

## 5.7 PLL

This is a PLL control the MIPI output frequency.

## 5.8 PMU

The PMU (Power Management Unit) is responsible for putting SSD2829 into deep-sleep mode, cutting the power consumption to ultra-low level. Internally, it uses APB interface for register programming

# 6 PIN ARRANGEMENT

## 6.1 128 pins LQFP

**Figure 6-1 : Pinout Diagram – 128 pins LQFP (Topview)**

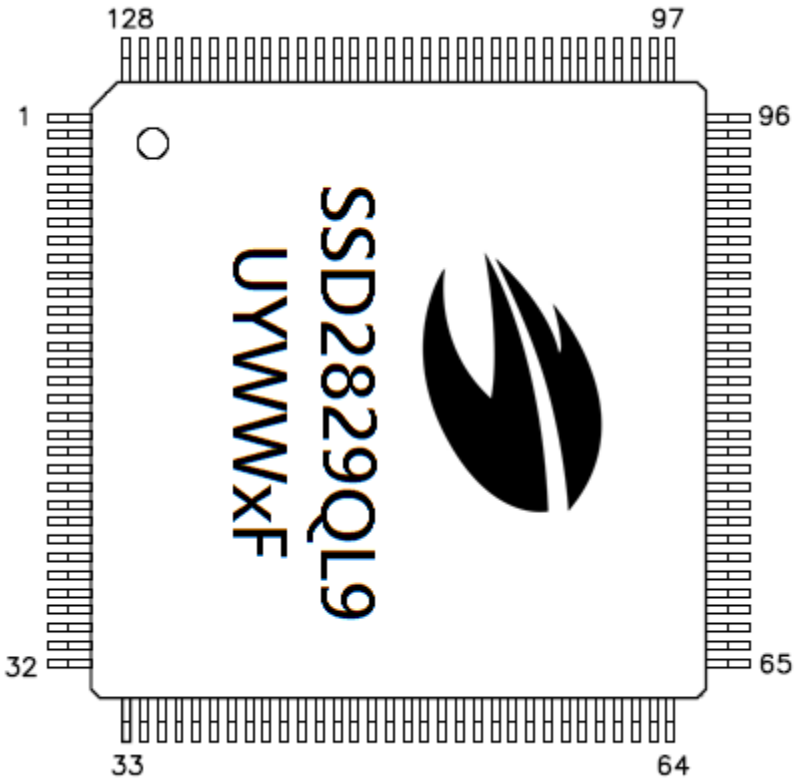


**Table 6-1: LQFP Pin Assignment Table**

| Pin # | QFP Pin name | Pin # | QFP Pin name | Pin # | QFP Pin name | Pin # | QFP Pin name |
|---|---|---|---|---|---|---|---|
| 1 | AVDD | 33 | DATA0_13 | 65 | DATA0_44 | 97 | AVDD |
| 2 | VSS | 34 | DATA0_14 | 66 | DATA0_45 | 98 | AVDD_CDR |
| 3 | VDD_CORE | 35 | DATA0_15 | 67 | DATA0_46 | 99 | TXB_DN3 |
| 4 | IF_SEL0 | 36 | DATA0_16 | 68 | DATA0_47 | 100 | TXB_DP3 |
| 5 | VDDIO | 37 | DATA0_17 | 69 | DATA0_48 | 101 | TXB_DN0 |
| 6 | VSS | 38 | DATA0_18 | 70 | DATA0_49 | 102 | TXB_DP0 |
| 7 | VSS | 39 | DATA0_19 | 71 | DATA0_50 | 103 | VDRV |
| 8 | VSS | 40 | DATA0_20 | 72 | DATA0_51 | 104 | AVDD |
| 9 | RST_IN | 41 | DATA0_21 | 73 | DATA0_52 | 105 | TXB_CN |
| 10 | VDDIO | 42 | DATA0_22 | 74 | DATA0_53 | 106 | TXB_CP |
| 11 | VDD_CORE | 43 | DATA0_23 | 75 | SDO | 107 | TXB_DN1 |
| 12 | VSS | 44 | TE_OUT_0 | 76 | SDI | 108 | VDRV |
| 13 | CSX0 | 45 | TE_OUT_1 | 77 | SCK | 109 | TXB_DP1 |
| 14 | CLK_IN | 46 | VDDIO | 78 | DEN | 110 | TXB_DN2 |
| 15 | PD_N | 47 | XTAL_IN | 79 | PCLK | 111 | TXB_DP2 |
| 16 | VSYNC | 48 | XTAL_OUT | 80 | HSYNC | 112 | AVDD_RC |
| 17 | HSYNC | 49 | VSS | 81 | VSYNC | 113 | VCIP |
| 18 | PCLK | 50 | VDD_CORE | 82 | TE_IN_1_CM | 114 | VDRV_REG |
| 19 | DEN | 51 | DATA0_30 | 83 | TE_IN_0_CM | 115 | AVSS |
| 20 | DATA0_0 | 52 | DATA0_31 | 84 | VSS | 116 | TXA_DN3 |
| 21 | DATA0_1 | 53 | DATA0_32 | 85 | VDD_CORE | 117 | TXA_DP3 |
| 22 | DATA0_2 | 54 | DATA0_33 | 86 | VDDIO | 118 | TXA_DN0 |
| 23 | DATA0_3 | 55 | DATA0_34 | 87 | SDC | 119 | TXA_DP0 |
| 24 | DATA0_4 | 56 | DATA0_35 | 88 | PS4 | 120 | VDRV |
| 25 | DATA0_5 | 57 | DATA0_36 | 89 | PS3 | 121 | AVDD |
| 26 | DATA0_6 | 58 | DATA0_37 | 90 | PS2 | 122 | TXA_CN |
| 27 | DATA0_7 | 59 | DATA0_38 | 91 | PS1 | 123 | TXA_CP |
| 28 | DATA0_8 | 60 | DATA0_39 | 92 | PS0 | 124 | TXA_DN1 |
| 29 | DATA0_9 | 61 | DATA0_40 | 93 | INT_B | 125 | VDRV |
| 30 | DATA0_10 | 62 | DATA0_41 | 94 | VDD_CORE | 126 | TXA_DP1 |
| 31 | DATA0_11 | 63 | DATA0_42 | 95 | VSS | 127 | TXA_DN2 |
| 32 | DATA0_12 | 64 | DATA0_43 | 96 | AVDD | 128 | TXA_DP2 |

# 7 PIN DESCRIPTIONS

**Key:**

I = Input
O =Output
IO = Bi-directional (input/output)
P = Power pin

## 7.1 Power Supply Pin

**Table 7-1: Power Supply Pin Description**

| Pin name | Type | Connect to | Description | When not in use |
|---|---|---|---|---|
| VDD | P | Power | Core Power Supply, 1.2V | - |
| VDDIO | P | Power | I/O Power Supply, 1.8V or 3.3V | - |
| VSS | P | GND | Ground | - |
| AVDD_CDR | P | Power | Analog Regulator Output for DPHY CDR, 1.2V | - |
| AVDD_RC | P | Power | Analog Power Supply 1.2V | - |
| AVDD | P | Power | Analog Power Supply 1.2V | - |
| AVDD_CORE | P | Power | Analog Core Power Supply 1.2V | - |
| AVSS | P | GND | Ground | - |
| VCIP | P | Power | Power for Bandgap, 3.3V | - |
| VDRV_REG | P | Power | LV Regulator Output | - |
| VDRV | P | Power | Power for MIPI TX Driver (to be connected to VDRV_REG, 0.5V) | - |

## 7.2 MIPI Pin

**Table 7-2: MIPI Pin Description**

| Pin name | Type | Connect to | Description | When not in use |
|---|---|---|---|---|
| TX0_DP0 | I/O | MIPI Tx | TX0 DSI Data Lane Positive 0 | Open |
| TX0_DN0 | I/O | | TX0 DSI Data Lane Negative 0 | |
| TX0_DP1 | O | | TX0 DSI Data Lane Positive 1 | |
| TX0_DN1 | O | | TX0 DSI Data Lane Negative 1 | |
| TX0_DP2 | O | | TX0 DSI Data Lane Positive 2 | |
| TX0_DN2 | O | | TX0 DSI Data Lane Negative 2 | |
| TX0_DP3 | O | | TX0 DSI Data Lane Positive 3 | |
| TX0_DN3 | O | | TX0 DSI Data Lane Negative 3 | |
| TX0_CP | O | | TX0 DSI Clock Lane Positive | |
| TX0_CN | O | | TX0 DSI Clock Lane Negative | |
| TX1_DP0 | I/O | | TX1 DSI Data Lane Positive 0 | |
| TX1_DN0 | I/O | | TX1 DSI Data Lane Negative 0 | |
| TX1_DP1 | O | | TX1 DSI Data Lane Positive 1 | |
| TX1_DN1 | O | | TX1 DSI Data Lane Negative 1 | |
| TX1_DP2 | O | | TX1 DSI Data Lane Positive 2 | |

| | | | | |
|---|---|---|---|---|
| TX1_DN2 | O | | TX1 DSI Data Lane Negative 2 | |
| TX1_DP3 | O | | TX1 DSI Data Lane Positive 3 | |
| TX1_DN3 | O | | TX1 DSI Data Lane Negative 3 | |
| TX1_CP | O | | TX1 DSI Clock Lane Positive | |
| TX1_CN | O | | TX1 DSI Clock Lane Negative | |

## 7.3 Control Signal Pin

**Table 7-3: Control Signal Pin Description**

| Pin name | Type | Connect to | Description | When not in use |
|---|---|---|---|---|
| RESET | I | VDDIO / GND | System Reset signal to the whole chip, active low | VDDIO |
| INT_B | O | - | Output Interrupt Signal | Open |
| PD_N | I | VDDIO / GND | Power Down, active low | VDDIO |
| IF_SEL[0] | I | VDDIO / GND | Interface selection signals<br>- 0 : A combination of RGB and SPI interface is selected<br>- 1 : MCU interface is selected | VDDIO / GND |
| PS[4:0] | I | VDDIO / GND | Interface selection signal<br>PS[1:0] is for SPI interface<br>- 00: 3 wire 24-bit SPI interface<br>- 01: 3 wire 8-Bit SPI interface<br>- 10: 4 wire 8-Bit SPI interface<br>- 11: Reserved<br><br>PS[4:2] is for the MCU interface<br>When if_sel is '01'<br>- 000: 8-Bit MCU interface (MIPI DBI type B)<br>- 001: 16-bit MCU interface (MIPI DBI type B)<br>- 010: 8-Bit MCU interface (MIPI DBI type A, fixed E or clocked E mode)<br>- 011: 16-bit MCU interface (MIPI DBI type A, fixed E or clocked E mode)<br>- 100: 24-bit MCU interface (MIPI DBI type B)<br>- 110: 24-bit MCU interface (MIPI DBI type A, fixed E or clocked E mode)<br>- 101: reserved<br>- 111: reserved | VDDIO / GND |
| CLK_IN | I | - | Reserved | Open |
| XTAL_OUT | I | - | Crystal inout for System PLL | Open |
| XTAL_IN | I | External CLK | Crystal in for System PLL<br>Frequency range: 8MHz to 30MHz | - |

## 7.4 Interface Logic Pin

**Table 7-4: MCU/RGB Interface Description**

| Pin name | Type | Connect to | Description | When not in use |
|---|---|---|---|---|
| DATA0[53:30] | I/O | MCU/RGB Signals | RGB data for RGB Interface | Open |
| DATA0[23:0] | I/O | | RGB data for RGB Interface<br>MCU data for MCU interface | Open |
| VSYNC / E / WRX | I | | - Vsync for lower RGB interface<br>- E clock signal for MCU interface<br>(This is for MIPI DBI type A interface)<br>- Write enable signal for MCU interface. Enabled when low. (This is for MIPI DBI type B interface) | VDDIO / GND |
| PCLK / RWX / RDX | I | | - PCLK for lower RGB interface<br>- Read/Write selection signal for MCU interface. Read cycle when high, write cycle when low.<br>(This is for MIPI DBI type A interface)<br>- Read enable signal for MCU interface. Enabled when low. (This is for MIPI DBI type B interface.) | VDDIO / GND |
| HSYNC | I | | - Hsync for lower RGB interface | VDDIO / GND |
| DEN; DCX | I | | - Den for lower RGB interface<br>- Data or command signal for MCU interface | VDDIO / GND |
| TE_IN_0 | I | | Input Tearing Effect Signal for MCU | VDDIO / GND |
| TE_OUT_0 | O | | Output Tearing Effect Signal for MCU | Open |
| TE_IN_1 | I | | Reserved | VDDIO / GND |
| TE_OUT_1 | O | | Reserved | Open |

**Table 7-5: SPI Interface Description**

| Pin name | Type | Connect to | Description | When not in use |
|---|---|---|---|---|
| CSX0 | I | SPI Signal | Chip Select for SPI interface | VDDIO |
| SDC | I | | Data or Command for SPI interface (for 8-bit 4 wire) | VDDIO / GND |
| SCK | I | | Serial clock for SPI interface | VDDIO / GND |
| SDI | I | | Serial data input for SPI interface | VDDIO / GND |
| SDO | O | | Serial data output for SPI interface | Open |

# 8 COMMAND TABLE

## 8.1 Local Registers (non-APB) Descriptions

### 8.1.1 RGB Interface Control Register 1

Offset Address

**RICR1** **RGB Interface Control Register 1** **0xB1**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VSA** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x02 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HSA** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x0A | | | | | | | |
| | | | | | | | | |

**Table 8-1: RGB Interface Control Register 1 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **VSA**<br>Bit 15-8 | **VSA** – Vertical Sync Active Period<br><br>These bits specify the Vsync active period. The Hsync active period is from the Vsync falling edge to rising edge, in terms of Hsync lines. It is only used in non-burst mode with Sync pulses.<br>(The minimum value is 1) | Per Application Condition |
| **HSA**<br>Bit 7-0 | **HSA** – Horizontal Sync Active Period<br><br>These bits specify the Hsync active period. The Hsync active period is from the Hsync falling edge to rising edge, in terms of pclk. It is only used in non-burst mode with Sync pulses.<br>(The minimum value is 1) | Per Application Condition |

## 8.1.2 RGB Interface Control Register 2

Offset Address

**RICR2** **RGB Interface Control Register 2** **0xB2**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **VBP[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **HBP[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VBP[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x02 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HBP[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x14 | | | | | | | |
| | | | | | | | | |

**Table 8-2: RGB Interface Control Register 2 Description**

| Name | Description | Setting |
|---|---|---|
| **VBP[15:8]**<br>Bit 31-24 | **VBP** – Vertical Back Porch Period High Byte<br><br>Refer to VBP[7:0] for description | Per Application Condition |
| **HBP[15:8]**<br>Bit 23-16 | **HBP** – Horizontal Back Porch Period High Byte<br><br>Refer to HBP[7:0] for description | Per Application Condition |
| **VBP[7:0]**<br>Bit 15-8 | **VBP** – Vertical Back Porch Period Low Byte<br><br>These bits specify the vertical back porch period in terms of Hsync pulses. The vertical back porch period depends on the video mode setting.<br><br>If the mode is non-burst mode with Sync pulses, it is from the Vsync rising edge to the Hsync of the first line of active display.<br>If the mode is non-burst mode with Sync events or burst mode, it is from the Vsync falling edge to the Hsync of the first line of active display. (The minimum value is 1) | Per Application Condition |
| **HBP[7:0]**<br>Bit 7-0 | **HBP** – Horizontal Back Porch Period Low Byte<br><br>These bits specify the horizontal back porch period in terms of pclk. The horizontal back porch period depends on the non-burst mode setting. | Per Application Condition |

| Name | Description | Setting |
|---|---|---|
| | If the mode is non-burst mode with Sync pulses, it is from the Hsync rising edge to the start of the valid display pixel.<br>If the mode is non-burst mode with Sync events or burst mode, it is from the Hsync falling edge to the start of the valid display pixel.<br>(The minimum value is 1) | |

### 8.1.3 RGB Interface Control Register 3

Offset Address

**RICR3** **RGB Interface Control Register 3** **0xB3**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **VFP[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **HFP[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VFP[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x02 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HFP[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x14 | | | | | | | |
| | | | | | | | | |

**Table 8-3: RGB Interface Control Register 3 Description**

| Name | Description | Setting |
|---|---|---|
| **VFP[15:8]**<br>Bit 31-24 | **VFP** – Vertical Front Porch Period High Byte<br><br>Refer to VFP[7:0] for description | Per Application Condition |
| **HFP[15:8]**<br>Bit 23-16 | **HFP** – Horizontal Front Porch Period High Byte<br><br>Refer to HFP[7:0] for description | Per Application Condition |
| **VFP[7:0]**<br>Bit 15-8 | **VFP** – Vertical Front Porch Period Low Byte<br><br>These bits specify the vertical front porch period in terms of Hsync pulses. The vertical front porch period is from the first Hsync after the last line of active display to the next Vsync falling edge. | Per Application Condition |
| **HFP[7:0]**<br>Bit 7-0 | **HFP** – Horizontal Front Porch Period Low Byte<br><br>These bits specify the horizontal front porch period in terms of pclk. The horizontal front porch period is from the end of the valid display pixel to the next Hsync falling edge. | Per Application Condition |

### 8.1.4 RGB Interface Control Register 4

Offset Address

**RICR4** **RGB Interface Control Register 4** **0xB4**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **HACT[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x07 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HACT[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x80 | | | | | | | |
| | | | | | | | | |

**Table 8-4: RGB Interface Control Register 4 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **HACT**<br>Bit 15-0 | **HACT** – Horizontal Active Period<br><br>These bits specify the horizontal active period in terms of pclk. During the horizontal active period, the den signal should always be high. | Per Application Condition |

### 8.1.5 RGB Interface Control Register 5

Offset Address

**RICR5** **RGB Interface Control Register 5** **0xB5**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VACT[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x04 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VACT[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x38 | | | | | | | |
| | | | | | | | | |

**Table 8-5: RGB Interface Control Register 5 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **VACT**<br>Bit 15-0 | **VACT** – Vertical Active Period<br><br>These bits specify the vertical active period in terms of Hsync pulses. | Per Application Condition |

## 8.1.6 RGB Interface Control Register 6

Offset Address

**RICR6** **RGB Interface Control Register 6** **0xB6**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **VSD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **HSD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x02 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VS_P** | **HS_P** | **PCLK_P** | **SDR** | **RGB_PACK_SEQ** | | **VPF_EXT** | **CBM** |
| TYPE | RW | RW | RW | RW | RW | | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x1 | 0x0 | | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **NVB** | **NVD** | **BLLP** | **VCS** | **VM** | | **VPF** | |
| TYPE | RW | RW | RW | RW | RW | | RW | |
| RESET | 0x0 | 0x0 | 0x1 | 0x0 | 0x1 | | 0x0 | |
| | | | | | | | | |

**Table 8-6: RGB Interface Control Register 6 Description**

| Name | Description | Setting |
|---|---|---|
| **VSD**<br>Bit 31-24 | **VSD** – Vertical Sync Delay<br><br>These bits control the internal pipeline delay of the Vsync input. | Per Application Condition |
| **HSD**<br>Bit 23-16 | **HSD** – Horizontal Sync Delay<br><br>These bits control the internal pipeline delay of the Hsync input. | Per Application Condition |
| **VS_P**<br>Bit 15 | **VS_P** – Vertical Sync Polarity<br><br>This bit control the polarity of the Vsync pulse input. | 0 – Vsync Pulse is active low<br>1 – Vsync Pulse is active high |
| **HS_P**<br>Bit 14 | **HS_P** – Horizontal Sync Polarity<br><br>This bit control the polarity of the Hsync pulse input. | 0 – Hsync Pulse is active low<br>1 – Hsync Pulse is active high |
| **PCLK_P**<br>Bit 13 | **PCLK_P** – Pixel Clock Polarity<br><br>This bit control the polarity of the PCLK input. This bit is valid when **SDR** is 1. | 0 – Data is launch at falling edge, SSD2829 latch data at rising edge<br>1 – Data is launch at rising edge, SSD2829 latch data at falling edge |
| **SDR**<br>Bit 12 | **SDR** - Single Data Rate<br><br>This bit control whether the RGB input is single data rate or dual data rate. | 0 – Data is launch at both rising and falling edge<br>1 – Data is launch at either rising or falling edge, depends on the PCLK_P bit |

| Name | Description | Setting |
|---|---|---|
| **RGB_PACK_SEQ**<br>Bit 11-10 | **RGB_PACK_SEQ** - RGB Packing Sequence<br><br>This is applicable for 1 RGB to 2 DSI_TX output(1 to 2) configurations. | For RGB input and 2 DSI_TX output(1 to 2)<br>0 - Odd/Even split. RGB lower order pixel = pixel[0] on DSI_TX0, RGB higher order pixel = pixel[1] on DSI_TX1.<br>1 - Left/Right split. pixel[0] to pixel[n/2-1] on DSI_TX0, pixel[n/2] to pixel[n-1] on DSI_TX1.<br>2 - Broadcast split. DSI_TX0 is duplicated to DSI_TX1.<br>3 - Reserved |
| **VPF_EXT**<br>Bit 9 | **VPF_EXT** - Video Pixel Format Extension<br><br>This bit is used in conjunction with the **VPF[1:0]** bits to define the output pixel format. | [**VPF_EXT, VPF**]<br>000 - 16-bit<br>001 - 18-bit<br>010 - 18-bit loosely<br>011 - 24-bit<br>100 - Reserved<br>111 - Compressed pixel |
| **CBM**<br>Bit 8 | **CBM** – Compress Burst Mode Control<br><br>If the video mode is burst(VM=0x2) and this bit is 1, MIPITX will send video packet in compressed burst mode (i.e. no blanking packet after horizontal sync packet) | 0 – Video with blanking packet.<br>1 – Video with no blanking packet |
| **NVB**<br>Bit 7 | **NVB** – Non Video Data Burst Mode Control<br><br>This bit specifies how non video data will be interleaved with video data transmission in burst mode. | 0 - Non video data will be transmitted during any BLLP period<br>1 - Non video data will only be transmitted during vertical blanking period |
| **NVD**<br>Bit 6 | **NVD** – Non Video Data Transmission Control<br><br>This bit specifies how non video data will be interleaved with video data transmission.<br>The SSD2829 will send non video data (written from the SPI interface) during the vertical blanking period (non burst mode) or any BLLP period in burst mode (depends on NVB setting). The data can be sent either in high speed mode or low power mode. This bit selects which mode to use. If LP mode is selected, the data lane will enter LP mode for BLLP period, even if there is no non-video data to send.<br>Please note that sending data in LP mode is much slower than HS mode. It is the responsibility of the host processor to make sure that the duration is long enough to finish the data transfer and the timing of Hsync and Vsync is not affected. | 0 – Non video data will be transmitted using HS mode<br>1 – Non video data will be transmitted using LP mode |
| **BLLP**<br>Bit 5 | **BLLP** – Blanking and Low Power Control<br><br>This bit specifies the SSD2829 operation during BLLP period. This bit takes effect only for non burst mode and **NVD** being 0. | 0 – Blanking packet will be sent during BLLP period<br>1 – LP mode will be used during BLLP period |

| Name | Description | Setting |
|---|---|---|
| | When the video mode is burst mode, the SSD2829 will not send any blanking packet during BLLP. It will enter LP mode.<br>When **NVD** is 1 in non burst mode, the SSD2829 will stay in LP mode after sending the non video data (if there is any), until the BLLP period ends.<br><br>When **NVD** is 0 in non burst mode, the SSD2829 will use this bit to decide whether to send blanking packet or enter LP mode after sending non video data (if there is any), until the BLLP period ends.<br>Please note that entering and exiting from LP mode needs more time, as the speed of LP mode is slow. It is the responsibility of the host processor to make sure that the period is long enough to finish the data transfer and the timing of Hsync and Vsync is not affected. | |
| **VCS**<br>Bit 4 | **VCS** – Video Clock Suspend<br><br>This bit specifies how non video data will be interleaved with video data transmission in burst mode.<br>This bit specifies the clock lane behavior | 0 – During burst mode, the clock lane remains in HS mode, when there is no data to transmit. During non burst mode, the clock lane will remain in HS mode all the time.<br>1 – During burst mode, the clock lane enters LP mode when there is no data to transmit. During non burst mode, the clock lane enters LP mode during vertical blanking period. |
| **VM**<br>Bit 3-2 | **VM** – Video Mode<br><br>These bits specify the video mode when RGB interface is selected. | 00 – Non burst mode with sync pulses<br>01 – Non burst mode with sync events<br>10 – Burst mode<br>11 – Reserved |
| **VPF**<br>Bit 1-0 | **VPF** – Video Pixel Format<br><br>This bit is used in conjunction with the **VPF_EXT** bit to define the output pixel format. | [**VPF_EXT, VPF**]<br>000 - 16-bit<br>001 - 18-bit<br>010 - 18-bit loosely<br>011 - 24-bit<br>100 - Reserved<br>111 - Compressed pixel |

**Table 8-7: RGB data arrangement**

| | D23 | D22 | D21 | D20 | D19 | D18 | D17 | D16 | D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 24bpp | R7 | R6 | R5 | R4 | R3 | R2 | R1 | R0 | G7 | G6 | G5 | G4 | G3 | G2 | G1 | G0 | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
| 18bpp | X | X | X | X | X | X | R5 | R4 | R3 | R2 | R1 | R0 | G5 | G4 | G3 | G2 | G1 | G0 | B5 | B4 | B3 | B2 | B1 | B0 |
| 16bpp | X | X | X | X | X | X | X | X | R4 | R3 | R2 | R1 | R0 | G5 | G4 | G3 | G2 | G1 | G0 | B4 | B3 | B2 | B1 | B0 |

## 8.1.7 Configuration Register

Offset Address

**CFR** **Configuration Register** **0xB7**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | **VEN_CTR** | **SCR_EN** | **OTHER_ CMD** | **TXD** | **LPE** | **EOT** | **ECD** |
| TYPE | RO | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x1 | 0x1 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **REN** | **DCS** | **CSS** | **HCLK** | **VEN** | **SLP** | **CKE** | **HS** |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-8: Configuration Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-15 | **Reserved** | Not Applicable |
| **VEN_CTR**<br>Bit 14 | **VEN_CTR** – Video Enable Control<br><br>This bit specifies whether the SSD2829 will extend the internal video enable bit until frame boundary. | 0 – Internal video enable follows the **VEN** bit<br>1 – Internal video enable extends until the frame boundary |
| **SCR_EN**<br>Bit 13 | **SCR_EN** – Scrambler Mode Enable<br><br>This bit specifies whether the SSD2829 will send the long packets with scrambled data. SSD2829 will send the Scrambling Mode Packet prior to the Long packet. | 0 – Scrambling is disable<br>1 – Scrambling is enable |
| **OTHER_C MD**<br>Bit 12 | **OTHER_CMD** – Other Command<br><br>This bit defines how DCS, Generic, PPS or Compression Mode packet is sent. | 0 – **DCS** bit defines DCS or Generic command.<br>1 – **DCS** bit defines PPS or Compression mode packet. |
| **TXD**<br>Bit 11 | **TXD** –Transmit Disable<br><br>This bit specifies whether the SSD2829 will disable the sending of MIPI Packets stored in the buffers. Software can enable TXD, fill out the buffers and then disable it to send all packets out in 1 burst. | 0 – Transmit on<br>1 – Transmit halt |
| **LPE** | **LPE** –Long Packet Enable | 0 – Short Packet |

| Name | Description | Setting |
|---|---|---|
| Bit 10 | This bit specifies whether the SSD2829 will send out a Generic Long Write Packet or Generic Short Write Packet when the payload is no more than 2 bytes.<br>It also specifies whether the SSD2829 will send out a DCS Long Write Packet or DCS Short Write Packet when the payload is no more than 1 byte. | 1 – Long Packet |
| **EOT**<br>Bit 9 | **EOT** – EOT Packet Enable<br><br>This bit specifies whether the SSD2829 will send out the EOT packet at the end of HS transmission or not. | 0 – Do not send<br>1 – Send |
| **ECD**<br>Bit 8 | **ECD** – ECC CRC Check Disable<br><br>This bit specifies whether SSD2829 will perform ECC and CRC checking for the packets received from the MIPI slave. | 0 – Enable<br>1 – Disable |
| **REN**<br>Bit 7 | **REN** – Read Enable<br><br>This bit specifies whether the next operation is a write or read operation. | 0 – Write operation<br>1 – Read operation |
| **DCS**<br>Bit 6 | **DCS** – DCS or Generic<br><br>This bit specifies whether the packet to be sent is DCS packet or generic packet. This bit applies for both write and read operation. When **OTHER_CMD** bit is set, this bit specifies whether the packet to be sent is PPS or Compress Mode packet. | When **OTHER_CMD** is 0,<br><br>0 – Generic packet (The packet can be any one of Generic Long Write, Generic Short Write, Generic Read packet, depending on the configuration.)<br>1 – DCS packet (The packet can be any one of DCS Long Write, DCS Short Write, DCS Read packet, depending on the configuration.)<br><br>When **OTHER_CMD** is 1,<br><br>0 – Picture Parameter Setting Packet<br>1 – Compress Mode Packet |
| **CSS**<br>Bit 5 | **CSS** – Clock Source Select<br><br>This bit selects the clock source for the PLL. The **CSS** setting should be programmed only when **PEN** is 0. It has no effect when PEN is 1. | 0 – The clock source is XTAL_IN<br>1 – The clock source is pclk |
| **HCLK**<br>Bit 4 | **HCLK** – High Speed Clock Disable<br><br>This bit controls the clock lane behavior during the reverse direction communication. This bit takes effect only when **CKE** is 0 and **VEN** is 0. | 0 – HS clock is enabled<br>1 – HS clock is disabled |
| **VEN**<br>Bit 3 | **VEN** – Video Mode Enable | 0 – Video mode is disabled<br>1 – Video mode is enabled |

| Name | Description | Setting |
|---|---|---|
| | This bit controls the video mode operation. Only after this bit is set to 1, video mode is enabled. | |
| **SLP** Bit 2 | **SLP** – Sleep Mode Enable<br><br>This bit controls the sleep mode operation. When this bit is set to 1, the HS bit will be cleared to 0 automatically. | 0 – Sleep mode is disabled<br>1 – Sleep mode is enabled Only the register interface is active |
| **CKE** Bit 1 | **CKE** – Clock Lane Enable<br><br>This bit controls the clock lane mode when data lane enters LP mode. | 0 – Clock lane will enter LP mode, if it is not in reverse direction communication.<br>Clock lane will follow the setting of HCLK, if it is in reverse direction communication.<br>1 – Clock lane will enter HS mode for all the cases |
| **HS** Bit 0 | **HS** – High Speed Mode<br><br>This bit controls whether the SSD2829 is using HS or LP mode to send data.<br>This bit can be affected by the SLP bit value. | 0 – LP mode<br>1 – HS mode |

## 8.1.8 Virtual Channel Control Register

Offset Address

**VCCR** **Virtual Channel Control Register** **0xB8**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VCM** | | **VCE** | | **VC2** | | **VC1** | |
| TYPE | RW | | RW | | RW | | RW | |
| RESET | 0x1 | | 0x0 | | 0x1 | | 0x1 | |
| | | | | | | | | |

**Table 8-9: Virtual Channel Control Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-8 | **Reserved** | Not Applicable |
| **VCM**<br>Bit 7-6 | **VCM** – Virtual Channel ID for Maximum Return Size Packet<br><br>These bits specify the VC ID for the Maximum Return Size Packet sent by SSD2829.<br>This register field is included as the VC ID for this packet might be different from the VC ID for the packets carrying the actual data. | Per Application Condition |
| **VCE**<br>Bit 5-4 | **VCE** – Virtual Channel ID for EOT Packet<br><br>These bits specify the VC ID for the EOT Packet sent by SSD2829.<br>This register field is included as the VC ID for this packet might be different from the VC ID for the packets carrying the actual data. | Per Application Condition |
| **VC2**<br>Bit 3-2 | **VC2** – Virtual Channel ID for SPI Interface<br><br>These bits specify the VC ID for the packets written in through the SPI interface, when the interface setting is RGB + SPI(if_sel[0] = 0). | Per Application Condition |
| **VC1**<br>Bit 1-0 | **VC1** – Virtual Channel ID for RGB and MCU Interface<br><br>These bits specify the VC ID for the packets written in through the RGB | Per Application Condition |

| Name | Description | Setting |
|---|---|---|
| | interface, when the interface is RGB + SPI(if_sel[0] = 0).<br>These bits specify the VC ID for the packets written in through the MCU interface, when the interface setting is MCU(if_sel[0] = 1). | |

### 8.1.9 PLL Control Register

Offset Address

**PCR** **PLL Control Register** **0xB9**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | | | | | | | | **PEN** |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-10: PLL Control Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-1 | **Reserved** | Not Applicable |
| **PEN**<br>Bit 0 | **PEN** – PLL Enable<br><br>This bit controls the PLL operation. | 0 – PLL power down<br>1 – PLL enable |

Note: Frequency of PLL can only be changed during PEN=0

## 8.1.10 PLL Configuration Register

Offset Address

**PCFR** **PLL Configuration Register** **0xBA**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **FR** | | **PLL_TEST** | **MS** | | | | |
| TYPE | RW | | RW | RW | | | | |
| RESET | 0x3 | | 0x0 | 0x01 | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **NS** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x20 | | | | | | | |
| | | | | | | | | |

**Table 8-11: PLL Configuration Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **FR**<br>Bit 15-14 | **PEN** – Frequency Range<br><br>These bits select the range of the output clock. | 00 – 62.5 to 125<br>01 – 126 to 250<br>10 – 251 to 500<br>11 – 501 to 1250 |
| **PLL_TEST**<br>Bit 13 | **PLL_TEST** – PLL Test Mode<br><br>This bit set the TEST_MODE pin of the PLL. It should be set to 0 in normal mode. | Not Applicable |
| **MS**<br>Bit 12-8 | **MS** – PLL Divider<br><br>These bits specify the PLL pre-divider value, **MS**. | 0x00 - Reserved<br>0x01 - MS=1<br>0x02 - MS=2<br>…<br>0x1F - MS=31 |
| **NS**<br>Bit 7-0 | **NS** – PLL Multiplier<br><br>These bits specify the PLL output frequency multiplier value, **NS**. | 0x00 - NS=1<br>0x01 - NS=1<br>0x02 - NS=2<br>…<br>0xFF - NS=255 |

e.g. XTAL_IN = 20MHz, 0xBAh = 0xC132h

PLL = 50 x 20 / 1 = 1Gbps

## 8.1.11 Clock Control Register

Offset Address

**CCR** **Clock Control Register** **0xBB**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | **BYP_BIT _DIV** | | | | |
| TYPE | RO | RO | RO | RW | RW | | | |
| RESET | 0x0 | 0x0 | 0x0 | 0x1 | 0x6 | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x07 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **TX_LPD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x03 | | | | | | | |
| | | | | | | | | |

**Table 8-12: Clock Control Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved** Bit 31-29 | **Reserved** | Not Applicable |
| **Reserved** Bit 28 | **Reserved** | Not Applicable |
| **Reserved** Bit 27-24 | **Reserved** | Not Applicable |
| **Reserved** Bit 23-16 | **Reserved** | Not Applicable |
| **Reserved** Bit 15-8 | **Reserved** | Not Applicable |
| **TX_LPD** Bit 7-0 | **TX_LPD** – LP Clock Divider for MIPITX<br><br>These bits give the divider value for generating the LP mode clock from the system clock. | 0x0 – Divide by 1<br>0x1 – Divide by 2<br>0x2 – Divide by 3<br>…<br>0x3F – Divide by 64 |

Remark: e.g. LPD = 0x9

PLL = 1Gbps

LP clock = 1Gbps / (LPD+1) / 8 = 1000 / 10 / 8 = 12.5MHz

## 8.1.12 Packet Size Control Register 1

Offset Address

**PSCR1** **Packet Size Control Register 1** **0xBC**

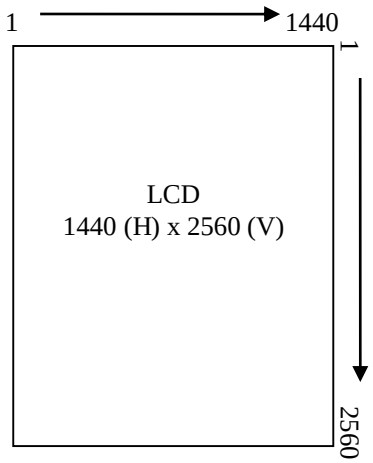
| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **TDC_L[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **TDC_L[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-13: Packet Size Register 1 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **TDC_L**<br>Bit 15-0 | **TDC_L** – Transmit Data Count Low<br><br>These bits, together with **TDC_H**, forms the 32 bit value for **TDC.**<br><br>**TDC** set the total number of data bytes to be transmitted by the SSD2829 in the next operation. The SSD2829 will use the value in this field to decide what type of packet to send out.<br><br>The settings of **TDC** and **PST** will configure the transfer mode into partition<br>and non-partition mode when the command is 0x2C or 0x3C.<br><br>Partition mode(**TDC** > **PST**) - For DCS Long Write packet with DCS command being 0x2C or 0x3C, there is no<br>limit in the maximum number of bytes to be transmitted in 1 write. The **PST** value can<br>be set to maximum of 8191 bytes. The SSD2829 will auto insert 0x3C command at these boundaries. This is valid in RGB+SPI mode. | Per Application Condition |

| Name | Description | Setting |
|---|---|---|
| | Non-Partition mode(**TDC** <= **PST**) For DCS Long Write packet with DCS command being 0x2C or 0x3C, the maximum number of bytes to be transmitted in 1 write is 10560 bytes for MIPITX0 and 8544 bytes for MIPITX1. In this mode, the PST value is the same or greater than the TDC value. | |

e.g.




Total transmitted data per frame = 1440 x 2560 x 3 = 11,059,200 (= 0xA8C000h)

1 line data = 1440 x 3 =4,320 (=0x10E0h)

0xBC = 0xC000 //1 frame RAM (lower 16bit)

0xBD = 0x00A8 //1 frame RAM (higher 16bit)

0xBE = 0x10E0 //1 line RAM

## 8.1.13 Packet Size Control Register 2

Offset Address

**PSCR2** **Packet Size Control Register 2** **0xBD**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **TDC_H[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **TDC_H[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-14: Packet Size Register 2 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **TDC_H**<br>Bit 15-0 | **TDC_H** – Transmit Data Count High<br><br>Please see TDC_L for description. | Per Application Condition |

## 8.1.14 Packet Size Control Register 3

Offset Address

**PSCR3** **Packet Size Control Register 3** **0xBE**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | **PST[12:8]** | | | | |
| TYPE | RO | RO | RO | RW | | | | |
| RESET | 0x0 | 0x0 | 0x0 | 0x1F | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **PST[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0xFF | | | | | | | |
| | | | | | | | | |

**Table 8-15: Packet Size Register 3 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-13 | **Reserved** | Not Applicable |
| **PST**<br>Bit 12-0 | **PST** – Packet Size Threshold<br><br>These bits give the threshold value for partitioning the incoming long packet data into smaller packets. The partitioning only applies to the DCS Long Write packet with DCS command being 0x2C or 0x3C in Command mode(if_sel[1:0]=01). The payload will be partitioned into multiple packets. The PST represents the threshold in term of bytes. | Per Application Condition |

## 8.1.15 Packet Drop Register

Offset Address

**PDR** **Packet Drop Register** **0xBF**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **PD[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **PD[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-16: Packet Drop Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **PD**<br>Bit 15-0 | **PD** – Packet Drop<br><br>This register is not a true register. It is the entry point for the internal buffer. The payload of the generic packets (Generic Short Write, Generic Long Write, Generic Read, DCS Short Write, DCS Long Write, DCS Read, PPS Long Write, Compress Mode Write).<br><br>The application processor can treat this register as an FIFO and continuously write data into it.<br><br>Since the register is only the entry point of the internal buffer, the application processor is not able to read the data written into the buffer. | Per Application Condition |

## 8.1.16 Operational Control Register

Offset Address

**OCR** **Operational Control Register** **0xC0**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | **SWR** |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | | | | | | | | **COP** |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-17: Operational Control Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-9 | **Reserved** | Not Applicable |
| **SWR**<br>Bit 8 | **SWR -** Software Reset<br><br>Writing a '1' to this bit will reset the entire module. This bit will be cleared after the reset is completed.<br>Writing a '1' to this bit will cause the MIPI link enters TX stop state immediately and any outgoing MIPI packet will be terminated immediately. | Per Application Condition |
| **Reserved**<br>Bit 7-1 | **Reserved** | Not Applicable |
| **COP**<br>Bit 0 | **COP –** Cancel Operation<br><br>This bit is to cancel the current operation. When this bit is set to 1, the SSD2829 will still finish transmitting the current packet. (Otherwise, the serial link operation will lose sync.) Afterwards, the SSD2829 will stop any further transmission. It will clear its internal buffer such that all the data being written in and not sent out yet will be cleared. It will also bring the state machine to its initial state.<br>Once this process is finished, the **COP** bit will be automatically set to 0. At the same time, the **PO** bit of the status register will be | Per Application Condition |

| Name | Description | Setting |
|---|---|---|
| | set to 1 too. At this stage, there is no data in the internal buffer. The application processor can start a new operation. This operation is not valid in video mode(**VEN=1)**. | |

## 8.1.17 Maximum Return Size Register

Offset Address

**MRSR** **Maximum Return Size Register** **0xC1**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **MRS[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **MRS[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x01 | | | | | | | |
| | | | | | | | | |

**Table 8-18: Maximum Return Size Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **MRS**<br>Bit 15-0 | **MRS –** Maximum Return Size<br><br>These bits set the maximum return size of the read response packet returned by the MIPI slave. The SSD2829 will automatically send out the Set Maximum Return Size packet using the value in this field, before every read operation. It informs the MIPI slave about the limit of the SSD2829. The application processor does not need to program the register before every read operation, if the maximum return size does not change. The Set Maximum Return Size packet will always be sent. | Per Application Condition |

## 8.1.18 Return Data Count Register

Offset Address

**RDCR** **Return Data Count Register** **0xC2**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **RDC1[15:8]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **RDC1[7:0]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **RDC0[15:8]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **RDC0[7:0]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-19: Return Data Count Register Description**

| Name | Description | Setting |
|---|---|---|
| **RDC1**<br>Bit 31-16 | **RDC1 –** Return Data Count from MIPITX1<br><br>These bits reflect the number of data bytes received from the MIPI slave read response packet from MIPITX1. This register can only be updated by the SSD2829. | Per Application Condition |
| **RDC0**<br>Bit 15-0 | **RDC0 –** Return Data Count from MIPITX0<br><br>These bits reflect the number of data bytes received from the MIPI slave read response packet from MIPITX0. This register can only be updated by the SSD2829. | Per Application Condition |

## 8.1.19 Acknowledge Response Status Register

Offset Address

**ARSR** **Acknowledge Response Status Register** **0xC3**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **ACK1[15:8]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **ACK1[7:0]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **ACK0[15:8]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **ACK0[7:0]** | | | | | | | |
| TYPE | RO | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-20: Acknowledge Response Status Register Description**

| Name | Description | Setting |
|---|---|---|
| **ACK1**<br>Bit 31-16 | **ACK1 –** ACK Response from MIPITX1<br><br>These bits contain the ACK response from the MIPI slave from MIPITX1. The register will be updated when ACK with Error Report packet is received. Otherwise, the value will be set to 0. The bits in this register follow the definition in MIPI DSI.<br>This register can only be updated by the SSD2829. | Per Application Condition |
| **ACK0**<br>Bit 15-0 | **ACK0 –** ACK Response from MIPITX0<br><br>These bits contain the ACK response from the MIPI slave from MIPITX0. The register will be updated when ACK with Error Report packet is received. Otherwise, the value will be set to 0. The bits in this register follow the definition in MIPI DSI.<br>This register can only be updated by the SSD2829. | Per Application Condition |

## 8.1.20 Line Control Register

Offset Address

**LCR** **Line Control Register** **0xC4**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | **RT1** | **RTB1** | **FBC1** | **FBT1** | **FBW1** |
| TYPE | RO | RO | RO | RWAC | RWAC | RWAC | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | | | | **RT0** | **RTB0** | **FBC0** | **FBT0** | **FBW0** |
| TYPE | RO | RO | RO | RWAC | RWAC | RWAC | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-21: Line Control Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-21 | **Reserved** | Not Applicable |
| **RT1**<br>Bit 20 | **RT1 –** Reset Trigger for MIPITX1<br><br>This bit is to send a Reset Trigger Message. When this bit is set to 1, the SSD2829 will send a Reset Trigger Message. It is recommended to enter LP mode and send this trigger message. If this bit is programmed during vertical active data is being sent on MIPI link, the reset trigger will be delayed to next vertical blanking period so that the reset trigger message will not disturb the video timing on the MIPI link. Once the Reset Trigger Message is sent out, **RT1** bit will be automatically set to 0. | 0 – Do not send<br>1 – Send Reset Trigger |
| **RTB1**<br>Bit 19 | **RTB1 –** Register Triggered BTA for MIPITX1<br><br>This bit automatically perform Bus Turnaround(BTA) when link is not used. When bus is returned back from the slave, the link will remains in Low Power state until a new request come in where HS bit determination the transfer mode. | 0 – Do not send<br>1 – Send BTA |
| **FBC1**<br>Bit 19 | **FBC1 –** Force Bus Contention for MIPITX1<br><br>This bit controls whether to force a bus contention on the | 0 – Do not force<br>1 – Force Bus Contention |

| Name | Description | Setting |
|---|---|---|
| | data lane. This bit will be changed to 0, after the bus contention is not detected. | |
| **FBT1**<br>Bit 18 | **FBT1 –** Force Bus Turnaround Tearing for MIPITX1<br><br>This bit controls whether to perform automatic BTA after previous BTA so as to get the TE response from MIPI slave. | 0 – Do not force<br>1 – Force BTA for Tearing |
| **FBW1**<br>Bit 17 | **FBW1 –** Force Bus Turnaround after write for MIPITX1<br><br>This bit controls whether to automatically generate a BTA after a write operation. It is only valid for write operation.<br>After performing BTA, the bus authority has been passed to the MIPI slave. The SSD2829 is not able to send any data to the MIPI slave before the bus authority is passed back. It is the responsibility of the application processor to check the status of the bus before sending any data. | 0 – Do not force<br>1 – Force BTA after write |
| **Reserved**<br>Bit 16-5 | **Reserved** | Not Applicable |
| **RT0**<br>Bit 4 | **RT0 –** Reset Trigger for MIPITX0<br><br>This bit is to send a Reset Trigger Message. When this bit is set to 1, the SSD2829 will send a Reset Trigger Message. It is recommended to enter LP mode and send this trigger message. If this bit is programmed during vertical active data is being sent on MIPI link, the reset trigger will be delayed to next vertical blanking period so that the reset trigger message will not disturb the video timing on the MIPI link. Once the Reset Trigger Message is sent out, **RT1** bit will be automatically set to 0. | 0 – Do not send<br>1 – Send Reset Trigger |
| **RTB0**<br>Bit 3 | **RTB0 –** Register Triggered BTA for MIPITX0<br><br>This bit automatically perform Bus Turnaround(BTA) when link is not used.<br>When bus is returned back from the slave, the link will remains in Low Power state until a new request come in where HS bit determination the transfer mode. | 0 – Do not send<br>1 – Send BTA |
| **FBC0**<br>Bit 2 | **FBC0 –** Force Bus Contention for MIPITX0<br><br>This bit controls whether to force a bus contention on the<br>data lane. This bit will be changed to 0, after the bus contention is not detected. | 0 – Do not force<br>1 – Force Bus Contention |
| **FBT0**<br>Bit 1 | **FBT0 –** Force Bus Turnaround Tearing for MIPITX0<br><br>This bit controls whether to perform automatic BTA after previous BTA so as to get the TE response from MIPI slave. | 0 – Do not force<br>1 – Force BTA for Tearing |

| Name | Description | Setting |
|---|---|---|
| **FBW0**<br>Bit 0 | **FBW0 –** Force Bus Turnaround after write for MIPITX0<br><br>This bit controls whether to automatically generate a BTA after a write operation. It is only valid for write operation.<br>After performing BTA, the bus authority has been passed to the MIPI slave. The SSD2829 is not able to send any data to the MIPI slave before the bus authority is passed back. It is the responsibility of the application processor to check the status of the bus before sending any data. | 0 – Do not force<br>1 – Force BTA after write |

## 8.1.21 Interrupt Control Register

Offset Address

**ICR** **Interrupt Control Register** **0xC5**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **CBEE1** | **CBAE1** | | | | | **MLEE1** | **MLAE1** |
| TYPE | RW | RW | RO | RO | RO | RO | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | **LPTOE1** | **HSTOE1** | | **ARRE1** | **BTARE1** | | **RDRE1** |
| TYPE | RO | RW | RW | RO | RW | RW | RO | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **CBEE0** | **CBAE0** | | | | | **MLEE0** | **MLAE0** |
| TYPE | RW | RW | RO | RO | RO | RO | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **PLSE** | **LPTOE0** | **HSTOE0** | | **ARRE0** | **BTARE0** | | **RDRE0** |
| TYPE | RW | RW | RW | RO | RW | RW | RO | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-22: Interrupt Control Register Description**

| Name | Description | Setting |
|---|---|---|
| **CBEE1**<br>Bit 31 | **CBEE1 –** Command Buffer Empty Enable for MIPITX1<br><br>This bit enables the mapping of **CBE1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **CBAE1**<br>Bit 30 | **CBAE1 –** Command Buffer Available Enable for MIPITX1<br><br>This bit enables the mapping of **CBA1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **Reserved**<br>Bit 29-26 | **Reserved** | Not Applicable |
| **MLEE1**<br>Bit 25 | **MLEE1 –** MCU Long Buffer Empty Enable for MIPITX1<br><br>This bit enables the mapping of **MLE1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **MLAE1**<br>Bit 24 | **MLAE1 –** MCU Long Buffer Available Enable for MIPITX1<br><br>This bit enables the mapping of **MLA1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **Reserved**<br>Bit 23 | **Reserved** | Not Applicable |
| **LPTOE1**<br>Bit 22 | **LPTOE1** – LP RX Time Out Enable for MIPITX1 | 0 – Do not enable<br>1 – Enable |

| Name | Description | Setting |
|---|---|---|
| | This bit enables the mapping of **LPTO1** interrupt to the interrupt pin, INT_B. | |
| **HSTOE1**<br>Bit 21 | **HSTOE1** – HP TX Time Out Enable for MIPITX1<br><br>This bit enables the mapping of **HSTO1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **Reserved**<br>Bit 20 | **Reserved** | Not Applicable |
| **ARRE1**<br>Bit 19 | **ARRE1** – ACK Response Ready Enable for MIPITX1<br><br>This bit enables the mapping of **ARR1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **BTARE1**<br>Bit 18 | **BTARE1** – Bus Turnaround Response Enable for MIPITX1<br><br>This bit enables the mapping of **BTAR1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **Reserved**<br>Bit 17 | **Reserved** | Not Applicable |
| **RDRE1**<br>Bit 16 | **RDRE1** – Read Data Ready Enable for MIPITX1<br><br>This bit enables the mapping of **RDR1** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **CBEE0**<br>Bit 15 | **CBEE0 –** Command Buffer Empty Enable for MIPITX0<br><br>This bit enables the mapping of **CBE0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **CBAE0**<br>Bit 14 | **CBAE0 –** Command Buffer Available Enable for MIPITX0<br><br>This bit enables the mapping of **CBA0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **Reserved**<br>Bit 13-10 | **Reserved** | Not Applicable |
| **MLEE0**<br>Bit 9 | **MLEE0 –** MCU Long Buffer Empty Enable for MIPITX0<br><br>This bit enables the mapping of **MLE0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **MLAE0**<br>Bit 8 | **MLAE0 –** MCU Long Buffer Available Enable for MIPITX0<br><br>This bit enables the mapping of **MLA0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **PLSE**<br>Bit 7 | **PLSE** – PLL Lock Status Enable<br><br>This bit enables the mapping of **PLS** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **LPTOE0**<br>Bit 6 | **LPTOE0** – LP RX Time Out Enable for MIPITX0 | 0 – Do not enable<br>1 – Enable |

| Name | Description | Setting |
|---|---|---|
| | This bit enables the mapping of **LPTO0** interrupt to the interrupt pin, INT_B. | |
| **HSTOE0**<br>Bit 5 | **HSTOE0** – HP TX Time Out Enable for MIPITX0<br><br>This bit enables the mapping of **HSTO0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **Reserved**<br>Bit 4 | **Reserved** | Not Applicable |
| **ARRE0**<br>Bit 3 | **ARRE0** – ACK Response Ready Enable for MIPITX0<br><br>This bit enables the mapping of **ARR0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **BTARE0**<br>Bit 2 | **BTARE0** – Bus Turnaround Response Enable for MIPITX0<br><br>This bit enables the mapping of **BTAR0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |
| **Reserved**<br>Bit 1 | **Reserved** | Not Applicable |
| **RDRE0**<br>Bit 0 | **RDRE0** – Read Data Ready Enable for MIPITX0<br><br>This bit enables the mapping of **RDR0** interrupt to the interrupt pin, INT_B. | 0 – Do not enable<br>1 – Enable |

## 8.1.22 Interrupt Status Register

Offset Address

**ISR** **Interrupt Status Register** **0xC6**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **CBE1** | **CBA1** | | | **CLS1** | **DLS1** | **MLE1** | **MLA1** |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | **LPTO1** | **HSTO1** | **ATR1** | **ARR1** | **BTAR1** | | **RDRE1** |
| TYPE | RO | RESW1C | RESW1C | RESW1C | RESW1C | RESW1C | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **CBE0** | **CBA0** | | | **CLS0** | **DLS0** | **MLE0** | **MLA0** |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **PLS** | **LPTO0** | **HSTO0** | **ATR0** | **ARR0** | **BTAR0** | | **RDR0** |
| TYPE | RO | RESW1C | RESW1C | RESW1C | RESW1C | RESW1C | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-23: Interrupt Status Register Description**

| Name | Description | Setting |
|---|---|---|
| **CBE1**<br>Bit 31 | **CBE1 –** Command Buffer Empty for MIPITX1<br><br>This bit reflects the status of the internal command buffer of the SPI/MCU interface. If the command buffer is empty, this bit will be set to 1. The application processor can write up to the maximum size of the command buffer. | 0 – The command buffer is not empty<br>1 – The command buffer is empty |
| **CBA1**<br>Bit 30 | **CBA1 –** Command Buffer Available for MIPITX1<br><br>This bit reflects the status of the internal command buffer of the SPI/MCU interface. If the command buffer is not full, this bit will be set to 1. The application processor can write at least 1 packet to the command buffer. | 0 – The command buffer is not available<br>1 – The command buffer is available |
| **Reserved**<br>Bit 29-28 | **Reserved** | Not Applicable |
| **CLS1**<br>Bit 27 | **CLS1 –** Clock Lane Status for MIPITX1<br><br>This bit reflects the status at the MIPI Clock lane in MIPITX1. | 0 – Clock lane is not in LP-11<br>1 – Clock lane is in LP-11 |
| **DLS1**<br>Bit 26 | **DLS1 –** Data Lane Status for MIPITX1<br><br>This bit reflects the status at the MIPI Clock lane in MIPITX1. | 0 – Data lanes are not in LP-11<br>1 – Data lanes are in LP-11 |
| **MLE1**<br>Bit 25 | **MLE1 –** MCU Long Buffer Empty for MIPITX1 | 0 – The long buffer is not empty<br>1 – The long buffer is empty |

| Name | Description | Setting |
|---|---|---|
| | This bit reflects the status of the internal long buffer of the MCU interface. If the long buffer is empty, this bit will be set to 1. The application processor can write up to the maximum size of the long buffer for DCS command 0x2C and 0x3C. | |
| **MLA1**<br>Bit 24 | **MLA1 –** MCU Long Buffer Available for MIPITX1<br><br>This bit reflects the status of the internal long buffer of the MCU interface. If the long buffer is not full, this bit will be set to 1. The application processor can write at least 1<br>packet to the long buffer for DCS command 0x2C and 0x3C. | 0 – The long buffer is not available<br>1 – The long buffer is available |
| **Reserved**<br>Bit 23 | **Reserved** | Not Applicable |
| **LPTO1**<br>Bit 22 | **LPTO1** – LP RX Time Out for MIPITX1<br><br>This bit reflects the status of the LP RX timer.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – The LP RX timer has expired<br>1 – The LP RX timer has not expired |
| **HSTO1**<br>Bit 21 | **HSTO1** – HP TX Time Out for MIPITX1<br><br>This bit reflects the status of the HS TX timer.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – The HS TX timer has expired<br>1 – The HS TX timer has not expired |
| **ATR1**<br>Bit 20 | **ATR1** – ACK Trigger Response for MIPITX1<br><br>This bit reflects whether the ACK trigger message has been received or not.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – ACK trigger message has not been received<br>1 – ACK trigger message has been received |
| **ARR1**<br>Bit 19 | **ARR1** – ACK Response Ready for MIPITX1<br><br>This bit reflects whether the ACK response has been received or not. The ACK response can be an ACK trigger message or ACK with Error Report packet.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – Response has not been received<br>1 – Response has been received |
| **BTAR1**<br>Bit 18 | **BTAR1** – Bus Turnaround Response for MIPITX1<br><br>This bit reflects the data lane status after SSD2829 has made a BTA.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – The MIPI slave has not passed the lane authority back<br>1 – The MIPI slave has passed the lane authority back |
| **Reserved**<br>Bit 17 | **Reserved** | Not Applicable |
| **RDR1** | **RDR1** – Read Data Ready for MIPITX1 | 0 – Not ready |

| Name | Description | Setting |
|---|---|---|
| Bit 16 | This bit reflects whether the data from the MIPI slave is ready for read by the application processor. This bit is valid only during the read operation.<br><br>This bit will be automatically cleared when all the received data are read out. | 1 – Ready |
| **CBE0**<br>Bit 15 | **CBE0 –** Command Buffer Empty for MIPITX0<br><br>This bit reflects the status of the internal command buffer of the SPI/MCU interface. If the command buffer is empty, this bit will be set to 1. The application processor can write up to the maximum size of the command buffer. | 0 – The command buffer is not empty<br>1 – The command buffer is empty |
| **CBA0**<br>Bit 14 | **CBA0 –** Command Buffer Available for MIPITX0<br><br>This bit reflects the status of the internal command buffer of the SPI/MCU interface. If the command buffer is not full, this bit will be set to 1. The application processor can write at least 1 packet to the command buffer. | 0 – The command buffer is not available<br>1 – The command buffer is available |
| **Reserved**<br>Bit 13-12 | **Reserved** | Not Applicable |
| **CLS0**<br>Bit 11 | **CLS0 –** Clock Lane Status for MIPITX0<br><br>This bit reflects the status at the MIPI Clock lane in MIPITX0. | 0 – Clock lane is not in LP-11<br>1 – Clock lane is in LP-11 |
| **DLS0**<br>Bit 10 | **DLS0 –** Data Lane Status for MIPITX0<br><br>This bit reflects the status at the MIPI Clock lane in MIPITX0. | 0 – Data lanes are not in LP-11<br>1 – Data lanes are in LP-11 |
| **MLE0**<br>Bit 9 | **MLE0 –** MCU Long Buffer Empty for MIPITX0<br><br>This bit reflects the status of the internal long buffer of the MCU interface. If the long buffer is empty, this bit will be set to 1. The application processor can write up to the maximum size of the long buffer for DCS command 0x2C and 0x3C. | 0 – The long buffer is not empty<br>1 – The long buffer is empty |
| **MLA0**<br>Bit 8 | **MLA0 –** MCU Long Buffer Available for MIPITX0<br><br>This bit reflects the status of the internal long buffer of the MCU interface. If the long buffer is not full, this bit will be set to 1. The application processor can write at least 1 packet to the long buffer for DCS command 0x2C and 0x3C. | 0 – The long buffer is not available<br>1 – The long buffer is available |
| **PLS**<br>Bit 7 | **PLS** – PLL Lock Status<br><br>This bit reflects the status of the PLL. Before the PLL is locked, the whole system is running at the reference clock input of the PLL, as the PLL | 0 – PLL is not locked<br>1 – PLL is locked |

| Name | Description | Setting |
|---|---|---|
| | has no output before getting lock. Hence, the application processor must access the registers using slow speed. | |
| **LPTO0**<br>Bit 6 | **LPTO0** – LP RX Time Out for MIPITX0<br><br>This bit reflects the status of the LP RX timer.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – The LP RX timer has expired<br>1 – The LP RX timer has not expired |
| **HSTO0**<br>Bit 5 | **HSTO0** – HP TX Time Out for MIPITX0<br><br>This bit reflects the status of the HS TX timer.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – The HS TX timer has expired<br>1 – The HS TX timer has not expired |
| **ATR0**<br>Bit 4 | **ATR0** – ACK Trigger Response for MIPITX0<br><br>This bit reflects whether the ACK trigger message has been received or not.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – ACK trigger message has not been received<br>1 – ACK trigger message has been received |
| **ARR0**<br>Bit 3 | **ARR0** – ACK Response Ready for MIPITX0<br><br>This bit reflects whether the ACK response has been received or not. The ACK response can be an ACK trigger message or ACK with Error Report packet.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – Response has not been received<br>1 – Response has been received |
| **BTAR0**<br>Bit 2 | **BTAR0** – Bus Turnaround Response for MIPITX0<br><br>This bit reflects the data lane status after SSD2829 has made a BTA.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – The MIPI slave has not passed the lane authority back<br>1 – The MIPI slave has passed the lane authority back |
| **Reserved**<br>Bit 1 | **Reserved** | Not Applicable |
| **RDR1**<br>Bit 0 | **RDR0** – Read Data Ready for MIPITX0<br><br>This bit reflects whether the data from the MIPI slave is ready for read by the application processor. This bit is valid only during the read operation.<br><br>This bit will be automatically cleared when all the received data are read out. | 0 – Not ready<br>1 – Ready |

## 8.1.23 Error Status Register

Offset Address

**ESR** **Error Status Register** **0xC7**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | **CRCE1** | **ECCE2_1** | **ECCE1_1** |
| TYPE | RO | RO | RO | RO | RO | RESW1C | RESW1C | RESW1C |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **CBO1** | | | **MLO1** | | **CONT1** | | **VMM1** |
| TYPE | RESW1C | RO | RO | RESW1C | RO | RO | RO | RESW1C |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | **CRCE0** | **ECCE2_0** | **ECCE1_0** |
| TYPE | RO | RO | RO | RO | RO | RESW1C | RESW1C | RESW1C |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **CBO0** | | | **MLO0** | | **CONT0** | | **VMM0** |
| TYPE | RESW1C | RO | RO | RESW1C | RO | RO | RO | RESW1C |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-24: Error Status Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-27 | **Reserved** | Not Applicable |
| **CRCE1**<br>Bit 26 | **CRCE1 –** CRC Error for MIPITX1<br><br>This bit reflects the status of CRC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** bit is set to 0. Once a CRC error occurs, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – No CRC error since this bit is cleared<br>1 – At least 1 CRC error since this bit is cleared |
| **ECCE2_1**<br>Bit 25 | **ECCE2_1 –** ECC Multi Bit Error for MIPITX1<br><br>This bit reflects the status of ECC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** bit is set to 0. Once an ECC multi-bit error occurs, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – No ECC multi-bit error since this bit is cleared<br>1 – At least 1 ECC multi-bit error since this bit is cleared |
| **ECCE1_1**<br>Bit 24 | **ECCE1_1 –** ECC Single Bit Error for MIPITX1<br><br>This bit reflects the status of ECC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** bit is set to 0. | 0 – No ECC single bit error since this bit is cleared<br>1 – At least 1 ECC single bit error since this bit is cleared |

| Name | Description | Setting |
|---|---|---|
| | Once an ECC single-bit error occurs, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | |
| **CBO1**<br>Bit 23 | **CBO1 –** Command Buffer Overflow for MIPITX1<br><br>This bit reflects the status of internal command buffer of the SPI/MCU interface. If the command buffer has overflowed, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – Overflow has not occurred<br>1 – Overflow has occurred |
| **Reserved**<br>Bit 22-21 | **Reserved** | Not Applicable |
| **MLO1**<br>Bit 20 | **MLO1 –** MCU Long Buffer Overflow for MIPITX1<br><br>This bit reflects the status of internal long buffer of the MCU interface. If the long buffer has overflowed, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – Overflow has not occurred<br>1 – Overflow has occurred |
| **Reserved**<br>Bit 19 | **Reserved** | Not Applicable |
| **CONT1**<br>Bit 18 | **CONT1** – Contention Detected for MIPITX1<br><br>This bit reflects the status of the data lane contention detector. | 0 – No contention<br>1 – Contention has occurred |
| **Reserved**<br>Bit 17 | **Reserved** | Not Applicable |
| **VMM1**<br>Bit 16 | **VMM1** – VC Mis-Match for MIPITX1<br><br>This bit reflects whether there is a mismatch between the VC ID transmitted by the SSD2829 and the VC ID received from the MIPI slave.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – No mismatch<br>1 – Mismatch has occurred |
| **Reserved**<br>Bit 15-11 | **Reserved** | Not Applicable |
| **CRCE0**<br>Bit 10 | **CRCE0 –** CRC Error for MIPITX0<br><br>This bit reflects the status of CRC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** bit is set to 0. Once a CRC error occurs, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – No CRC error since this bit is cleared<br>1 – At least 1 CRC error since this bit is cleared |
| **ECCE2_0**<br>Bit 9 | **ECCE2_0 –** ECC Multi Bit Error for MIPITX0 | 0 – No ECC multi-bit error since this bit is cleared |

| Name | Description | Setting |
|---|---|---|
| | This bit reflects the status of ECC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** bit is set to 0. Once an ECC multi-bit error occurs, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 1 – At least 1 ECC multi-bit error since this bit is cleared |
| **ECCE1_0**<br>Bit 8 | **ECCE1_0 –** ECC Single Bit Error for MIPITX0<br><br>This bit reflects the status of ECC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** bit is set to 0. Once an ECC single-bit error occurs, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – No ECC single bit error since this bit is cleared<br>1 – At least 1 ECC single bit error since this bit is cleared |
| **CBO0**<br>Bit 7 | **CBO0 –** Command Buffer Overflow for MIPITX0<br><br>This bit reflects the status of internal command buffer of the SPI/MCU interface. If the command buffer has overflowed, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – Overflow has not occurred<br>1 – Overflow has occurred |
| **Reserved**<br>Bit 6-5 | **Reserved** | Not Applicable |
| **MLO0**<br>Bit 4 | **MLO0 –** MCU Long Buffer Overflow for MIPITX0<br><br>This bit reflects the status of internal long buffer of the MCU interface. If the long buffer has overflowed, this bit will be set to 1.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – Overflow has not occurred<br>1 – Overflow has occurred |
| **Reserved**<br>Bit 3 | **Reserved** | Not Applicable |
| **CONT0**<br>Bit 2 | **CONT0** – Contention Detected for MIPITX0<br><br>This bit reflects the status of the data lane contention detector. | 0 – No contention<br>1 – Contention has occurred |
| **Reserved**<br>Bit 1 | **Reserved** | Not Applicable |
| **VMM0**<br>Bit 0 | **VMM0** – VC Mis-Match for MIPITX0<br><br>This bit reflects whether there is a mismatch between the VC ID transmitted by the SSD2829 and the VC ID received from the MIPI slave.<br><br>It will remain as 1 until the application processor writes 1 to clear it. | 0 – No mismatch<br>1 – Mismatch has occurred |

## 8.1.24 Compressed Register

Offset Address

**CR** **Compressed Register** **0xC8**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VID_COMPRESSED_BYTE_COUNT[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VID_COMPRESSED_BYTE_COUNT[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-25: Compressed Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 15-0 | **Reserved** | Not Applicable |

## 8.1.25 Delay Adjustment Register 1

Offset Address

**DAR1** **Delay Adjustment Register 1** **0xC9**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **HZD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x14 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HPD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x02 | | | | | | | |
| | | | | | | | | |

**Table 8-26: Delay Adjustment Register 1 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **HZD**<br>Bit 15-8 | **HZD –** HS Zero Delay<br><br>These bits specifies the number of system clock for HS zero delay period THS-ZERO.<br><br>The minimum value is 1. | Per Application Condition |
| **HPD**<br>Bit 7-0 | **HPD –** HS Prepare Delay<br><br>These bits specifies the number of system clock for HS prepare delay period THS-PREPARE.<br><br>The minimum value is 1. | Per Application Condition |

## 8.1.26 Delay Adjustment Register 2

Offset Address

**DAR2** **Delay Adjustment Register 2** **0xCA**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **CZD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x28 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **CPD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x03 | | | | | | | |
| | | | | | | | | |

**Table 8-27: Delay Adjustment Register 2 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **CZD**<br>Bit 15-8 | **CZD** – Clock Zero Delay<br><br>These bits specifies the number of system clock for HS zero delay period TCLK-ZERO.<br><br>The minimum value is 1. | Per Application Condition |
| **CPD**<br>Bit 7-0 | **CPD –** Clock Prepare Delay<br><br>These bits specifies the number of system clock for HS prepare delay period TCLK-PREPARE.<br><br>The minimum value is 1. | Per Application Condition |

## 8.1.27 Delay Adjustment Register 3

Offset Address

**DAR3** **Delay Adjustment Register 3** **0xCB**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **CPED** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x04 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **CPTD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x16 | | | | | | | |
| | | | | | | | | |

**Table 8-28: Delay Adjustment Register 3 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **CPED**<br>Bit 15-8 | **CPED** – Clock Pre Delay<br><br>These bits specifies the number of system clock for CLK pre delay period TCLK-PRE.<br><br>The minimum value is 1. | Per Application Condition |
| **CPTD**<br>Bit 7-0 | **CPTD –** Clock Prepare Delay<br><br>These bits specifies the number of system clock for CLK post delay period TCLK-POST.<br><br>The minimum value is 1. | Per Application Condition |

## 8.1.28 Delay Adjustment Register 4

Offset Address

**DAR4** **Delay Adjustment Register 4** **0xCC**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **CTD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x0A | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HTD** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x0A | | | | | | | |
| | | | | | | | | |

**Table 8-29: Delay Adjustment Register 4 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **CTD**<br>Bit 15-8 | **CTD** – Clock Pre Delay<br><br>These bits specifies the number of system clock for CLK trail delay period TCLK-TRAIL.<br><br>The minimum value is 1. | Per Application Condition |
| **HTD**<br>Bit 7-0 | **HTD –** Clock Prepare Delay<br><br>These bits specifies the number of system clock for HS trail delay period THS-TRAIL.<br><br>The minimum value is 1. | Per Application Condition |

## 8.1.29 Delay Adjustment Register 5

Offset Address

**DAR5** **Delay Adjustment Register 5** **0xCD**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **WUD[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x10 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **WUD[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-30: Delay Adjustment Register 5 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **WUD**<br>Bit 15-0 | **WUD** – Wake Up Delay<br><br>These bits specifies the number of clock cycles for wake up delay period TWAKEUP. The delay is used to wake up the MIPI slave from ULPS state. The clock is the low power clock. | Per Application Condition |

## 8.1.30 Delay Adjustment Register 6

Offset Address

**DAR6** **Delay Adjustment Register 6** **0xCE**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | **TGO** | | | |
| TYPE | RO | RO | RO | RO | RW | | | |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x4 | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | | | | | **TGET** | | | |
| TYPE | RO | RO | RO | RO | RW | | | |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x5 | | | |
| | | | | | | | | |

**Table 8-31: Delay Adjustment Register 6 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-12 | **Reserved** | Not Applicable |
| **TGO**<br>Bit 11-8 | **TGO** – TA Go Delay<br><br>These bits specifies the number of TLPX for TA go delay period TTA-GO. | Per Application Condition |
| **Reserved**<br>Bit 7-4 | **Reserved** | Not Applicable |
| **TGET**<br>Bit 3-0 | **TGET** – TA Get Delay<br><br>These bits specifies the number of TLPX for TA get delay period TTA-GET. | Per Application Condition |

## 8.1.31 HS TX Timer Register 1

Offset Address

**HTTR1** **HS TX Timer Register 1** **0xCF**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **HTT_L[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HTT_L[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-32: HS TX Timer Register 1 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **HTT_L**<br>Bit 15-0 | **HTT_L** – HS TX Timer Low<br><br>These bits are the lower 16 bits of the HTT.<br><br>These bits specify the HS TX timer timeout value. PLL reference clock is used to increment an internal timer. The timer starts when the SSD2829 enters HS transmit mode. When the SSD2829 exits from HS transmit mode, the timer will be reset. If the timer expires before the end of HS transmission, the SSD2829 will signal an error and switch to LP mode to continue the transmission. At the same time, the HS bit will be cleared to 0. Software intervention is required so that the SSD2829 can go back to proper HS transmission mode. | Per Application Condition |

## 8.1.32 HS TX Timer Register 2

Offset Address

**HTTR2** **HS TX Timer Register 2** **0xD0**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **HTT_H[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **HTT_H[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-33: HS TX Timer Register 2 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **HTT_H**<br>Bit 15-0 | **HTT_H** – HS TX Timer High<br><br>These bits are the upper 16 bits of the HTT.<br><br>These bits specify the HS TX timer timeout value. PLL reference clock is used to increment an internal timer. The timer starts when the SSD2829 enters HS transmit mode. When the SSD2829 exits from HS transmit mode, the timer will be reset. If the timer expires before the end of HS transmission, the SSD2829 will signal an error and switch to LP mode to continue the transmission. At the same time, the HS bit will be cleared to 0. Software intervention is required so that the SSD2829 can go back to proper HS transmission mode. | Per Application Condition |

## 8.1.33 TE Status Register

Offset Address

**TESR** **TE Status Register** **0xD3**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | | **CMD_BC** | **TE_OUT_SEL1** | **TE_OUT_SEL0** | **TE_IN_SEL1** | **TE_IN_SEL0** | **TER1** | **TER0** |
| TYPE | RO | RW | RW | RW | RW | RW | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |

**Table 8-34: TE Status Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-7 | **Reserved** | Not Applicable |
| **CMD_BC**<br>Bit 6 | **CMD_BC** – Command Broadcast<br><br>This bit select whether to broadcast the 2C/3C content from MIPITX0 to MIPITX1 when the configuration is 1x2(**TX_DUAL**=1) for MCU or DSI input. | 0 - No Broadcast<br>1 - Broadcast |
| **TE_OUT_SEL1**<br>Bit 5 | **Reserved** | Not Applicable |
| **TE_OUT_SEL0**<br>Bit 4 | **TE_OUT_SEL0** – TE output from MIPITX0<br><br>This bit select whether to send the TE output from MIPITX0 via TE_OUT0 pin. | 0 - Internal TE is sent to TE_OUT_0 pin<br>1 - Reserved |
| **TE_IN_SEL1**<br>Bit 3 | **Reserved** | Not Applicable |
| **TE_IN_SEL0**<br>Bit 2 | **TE_OUT_SEL0** – TE output from MIPITX0<br><br>This bit select which TE input to be used for pulse shaping. | 0 - TE is taken from TE_IN_0 pin<br>1 - TE is taken from MIPITX0 |
| **TE_RESP1**<br>Bit 1 | **Reserved** | Not Applicable |
| **TE_RESP0**<br>Bit 0 | **TE_RESP0** – TE Response from MIPITX0 | 0 –TE response has not been received |

| Name | Description | Setting |
|---|---|---|
| | This bit reflects whether a TE response has been received or not. Once a TE response is received, this bit will be set to 1. At the same time, the output TE signal will go high. The host processor can write 1 to clear this bit. Once the bit is cleared, the TE signal will go low. | 1 – TE response has been received |

## 8.1.34 SPI Read Register

Offset Address

**SRR** **SPI Read Register** **0xD4**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **RRA** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0xFA | | | | | | | |
| | | | | | | | | |

**Table 8-35: SPI Read Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-8 | **Reserved** | Not Applicable |
| **RRA**<br>Bit 7-0 | **RRA** – Register Read Address<br><br>These bits specify the address of the register to be read through the SPI interface, when the interface is SPI 8-bit (either 3 wire or 4 wire). | Per Application Condition |

## 8.1.35 PLL Lock Register

Offset Address

**PLR** **PLL Lock Register** **0xD5**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **LOCK[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x14 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **LOCK[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x50 | | | | | | | |
| | | | | | | | | |

**Table 8-36: PLL Lock Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-16 | **Reserved** | Not Applicable |
| **LOCK**<br>Bit 15-0 | **LOCK** – Lock Counter<br><br>These bits specify the PLL lock range in term of PLL reference frequency. The maximum PLL lock period is 500us and the default setting assumed the reference clock is 10Mhz. | Per Application Condition |

## 8.1.36 Test Register

Offset Address

**TR** **Test Register** **0xD6**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | **EHS** | | | | |
| TYPE | RO | RO | RO | RW | RW | | | |
| RESET | 0x0 | 0x0 | 0x0 | 0x1 | 0x0 | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | **COMP_SLICE** | | | |
| TYPE | RW | | | RO | RW | | | |
| RESET | 0x0 | | | 0x0 | 0x0 | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **TM** | | **EIC** | | | | | **DIS_CONT** |
| TYPE | RW | | RW | | | | | RW |
| RESET | 0x0 | | 0x00 | | | | | 0x1 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **PNB** | | | | | | **END** | **CO** |
| TYPE | RW | | | | | | RW | RW |
| RESET | 0x01 | | | | | | 0x0 | 0x1 |
| | | | | | | | | |

**Table 8-37: Test Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-29 | **Reserved** | Not Applicable |
| **EHS**<br>Bit 28 | **EHS** – Early High Speed clock<br><br>This bit is used to allow MIPI output to enter HS 1 line earlier before the video data transmission. | 0 - disable<br>1 - enable |
| **Reserved**<br>Bit 27-21 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 20 | **Reserved** | Not Applicable |
| **COMP_SLICE**<br>Bit 19-16 | **COMP_SLICE** – Number of Compressed Slice per Line<br><br>These bits indicate the number of slice per line when the input DSI video stream is compressed data. | 0 – 1 slice per line<br>..<br>15 –16 slices per line |
| **TM**<br>Bit 15-14 | **TM** – Test Mode<br><br>These bits selects whether to inject CRC/ECC error for the outgoing streams. They are used for debugging purpose only. They should be set to 00 in normal mode. | 00 – Normal mode<br>01 – Inject CRC error<br>10 – Inject 1 bit ECC error<br>11 – Inject 2 bit ECC error |
| **EIC**<br>Bit 13-9 | **EIC** – Error Injection Control | Per Application Condition |

| Name | Description | Setting |
|---|---|---|
| | These bits control the position of the error being injected for testing. It is only applicable when TM is 01. | |
| **DIS_CONT**<br>Bit 8 | **DIS_CONT** – Disable Contention input from Analog<br><br>This bit selects whether to ignore the error contention signals output from the Phy. | 0 – Enable<br>1 – Disable |
| **PNB**<br>Bit 7-2 | **PNB** – Packet Number during Blanking<br><br>These bits control the number of packet to send during video mode blanking period. | Per Application Condition |
| **END**<br>Bit 1 | **END** – Endian-ness<br><br>This bit specifies the endian-ness of the data transmitted over the serial link. During video mode transmission, this bit must be set to 0 so as to follow the MIPI DSI specification. | 0 – Least significant byte sent first<br>1 – Most significant byte sent first |
| **CO**<br>Bit 0 | **CO** – Color Order<br><br>This bit specifies the order of the color component in the pixel. During video mode transmission, this bit must be set to 1 so as to follow the MIPI DSI specification | 0 – RGB. R is in the higher portion of the pixel<br>1 – BGR. B is in the higher portion of the pixel |

## 8.1.37 TE Count Register

Offset Address

**TECR** **TE Count Register** **0xD7**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **TEC1[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **TEC1[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x01 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **TEC0[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **TEC0[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x01 | | | | | | | |
| | | | | | | | | |

**Table 8-38: TE Count Register Description**

| Name | Description | Setting |
|---|---|---|
| **TEC1**<br>Bit 31-16 | **TEC1** – TE Counter for MIPITX1<br><br>These bits determines the pulse width of the output TE signal. A counter will be started after the TE signal goes to 1. When the counter reaches the value in **TEC1** field, the TE signal will be set to 0. The counter uses the PLL reference clock to do counting.<br><br>The minimum value is 1. | Per Application Condition |
| **TEC0**<br>Bit 15-0 | **TEC0** – TE Counter for MIPITX0<br><br>These bits determines the pulse width of the output TE signal. A counter will be started after the TE signal goes to 1. When the counter reaches the value in **TEC0** field, the TE signal will be set to 0. The counter uses the PLL reference clock to do counting.<br><br>The minimum value is 1. | Per Application Condition |

## 8.1.38 Analog Control Register 1

Offset Address

**ACR1** **Analog Control Register 1** **0xD8**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **THFT_TLFT1** | **THFT_TLFT0** | **EN_REG** | **HSTX_RO_IN** | | | | **LPTX_DS[2]** |
| TYPE | RW | RW | RW | RW | | | | RW |
| RESET | 0x0 | 0x0 | 0x1 | 0x5 | | | | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **LPTX_DS[1:0]** | | **BG_TRIM_V0P6** | | | | | |
| TYPE | RW | | RW | | | RW | | |
| RESET | 0x0 | | 0x3 | | | 0x4 | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **BG_TC** | | | **BG_TEN** | **BG_TRIM_0P5** | | | **BG_IDUTY[2]** |
| TYPE | RW | | | RW | RW | | | RW |
| RESET | 0x4 | | | 0x0 | 0x3 | | | 0x1 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **BG_IDUTY[1:0]** | | **BG_IREG** | | **BG_ISEL** | | | **EN_BG** |
| TYPE | RW | | RW | | RW | | | RW |
| RESET | 0x0 | | 0x1 | | 0x4 | | | 0x1 |
| | | | | | | | | |

**Table 8-39: Analog Control Register 1 Description**

| Name | Description | Setting |
|---|---|---|
| **THFT_TLFT1** Bit 31 | **THFT_TLFT1** – Low Power Receiver Input Threshold High/Low Adjust Bit 1 | Per Application Condition |
| **THFT_TLFT0** Bit 30 | **THFT_TLFT0** – Low Power Receiver Input Threshold High/Low Adjust Bit 0 | Per Application Condition |
| **EN_REG** Bit 29 | **EN_REG** – 0.5V LDO enable | Per Application Condition |
| **HSTX_RO_IN** Bit 28-25 | **HSTX_RO_IN** – Driver output resistance control | Per Application Condition |
| **LPTX_DS** Bit 24-22 | **LPTX_DS** – Low Power Transmitter Drive Strength | Per Application Condition |
| **BG_TRIM_V0P6** Bit 21-19 | **BG_TRIM_V0P6** – Voltage trimming bits | Per Application Condition |
| **Reserved** Bit 18-16 | **Reserved** | Not Applicable |
| **BG_TC** Bit 15-13 | **BG_TC** – Temperature coefficient programming bits of bandgap | Per Application Condition |
| **BG_TEN** Bit 12 | **BG_TEN** – Bandgap Test Enable | Per Application Condition |
| **BG_TRIM_0P5** Bit 11-9 | **BG_TRIM_0P5** – Voltage trimming bits of 0.5V LDO | Per Application Condition |

| Name | Description | Setting |
|---|---|---|
| **BG_IDUTY**<br>Bit 8-6 | **BG_IDUTY** – Biasing current adjustment of duty cycle regulation circuit | Per Application Condition |
| **BG_IREG**<br>Bit 5-4 | **BG_IREG** – Biasing current adjustment of 0.5V LDO | Per Application Condition |
| **BG_ISEL**<br>Bit 3-1 | **BG_ISEL** – Biasing current adjustment of contention detection | Per Application Condition |
| **EN_BG**<br>Bit 0 | **EN_BG** – Bandgap Enable | Per Application Condition |

## 8.1.39 RGB Interface Control Register 7

Offset Address

**RICR7** **RGB Interface Control Register 7** **0xDD**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | **VEC** | **XEQ1** | **XEQ0** |
| TYPE | RO | RO | RO | RO | RO | RW | RWAC | RWAC |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VBN** | | | | **VFN** | | | |
| TYPE | RW | | | | RW | | | |
| RESET | 0x0 | | | | 0x0 | | | |
| | | | | | | | | |

**Table 8-40: RGB Interface Control Register 7 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-11 | **Reserved** | Not Applicable |
| **VEC**<br>Bit 10 | **VEC –** Video Enable Control<br><br>This bit controls how command is handled during video mode, **VEN**. | 0 - Any command received will output immediately if there is no video packet to send.<br><br>1 - Any command received will only be output during the blanking period after first video packet is received. If there is no video packet to be sent, the command is held. |
| **XEQ1**<br>Bit 9 | **XEQ1 –** Execute Queue for MIPITX1<br><br>This bit will cause SSD2829 to generate Execute Queue Packet(0x16) at the last line of the frame, after hsync packet. It will be cleared when Execute Queue Packet is sent out.<br><br>This bit is auto cleared by hardware. | 0 - Do nothing<br>1 - Send Execute Queue Packet(0x16) at the last line of the current frame, after hsync. |
| **XEQ0**<br>Bit 8 | **XEQ –** Execute Queue for MIPITX0<br><br>This bit will cause SSD2829 to generate Execute Queue Packet(0x16) at the last line of the frame, after hsync packet. It will be cleared when Execute Queue Packet is sent out.<br><br>This bit is auto cleared by hardware. | 0 - Do nothing<br>1 - Send Execute Queue Packet(0x16) at the last line of the current frame, after hsync. |

| Name | Description | Setting |
|---|---|---|
| **VBN**<br>Bit 7-4 | **VBN –** Vertical Back Porch Non Video Data Window<br><br>These fields specify the number of vertical back porch counting backward from the first vertical active line in which non-video data is not allowed to be sent via MIPI link.<br>This field is only valid when **VEN** is 1 and the interface setting is RGB + SPI(if_sel[0] = 0).<br><br>This field should not larger than VBP | Per Application Condition |
| **VFN**<br>Bit 3-0 | **VFN –** Vertical Front Porch Non Video Data Window<br><br>These fields specify the number of vertical front porch counting forward from the last vertical active line in which non-video data is not allowed to be sent via MIPI link.<br>This field is only valid when **VEN** is 1 and the interface setting is RGB + SPI(if_sel[0] = 0).<br><br>This field should not larger than VFP. | Per Application Condition |

## 8.1.40 INOUT Configuration Control Register

Offset Address

**IOCR** **INOUT Configuration Register** **0xDE**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | **MCU_SWAP1** | **MCU_SWAP0** | **BIT_SWAP1** |
| TYPE | RO | RO | RO | RO | RO | RW | RW | RW |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **BIT_SWAP0** | **PIXEL_SWAP1** | **PIXEL_SWAP0** | | | | | |
| TYPE | RW | RW | RW | RW | RW | | RW | |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | | 0x0 | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **TX_READ** | **TX_WRITE** | | **TX_DUAL** | **TX_LS1** | | **TX_LS0** | |
| TYPE | RW | RW | | RW | RW | | RW | |
| RESET | 0x0 | 0x1 | | 0x0 | 0x0 | | 0x0 | |
| | | | | | | | | |

**Table 8-41: INOUT Configuration Register Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-27 | **Reserved** | Not Applicable |
| **MCU_SWAP1**<br>Bit 26 | **Reserved** | Not Applicable |
| **MCU_SWAP0**<br>Bit 25 | **MCU_SWAP0 –** MCU data swap for MCU<br><br>This bit defines how the SSD2829 interpret the MCU input. | 0 – No Swap<br>1 –Swap MCU data MSB to LSB |
| **BIT_SWAP1**<br>Bit 24 | **Reserved** | Not Applicable |
| **BIT_SWAP0**<br>Bit 23 | **BIT_SWAP0 –** RGB bit swap for RGB<br><br>This bit defines how the SSD2829 interpret the RGB video input. | 0 – No Swap<br>1 –Swap RGB pixel MSB to LSB |
| **PIXEL_SWAP1**<br>Bit 22 | **Reserved** | Not Applicable |
| **PIXEL_SWAP0**<br>Bit 21 | **BIT_SWAP0 –** RGB pixel swap for RGB<br><br>This bit defines how the SSD2829 interpret the RGB video input. | 0 – No Swap<br>1 –Swap RGB pixel upper and lower |
| **Reserved**<br>Bit 20 | **Reserved** | Not Applicable |

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 19-18 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 17-16 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 15-8 | **Reserved** | Not Applicable |
| **TX_READ**<br>Bit 7 | **TX_READ** – TX read<br><br>This bit determines the destination of the read command when SSD2829 is configured as 1 input with 2 output(MIPITX). | 0 - Read from MIPITX0<br>1 - Read from MIPITX1 |
| **TX_WRITE**<br>Bit 6-5 | **TX_WRITE** – TX write<br><br>This bit determines the destination of the write command when SSD2829 is configured as 1 input with 2 output(MIPITX). | 00 - Discard<br>01 - Write to MIPITX0<br>10 - Write to MIPITX1<br>11 - Write to MIPITX0 and MIPITX1 |
| **TX_DUAL**<br>Bit 4 | **TX_DUAL –** MIPITX dual mode<br><br>This bit defines the dual/single mode of MIPITX. | 0 – Single mode<br>1 – Dual mode |
| **TX_LS1**<br>Bit 3-2 | **TX_LS1** – Lane Select for MIPITX1<br><br>These bits define the number of lane to be used for MIPITX1 in SSD2829. | 00 – 1 lane mode<br>01 – 2 lane mode<br>10 – 3 lane mode<br>11 – 4 lane mode |
| **TX_LS0**<br>Bit 1-0 | **TX_LS0** – Lane Select for MIPITX0<br><br>These bits define the number of lane to be used for MIPITX0 in SSD2829. | 00 – 1 lane mode<br>01 – 2 lane mode<br>10 – 3 lane mode<br>11 – 4 lane mode |

## 8.1.41 APB Write Register

Offset Address

**AWR** **APB Write Register** **0xE0**

| BIT | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 |
|---|---|---|---|---|---|---|---|---|
| NAME | **DATA[31:24]** | | | | | | | |
| TYPE | W | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 39 | 38 | 37 | 36 | 35 | 34 | 33 | 32 |
| NAME | **DATA[23:16]** | | | | | | | |
| TYPE | W | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
| NAME | **DATA[15:8]** | | | | | | | |
| TYPE | W | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **DATA[7:0]** | | | | | | | |
| TYPE | W | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **ADDR[15:8]** | | | | | | | |
| TYPE | W | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **ADDR[7:0]** | | | | | | | |
| TYPE | W | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-42: Delay Adjustment Register Description**

| Name | Description | Setting |
|---|---|---|
| **DATA**<br>Bit 47-16 | **DATA –** 32-bit write data for APB | Refer to APB write section |
| **ADDR**<br>Bit 15-0 | **ADDR –** 16-bit address for APB write/read | Refer to APB write section |

## 8.1.42 APB Read Register

Offset Address

**ARR** **APB Read Register** **0xE1**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **DATA[31:24]** | | | | | | | |
| TYPE | R | | | | | | | |
| RESET | 0x0 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **DATA[23:16]** | | | | | | | |
| TYPE | R | | | | | | | |
| RESET | 0x0 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **DATA[15:8]** | | | | | | | |
| TYPE | R | | | | | | | |
| RESET | 0x0 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **DATA[7:0]** | | | | | | | |
| TYPE | R | | | | | | | |
| RESET | 0x0 | | | | | | | |
| | | | | | | | | |

**Table 8-43: APB Read Register Description**

| Name | Description | Setting |
|---|---|---|
| **DATA**<br>Bit 31-0 | **DATA** – 32 bits read data | Refer to APB read section |

## 8.2 Local (APB) Register Descriptions

The APB registers have 16 bit address, of which the most significant 4 bits represent the base address of the APB peripheral, and the least significant 16 bits represent the offset within the APB peripheral. The table below shows the APB peripherals and their respective base address & address range.

| Address | Module |
|---|---|
| 0x0000 – 0x0FFF | SCM |
| 0x1000 – 0x1FFF | Reserved |
| 0x2000 – 0x2FFF | Reserved |
| 0x3000 – 0x3FFF | Reserved |
| 0x4000 – 0x4FFF | Reserved |
| 0x5000 – 0x7FFF | Video BIST |
| 0x6000 – 0x6FFF | Pixel Peek |

## 8.2.1 SCM Registers Descriptions

### 8.2.1.1 SCM Miscellaneous Control Register

Offset Address

**SCM_MISC** **SCM Miscellaneous Control Register** **0x0010**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | **ckmon** | lpstate | | | | |
| TYPE | RO | RO | RW | RW | RW | RW | RW | |
| RESET | 0x0 | 0x0 | 0x1 | 0x0 | 0x1 | 0x1 | 0x0 | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | **v2c** |
| TYPE | RW | | RW | | RW | RW | RW | RW |
| RESET | 0x2 | | 0x1 | | 0x0 | 0x0 | 0x0 | 0x0 |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RW | | | | RW | | RW | RW |
| RESET | 0x0 | | | | 0x3 | | 0x1 | 0x0 |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **flip1** | **flip0** | | | | | | |
| TYPE | RW | RW | RW | | RW | RW | | RW |
| RESET | 0x0 | 0x0 | 0x0 | | 0x0 | 0x0 | | 0x0 |
| | | | | | | | | |

**Table 8-44: SCM Miscellaneous Control Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit31-30 | **Reserved** | Not Applicable |
| **clkmon**<br>Bit29 | **Clock Monitor on TE_OUT0 –**<br>This bit enable the TE_OUT0 to be used as clkout monitor signals | Per Application Condition |
| **lpstate**<br>Bit28 | **LP State –**<br>This bit controls the state of the link when the chip is powered down | 0=LP00<br>1=LP11 |
| **Reserved**<br>Bit 27-17 | **Reserved** | Not Applicable |
| **v2c**<br>Bit 16 | **Video to Cmd** – Indicates whether video is converted to 2C/3C commands. | 0=Disable<br>1=Enable |
| **Reserved**<br>Bit 15-12 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 11-9 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 8 | **Reserved** | Not Applicable |
| **mtx_flip1**<br>Bit 7 | **MIPI Left/Right Flip for DSI1** –<br>This bit configures the left and right image swap at the MIPI output. | 0x0 = Data received from read from left to right (Default)<br>0x1 = Data received from read from right to left |
| **mtx_flip0**<br>Bit 6 | **MIPI Left/Right Flip for DSI0** –<br>This bit configures the left and right image swap at the MIPI output. | 0x0 = Data received from read from left to right (Default)<br>0x1 = Data received from read from right to left |
| **Reserved**<br>Bit 5-4 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 3 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 2-1 | **Reserved** | Not Applicable |
| **Reserved**<br>Bit 0 | **Reserved** | Not Applicable |

### 8.2.1.2 SCM Scratch Register

Offset Address

**SCM_SCRATCH** **SCM Scratch Register** **0x0014**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **scratch[31:24]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **scratch[23:16]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **scratch[15:8]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **scratch0[7:0]** | | | | | | | |
| TYPE | RW | | | | | | | |
| RESET | 0x00 | | | | | | | |
| | | | | | | | | |

**Table 8-45: SCM Scratch Register Description**

| Name | Description | Setting |
|---|---|---|
| **scratch**<br>Bit 31-0 | **Scratch Register –** User can use this register to store any value | Per Application Condition |

## 8.2.2 Video BIST Register Descriptions

### 8.2.2.1 Video BIST Register 0

Offset Address

**VBISTR0** **Video BIST Register 0** **0x000**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **VB_REPEAT_CNT[15:8]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **VB_REPEAT_CNT[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VB_MODE{3:0}** | | | | **VB_CFG_MODE[1:0]** | | **VB_CSPF** | **VB_EN** |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-46: Video BIST Register 0 Description**

| Name | Description | Setting |
|---|---|---|
| **VB_REPEAT_CNT[15:0]**<br>Bit 31-16 | **Video BIST Repeat Count –** These bits set the repeat count for video pattern generation. It will have different meaning for different **VB_MODE.** | Video BIST section for more information. |
| **Reserved**<br>Bit15-8 | **Reserved** | Not Applicable |
| **VB_MODE[3:0]**<br>Bit 7-4 | **Video BIST Mode –** These bits select the Video BIST image pattern. | Video BIST section for more information. |
| **VB_CFG_MODE[1:0]**<br>Bit 3-2 | **Video BIST Config Mode –** These bits select the mode that video bist generate the pattern | **0** – Single data buffer mode (Normal)<br>**1** – Reserved<br>**2** – Reserved<br>**3** – Dual data buffer mode (Broadcast) |
| **VB_CSPF**<br>Bit 1 | **Video BIST Color Swap per Frame –** This bit enable the color swap per frame | **0** – Disable<br>**1** – Enable |
| **VB_EN**<br>Bit 0 | **Video BIST Enable –** This bit enable the Video BIST function | **0** – Disable<br>**1** – Enable |

### 8.2.2.2 Video BIST Register 1

Offset Address

**VBISTR1** **Video BIST Register 1** **0x004**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | **VB_COLOR_X_R[9:8]** | |
| TYPE | RO | RO | RO | RO | RO | RO | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VB_COLOR_X_R[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-47: Video BIST Register 1 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31-10 | **Reserved** | Not Applicable |
| **VB_COLOR_X_R[ 9:0]**<br>Bit 9-0 | **Video BIST Color X Red Component** – These bits specify the Color X Red Component. | 30bpp – [9:0]<br>24bpp – [7:0] |

### 8.2.2.3 Video BIST Register 2

Offset Address

**VBISTR2** **Video BIST Register 2** **0x008**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | **VB_COLOR_X_G[9:8]** | |
| TYPE | RO | RO | RO | RO | RO | RO | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **VB_COLOR_X_G[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | **VB_COLOR_X_B[9:8]** | |
| TYPE | RO | RO | RO | RO | RO | RO | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VB_COLOR_X_B[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-48: Video BIST Register 2 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31–26 | **Reserved** | Not Applicable |
| **VB_COLOR_X_G[ 9:0]**<br>Bit 25–16 | **Video BIST Color X Green Component –** These bits specify the Color X Red Component. | 30bpp – [9:0]<br>24bpp – [7:0] |
| **Reserved**<br>Bit 15–10 | **Reserved** | Not Applicable |
| **VB_COLOR_X_B[ 9:0]**<br>Bit 9-0 | **Video BIST Color X Blue Component –** These bits specify the Color X Red Component. | 30bpp – [9:0]<br>24bpp – [7:0] |

### 8.2.2.4 Video BIST Register 3

Offset Address

**VBISTR3** **Video BIST Register 3** **0x00C**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | **VB_COLOR_Y_R[9:8]** | |
| TYPE | RO | RO | RO | RO | RO | RO | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VB_COLOR_Y_R[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-49: Video BIST Register 3 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31–10 | **Reserved** | Not Applicable |
| **VB_COLOR_Y_R[ 9:0]**<br>Bit 9-0 | **Video BIST Color X Red Component –** These bits specify the Color X Red Component. | 30bpp – [9:0]<br>24bpp – [7:0] |

### 8.2.2.5 Video BIST Register 4

Offset Address

**VBISTR4** **Video BIST Register 4** **0x010**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | **VB_COLOR_Y_G[9:8]** | |
| TYPE | RO | RO | RO | RO | RO | RO | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **VB_COLOR_Y_G[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | **VB_COLOR_Y_B[9:8]** | |
| TYPE | RO | RO | RO | RO | RO | RO | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VB_COLOR_Y_B[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-50: Video BIST Register 4 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31–26 | **Reserved** | Not Applicable |
| **VB_COLOR_Y_G[ 9:0]**<br>Bit 25–16 | **Video BIST Color Y Green Component –** These bits specify the Color Y Red Component. | 30bpp – [9:0]<br>24bpp – [7:0] |
| **Reserved**<br>Bit 15–10 | **Reserved** | Not Applicable |
| **VB_COLOR_Y_B[ 9:0]**<br>Bit 9-0 | **Video BIST Color Y Blue Component –** These bits specify the Color Y Red Component. | 30bpp – [9:0]<br>24bpp – [7:0] |

### 8.2.2.6 Video BIST Register 5

Offset Address

**VBISTR5** **Video BIST Register 5** **0x014**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **VB_X_START[15:8]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **VB_X_START[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VB_X_END[15:8]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VB_X_END[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-51: Video BIST Register 5 Description**

| Name | Description | Setting |
|---|---|---|
| **VB_X_START[15:0]** Bit 31–16 | **Video BIST X Start** – These bits define the X starting position. | Per Application Condition |
| **VB_X_END[15:0]** Bit 15–0 | **Video BIST X End** – These bits define the X end position. | Per Application Condition |

### 8.2.2.7 Video BIST Register 6

Offset Address

**VBISTR6** **Video BIST Register 6** **0x018**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **VB_Y_START[15:8]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **VB_Y_START[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **VB_Y_END[15:8]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **VB_Y_END[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-52: Video BIST Register 6 Description**

| Name | Description | Setting |
|---|---|---|
| **VB_Y_START[15:0]** Bit 31–16 | **Video BIST Y Start** – These bits define the Y starting position. | Per Application Condition |
| **VB_Y_END[15:0]** Bit 15–0 | **Video BIST Y End** – These bits define the Y end position. | Per Application Condition |

## 8.2.3 Pixel Peek Registers Descriptions

### 8.2.3.1 Pixel Peek Register 0

Offset Address

**PIXELPEEKR0** **Pixel Peek Register 0** **0x000**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **CURSOR_POSITION_Y[15:8]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **CURSOR_POSITION_Y[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **CURSOR_POSITION_X[15:8]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **CURSOR_POSITION_X[7:0]** | | | | | | | |
| TYPE | RW | RW | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-53: Pixel Peek Register 0 Description**

| Name | Description | Setting |
|---|---|---|
| **CURSOR_POSITION_Y [15:0]** Bit 31–16 | **Cursor Position Y –** These bits set the y-coordinate to measure | Per Application Condition |
| **CURSOR_POSITION_X [15:0]** Bit 15–0 | **Cursor Position X –** These bits set the x-coordinate to measure | Per Application Condition |

### 8.2.3.2 Pixel Peek Register 1

Offset Address

**PIXELPEEKR1** **Pixel Peek Register 1** **0x004**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | | | **PREVENT_UPDATE** | **CURSOR_VISIBLE** | **CURSOR_COLOR_DYNAMIC** | **CURSOR_COLOR_RGB[2:0]** | | |
| TYPE | RO | RO | RW | RW | RW | RW | RW | RW |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-54: Pixel Peek Register 1 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31–6 | **Reserved** | Not Applicable |
| **PREVENT_UPDATE**<br>Bit 5 | **Prevent Update** – Set this bit before reading the values. | Per Application Condition |
| **CURSOR_VISIBLE**<br>Bit 4 | **Cursor Visible** – This bit enable cursor display. | 0 = disable<br>1 = enable |
| **CURSOR_COLOR_DYNAMIC**<br>Bit 3 | **Cursor Color Dynamic** – This bit enable dynamic color for cursor | 0 = disable<br>1 = enable |
| **CURSOR_COLOR_BGR[2:0]**<br>Bit 2–0 | **Cursor Color BGR** – These bits set the color for cursor (BGR, B on MSB). | Per Application Condition |

### 8.2.3.3 Pixel Peek Register 2

Offset Address

**PIXELPEEKR2** **Pixel Peek Register 2** **0x008**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x00 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x00 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **MEAS_VALUE_B[15:8]** | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x00 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **MEAS_VALUE_B[7:0]** | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x00 | | | | | | | |

**Table 8-55: Pixel Peek Register 2 Description**

| Name | Description | Setting |
|---|---|---|
| **Reserved**<br>Bit 31–16 | **Reserved** | Not Applicable |
| **MEAS_VALUE_B[15:0]**<br>Bit 15–0 | **Measure Value B** – Measured value for Blue. | Per Application Condition |

### 8.2.3.4 Pixel Peek Register 3

Offset Address

**PIXELPEEKR3** **Pixel Peek Register 3** **0x00C**

| BIT | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| NAME | **MEAS_VALUE_G[15:8]** | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| NAME | **MEAS_VALUE_G[7:0]** | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| NAME | **MEAS_VALUE_R[15:8]** | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NAME | **MEAS_VALUE_R[7:0]** | | | | | | | |
| TYPE | RO | RO | RO | RO | RO | RO | RO | RO |
| RESET | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0x0 | | | | | | | |

**Table 8-56: Pixel Peek Register 3 Description**

| Name | Description | Setting |
|---|---|---|
| **MEAS_VALUE_G[15:0]** Bit 31–16 | **Measure Value G** – Measured value for Green. | Per Application Condition |
| **MEAS_VALUE_R[15:0]** Bit 15–0 | **Measure Value R** – Measured value for Red. | Per Application Condition |

# 9 MAXIMUM RATINGS

**Table 9-1: Maximum Ratings (Voltage Referenced to $V_{SS}$)**

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| AVDD | Analog Power Supply | -0.3 | - | 1.44 | V |
| AVDD_RC | Analog Power Supply | -0.3 | - | 1.44 | V |
| VCIP | Analog Power for Bandgap, 3.3V | -0.3 | - | 3.96 | V |
| VDRV | Analog Power for MIPI TX Driver | -0.3 | - | 1.44 | V |
| VDD_CORE | Digital Core Power Supply | -0.3 | - | 1.44 | V |
| VDDIO | I/O Power Supply | -0.3 | - | 3.96 | V |
| $T_{STG}$ | Storage Temperature | -40 | - | 150 | °C |

Maximum ratings are those values beyond which damages to the device may occur. Functional operation should be restricted to the limits in the Electrical Characteristics tables or Pin Description section

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that $V_{CI}$ and $V_{OUT}$ be constrained to the range $V_{SS} < V_{DD} \leq V_{CI} < V_{OUT}$. Reliability of operation is enhanced if unused input is connected to an appropriate logic voltage level (e.g., either $V_{SS}$ or $V_{DD}$). Unused outputs must be left open. This device may be light sensitive. Caution should be taken to avoid exposure of this device to any light source during normal operation. This device is not radiation protected.

# 10 DC OPERATING CONDITIONS

**Table 10-1 : Recommended Operating Conditions**

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| AVDD | Analog Power Supply | 1.08 | 1.2 | 1.32 | V |
| AVDD_RC | Analog Power Supply | 1.08 | 1.2 | 1.32 | V |
| VDD_CORE | Digital Core Power Supply | 1.08 | 1.2 | 1.32 | V |
| VDDIO | I/O Power Supply | 1.62 | 1.8 | 1.98 | V |
| | I/O Power Supply | 2.7 | 3.3 | 3.6 | V |
| VCIP | Analog Power for Bandgap | 2.7 | 3.3 | 3.6 | V |
| $T_A$ | Operating Temperature | -40 | 25 | 85 | °C |

## 10.1 DC CHARACTERISTICS

| Symbol | Parameter | Test Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $I_{DD\_1.2V\_active}$ | Current for all 1.2V supplies for active mode | Active Mode (Video Mode) 2560x1600@60Hz, MIPITX-D@1.25Gbps x 8 lanes | - | 150 | 250 | mA |
| $I_{DD\_1.8V\_active}$ | Current for VDDIO (1.8/3.3V) supplies for active mode | | - | 7 | 15 | mA |
| $I_{DD\_3.3V\_active}$ | Current for VCIP supply for active mode | | - | 100 | 200 | uA |
| $I_{DD\_1.2V\_powerdown}$ | Current for all 1.2V supplies for power down mode | Power Down mode MIPIRX off MIPITX off | - | 110 | 250 | μA |
| $I_{DD\_1.8V\_powerdown}$ | Current for VDDIO (1.8/3.3V) supplies for power down mode | | - | 300 | 600 | μA |
| $I_{DD\_3.3V\_powerdown}$ | Current for VCIP supply for power down mode | | - | 0 | 10 | μA |
| $V_{OH\ (CMOS)}$ | Output High Voltage (CMOS) | $I_{OH}$ = -2 ~ -16 mA | VDDIO*80% | - | - | V |
| $V_{OL\ (CMOS)}$ | Output Low Voltage (CMOS) | $I_{OL}$= 2 ~ 16 mA | - | - | VDDIO*15% | V |
| $V_{IH\ (CMOS)}$ | Input High Voltage (CMOS) | - | VDDIO*70% | - | - | V |
| $V_{IL\ (CMOS)}$ | Input Low Voltage (CMOS) | - | - | - | VDDIO*20% | V |
| $I_{OZ}$ | Tri-state Output Leakage Current | - | -1 | - | +1 | μA |
| $I_{IN}$ | Input Leakage Current | - | -1 | - | +1 | μA |

# 11 AC CHARACTERISTICS

## 11.1 Power Up Timing

| Symbol | Parameters | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $T_{VDD12\_RISE}$ | Rise time for all 1.2V VDD supplies (10% to 90%) | 1 | - | 10 | ms |
| $T_{VDD18\_RISE}$ | Rise time for all 1.8V VDD supplies (10% to 90%) | 1 | - | 10 | ms |
| $T_{VDD18\_ON}$ | Time from 1.2V supplies on to 1.8V and 3.3V supplies on.<br>Note: If VDD18 is applied before VDD12 is applied, there could be up to 1mA of additional leakage during the period when VDD12 is still not available. | -5 | - | 30 | ms |

## 11.2 RESET Timing

| Symbol | Parameters | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $T_{RESET}$ | RESET_IN “Low” Pulse Width | 10 | - | - | us |

## 11.3 Interface Timing

### 11.3.1 MCU Interface (Type A) Timing

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $T_{MC}$ | MCU clock frequency | - | - | 160 | Mhz |
| $t_{CYCLE\_WR}$ | Clock Cycle Time(Write) | 1 | - | - | $T_{MC}$ |
| $t_{CYCLE\_RD}$ | Clock Cycle Time(Read) | 8 | - | - | $T_{MC}$ |
| $pw_{CSL}$ | Control Pulse Low Width | 0.5 | - | - | $T_{MC}$ |
| $pw_{CSH}$ | Control Pulse High Width | 0.5 | - | - | $T_{MC}$ |
| $t_{AS}$ | Address Setup Time | 1.6 | - | - | ns |
| $t_{AH}$ | Address Hold Time | 1.6 | - | - | ns |
| $t_{DSW}$ | Data Setup Time | 1.6 | - | - | ns |
| $t_{DHW}$ | Data Hold Time | 1.6 | - | - | ns |
| $t_{ACC}$ | Data Access Time | 12 | - | - | ns |
| $t_{DHR}$ | Read Data Hold time | $t_{CYCLE\_RD}$ /2 | - | - | ns |
| $t_R$ | Rise time | 1 | - | - | ns |
| $t_F$ | Fall time | 1 | - | - | ns |

Table 11-1 MCU Interface (Type A) Timing Characteristics

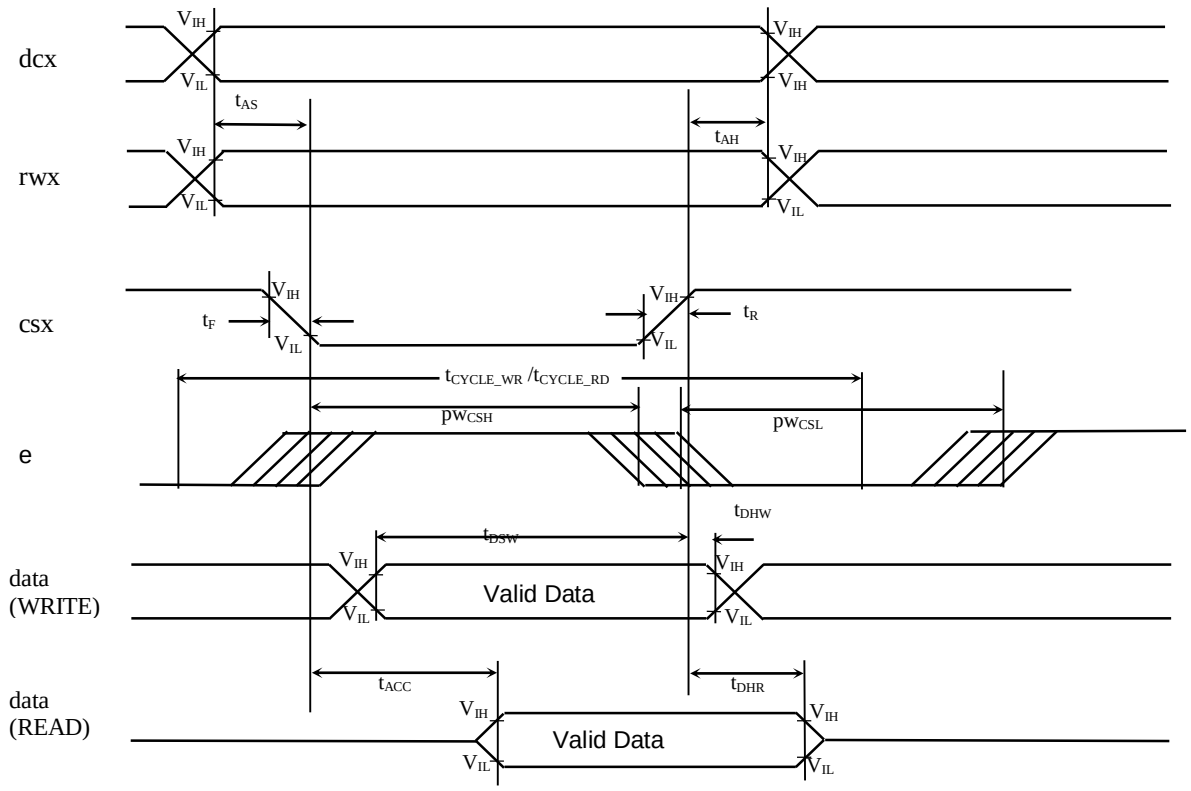



Figure 11-1 MCU Interface (Type A) Timing Diagram

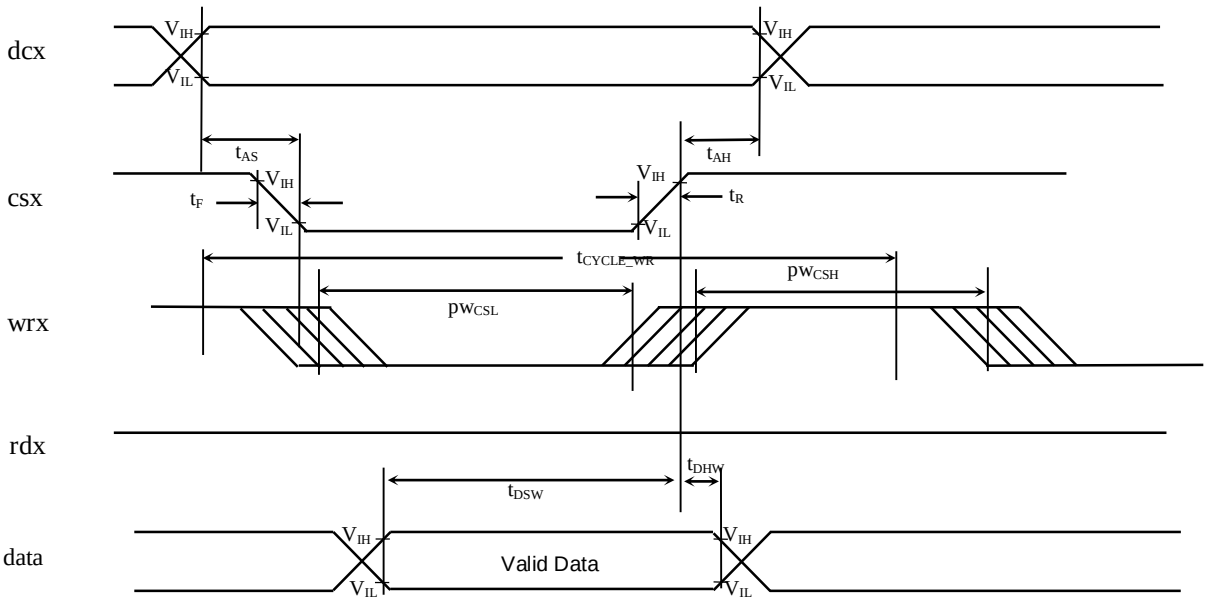
Note: $V_{IL}$ and $V_{IH}$ refers to DC CHARACTERISTICS

## 11.3.2 MCU Interface (Type B) Timing

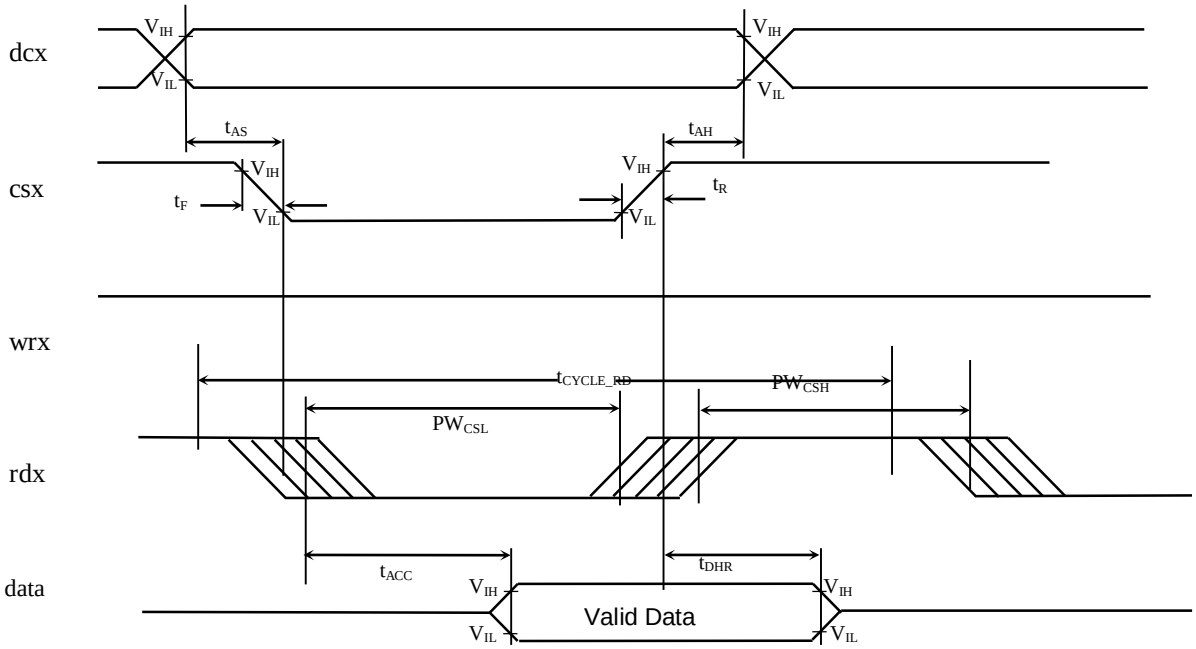
| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $T_{MC}$ | MCU Clock Frequency | - | - | 160 | Mhz |
| $t_{CYCLE\_WR}$ | Clock Cycle Time(Write) | 1 | - | - | $T_{MC}$ |
| $t_{CYCLE\_RD}$ | Clock Cycle Time(Read) | 8 | - | - | $T_{MC}$ |
| $pw_{CSL}$ | Control Pulse Low Width | 0.5 | - | - | $T_{MC}$ |
| $pw_{CSH}$ | Control Pulse High Width | 0.5 | - | - | $T_{MC}$ |
| $t_{AS}$ | Address Setup Time | 1.6 | - | - | ns |
| $t_{AH}$ | Address Hold Time | 1.6 | - | - | ns |
| $t_{DSW}$ | Data Setup Time | 1.6 | - | - | ns |
| $t_{DHW}$ | Data Hold Time | 1.6 | - | - | ns |
| $t_{ACC}$ | Data Access Time | 12 | - | - | ns |
| $t_{DHR}$ | Read Data Hold time | $t_{CYCLE\_RD}$ /2 | - | - | ns |
| $t_R$ | Rise time | 1 | - | - | Ns |
| $t_F$ | Fall time | 1 | - | - | Ns |

Table 11-2: MCU Interface (Type B) Timing Characteristics

**Write**



**Read**




Note: $V_{IL}$ and $V_{IH}$ refers to DC CHARACTERISTICS

## 11.3.3 **SPI Interface Timing**

| Symbol | Parameters | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $T_{MC}$ | SPI Clock Frequency | - | - | 50 | Mhz |
| $t_{CYCLE\_WR}$ | Clock Cycle Time(Write) | 1 | - | - | $T_{MC}$ |
| $t_{CYCLE\_RD}$ | Clock Cycle Time(Read) | 4 | - | - | $T_{MC}$ |
| $t_{CSS}$ | Chip Select Setup Time | 2 | - | - | ns |
| $t_{CSH}$ | Chip Select Hold Time | 4 | - | - | ns |
| $t_{DCS}$ | Chip Select Setup Time(for 4 wire 8 bit mode) | 2 | - | - | ns |
| $t_{DCH}$ | Chip Select Hold Time(for 4 wire 8 bit mode) | 4 | - | - | ns |
| $t_{DSW}$ | Write Data Setup Time | 2 | - | - | ns |
| $t_{OHW}$ | Write Data Hold Time | 4 | - | - | ns |
| $t_{ACC}$ | Read Data Access Time | 8 | - | - | ns |
| $t_{DHR}$ | Read Data Hold Time | $t_{CYCLE\_RD}$ /2 | - | - | ns |
| $t_{CSWD}$ | Chip Select Write Delay Time | 1 | - | - | $T_{BC}$ |
| $t_{CSRD}$ | Chip Select Read Delay Time | 1 | - | - | $T_{BC}$ |
| $t_R$ | Rise time | 1 | - | - | ns |
| $t_F$ | Fall time | 1 | - | - | ns |

Figure 11-2 SPI Interface Timing Characteristics

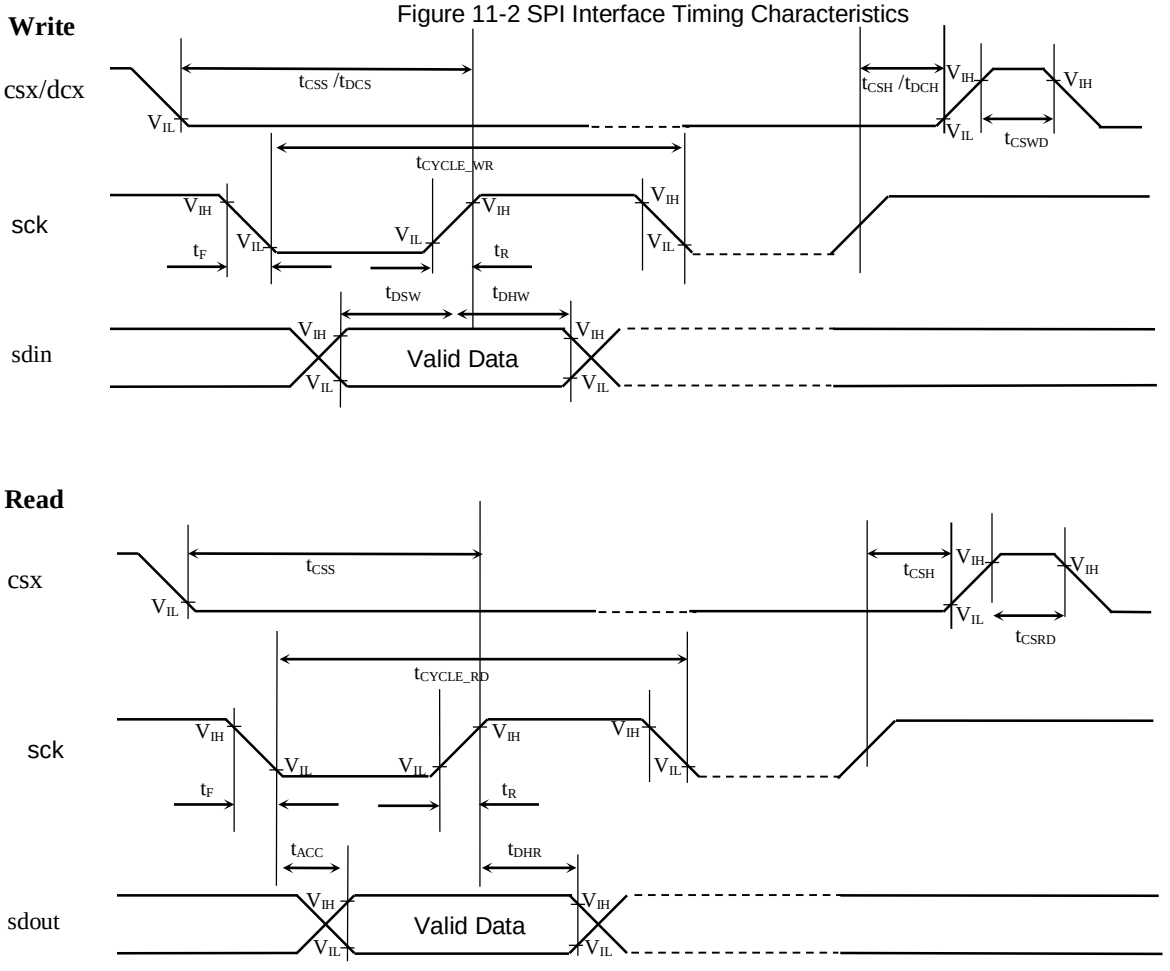



Figure 11-3: SPI Interface Timing Diagram

Note: $V_{IL}$ and $V_{IH}$ refers to DC CHARACTERISTICS

## 11.3.4 RGB Interface Timing

| Symbol | Parameters | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| $t_{pclk}$ | pclk Period | 5.5 | - | - | ns |
| $t_{vsys}$ | Vertical Sync Setup Time | 1.7 | - | - | ns |
| $t_{vsyh}$ | Vertical Sync Hold Time | 1.7 | - | - | ns |
| $t_{hsys}$ | Horizontal Sync Setup Time | 1.7 | - | - | ns |
| $t_{hsyh}$ | Horizontal Sync Hold Time | 1.7 | - | - | ns |
| $t_{hv}$ | Phase difference of Sync Signal Falling Edge | -1 | 0 | 1 | $t_{pclk}$ |
| $t_{ds}$ | Data Setup Time | 1.7 | - | - | ns |
| $t_{dh}$ | Data hold Time | 1.7 | - | - | ns |
| $t_{ch2ch}$ | Phase difference of channel 0 and 1 | -1 | 0 | 2 | $t_{pclk}$ |

Table 11-3 RGB Interface Timing Characteristics

**Note:** The link should run at greater or equal than the pclk frequency * bit per pixel (bpp).

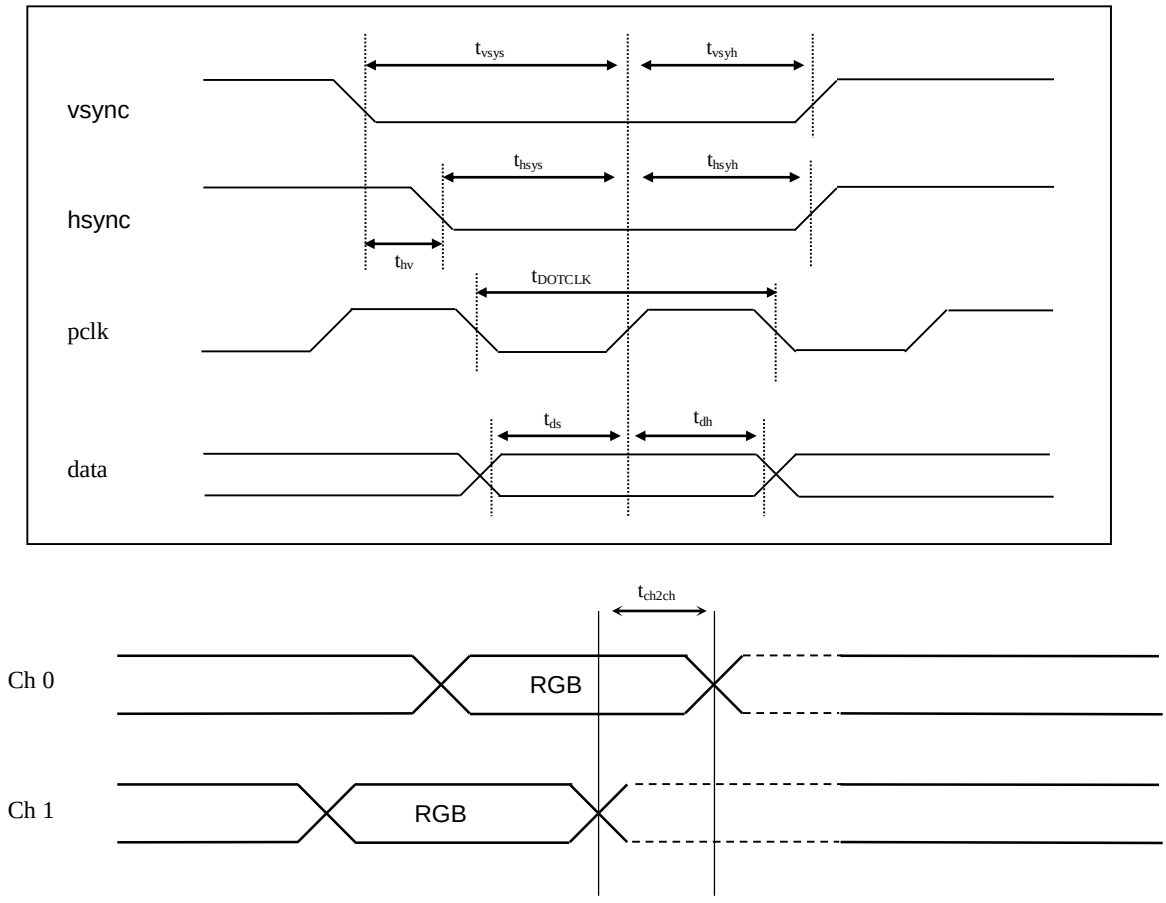
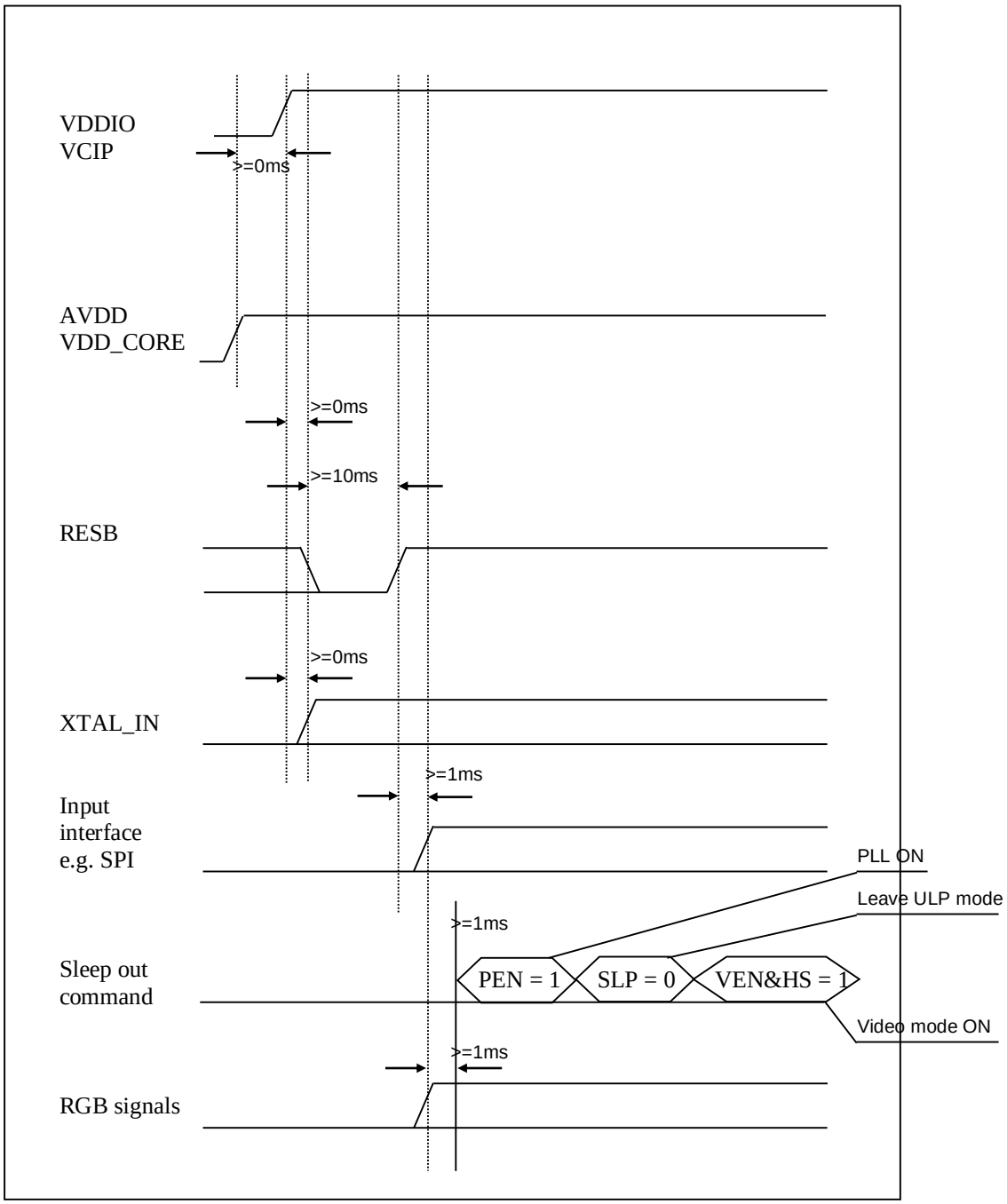
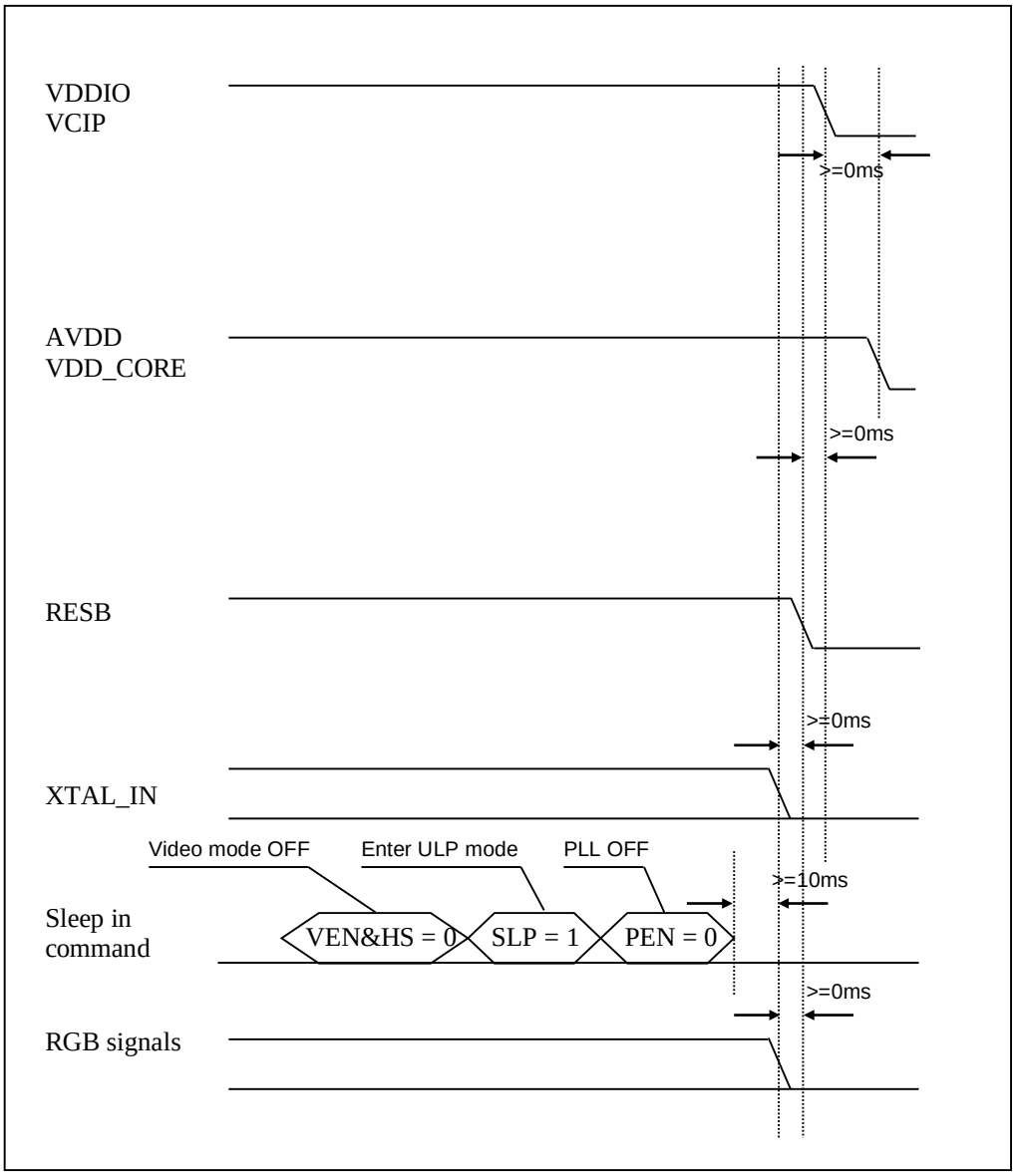



Figure 11-4: RGB Interface Timing Diagram

## 12 POWER UP SEQUENCE



## 13 POPWER OFF SEQUENCE



# 14 MIPI DPHY CHARACTERISTICS

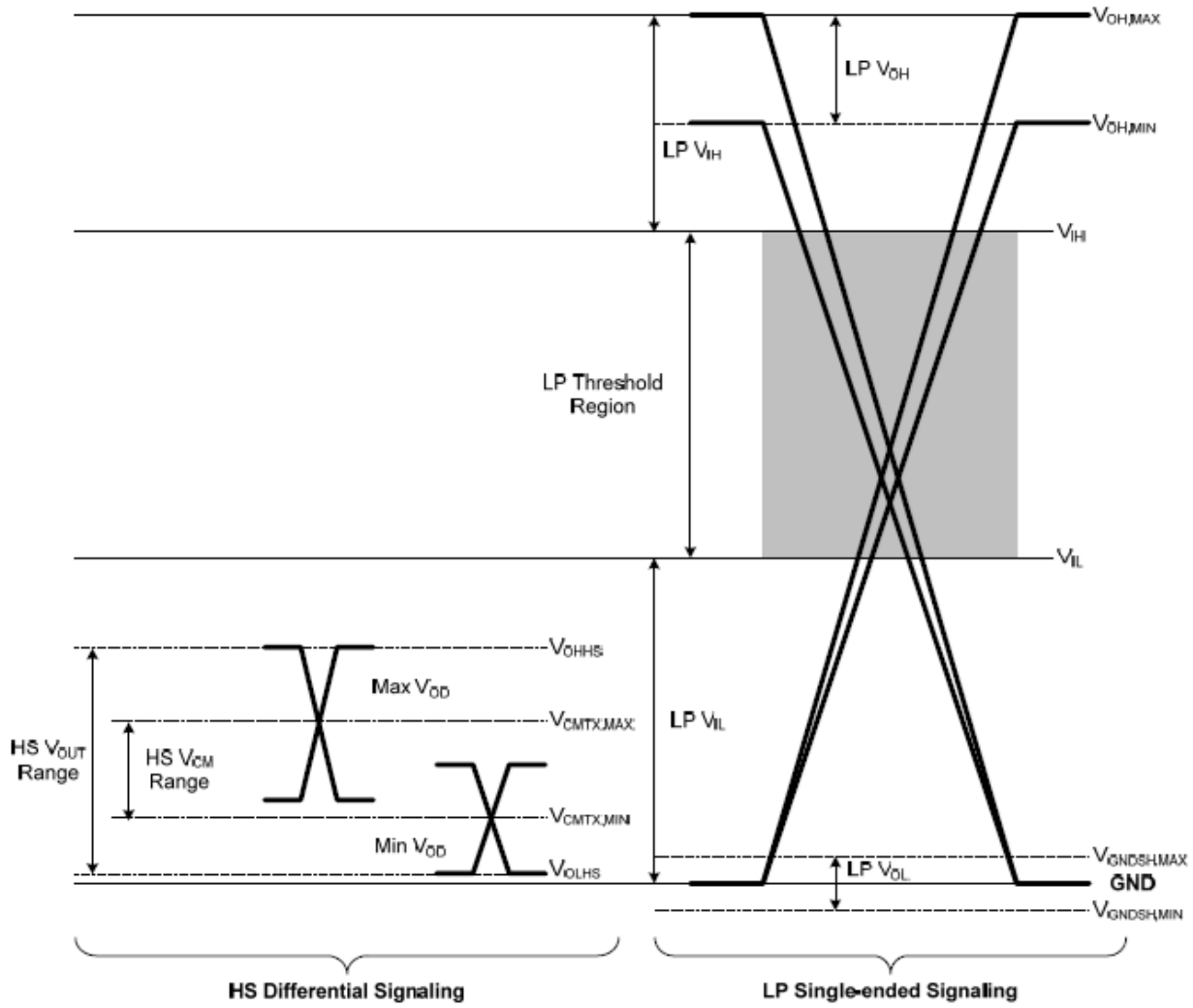



**Figure 14-1 D-PHY Signaling Levels**

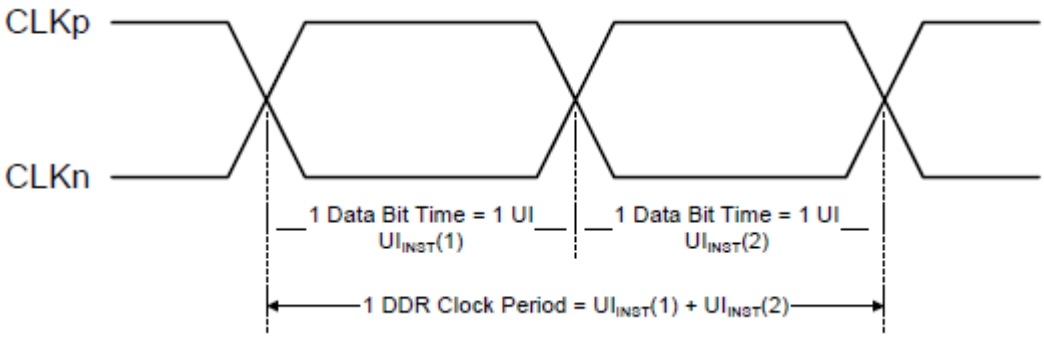



**Figure 14-2 DDR Clock Definition**

**Table 14-1 : Clock Signal Specification**

| Clock Parameter | Symbol | Min | Typ | Max | Units | Notes |
|---|---|---|---|---|---|---|
| UI instantaneous | $UI_{INST}$ | | | 12.5 | ns | 1,2 |
| UI variation | ΔUI | -10% | | 10% | UI | 3 |
| | | -5% | | 5% | UI | 4 |

*Notes:*

1. *This value corresponds to a minimum 80 Mbps data rate.*
2. *The minimum UI shall not be violated for any single bit period, i.e., any DDR half cycle within a data burst. The allowed instantaneous UI variation can cause instantaneous data rate variations. Therefore, devices should either accommodate these instantaneous variations with appropriate FIFO logic outside of the PHY or provide an accurate clock source to the Lane Module to eliminate these instantaneous variations.*
3. *When UI ≥ 1ns, within a single burst.*
4. *When UI < 1ns, within a single burst.*

## 14.1 MIPI DPHY HS CHARACTERISTICS

**Table 14-2 HS Transmitter DC Specifications**

| Parameter | Description | Min | Nom | Max | Units | Notes |
|---|---|---|---|---|---|---|
| $V_{CMTX}$ | HS transmit static common-mode voltage | 150 | 200 | 250 | mV | 1 |
| $\|\Delta V_{CMTX(1,0)}\|$ | $V_{CMTX}$ mismatch when output is Differential-1 or Differential-0 | | | 5 | mV | 2 |
| $\|V_{OD}\|$ | HS transmit differential voltage | 140 | 200 | 270 | mV | 1 |
| $\|\Delta V_{OD}\|$ | $V_{OD}$ mismatch when output is Differential-1 or Differential-0 | | | 14 | mV | 2 |
| $V_{OHHS}$ | HS output high voltage | | | 360 | mV | 1 |
| $Z_{OS}$ | Single ended output impedance | 40 | 50 | 62.5 | Ω | |
| $\Delta Z_{OS}$ | Single ended output impedance mismatch | | | 10 | % | |

*Notes:*

1. *Value when driving into load impedance anywhere in the $Z_{ID}$ range.*
2. *A transmitter should minimize $\Delta V_{OD}$ and $\Delta V_{CMTX(1,0)}$ in order to minimize radiation, and optimize signal integrity.*

**Table 14-3 HS Transmitter AC Specifications**

| Parameter | Description | Min | Nom | Max | Units | Notes |
|---|---|---|---|---|---|---|
| $\Delta V_{CMTX(HF)}$ | Common-level variations above 450MHz | | | 15 | $mV_{RMS}$ | |
| $\Delta V_{CMTX(LF)}$ | Common-level variation between 50-450MHz | | | 25 | $mV_{PEAK}$ | |
| $t_R$ and $t_F$ | 20%-80% rise time and fall time | | | 0.3 | UI | 1, 2 |
| | | | | 0.35 | UI | 1, 3 |
| | | 100 | | | ps | 4 |

*Notes:*

1. *UI is equal to 1/(2*fh). See Section 8.3 for the definition of fh.*
2. *Applicable when operating at HS bit rates ≤ 1 Gbps (UI ≥ 1 ns).*
3. *Applicable when operating at HS bit rates > 1 Gbps (UI < 1 ns).*
4. *Applicable for all HS bit rates. However, to avoid excessive radiation, bit rates ≤ 1 Gbps (UI ≥ 1 ns), should not use values below 150 ps.*

**Table 14-4 LP Transmitter DC Specifications**

| Parameter | Description | Min | Nom | Max | Units | Notes |
|---|---|---|---|---|---|---|
| $V_{OH}$ | Thevenin output high level | 1.1 | 1.2 | 1.3 | V | |
| $V_{OL}$ | Thevenin output low level | -50 | | 50 | mV | |
| $Z_{OLP}$ | Output impedance of LP transmitter | 110 | | | Ω | 1, 2 |

*Notes:*

1. *See Figure 42 and Figure 43.*
2. *Though no maximum value for $Z_{OLP}$ is specified, the LP transmitter output impedance shall ensure the $T_{RLP}/T_{FLP}$ specification is met.*

**Table 14-5 LP Transmitter AC Specifications**

| Parameter | Description | | Min | Nom | Max | Units | Notes |
|---|---|---|---|---|---|---|---|
| $T_{RLP}/T_{FLP}$ | 15%-85% rise time and fall time | | | | 25 | ns | 1 |
| $T_{REOT}$ | 30%-85% rise time and fall time | | | | 35 | ns | 1, 5, 6 |
| $T_{LP-PULSE-TX}$ | Pulse width of the LP exclusive-OR clock | First LP exclusive-OR clock pulse after Stop state or last pulse before Stop state | 40 | | | ns | 4 |
| | | All other pulses | 20 | | | ns | 4 |
| $T_{LP-PER-TX}$ | Period of the LP exclusive-OR clock | | 90 | | | ns | |
| $\delta V/\delta t_{SR}$ | Slew rate @ $C_{LOAD}$ = 0pF | | | | 500 | mV/ns | 1, 3, 7, 8 |
| | Slew rate @ $C_{LOAD}$ = 5pF | | | | 300 | mV/ns | 1, 3, 7, 8 |
| | Slew rate @ $C_{LOAD}$ = 20pF | | | | 250 | mV/ns | 1, 3, 7, 8 |
| | Slew rate @ $C_{LOAD}$ = 70pF | | | | 150 | mV/ns | 1, 3, 7, 8 |
| | Slew rate @ $C_{LOAD}$ = 0 to 70pF (Falling Edge Only) | | 30 | | | mV/ns | 1, 2, 3 |
| | Slew rate @ $C_{LOAD}$ = 0 to 70pF (Rising Edge Only) | | 30 | | | mV/ns | 1, 3, 9 |
| | Slew rate @ $C_{LOAD}$ = 0 to 70pF (Rising Edge Only) | | 30 – 0.075 * ($V_{O,INST}$ – 700) | | | mV/ns | 1, 3, 10, 11 |
| $C_{LOAD}$ | Load capacitance | | 0 | | 70 | pF | 1 |

*Notes:*

1. *$C_{LOAD}$ includes the low-frequency equivalent transmission line capacitance. The capacitance of TX and RX are assumed to always be <10pF. The distributed line capacitance can be up to 50pF for a transmission line with 2ns delay.*
2. *When the output voltage is between 400 mV and 930 mV.*
3. *Measured as average across any 50 mV segment of the output signal transition.*
4. *This parameter value can be lower than $T_{LPX}$ due to differences in rise vs. fall signal slopes and trip levels and mismatches between Dp and Dn LP transmitters. Any LP exclusive-OR pulse observed during HS EoT (transition from HS level to LP-11) is glitch behavior as described in Section 9.2.2.*
5. *The rise-time of $T_{REOT}$ starts from the HS common-level at the moment the differential amplitude drops below 70mV, due to stopping the differential drive.*
6. *With an additional load capacitance $C_{CM}$ between 0 and 60 pF on the termination center tap at RX side of the Lane*

# 15 OPERATING MODE

## 15.1 Programming Model

The table below shows the local register map summary in SSD2829.

**Table 15-1: SSD2829 Local Register Map**

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>0xB0 – 0xDF</td><td>Some of the commands would have additional data parameters added to support extension of certain register fields. For example VBP (Vertical back porch) is an 8-bit field. With the extension of the number of data parameters, VBP can now become a 16-bit field. The original register fields' location would be maintained in the first 2 data parameters for back-ward compatibility purpose. Only 2 data parameters would be added.</td></tr>
<tr><td>0xE0</td><td>This is the command for APB peripheral access (e.g. SCM)<br><br>If all 6 data parameters are given, SSD2829 would issue APB write access with the APB_ADDR and APB_DATA.<br><br>If only 2 data parameters are given, SSD2829 would store the APB_ADDR for read operation. Host can do a data read to read back the APB_DATA.<br><br>
<table>
<tr><td>Data Parameter</td><td>Description</td></tr>
<tr><td>1</td><td>APB_ADDR[7:0]</td></tr>
<tr><td>2</td><td>APB_ADDR[15:8]</td></tr>
<tr><td>3</td><td>APB_DATA[7:0]</td></tr>
<tr><td>4</td><td>APB_DATA[15:8]</td></tr>
<tr><td>5</td><td>APB_DATA[23:16]</td></tr>
<tr><td>6</td><td>APB_DATA[31:24]</td></tr>
</table>
</td></tr>
<tr><td>0xE1 – 0xFE</td><td>Reserved</td></tr>
</table>

### 15.1.1 Access Local (non-APB) Registers

The legacy registers (16-bit accessed) are accessed in term of 2 bytes per cycle for all MCU interfaces, except 8-bit format which requires 3 cycles to access (1 command, 2 data cycles)

In the first write cycle, only 8-bit data are written into the SSD2829, as the address can only be 8-bit. No matter whether the interface is 8-bit, 16-bit and 24-bit lower 8 bits are used. Please refer to the table below.

| Interface types | Data pins | | | |
|---|---|---|---|---|
| | D53-D30 | D23-D16 | D15-D8 | D7-D0 |
| 24-bit, 16-bit, 8-bit | Don't care | Don't care | Don't care | Address |

**Table 15-2: MCU Interface Data Pin Mapping for Command Cycle for Legacy Registers**

In the sub-sequent read or write cycle, the data width is only 16-bit or 2 bytes no matter the interface width is selected, except for 8-bit format. Please refer to the table below.

If there are 4 bytes in the legacy registers due to data parameters extension, there will be additional data cycles accordingly.

| Interface types | Cycle | Data pins | | | | Note |
|---|---|---|---|---|---|---|
| | | D53-D30 | D23-D16 | D15-D8 | D7-D0 | |
| 24-bit, 16-bit | 1st | Don't care | Don't care | Data Byte 1 | Data Byte 0 | (1) |
| | 2nd | Don't care | Don't care | Data Byte 3 | Data Byte 2 | (2) |
| 8-bit | 1st | Don't care | Don't care | Don't care | Data Byte 0 | (1) |
| | 2nd | Don't care | Don't care | Don't care | Data Byte 1 | (2) |
| | 3rd | Don't care | Don't care | Don't care | Data Byte 2 | (3) |
| | 4th | Don't care | Don't care | Don't care | Data Byte 3 | |

**Table 15-3: MCU Interface Data Pin Mapping for Legacy Register**

Note:

(1) If the local registers have 4 bytes of data, host can either write 2 bytes or 4 bytes of data.

(2) If the local registers have only 2-bytes of data, any extra write will be ignored

(3) For 8-bit interface, all writes must be in multiple of 2 cycles

### 15.1.2 Access Local (APB) Registers for Write

The APB peripheral's registers are accessed by 0xE0. The data content of 0xE0 are 6 bytes and arranged in the order of {addr low, addr high, data0, data1, data2, data3}, where addr low is sent first.

In the first write cycle, only 8-bit data are written into the SSD2829, as the address can only be 8-bit. No matter whether the interface is 8-bit, 16-bit and 24-bit, lower 8 bits are used. Please refer to the table below.

| Interface types | Data pins | | | |
|---|---|---|---|---|
| | D53-D30 | D23-D16 | D15-D8 | D7-D0 |
| 24-bit, 16-bit, 8-bit | Don't care | Don't care | Don't care | 0xE0 |

**Table 15-4: MCU Interface Data Pin Mapping for Command Cycle for Extended Registers Write**

In the sub-sequent write cycle, the data width is only 16-bit or 2 bytes, no matter the interface width is selected, excepted for 8-bit format. Please refer to the table below.

| Interface types | Cycle | Data pins | | | |
|---|---|---|---|---|---|
| | | D53-D30 | D23-D16 | D15-D8 | D7-D0 |
| 24-bit, 16-bit | 1st | Don't care | Don't care | Addr High | Addr Low |
| | 2nd | Don't care | Don't care | Data Byte 1 | Data Byte 0 |
| | 3rd | Don't care | Don't care | Data Byte 3 | Data Byte 2 |
| 8-bit | 1st | Don't care | Don't care | Don't care | Addr Low |
| | 2nd | Don't care | Don't care | Don't care | Addr High |
| | 3rd | Don't care | Don't care | Don't care | Data Byte 0 |
| | 4th | Don't care | Don't care | Don't care | Data Byte 1 |
| | 5th | Don't care | Don't care | Don't care | Data Byte 2 |
| | 6th | Don't care | Don't care | Don't care | Data Byte 3 |

**Table 15-5: MCU Interface Data Pin Mapping for Extended Registers Write**

### 15.1.3 Access APB Registers for Read

The APB peripheral's registers are accessed by 0xE0. The content of 0xE0 are 2 bytes and arranged in the order of {addr low, addr high}, where addr low is sent first.

In the first write cycle, only 8-bit data are written into the SSD2829, as the address can only be 8-bit. No matter whether the interface is 8-bit, 16-bit or 24-bit, lower 8 bits are used. Please refer to the table below.

| Interface types | Data pins | | | |
|---|---|---|---|---|
| | D53-D30 | D23-D16 | D15-D8 | D7-D0 |
| 24-bit, 16-bit, 8-bit | Don't care | Don't care | Don't care | 0xE0 |

**Table 15-6: MCU Interface Data Pin Mapping for Command Cycle for Extended Registers Read**

Prior to the read cycles, the host must set the address of the extended registers to be read through write 0xE1. In the sub-sequent write cycle, the data width is only 16-bit or 2 bytes no matter what interface width is selected, except for 8-bit format. Please refer to the table below.

| Interface types | Cycle | Data pins | | | |
|---|---|---|---|---|---|
| | | D53-D30 | D23-D16 | D15-D8 | D7-D0 |
| 24-bit, 16-bit | 1st | Don't care | Don't care | Addr High | Addr Low |
| 8-bit | 1st | Don't care | Don't care | Don't care | Addr Low |
| | 2nd | Don't care | Don't care | Don't care | Addr High |

**Table 15-7: MCU Interface Data Pin Mapping for Extended Registers Address Set**

For the read cycles, the host read the data in 16-bit or 2 bytes format no matter what interface width is selected except for 8-bit format. Please refer to the table below. Please refer to the table below.

| Interface types | Cycle | Data pins | | | |
|---|---|---|---|---|---|
| | | D53-D30 | D23-D16 | D15-D8 | D7-D0 |
| 24-bit, 16-bit | 1st | Don't care | Don't care | Read Data 1 | Read Data 0 |
| | 2nd | Don't care | Don't care | Read Data 3 | Read Data 2 |
| 8-bit | 1st | Don't care | Don't care | Don't care | Read Data 0 |
| | 2nd | Don't care | Don't care | Don't care | Read Data 1 |
| | 3rd | Don't care | Don't care | Don't care | Read Data 2 |
| | 4th | Don't care | Don't care | Don't care | Read Data 3 |

**Table 15-8: MCU Interface Data Pin Mapping for Extended Registers Address Set**

## 15.2 SPI Interface

SSD2829 supports three types of SPI interface,

8-Bit 3 wire (type C option 1, DBI 2.0)
8-Bit 4 wire (type C option 3, DBI 2.0)
24-bit 3 wire

The selection is controlled by ps[1:0] pins. The least significant byte should be written first

### 15.2.1 SPI Interface 8-Bit 4 Wire

This interface consists of sdc, sck, sdin, sdout and csx. It only supports 8-Bit data. Each cycle contains 8-Bit data. The first cycle should be a command write cycle to specify the register address for access. The subsequent cycles are read or write cycles for read or write operations.

The csx should be driven from 1 to 0 to start an operation and from 0 to 1 to end an operation. During 1 operation, the application processor can write or read multiple bytes.

sdcx indicates whether the operation is for data or command. When sdcx is 1, the operation is for data. When sdcx is 0, the operation is for command. sdcx is sampled at every 8$^{th}$ rising edge of sck during 1 operation.

During write operation, sdin will be sampled by SSD2829 at the rising edge of sck. The first rising edge of sck after the falling edge of csx samples the bit 7 of the 8-Bit data. The second rising edge of sck samples the bit 6 of the 8-Bit data, and so on. The value of sdcx is sampled at the 8$^{th}$ rising edge of sck, together with bit 0 of the 8-Bit data. Please see the diagram below for illustration. Optionally, the csx can be driven to 1 in between cycles.

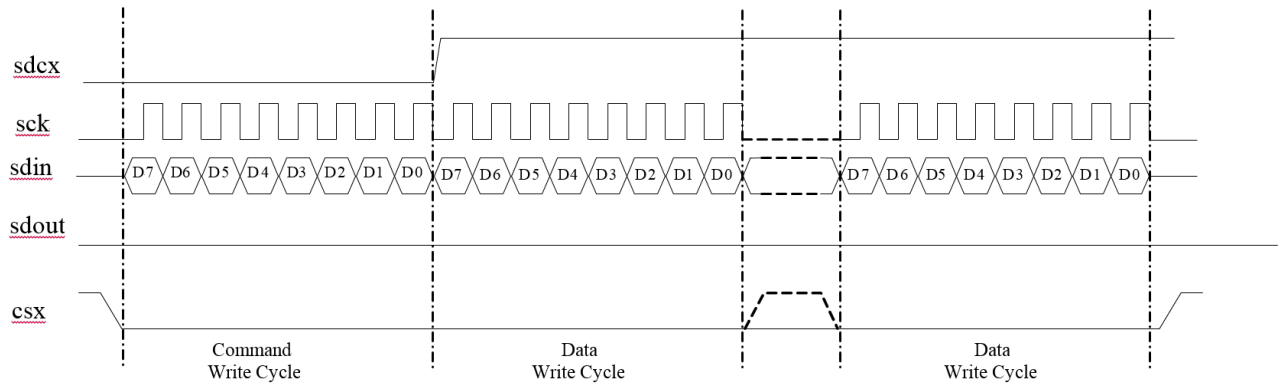



**Figure 15-1: SPI Interface 8-bit 4 wire for write**

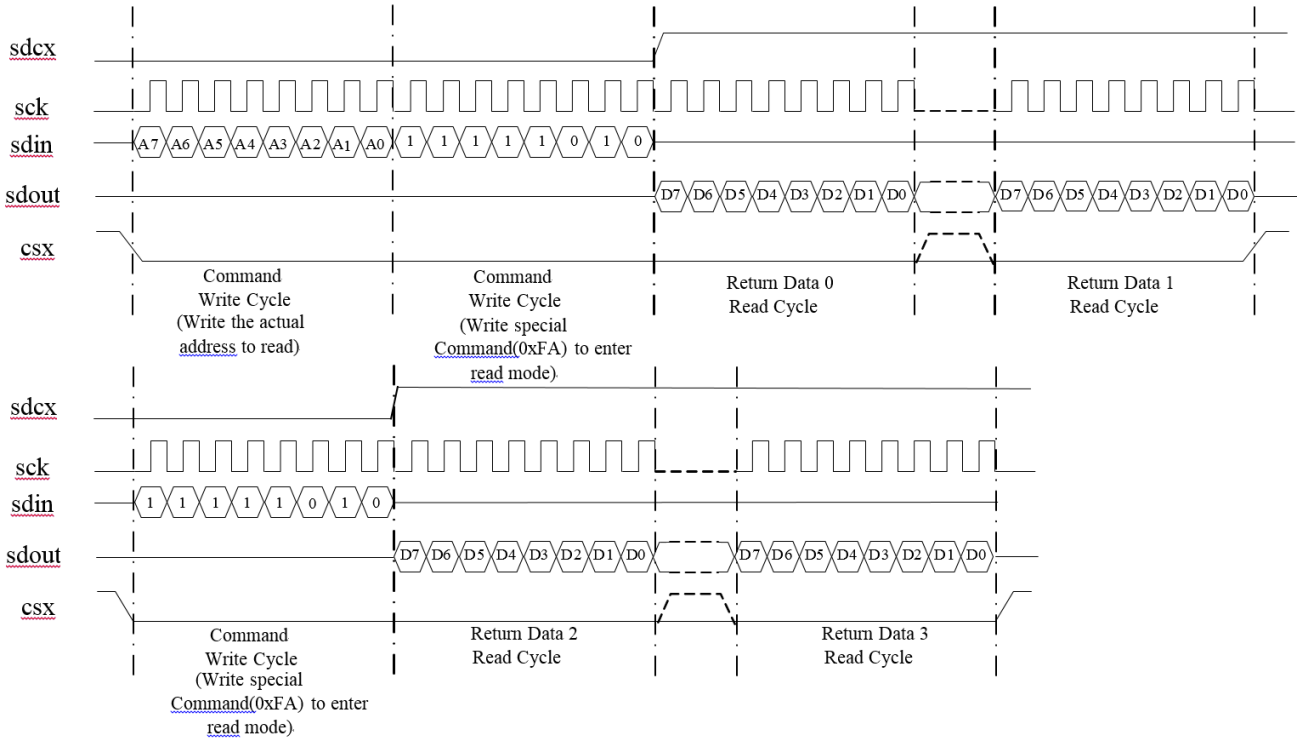



**Figure 15-2: SPI Interface 8-bit 4 wire for read**

### 15.2.2 SPI Interface 8-Bit 3 Wire

This interface consists of sck, sdin, sdout and csx. It only supports 8-Bit data. Each cycle contains 8-Bit data. The first cycle should be a write cycle to specify the register address for access. The subsequent cycles are read or write cycles for read or write operations.

The csx should be driven from 1 to 0 to start an operation and from 0 to 1 to end an operation. During 1 operation, the application processor can write or read multiple bytes.

Instead of sdcx, an sdcx bit is used to indicate whether the operation is for data or command. Each byte is associated with an sdcx bit. When sdcx is 1, the operation is for display data. When sdcx is 0, the operation is for command. The sdcx bit is sent priori to each byte. In other words, the sdcx bit is the first bit of every 9 bits during 1 operation.

During write operation, sdin will be sampled by SSD2829 at the rising edge of sck. The first rising edge of sck after the falling edge of csx samples the sdcx bit. The second rising edge samples bit 7 of the 8-Bit data. The third rising edge of sck samples the bit 6 of the 8-Bit data, and so on. Please see the diagram below for illustration. Optionally, the csx can be driven to 1 in between cycles.

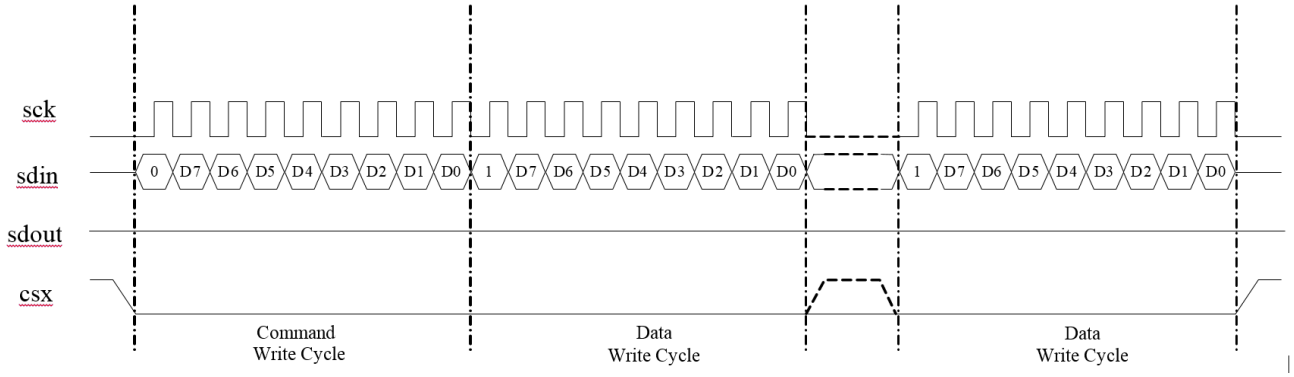



**Figure 15-3: SPI Interface 8-bit 3 wire for write**

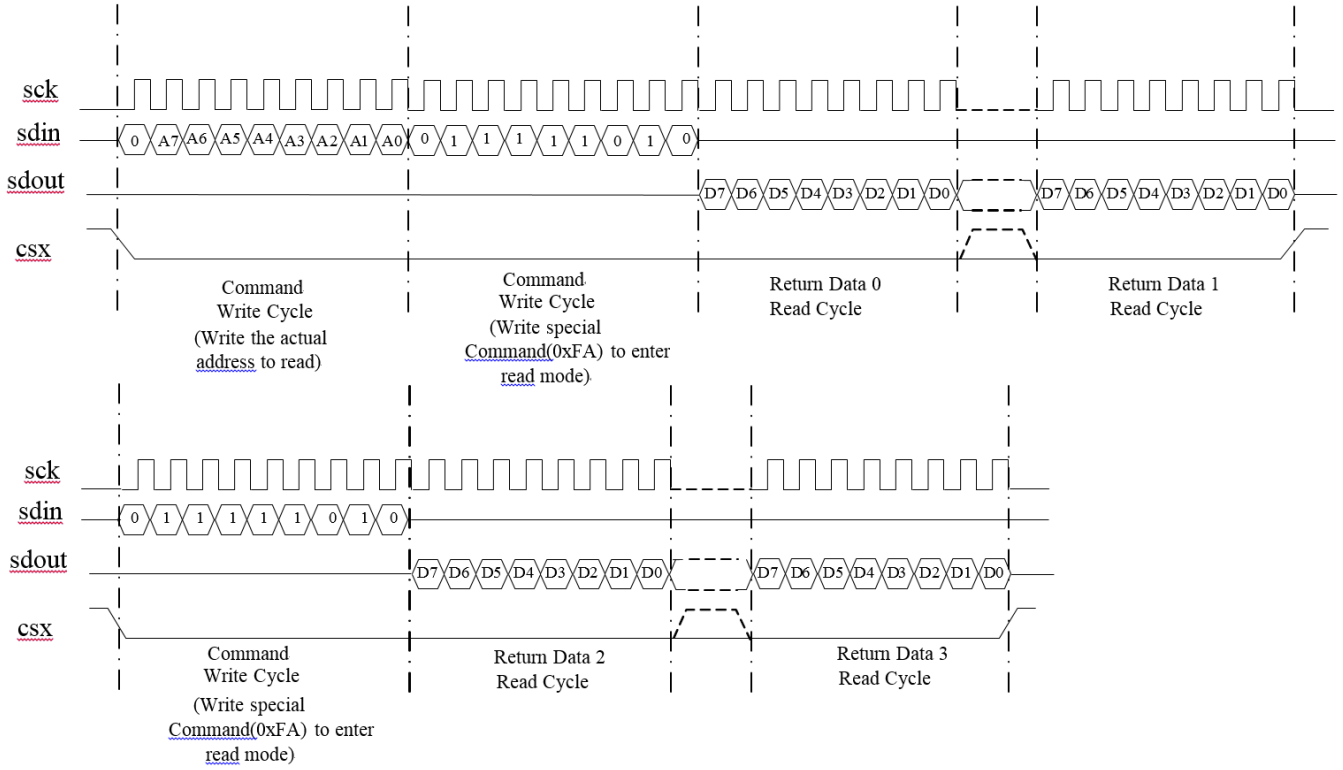



**Figure 15-4: SPI Interface 8-bit 3 wire for read**

### 15.2.3 SPI Interface 24-Bit 3 Wire

This interface consists of sck, sdin, sdout and csx. It only supports 16-bit data. Each cycle contains 16-bit data. The first cycle should be a write cycle to specify the register address for access. The subsequent cycles are read or write cycles for read or write operations.

The csx should be driven from 1 to 0 to start cycle and from 0 to 1 to end a cycle. During 1 operation, the application processor can have multiple write or read cycles. However, the csx must go from 0 to 1 at the end of each cycle.

Each cycle contains 24-bit data. Among the 24-bit data, the first 8-Bit are for control purpose and the next 16-bit are the actual data. The first 6 bits are the ID bit for SSD2829, which must be 011100. If this field does not match, the cycle will not be taken in. The 7th bit is the sdcx bit which is the same as the 8-Bit 3 wire interface. The 8th bit is the RW bit which indicates whether the current cycle is a read or write cycle. When RW is 1, the cycle is a read cycle. When RW is 0, the cycle is a write cycle.

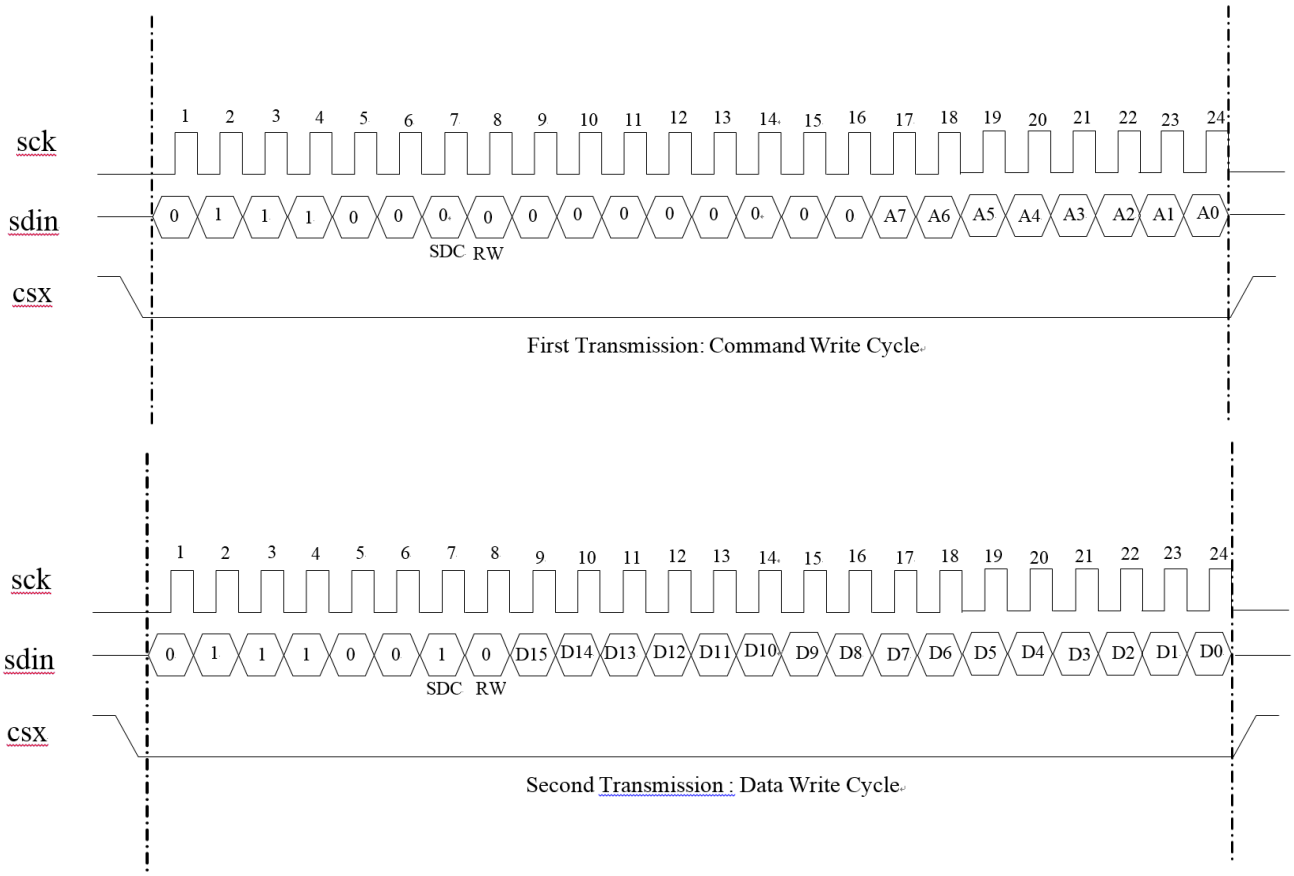



**Figure 15-5: SPI Interface 24-bit 3 wire for write**

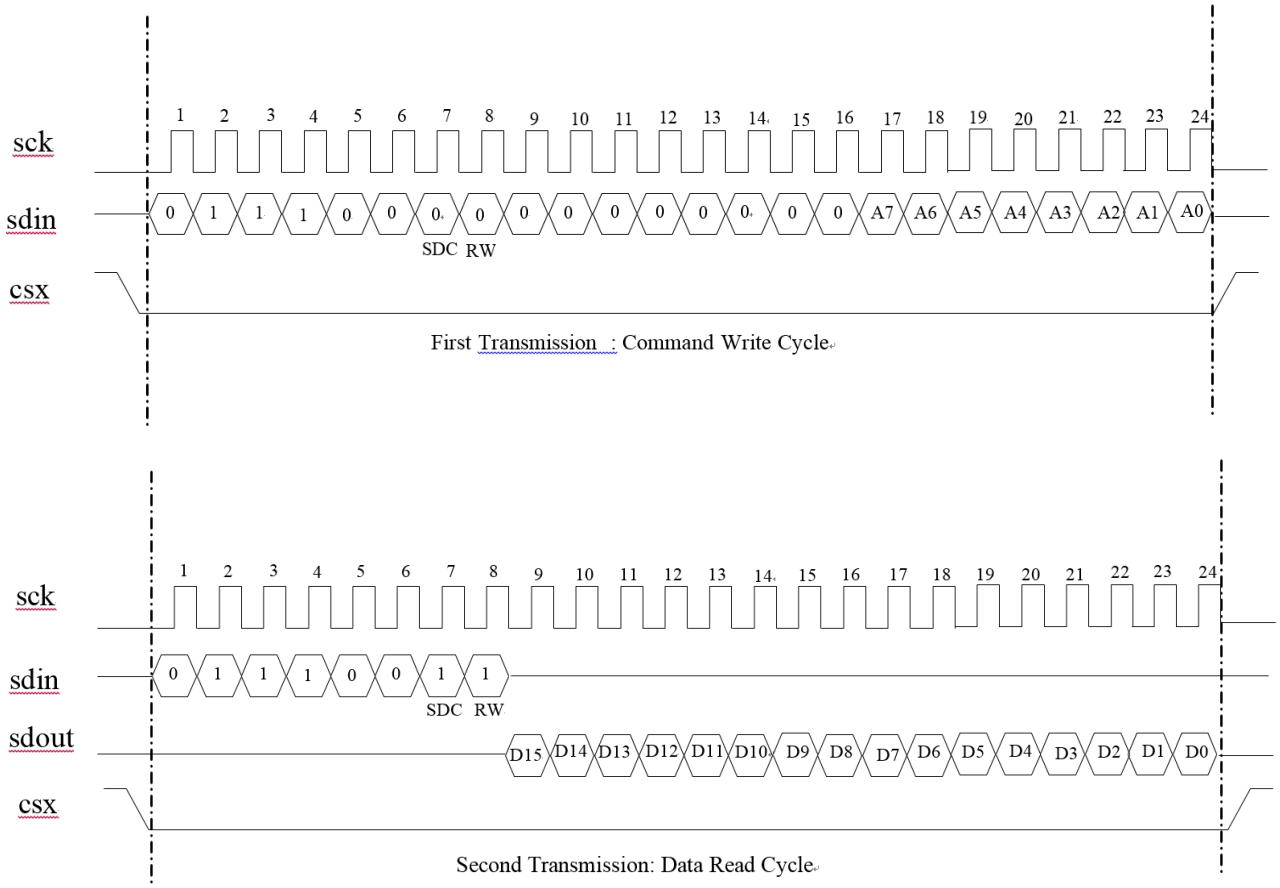



**Figure 15-6: SPI Interface 24-bit 3 wire for read**

## 15.3 MCU Interface

The SSD2829 supports three types of MCU interface,

Type A, fixed E mode, DBI 2.0
Type A, clocked E mode, DBI 2.0
Type B, DBI 2.0

The selection is controlled by ps[4:2] pins.

PS[4:2] is for the MCU interface

- 000: 8-Bit MCU interface (MIPI DBI type B)
- 001: 16-bit MCU interface (MIPI DBI type B)
- 010: 8-Bit MCU interface (MIPI DBI type A, fixed E or clocked E mode)
- 011: 16-bit MCU interface (MIPI DBI type A, fixed E or clocked E mode)
- 100: 24-bit MCU interface (MIPI DBI type B)
- 110: 24-bit MCU interface (MIPI DBI type A, fixed E or clocked E mode)
- 101: Reserved
- 111: Reserved

MCU interfaces support 8-bit, 16-bit and 24-bit data bus. Below are the data pins used for each interface. For 8-Bit interface, the least significant byte should be written first. For 16 or 24-bit interfaces, the lease significant word should be written first.

- data[15:0] for 16-bit interface
- data[23:0] for 24-bit interface

The local registers are always accessed in 16-bit data word for the data phase of the MCU cycle, irrespective of any bus width selection.

### 15.3.1 MCU Interface Type A, fixed E mode

This interface consists of data, rwx, dcx, e and csx. It supports 24-bit, 16-bit and 8-bit data bus. The first cycle should be a command write cycle to specify the register address for access. The subsequent cycles are read or write cycles for read or write operations.

'e' signal should be driven to 1 in this mode.

rwx indicates whether the operation is a read or a write operation. When rwx is 1, the operation is a read operation. When rwx is 0, the operation is a write operation.

During write operation, dcx indicates whether the operation is for data or command. When dcx is 1, the operation is for data. When dcx is 0, the operation is for command. During the read operation, the dcx should be 1.

During the write operation, data are sampled at the rising edge of csx. During read operation, data are provided at the falling edge of csx and the application processor should use the rising edge of csx to sample.

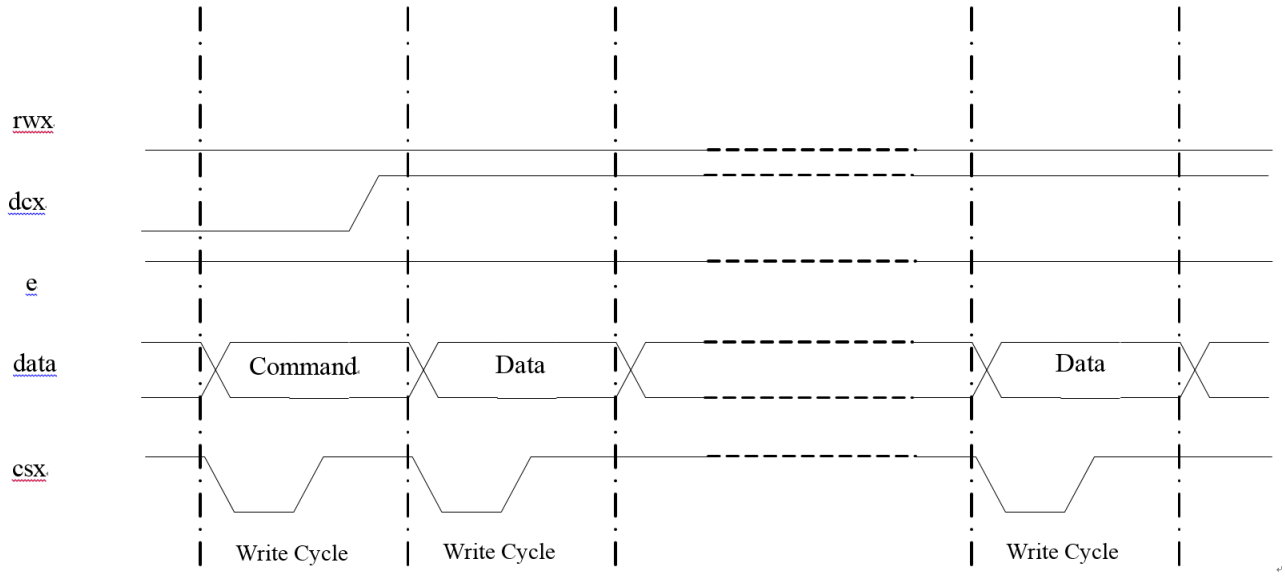



**Figure 15-7: Illustration of Write Operation for Type A, Fixed E Mode Interface**

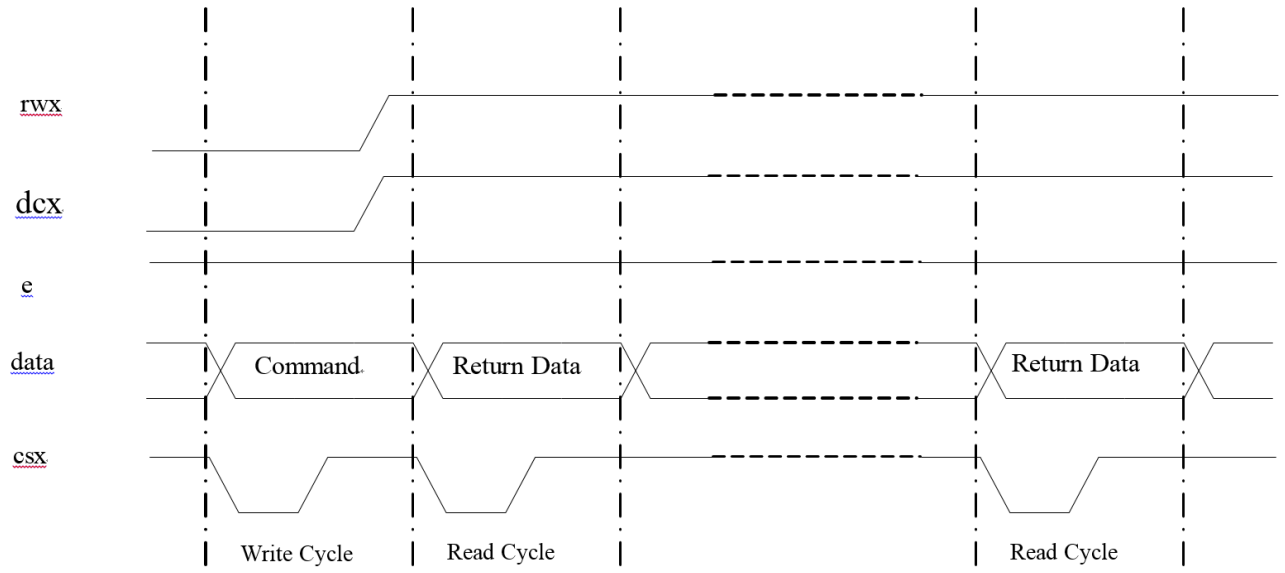



**Figure 15-8: Illustration of Read Operation for Type A, Fixed E Mode Interface**

### 15.3.2 MCU Interface Type A, Clocked E mode

This interface consists of data, rwx, dcx, e and csx. It supports 24-bit, 16-bit and 8-bit data bus. The first cycle should be a command write cycle to specify the register address for access. The subsequent cycles are read or write cycles for read or write operations.

csx should be driven to 0 in this mode.

rwx indicates whether the operation is a read or a write operation. When rwx is 0, the operation is a write operation. When rwx is 1, the operation is a read operation.

During write operation, dcx indicates whether the operation is for data or command. When dcx is 1, the operation is for data. When dcx is 0, the operation is for command. During the read operation, the dcx should be 1.

During the write operation, data are sampled at the falling edge of E. During read operation, data[23:0] are provided at the rising edge of e and the application processor should use the falling edge of e to sample.

Below is a diagram for illustration.

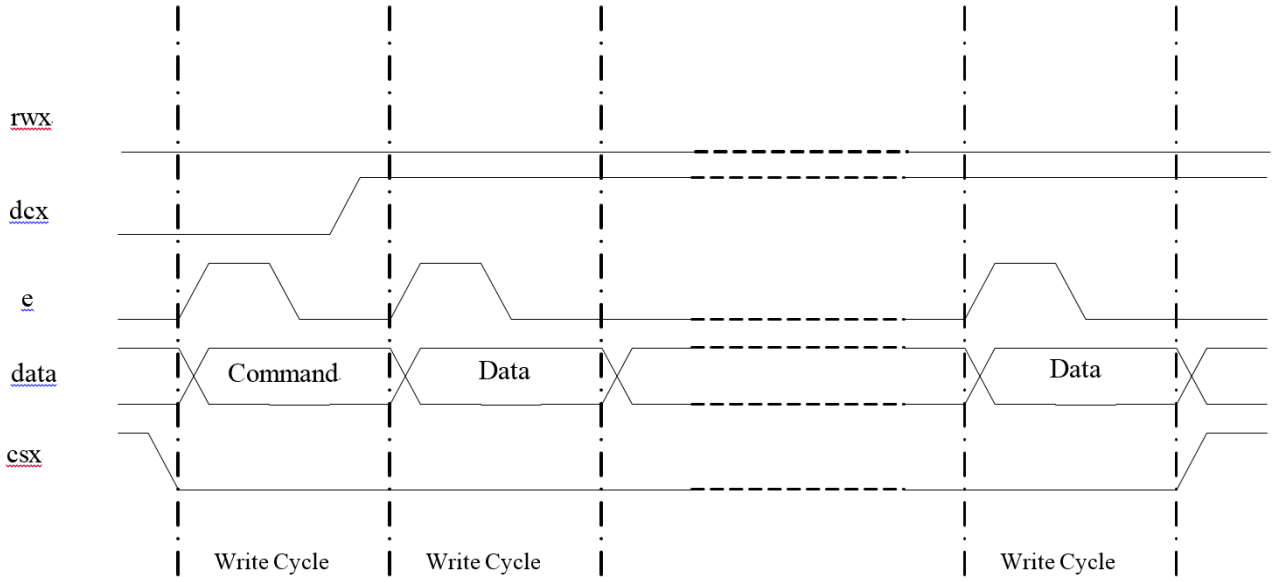



**Figure 15-9: Illustration of Write Operation for Type A, Clocked E Mode Interface**

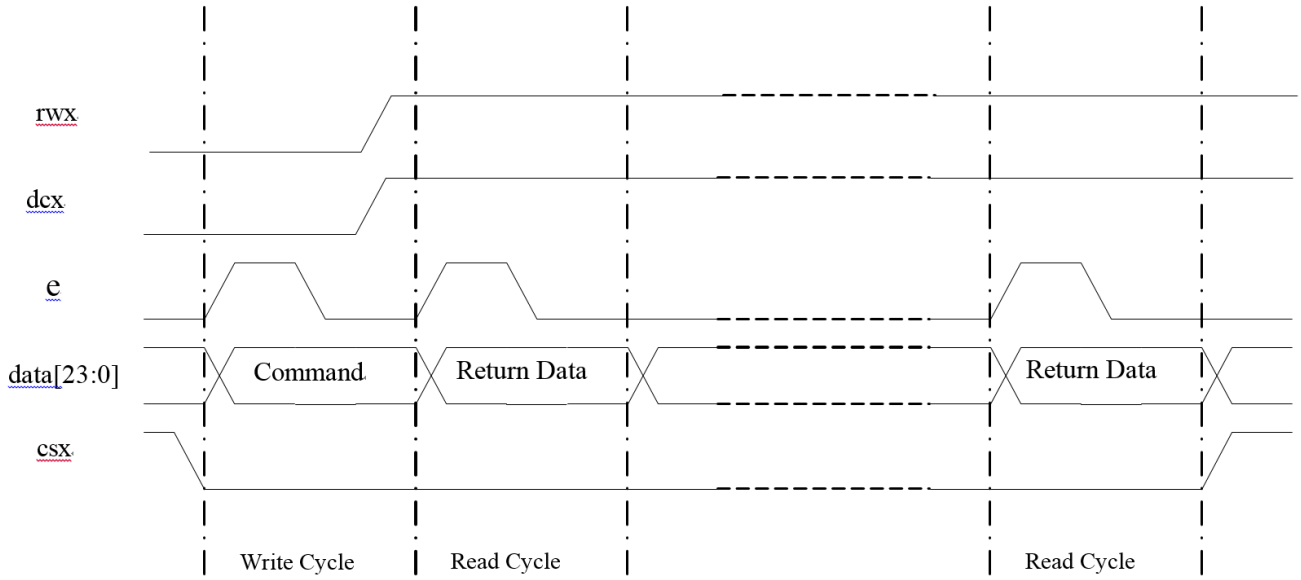



**Figure 15-10: Illustration of Read Operation for Type A, Clocked E Mode Interface**

### 15.3.3 MCU Interface Type B

This interface consists of data, rdx, wrx, dcx, and csx. It supports 24-bit,16-bit and 8-bit data bus. The first cycle should be a command write cycle to specify the register address for access. The subsequent cycles are read or write cycles for read or write operations.

csx should be driven to 0 in this mode.

When wrx is driven from 1 to 0 and 0 to 1, the operation is a write operation. When rdx is driven from 1 to 0 and 0 to 1, the operation is a read operation.

During write operation, dcx indicates whether the operation is for data or command. When dcx is 1, the operation is for data. When dcx is 0, the operation is for command. During the read operation, the dcx should be 1.

During the write operation, data are sampled at the rising edge of wrx. During read operation, data[23:0] are provided at the falling edge of rdx and the application processor should use the rising edge of rdx to sample.

Below is a diagram for illustration.

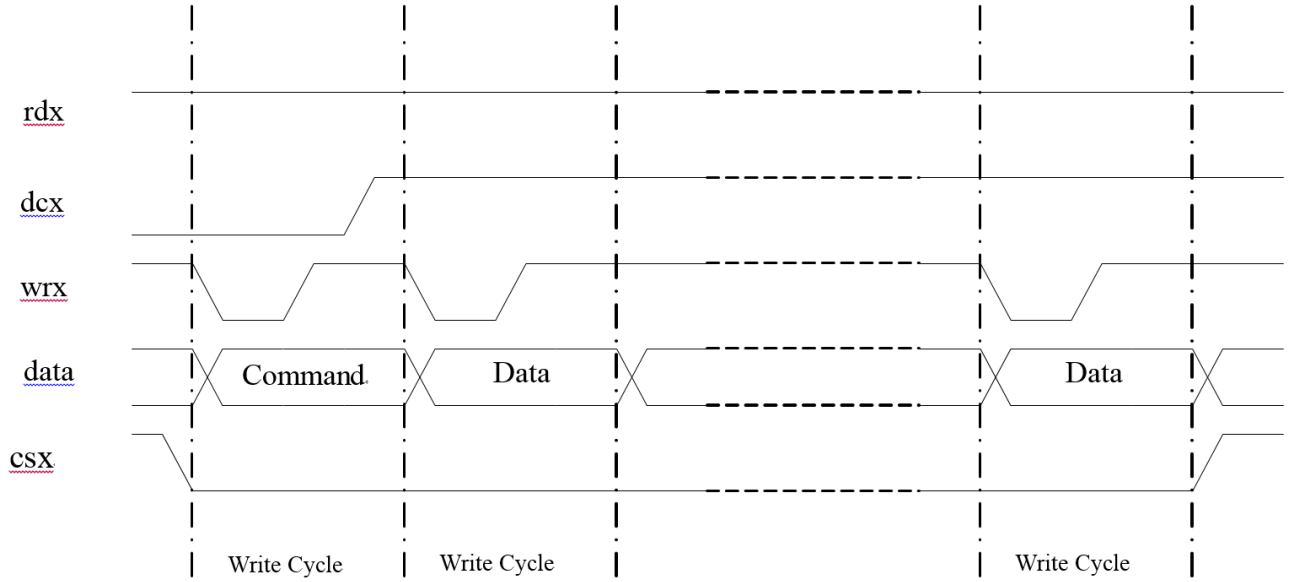



**Figure 15-11: Illustration of Write Operation for Type B Interface**

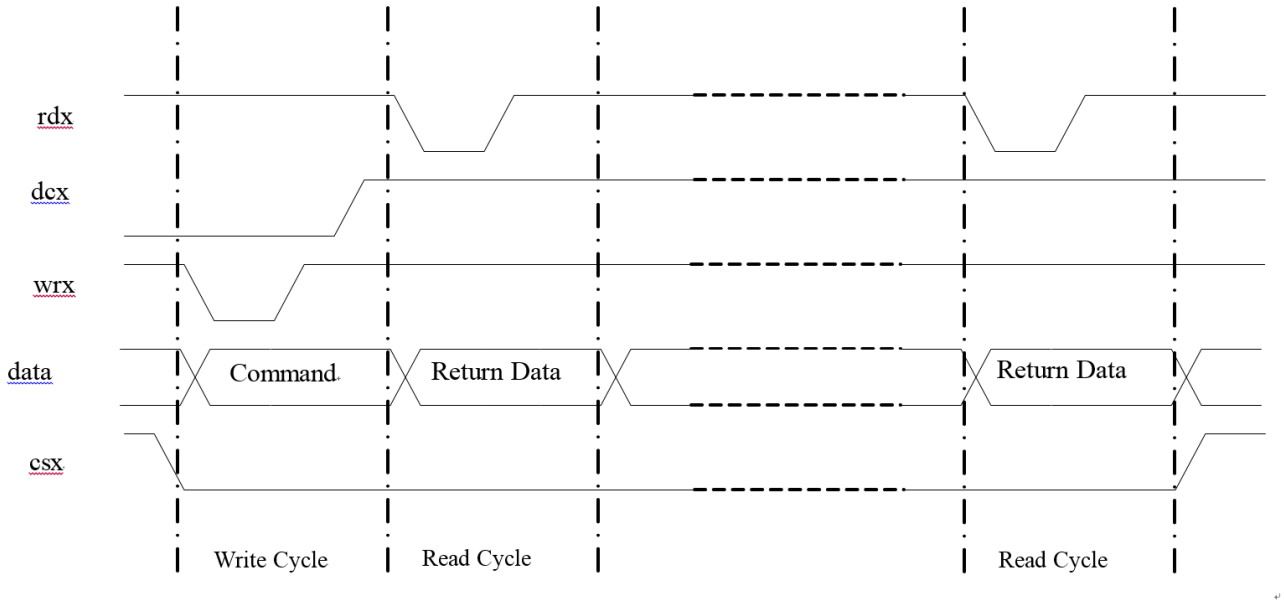



**Figure 15-12: Illustration of Read Operation for Type B Interface**

### 15.3.4 MCU Interface for MIPI Command Packet

Any write to the address ranges from 0x00 to 0xAF will be sent out as MIPI command packet. The type of packet, whether it is short or long packet, DCS or generic packet is determined by the SSD2829 local registers. Hence the user should program the local registers prior to any transmission at the MIPI link.

If the host wants to send any addresses in the range of 0xB1 to 0xFF to external MIPI receiver, it can do so using packet drop command in the 0xBF register.

In the first write cycle, only 8-Bit data are written into the SSD2829, as the command can only be 8-Bit. No matter whether the interface is 8-bit, 16-bit or 24-bit, lower 8-bits are used. Please refer to the table below.

| **Interface types** | **Data pins** | | | |
|---|---|---|---|---|
| | **D53-D30** | **D23-D16** | **D15-D8** | **D7-D0** |
| 24-bit | Don't care | Don't care | Don't care | Command |
| 16-bit | Don't care | Don't care | Don't care | Command |
| 8-bit | Don't care | Don't care | Don't care | Command |

**Table 15-9: MCU Interface Data Pin Mapping for Command Cycle**

In the sub-sequent read or write cycles, command parameters can be written into the SSD2829. Depending on the interface width, different data pin mapping is adopted. Please refer to the table below.

When the number of parameters is not an integer of the width of the interface type, the remaining bytes should be put on the lower data buses at the last data cycle.

| Interface types | Cycle | Data pins | | | |
|---|---|---|---|---|---|
| | | D53-D30 | D23-D16 | D15-D8 | D7-D0 |
| 24-bit | 1st | Don't care | Parameter 3 | Parameter 2 | Parameter 1 |
| | 2nd | Don't care | Parameter 6 | Parameter 5 | Parameter 4 |
| | 3rd | Don't care | Parameter 9 | Parameter 8 | Parameter 7 |
| 16-bit | 1st | Don't care | Don't care | Parameter 2 | Parameter 1 |
| | 2nd | Don't care | Don't care | Parameter 4 | Parameter 3 |
| | 3rd | Don't care | Don't care | Parameter 6 | Parameter 5 |
| 8-Bit | 1st | Don't care | Don't care | Don't care | Parameter 1 |
| | 2nd | Don't care | Don't care | Don't care | Parameter 2 |
| | 3rd | Don't care | Don't care | Don't care | Parameter 3 |

**Table 15-10: MCU Interface Data Pin Mapping for Parameter Cycles**

### 15.3.5 MCU Interface for Local Registers

The local registers for SSD2829 resided in the range from 0xB1 to 0xFF. There are 2 types of local registers, legacy registers and extended registers.

To expand certain field in the legacy registers, the data bits in that address is extended to 32-bit, or 4-bytes width from 2-bytes.

The extended registers can be accessed by indirect addressing through the address location 0xE0. The content of 0xE0 defines the 16-bits addresses and 32-bits data for the extended registers.

## 15.4 RGB Interface

SSD2829 supports RGB interface. The input is 48-bit wide and it supports 2-pixels per RGB module using SDR or DDR input pixel clock.

To support different bpp settings, the following data pins are used. For all cases, Red component should be at the upper bits and Blue component should be at the lower bits. The type of video packets supported for each RGB interface is shown below.

| Data Bus | RGB format |
|---|---|
| [15:0] | 16-bits per pixel for Pixel 1 |
| [45:30] | 16-bits per pixel for Pixel 2 |
| [17:0] | 18 bits per pixel for Pixel 1 |
| [47:30] | 18 bits per pixel for Pixel 2 |
| [23:0] | 24-bits per pixel for Pixel 1 |
| [53:30] | 24-bits per pixel for Pixel 2 |
| [23:0] | Compressed Stream Data(lower) |
| [53:30] | Compressed Stream Data(Upper) |

User can also send command mode data through SPI interface, during the video mode transmission. The data will be sent during the horizontal or vertical blanking period. Since the RGB and SPI interface are completely separated, the two interfaces can operate independently. The RGB interface is used to provide display data for the video mode. The SPI interface is used to program the local registers of SSD2829, or to send command across the link to the MIPI receiver.

**2 supporting data type for compressed stream data:**

- Compressed data - PPS (DT=0xA)
- Compression Mode (DT=0x7)

There is a register to be set if incoming MCU command bytes are to be packed as PPS, or Compression-Mode packet. All the parameters are carried in the PPS packet data, as defined by VESA standard.

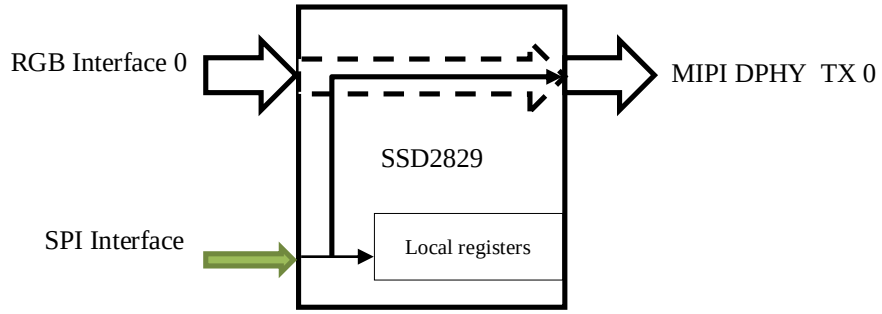
## 15.5 Video Mode Use Cases

### 15.5.1 RGB + SPI

For this mode, the user must set if_sel[1:0] to “00” to select the interface as a combination of RGB and SPI interface. The video data come from the RGB interface and the configuration is done through the SPI interface.

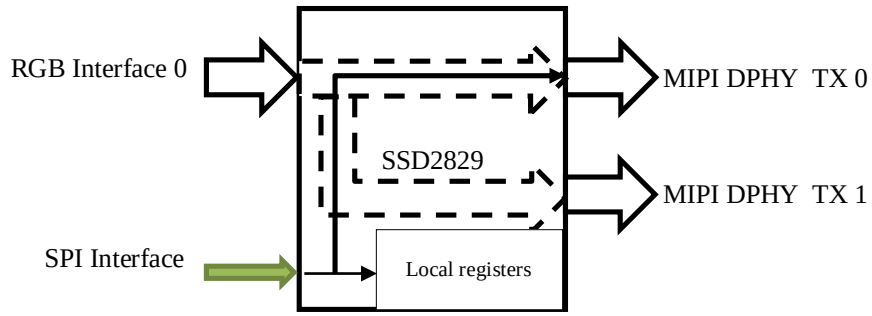
The possible video paths (and non-video command paths) supported in this mode are shown below. The SPI interface can be used to program local registers or transmit command packets during video blanking to MIPI output.

**RGB Interface Input, Single MIPI Video Output**




**RGB Interface Input, Dual MIPI Video Output**

This is either a split use case or a broadcast use case. For split use case, it can be either odd/even pixel split, or left/right image split



The user, first, needs to set the video parameters through registers programming with correct values. After programming those register fields, the user can turn on the RGB interface and enable the SSD2829 to start transmission. All three video mode sequence defined in the MIPI DSI specification are supported.

The PLL multiplication factor should be set such that the serial link data rate is faster than the incoming data rate. Please refer to the table below for the PLL settings.

Below is the diagram to illustrate the definition of all the fields.

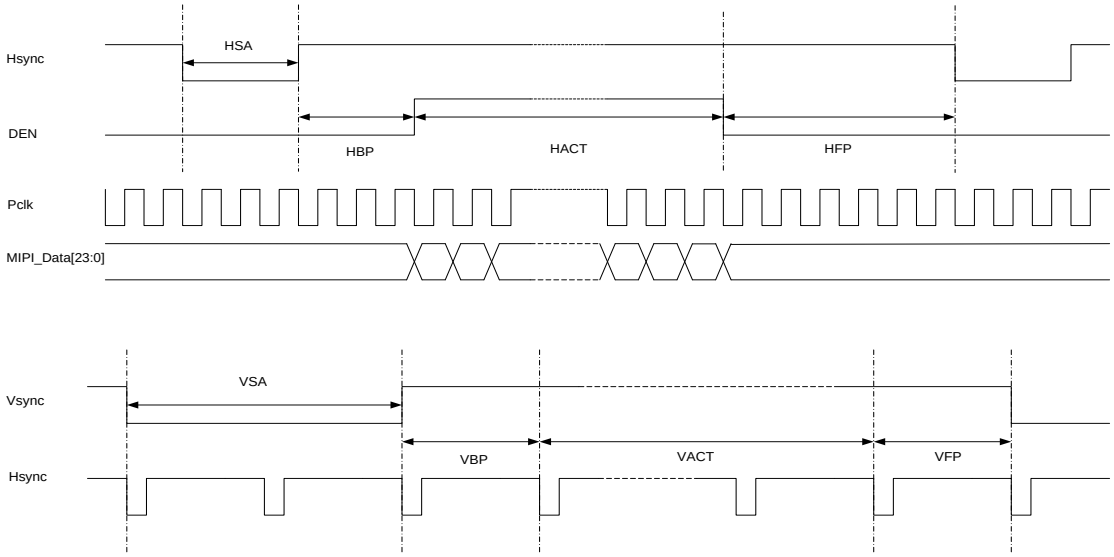



**Figure 15-13: Illustration of RGB Interface Parameters for Non-burst Mode with Sync Pulses**

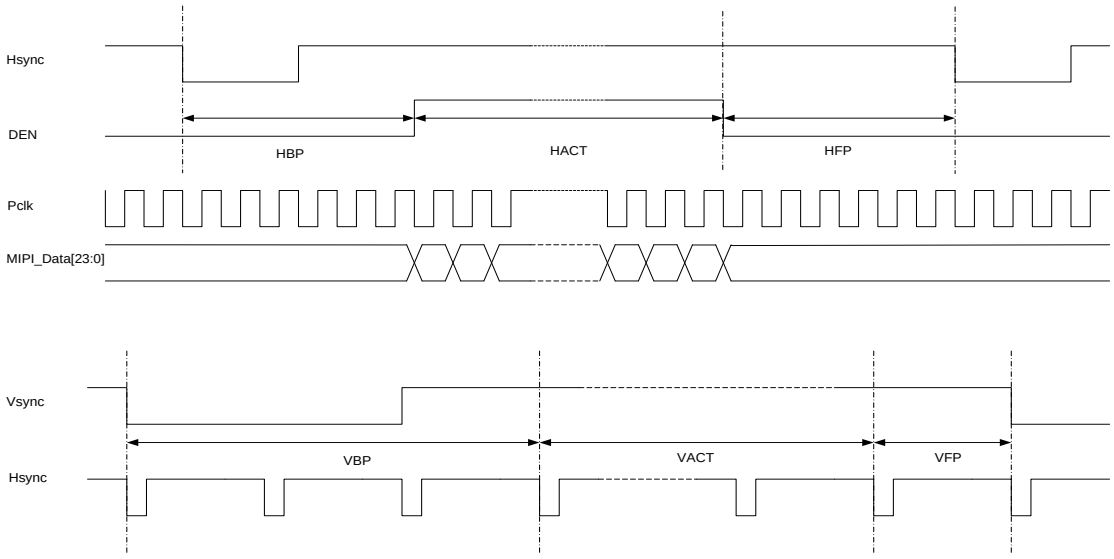



**Figure 15-14: Illustration of RGB Interface Parameters for Non-burst Mode with Sync Events and Burst Mode**

#### 15.5.1.1 Interleaving Non-Video Packets with Video Packets

Non-video data can be transmitted during the vertical blanking of the video frames, or when **nvb** (in 0xB6 register) is set, during any BLLP period (including those in the horizontal blanking). It is recommended to send non-video data during vertical blanking.

The **nvd** and **bllp** field (in 0xB6 command) determines how the non-video data is sent. See below table for illustration.

| NVD | BLLP | Non-burst mode | Burst mode |
|---|---|---|---|
| 0 | 0 | If there is no non-video data to send, the serial link will send blanking packet in HS mode during BLLP period.<br>If there is non-video data to send, the non-video data will be sent in HS mode. Afterwards, the serial link will send blanking packet in HS mode for the remaining period of BLLP period. | If there is no non-video data to send, the serial link will enter LP mode during BLLP period.<br>If there is non-video data to send, non-video data will be sent in HS mode. Afterwards, the serial link will enter LP mode for the remaining period of BLLP period. |
| 0 | 1 | If there is no non-video data to send, the serial link will enter LP mode during BLLP period.<br>If there is non-video data to send, non-video data will be sent in HS mode. Afterwards, the serial link will enter LP mode for the remaining period of BLLP period. | Same as non-burst mode. |
| 1 | x | The serial link will enter LP mode for BLLP mode. If there is non-video data to send, the data will be sent in LP mode at the beginning of BLLP period. | Same as non-burst mode. |

### 15.5.2 Interrupt Operation

An interrupt signal int has been provided to interrupt the application processor so that it does not need to poll the status all the time. This will save the processing time of the application processor. int can be programmed to active high or active low, when the event has happened.

There are many sources that can be mapped to the interrupt signal. The user can select different source to perform different task. If more than 1 source is selected, the int signal will be asserted when the event for 1 of the sources has happened. In this case, the user needs to read the register **ISR** to determine what event has happened. The different sources can be enabled/disabled through register **ICR**. Below is the list of available interrupt sources and their usage.

**RDR**

To indicate that return data from one of the MIPI slave is available for read.

**BTAR**

To indicate whether the SSD2829 has the bus authority or not. It can be used after SSD2829 makes a BTA. If the MIPI slave has returned the bus authority back to SSD2829, the interrupt will be set to indicate so. Please note that, on power up, the bus authority is already on the SSD2829. Hence, the SSD2829 will show that it has the bus authority.

**ARR**

To indicate whether the SSD2829 has received the acknowledge response from the MIPI slave. The acknowledge response can either report error or not error. This is to be determined by the **ATR** bit.

The above three interrupts are provided to the user to handle reading data from the MIPI slave or getting acknowledgement response from the MIPI slave.

**PLS**

To indicate whether the PLL has been locked or not. If the PLL is not locked, the programming speed at the external interface must be slow. After changing the PLL setting or changing the reference clock source, the user also needs to use this interrupt to determine the PLL status.

On power up, only **PLS** interrupt is enabled. This is to let the user determine the programming speed before configuring the SSD2829.

**LPTO**

To indicate that there is LP RX time out.

**HSTO**

To indicate that there is HS TX time out.

The above two interrupts are provided to the user for error handling.

**PO**

To indicate whether the SSD2829 is ready to accept any data from the user. The SSD2829 has several internal buffers to hold the data written by the user. When the user writes after than the serial link speed, those buffers will be full. If the user still writes data to SSD2829, those data will be lost. The length of the payload of the next packet that the user is going to write is determined by **TDC**, **PST**, and **DCS** fields. The SSD2829 will use these fields to decide whether the user can write the next packet or not. Hence, after programming the above mentioned fields, the user needs to check the interrupt status before writing.

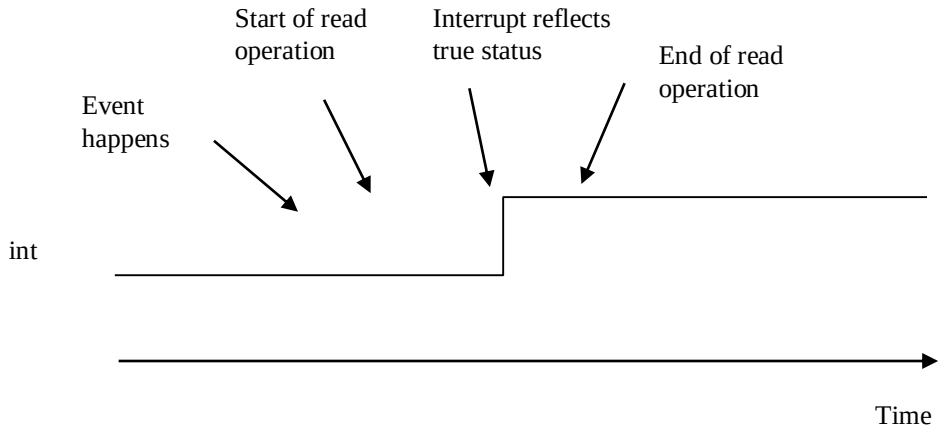
**CBE, CBA, MLE, MLA**

All these interrupts (CBE = command buffer empty, CBA = command buffer available, MLE = MCU line buffer empty, MLA = MCU line buffer available) are provided to indicate the status of the internal data

buffers. They are used if the user is familiar with the buffer management of the SSD2829. Otherwise, it is recommended to use the **PO** interrupt.

One important thing to note is the interrupt latency. The output interrupt signal does not change immediately after an operation. This is due to the internal processing of the SSD2829. For example, after changing the interrupt source from one to another, the output int level will remain at the old level for a short period after the programming is done. Another example is that after programming the **TDC** field, the interrupt will take a short period to reflect the correct **PO** status on int. There is always a delay between the actual event and the interrupt.

In order to guarantee that the user can get the correct interrupt, it is recommended that the user performs a read of any SSD2829 local register before taking in the interrupt signal or polling the interrupt status bits. The read operation will cover the interrupt latency period. Alternatively, the user can wait for certain amount of time to make sure the interrupt reflects the true status. Below is a diagram for illustration.



### 15.5.3 Internal Buffer Status

There are 2 types of buffers inside the SSD2829, which are MCU interface line buffer (ML) and MCU/SPI command interface buffer (CB).

The ML buffers are used to store the data (DCS command 0x2C and 0x3C) written through MCU interface when the if_sel is '01'. They are also used to store the video data written through RGB interface when the if_sel is '00'. However, since there is no flow control for the RGB interface video packets, the status is only valid for MCU interface.

For CB buffers, all command packets will be stored into them. They can store multiple packets, up to 4096 bytes in total. Below is a list of possible packets

- Generic Short Write Packet
- Generic Read Packet
- DCS Short Write Packet
- DCS Read Packet
- Generic Long Write Packet
- DCS Long Write Packet

In case of automatic partitioning, the packet length is determined by the **PST** field. It is not recommended to make the **PST** field so small.

When the if_sel is "00", the user can write the data through SPI interface. All packets will be written into the CB buffers. Hence, the user needs to check the corresponding interrupts. The usage of the interrupts is listed below.

**CBE**

To indicate that the Command buffer is empty.

**CBA**

To indicate that the Command buffer can hold at least 1 more packet. The user can write 1 such packet into CB buffer.

**MLE**

To indicate that MCU Long buffer is empty. Since the ML buffer can hold 2 packets, the user can write up to 2 such packets into ML buffer without needing to look at the interrupt status.

**MLA**

To indicate that the MCU Long buffer can hold at least 1 more packet. The user can write 1 such packet into ML buffer.

The interrupts mentioned here can be used as flow control between the application processor and the SSD2829. However, it requires the user to know the buffer operation well. The **PO** interrupt is a combination of the eight. It makes decision according to the parameters provided by the user for the next packet to be written. Hence, the user does not need to know which buffer is going to be used and how the buffer status is.

## 15.6 Command Mode Use Cases

MCU interface supports 8-bit, 16-bit and 24-bit data bus. To support different bus width, the following data pins of each MCU interface are used.

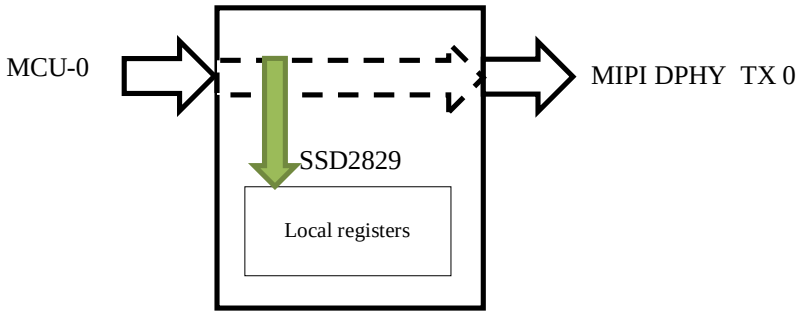
- data[7:0] for 8-bit interface.
- data[15:0] for 16-bit interface.
- data[23:0] for 24-bit interface.

The address range for the SSD2829 local register is from 0xB1 to 0xFF. The user can access the registers in this range to configure and control the SSD2829. For Generic packet that starts from 0xB1 to 0xFF, it can be written through the Packet Drop register. When the user writes data to it, the data will be sent over the serial link to the MIPI slave. The data packet sent will either be DCS or generic packet.

The following command mode use cases are possible:

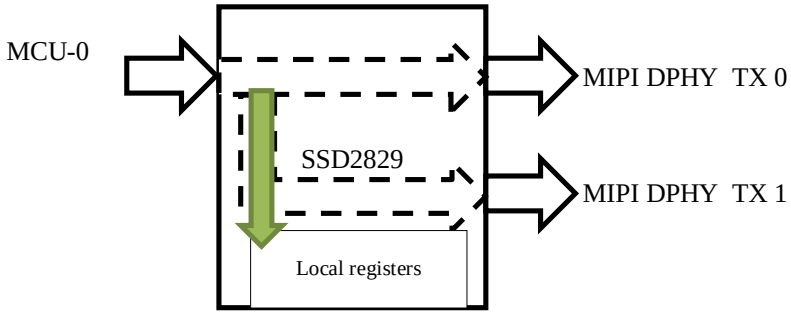
**MCU Input, Single MIPI Command/Video Output**

To select MCU(s) input, if_sel[1:0] needs to “01”.




**MCU Input, Dual MIPI Command Output**

This is a broadcast use case. To select MCU(s) input, if_sel[1:0] needs to “01”.



### 15.6.1 Write Operation

The SSD2829 can issue four kinds of packets for write operation, which are Generic Short Write Packet, Generic Long Write Packet, DCS Short Write Packet and DCS Long Write Packet. The VC ID of the outgoing packets can also be programmed through registers.

The SSD2829 needs to know the payload size of the outgoing packets. Hence, the user needs to program the corresponding control registers prior to sending the MIPI data.

To send a DCS or Generic Write Packet in address 0xB1 to 0xFF, the user needs to write the command/header and the payload to the Packet Data Drop register**.** If the size field is no more than 2 for Generic packet and 1 for DCS packet, the SSD2829 will send out DCS or Generic Short Write Packet with the correct type. Otherwise, DCS or Generic Long Write Packet will be sent out.

For DCS Write Packet, partition is supported for 0x2C or 0x3C DCS command. This is because the DCS command 0x2C and 0x3C are to write display data into the LCD panel display memory. The payload will be partitioned into a few packets where the payload of each packet is determined by the Partition register. The first byte is the DCS command and the following partition bytes are the payload. Only the last packet might contain less payload, as the total payload might not be integer multiple of partition size. If the incoming DCS command is 0x2C, the DCS command for the first packet is 0x2C and the DCS command for all other packets is 0x3C. If the incoming DCS command is 0x3C, the DCS command of all the packets is 0x3C.

For example, if the byte size field is 200 and partition field is 80, 3 packets will be sent. The first two have 80 bytes of payload. The last packet has 40 bytes of payload.

After performing a write operation, the user can optionally make a BTA to let the MIPI slave report its status. The SSD2829 will automatically make a BTA after each write operation.

### 15.6.2 Read Operation

The SSD2829 can issue two kinds of packets for read operation, which are Generic Read Packet, and DCS Read Packet. The bit **DCS** controls whether Generic Read Packet or DCS Read Packet will be sent out. The VC ID of the outgoing packets can also be programmed through registers.

Before the read packet is sent out, the SSD2829 will always send out the Set Maximum Return Size Packet. This is to limit the Read Response Packet sent by the MIPI slave such that there is no over flow. Two factors determine the maximum size. One is the limit of the SSD2829 and the other is the limit of the application processor. The user should choose the smaller one among these two limits to use as the maximum return size.

The parameter in the Set Maximum Return Size Packet is taken from local register. The user could program the Set Maximum Return Size Register before every read so that the correct value is sent through Set Maximum Return Size Packet. If the value is already the desired value, the user can choose not to program it. SSD2829 will always automatically send out Set Maximum Return Size Packet before the Read Packet.

To send a DCS Read or Generic Read Packet, the user just needs to write the DCS (as there is no parameter for DCS read) or Generic command, or write to Packet Drop Data register when the address is from 0xB1 to 0xFF.

Similar to the write operation, the Total Data Count Register field is used to determine the payload size of the outgoing packet. For DCS Read Packet, the payload is just the DCS command. There is no parameter associated. For Generic Read Packet, the SSD2829 will send out the correct packet type according to the Total Data Count value.

After sending out the read packet, the SSD2829 will automatically perform a BTA to wait for the Read Response Packet from the MIPI slave. The return data will be stored in a data register. No matter what read packet is sent out, there is only one packet returning data. Therefore, no matter whether the read is DCS read or Generic read, no matter what command is used in DCS read, the return data is always stored in the same data register. The user can read the data out when the read valid status bit is set to 1. After seeing read valid status bit been set to 1, the user should first check the number of bytes returned by the MIPI slave. By using this information, the user will know how many data should be read out from data register. After all the return data are read out, the read valid status bit will be set to 0 by the SSD2829.

Even the read valid status bit is set to 1, the user can choose not to read the data out from data register. The user can continue performing another operation. Once the user does so, the read valid status bit will be set to 0 by the SSD2829.

There might be Acknowledge and Error Report Packet sent by the MIPI slave at the same time.

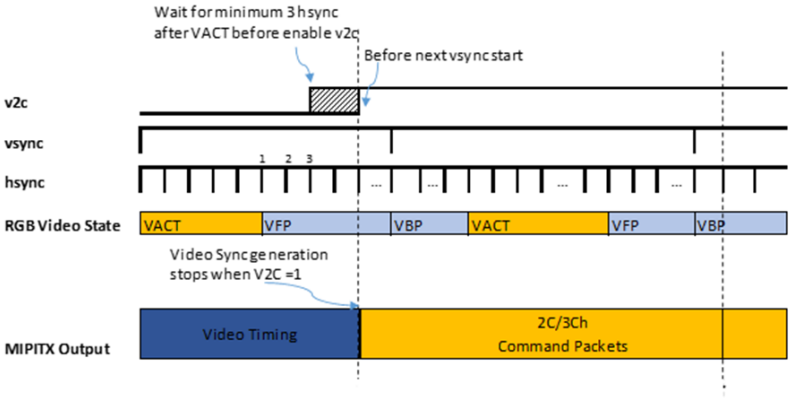
Under certain circumstance, the MIPI slave might only send back Acknowledge and Error Report Packet without any data. Thus, the read valid status bit will not be set. Therefore, it is recommended that the user check the bus turnaround bit first. The bus turnaround bit is to indicate whether the MIPI slave has passed the bus authority back to the SSD2829 or not. Only when the bus turnaround is 1, there might be return data. If there is no return data, the user should follow Acknowledgement Operation to handle the acknowledgement.

## 15.7 Video to Command Mode conversion

The MIPI TX output of SSD2830 can convert video packets to command mode packets (i.e. 0x2C command for the first video line, and 0x3C commands for the subsequent video lines).

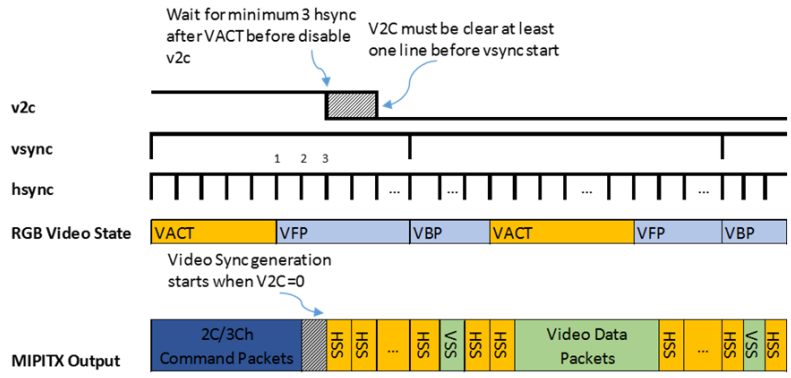
- Switching of Video mode to V2C mode
  1. Set SCM.0x0010.bit16 – v2c during VFP of incoming video timing

     To ensure all video data has been transmitted on the TX link, host shall wait for min of 3 hsync after VACT before setting v2c



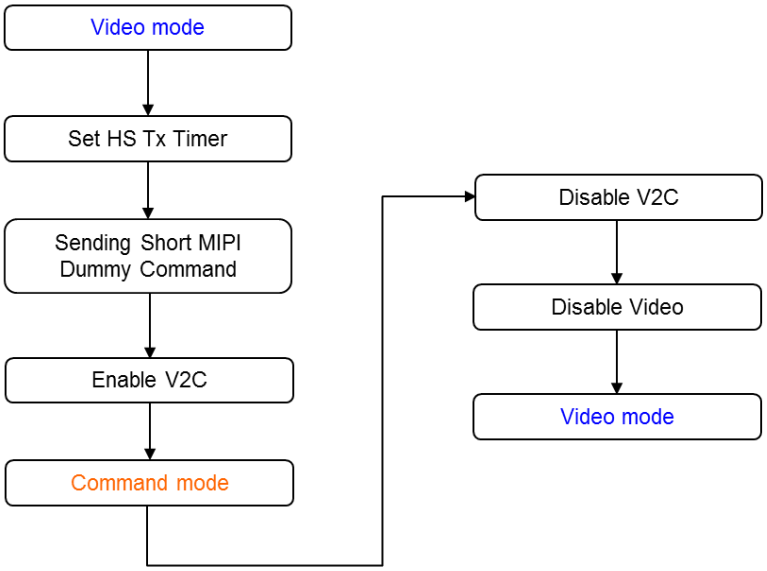

- Switching of V2C mode to Video mode
  1. Ensure 0xB7.bit3 – VEN is cleared
  2. Clear SCM.0x0010.bit16 – v2c during VFP of incoming video timing

     To ensure all video data has been transmitted on the TX link, host shall wait for min of 3 hsync after VACT before clearing v2c



### 15.7.1 Example of switching sequence

| Register adr | Data [31:24] | Data [23:16] | Data [15:8] | Data [7:0] | Description |
|---|---|---|---|---|---|
| 0xCF | 0x00 | 0x00 | 0x00 | 0x06 | // set HS TX timer |
| 0xD0 | 0x00 | 0x00 | 0x9A | 0x4F | // set HS TX timer |
| 0x0010 | 0x4C | 0x91 | 0x0E | 0x00 | // enable v2c |
| 0x0010 | 0x4C | 0x90 | 0x0E | 0x00 | // disable v2c |




**Note:**

For this feature, it is important to note that once Video-to-Command mode is turned on, the MIPI would be in HS link when there is active video input. It would remain in HS link until the mode is turned off. Since there is a HS timeout built-in SSD2830, user is recommended to switch off Video-to-command mode periodically (e.g. after a 2-3 frames of conversion)

## 15.8 State machine operation

The state machine controls the sending and receiving of the data packet over the serial link. It is triggered by an event from the application processor or the received data. Once a complete packet is written into the SSD2829 buffer, it will send it out through the serial link. The user can write 1 to bit COP (cancel-operation) at any time to cancel all the current operations.

When the SSD2829 is in high speed mode, the serial link is mainly used to send display data. If there is no data to send, it will send null packet to maintain the serial link timing. If the application processor does not have display data to send in a long period, it can turn the serial link into low power mode by setting the register bit HS to 0.

When the SSD2829 is in low power mode, the serial link is mainly used to send command and configuration data. If there are no data to be sent, the SSD2829 will be idle in LP TX stop mode.

The user can also enter sleep mode by writing 1 to SLP bit. Once the SLP bit is set to 1, the SSD2829 will automatically enter LP mode. If the HS bit is 1, the SSD2829 will clear the HS bit to 0 and switch from HS to LP mode. Afterwards, the SSD2829 will issue ULPS trigger message to the MIPI slave to enter Ultra Low Power State. During this state, the clock to SSD2829 can be switched off such that the SSD2829 only consumes leakage current. This will save the overall system power consumption. When exiting from the ULPS, the user can write 0 to SLP bit. However, the user should be aware that the time to exit from ULPS is relatively long. Hence, the user cannot perform any data transmission before the system exits from ULPS.64

During reception, the state machine will disassemble the incoming data packet and put the received register content into the internal buffer for reading out. Once all the data are put into the buffers, it will set the register bit RDY to 1 to indicate that the SSD2829 is ready for read. The total number of received bytes will also be stored in RDCR.
After the reception is completed, the SSD2829 will perform a bus turn around to enter the transmission mode. It will always come back to the LP TX stop mode before it enters any other

## 15.9 PHY controller Operation

PHY-controller controls the operation of the analog transceiver. It controls whether the serial link is in high speed or low power mode and whether it's in transmit or receive mode.
In transmit mode, the PHY controller will perform the handshaking procedure when switching between LP mode and HS mode according to the control from PCU. During HS mode, PHY controller will provide parallel data and clock to the analog transmitter for transmitting in differential signals serially. During LP mode, the PHY controller will provide the serial data to the analog transmitter.
In receive mode, the PHY controller will detect the handshaking sequence in LP mode and inform the PCU. Once entering escape mode, it will collect the serial data from analog receiver and put them in parallel form for the PCU to process.
Various timing parameter has been defined in MIPI DPHY specification. The timing parameters are a mixture of absolute time and cycle counts. Hence, for different operation speed, there is different timing requirement. The user can adjust the value in these registers to have different DPHY timing parameters. This gives maximum flexibility for different operation speed.

## 15.10 PLL Configuration

The PLL output frequency is calculated by the equations below,

$$f_{PRE} = \frac{f_{IN}}{MS}$$

$$f_{OUT} = f_{PRE} * NS$$

where the $f_{IN}$ is the input reference clock frequency and $f_{OUT}$ is the output clock frequency of the PLL.

The clock frequencies need to satisfy the constraint below.

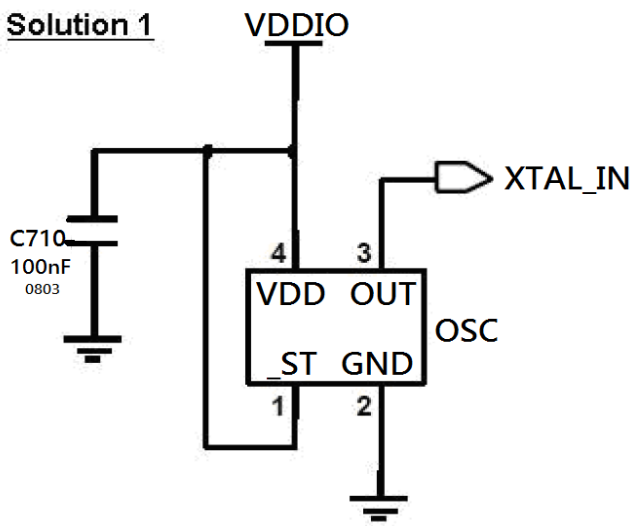
$5MHz < f_{IN} \leq 40MHz$

$5MHz < f_{REF} \leq 100MHz$

$62.5MHz < f_{OUT} \leq 1250MHz$

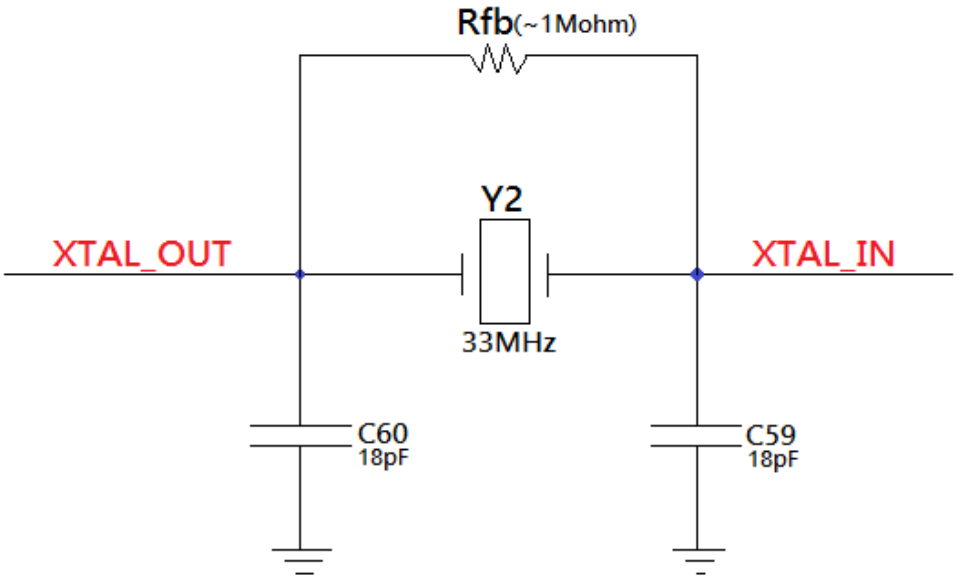
The value of FR, MS, and NS are controlled in the register **PLCR**.
All the values of FR, MS and NS can only be modified when the PLL is turned off (**PEN**=0). Hence, the sequence for modification is to turn off PLL, modify register value, and turn on PLL.

## 15.11 Clock Source Example




| Pin | Connection |
|---|---|
| XTAL_OUT | Open |

**Solution 2**



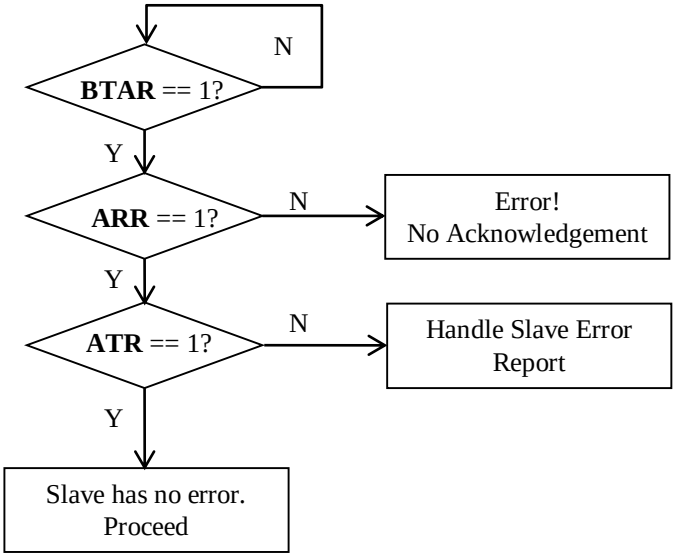
## 15.12 Acknowledgement Operation

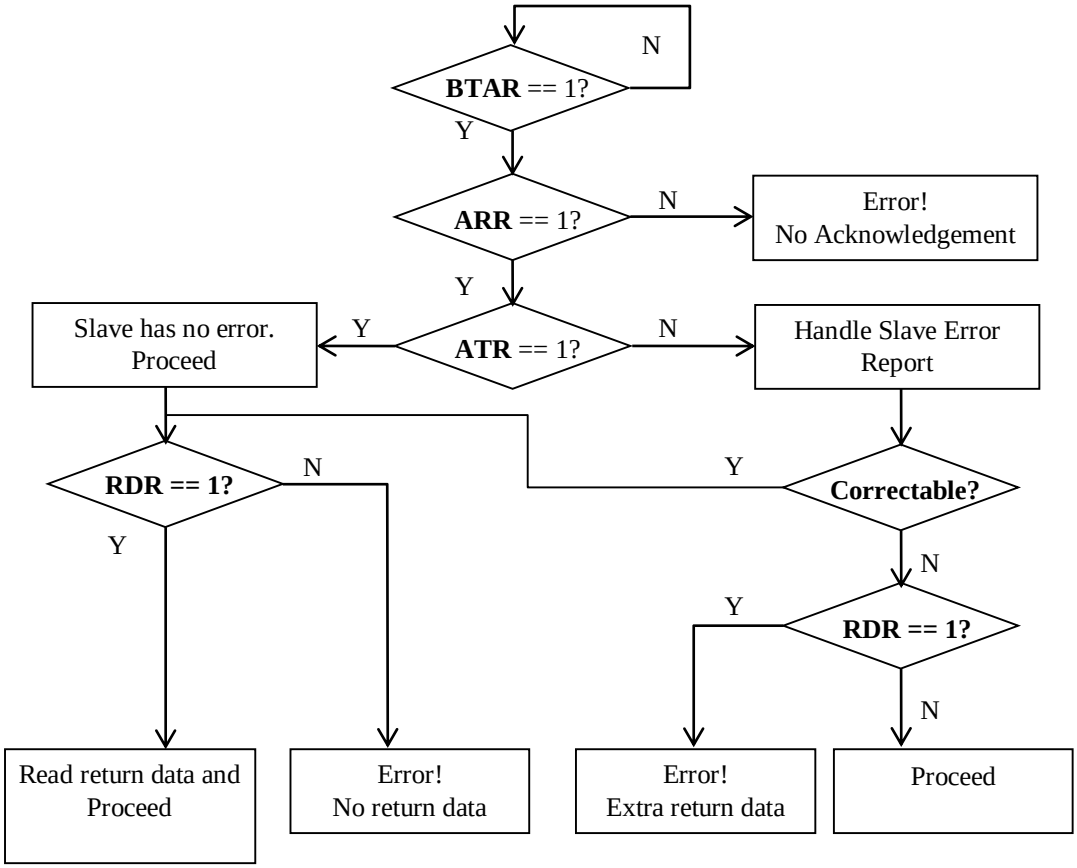
The SSD2829 can perform a BTA to give the bus authority to the MIPI slave and let it report its status. The BTA can be enabled by setting **FBW** bit to 1 and performing a write operation, or just performing a read operation. After the MIPI slave passes the bus authority back, the SDD2829 will set bit **BTAR** to 1.

If there is no error on the slave side, the MIPI slave will return ACK trigger message, if the packet before BTA is a write packet. The MIPI slave will return Read Response Packet, if the packet before BTA is a read packet. In this case, after receiving the response from the MIPI slave, SSD2829 will set bit **ARR** and **ATR** bits to 1. **ARR** indicates that response has been received from MIPI slave. **ATR** indicates that the MIPI slave has reported no error with ACK trigger message. Consequently, the register **ARSR** will be cleared to 0.

If there is error on the slave side, the MIPI slave will return Acknowledge and Error Report packet, if the packet before BTA is a write packet. The MIPI slave will return Read Response Packet (depending on the error type) and Acknowledge and Error Report Packet, if the packet before BTA is a read packet. In this case, after receiving the response from the MIPI slave, SSD2829 will set bit **ARR** bit to 1 and **ATR** bits to 0. **ARR** indicates that response has been received from MIPI slave. **ATR** indicates that the MIPI slave has sent Acknowledge and Error Report Packet instead of ACK trigger message. Therefore, the MIPI slave has reported error. The error reported by the MIPI slave will be stored in register **ARSR**. The user can read this register to see what error the MIPI slave has encountered.

For the detailed description of each error bit, please refer to MIPI DSI specification. Below are the flow charts of handling the MIPI slave acknowledgement. They are just for reference.

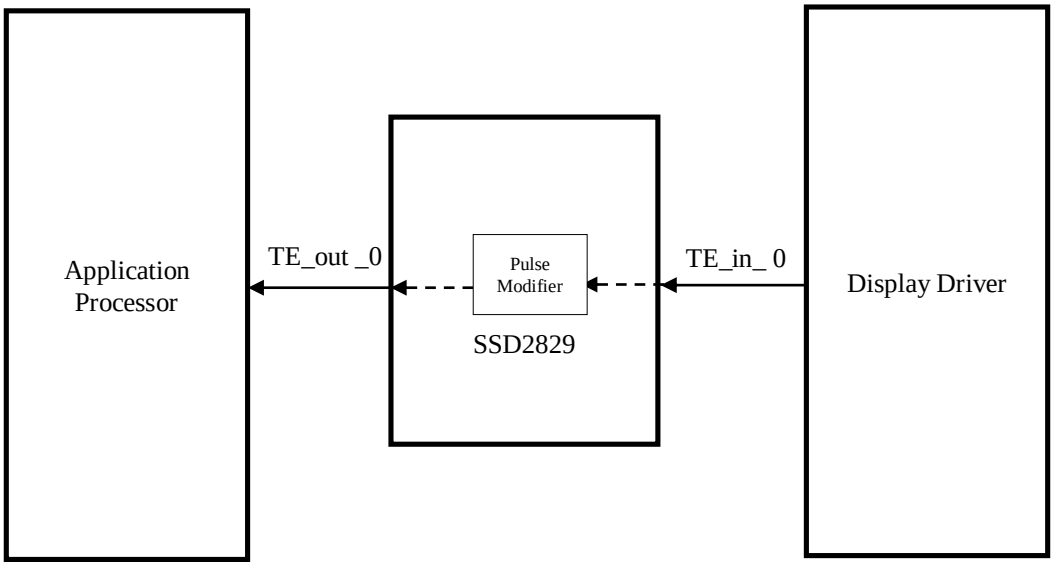



BTAR == 1?
N
Y
ARR == 1?
N
Error!
No Acknowledgement
Y
ATR == 1?
Y
Slave has no error.
Proceed
N
Handle Slave Error
Report
RDR == 1?
N
Y
Correctable?
Y
N
RDR == 1?
Y
N
Read return data and
Proceed
Error!
No return data
Error!
Extra return data
Proceed

## 15.13 Tearing Effect (TE) Operation

### 15.13.1 Using IO Pins

SSD2829 takes 1 TE_in pin, reshape them and output them to 1 TE_out pin. The programmable parameters are the pulse width, polarity, and delay.



### 15.13.2 Using MIPI Escape Mode

The TE operation is to perform a BTA following the previous BTA without transmitting anything in between. The bus is handed to the MIPI slave for providing TE information. After getting the TE event from display driver, the MIPI slave will pass the bus authority back to the SSD2829 by using BTA trigger message.

The TE operation can be enabled by setting bit **FBT** and **FBW** to 1 before writing the last command to the MIPI slave. Afterwards, the application processor can instruct the SSD2829 to send out the last command in a write packet. Since **FBW** is 1, the SSD2829 will automatically perform a BTA after the write operation. The MIPI slave will response and pass the bus authority back. Since **FBT** is 1, the SSD2829 will perform another BTA without sending any data. This makes the MIPI slave enter TE mode.

The MIPI slave will send a TE trigger message back when it gets the TE event. After getting the trigger message, the SSD2829 will set the TE pin to 1 to indicate that TE event has been received. At the same time, bit **TER** will be set to 1. The application processor can write 1 to this bit to clear it. As the TE trigger message only determines when the TE pin will be set to 1, a counter is used to determine when to set the TE pin to 0. The TE pin will be set to 0, once the counter reaches the value in **TEC**. The counter uses the reference clock to do counting.

If the MIPI slave does not send back the TE trigger message but just perform a BTA to pass the bus back, the SSD2829 will automatically perform another BTA to pass the bus to the MIPI slave again. It will continue do so until the MIPI slave respond with the TE trigger message, or the **FBT** bit is set to 0, or the LP RX timer expires.

If the MIPI slave does not send back the TE trigger message and still holds the bus, the user can set the bit **FBC** to 1 to force a bus contention. After bus contention is resolved, the slave will pass the bus back to SSD2829.

SSD2829 supports dual MIPI TX port. Hence there would be 2 TE outputs accordingly.

## 15.14 Contention Detection and Timer Operation

Two timers have been defined in SSD2829 to resolve the potential contention issue on the bus. The two timers are the HS TX timer and LP RX timer. Please see the register description for the detailed usage.

Whenever the SSD2829 sees a contention being detected, it will reset the state machine and enter the default mode, which is LP TX idle mode. The data line will be kept at LP11.

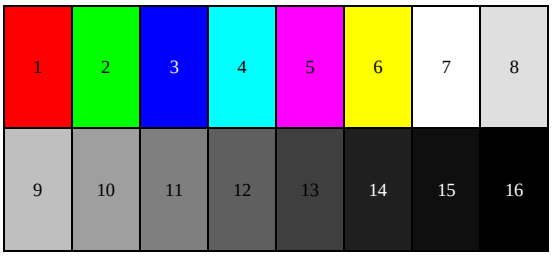
## 15.15 Video BIST

SSD2829 supports the following pattern generation for video BIST.

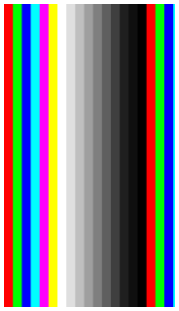
Parameter Required: 17 bytes

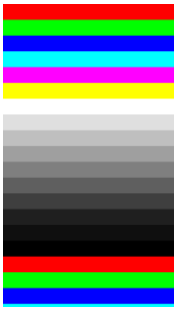
| Offset | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Default | R/W | Description |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | vb_mode | | | | | | vb_cspf | vb_en | 0x00 | R/W | vb_mode : VBIST test mode selection<br>vb_cspf: Enable color 1 & 2 swapping on every frame. (Note: This feature is intended to be used together with vb_mode=0xC. The usage in other vb_mode is not verified.)<br>vb_en : VBIST enable |
| 1 | vb_repeat_cnt_h | | | | | | | | 0x00 | R/W | Repeat count |
| 2 | vb_repeat_cnt_l | | | | | | | | 0x3C | R/W | |
| 3 | vb_r1 | | | | | | | | 0x00 | R/W | RGB value for color 1 |
| 4 | vb_g1 | | | | | | | | 0x00 | R/W | |
| 5 | vb_b1 | | | | | | | | 0x00 | R/W | |
| 6 | vb_r2 | | | | | | | | 0x00 | R/W | RGB value for color 2 |
| 7 | vb_g2 | | | | | | | | 0x00 | R/W | |
| 8 | vb_g2 | | | | | | | | 0x00 | R/W | |
| 9 | vb_x_start_h | | | | | | | | 0x00 | R/W | |
| 10 | vb_x_start_l | | | | | | | | 0x00 | R/W | |
| 11 | vb_x_end_h | | | | | | | | 0x00 | R/W | |
| 12 | vb_x_end_l | | | | | | | | 0x00 | R/W | |
| 13 | vb_y_start_h | | | | | | | | 0x00 | R/W | |
| 14 | vb_y_start_l | | | | | | | | 0x00 | R/W | |
| 15 | vb_y_end_h | | | | | | | | 0x00 | R/W | |
| 16 | vb_y_end_l | | | | | | | | 0x00 | R/W | |

| MODE 0x0 | | Parameter | Usage |
|---|---|---|---|
| Solid color loop in sequential order of 16 pre-defined color of (0XFF0000, 0X00FF00, 0X0000FF, 0X00FFFF, 0XFF00FF, 0XFFFF00, 0XFFFFFF, 0XDFDFDF, 0XBFBFBF, 0X9F9F9F, 0X7F7F7F, 0X5F5F5F, 0X3F3F3F, 0X1F1F1F, 0X0F0F0F, 0X000000) | 1 | vb_repeat_cnt_h | Number of frames pause between different colors |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | *Not used* |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | *Not used* |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | *Not used* |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

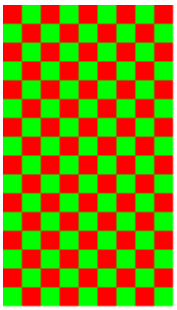


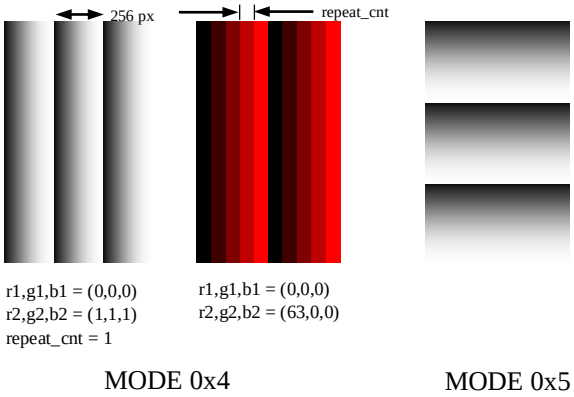

| MODE 0x1, 0x2 | | Parameter | Usage |
|---|---|---|---|
| Vertical (mode 0x1) / Horizontal (mode 0x2) repeating 16 colors bars with configurable bar width. Color repeating order are (0XFF0000, 0X00FF00, 0X0000FF, 0X00FFFF, 0XFF00FF, 0XFFFF00, 0XFFFFFF, 0XDFDFDF, 0XBFBFBF, 0X9F9F9F, 0X7F7F7F, 0X5F5F5F, 0X3F3F3F, 0X1F1F1F, 0X0F0F0F, 0X000000) | 1 | vb_repeat_cnt_h | Color bar width in pixels (MODE 0x1 only : Must be even number) |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | *Not used* |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | *Not used* |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | *Not used* |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

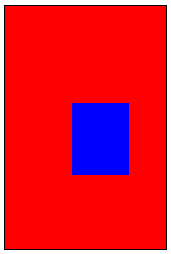


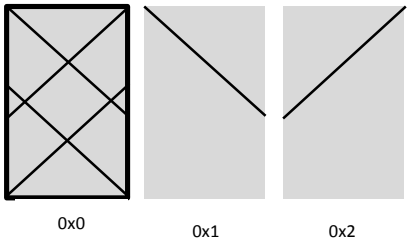


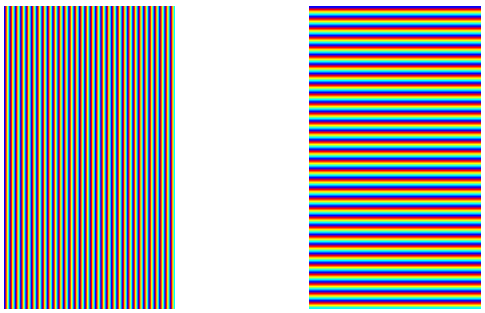
| MODE 0x3 | | Parameter | Usage |
|---|---|---|---|
| Checker box with configurable width (>= 2) and color. | 1 | vb_repeat_cnt_h | Box width (>=2, must be even number) |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Color 1 RGB value |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Color 2 RGB value |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | *Not used* |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

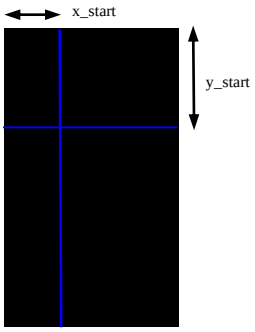


| MODE 0x4, 0x5 | | Parameter | Usage |
|---|---|---|---|
| Horizontal (0x4) or vertical (0x5) gradient ramp with programmable line width and color increment value.<br>  <br> | 1 | vb_repeat_cnt_h | For Mode 0x4 - # of pixels for each color step. NOTE: It must be an even number.<br>0 – 1 pixel.<br>2 – 2 pixels.<br>4 – 4 pixels.<br>…<br><br>For Mode 0x5 - # of lines for each color step.<br>0 – 1 line.<br>1 – 1 line.<br>2 – 2 lines.<br>… |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Start color (common for mode 0x4, 0x5) |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Color increment value for each step (common for mode 0x4, 0x5) |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | *Not used* |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

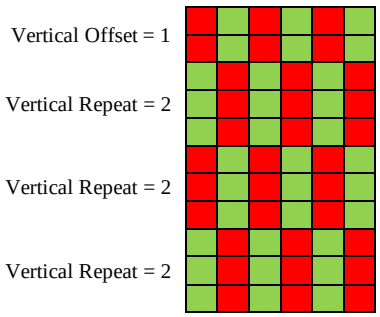
| MODE 0x6 | | Parameter | Usage |
|---|---|---|---|
| Solid filled rectangle with configurable position, size and foreground / background color<br>  | 1 | vb_repeat_cnt_h | *Not used* |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Foreground color |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Background color |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | Rectangle's left boundary (Must be even number) |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | Rectangle's right boundary (Must be even number) |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | Rectangle's top boundary |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | Rectangle's bottom boundary |
| | 16 | vb_y_end_l | |

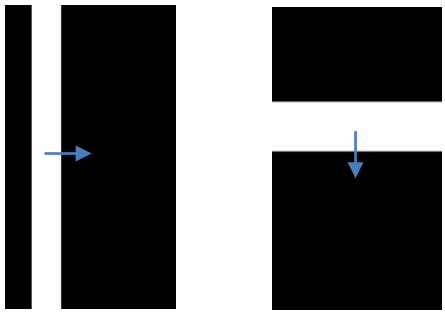
| MODE 0x7 | | Parameter | Usage |
|---|---|---|---|
| Single pixel width full size rectangle with two 45° cross touching screen corners. Or single 45° diagonal line drawn from top left / top right corner. Foreground and background color are configurable.   | 1 | vb_repeat_cnt_h | 0x0 : Original Box + Cross<br>0x1 : Line from (0,0) - (W,W)<br>0x2 : Line from (0,W) - (W,0) |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Foreground color RGB value |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Background color RGB value |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | *Not used* |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

| MODE 0x8, 0x9 | | Parameter | Usage |
|---|---|---|---|
| Vertical (0x8) or Horizontal (0x9) repeating color bars with width of 1 pixel per color. Color repeating order is [C1, !C2, !C1, C2]  | 1 | vb_repeat_cnt_h | *Not used* |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Color 1 RGB value |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Color 2 RGB value |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | *Not used* |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

| MODE 0xA | | Parameter | Usage |
|---|---|---|---|
| Single pixel width vertical and horizontal line with configurable foreground and background color   | 1 | vb_repeat_cnt_h | *No used* |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Foreground color RGB value |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Background color RGB value |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | x-coordinate of the vertical line |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | y-coordinate of the horizontal line |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

| MODE 0xB | | Parameter | Usage |
|---|---|---|---|
| This mode is not used. | 1 to 16 | | *Not used* |

| MODE 0xC | Parameter | | Usage |
|---|---|---|---|
| Check box with configurable color, vertical offset and vertical repeat count.<br>(Original single pixel checkbox can be obtained by setting offset=0 & repeat=0)<br>  <br> | 1 | vb_repeat_cnt_h | Vertical repeat in rows.<br>0: 1 line.<br>1: 2 lines.<br>… |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Color 1 RGB value |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Color 2 RGB value |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | Vertical Offset in rows. (Must be <= Vertical repeat)<br>0: "vertical repeat" - 0 lines.<br>1: "vertical repeat" - 1 lines.<br>…<br>Vertical repeat: 0 line. (Same as 0) |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

| MODE 0xD, 0xE | Parameter | | Usage |
|---|---|---|---|
| Vertical (0xD) or Horizontal (0xE) moving bar with configurable speed, width, step (with direction) and foreground / background color.<br>  | 1 | vb_repeat_cnt_h | Number of frames pause between steps |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Foreground color RGB value |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | Background color RGB value |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | Move step (For mode 0xD, must be even number, signed) |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | Bar width (For mode 0xD, must be even number) |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | Move step (For mode 0xE, singed) |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | Bar width (For mode 0xE) |
| | 16 | vb_y_end_l | |
| **MODE 0xF** | **Parameter** | | **Usage** |
| Full screen solid fill with configurable color. | 1 | vb_repeat_cnt_h | *Not used* |
| | 2 | vb_repeat_cnt_l | |
| | 3 | vb_r1 | Solid fill color RGB value |
| | 4 | vb_g1 | |
| | 5 | vb_b1 | |
| | 6 | vb_r2 | *Not used* |
| | 7 | vb_g2 | |
| | 8 | vb_g2 | |
| | 9 | vb_x_start_h | *Not used* |
| | 10 | vb_x_start_l | |
| | 11 | vb_x_end_h | *Not used* |
| | 12 | vb_x_end_l | |
| | 13 | vb_y_start_h | *Not used* |
| | 14 | vb_y_start_l | |
| | 15 | vb_y_end_h | *Not used* |
| | 16 | vb_y_end_l | |

## 15.16 Pixel Peek

SSD2829 supports pixel peek, which is allows user to peek at the pixel value on a programmable pixel location in the video frame. The pixel location can be configured to be marked out on the screen (through SSD2829 MIPI TX output) through a cursor (example shown below).

Note:

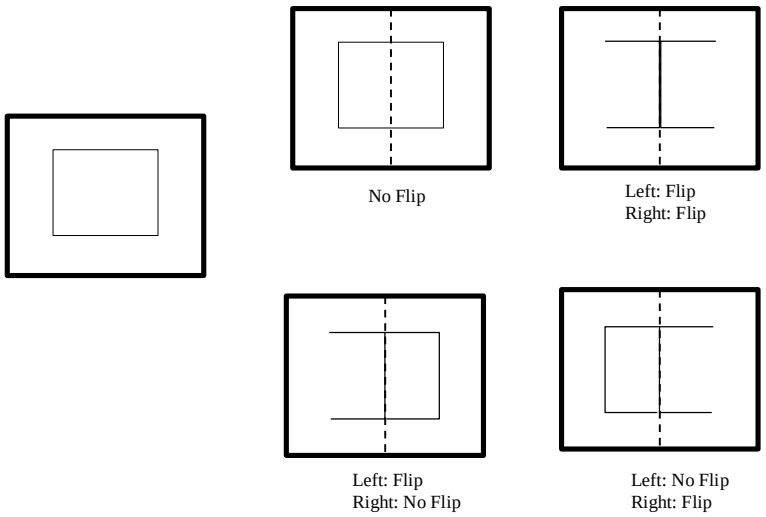
Pixel peek can only be supported for the following modes:

For single RGB0 input and 2 DSI_TX output(1 to 2)
RICR6.RGB_PACK_SEQ = 1: Left/Right split. pixel[0] to pixel[n/2-1] on DSI_TX0, pixel[n/2] to pixel[n-1] on DSI_TX1.

- The cursor shown crosses at the (x,y) location programmed by the user.
- The cursor is programmed to be visible, with 'blue'color.
- The actual pixel value of the location is stored in the register for user to read-back

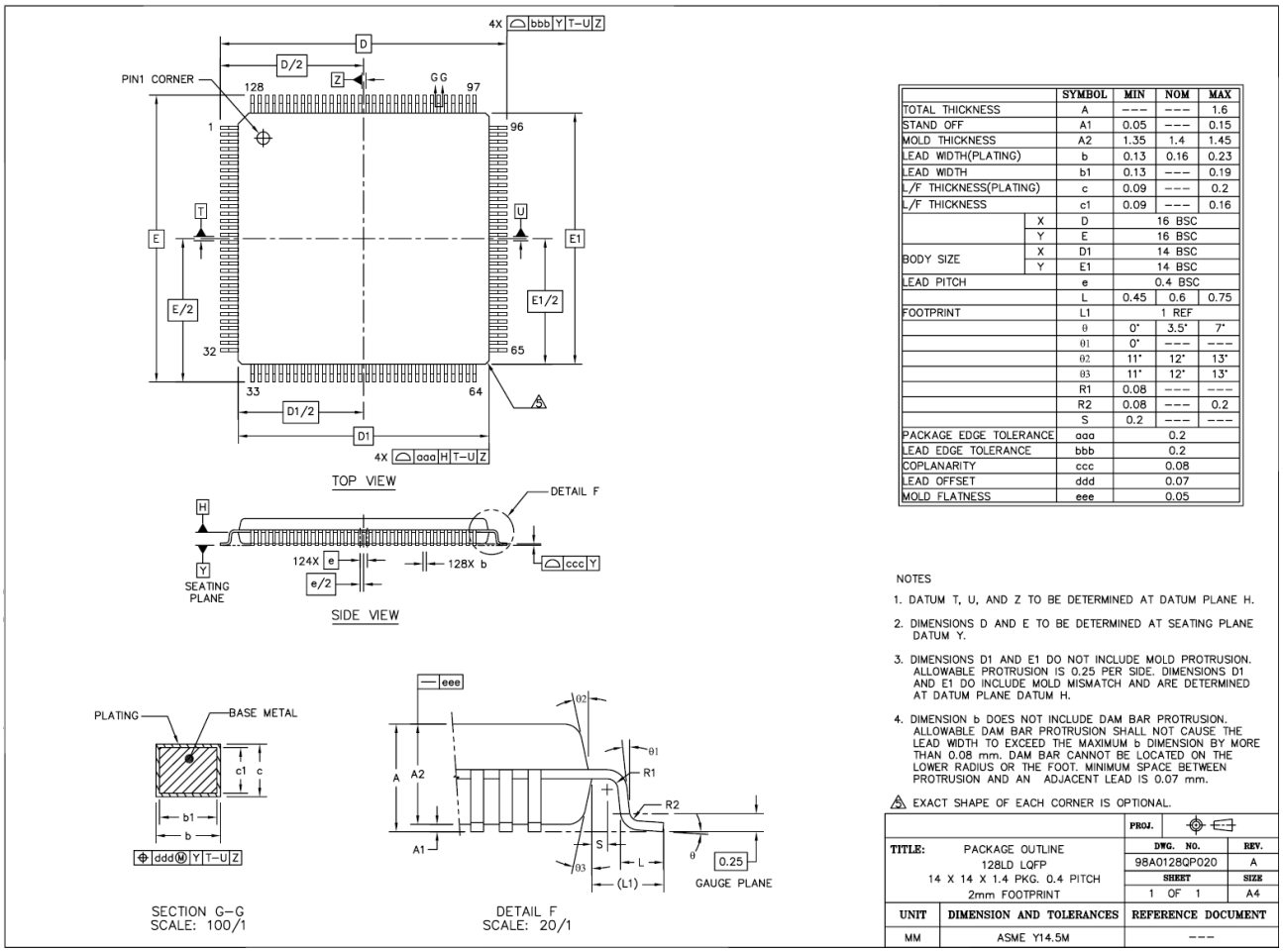
## 15.17 Image Flipping (Horizontal)

Each of the dual MIPI TX can be configured to perform horizontal flip independently of each other. For example:



# 16 PACKAGE INFORMATION

## 16.1 QFP 128 pins (14mm x 14mm)




| | | SYMBOL | MIN | NOM | MAX |
|---|---|---|---|---|---|
| TOTAL THICKNESS | | A | --- | --- | 1.6 |
| STAND OFF | | A1 | 0.05 | --- | 0.15 |
| MOLD THICKNESS | | A2 | 1.35 | 1.4 | 1.45 |
| LEAD WIDTH(PLATING) | | b | 0.13 | 0.16 | 0.23 |
| LEAD WIDTH | | b1 | 0.13 | --- | 0.19 |
| L/F THICKNESS(PLATING) | | c | 0.09 | --- | 0.2 |
| L/F THICKNESS | | c1 | 0.09 | --- | 0.16 |
| BODY SIZE | X | D | 16 BSC | | |
| | Y | E | 16 BSC | | |
| | X | D1 | 14 BSC | | |
| | Y | E1 | 14 BSC | | |
| LEAD PITCH | | e | 0.4 BSC | | |
| | | L | 0.45 | 0.6 | 0.75 |
| FOOTPRINT | | L1 | 1 REF | | |
| | | θ | 0° | 3.5° | 7° |
| | | θ1 | 0° | --- | --- |
| | | θ2 | 11° | 12° | 13° |
| | | θ3 | 11° | 12° | 13° |
| | | R1 | 0.08 | --- | --- |
| | | R2 | 0.08 | --- | 0.2 |
| | | S | 0.2 | --- | --- |
| PACKAGE EDGE TOLERANCE | | aaa | 0.2 | | |
| LEAD EDGE TOLERANCE | | bbb | 0.2 | | |
| COPLANARITY | | ccc | 0.08 | | |
| LEAD OFFSET | | ddd | 0.07 | | |
| MOLD FLATNESS | | eee | 0.05 | | |

NOTES

1. DATUM T, U, AND Z TO BE DETERMINED AT DATUM PLANE H.
2. DIMENSIONS D AND E TO BE DETERMINED AT SEATING PLANE DATUM Y.
3. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 PER SIDE. DIMENSIONS D1 AND E1 DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE DATUM H.
4. DIMENSION b DOES NOT INCLUDE DAM BAR PROTRUSION. ALLOWABLE DAM BAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM b DIMENSION BY MORE THAN 0.08 mm. DAM BAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND AN ADJACENT LEAD IS 0.07 mm.

5 EXACT SHAPE OF EACH CORNER IS OPTIONAL.

| | | | |
|---|---|---|---|
| TITLE: PACKAGE OUTLINE 128LD LQFP 14 X 14 X 1.4 PKG. 0.4 PITCH 2mm FOOTPRINT | | PROJ. | |
| | | DWG. NO. | REV. |
| | | 98A0128QP020 | A |
| | | SHEET | SIZE |
| | | 1 OF 1 | A4 |
| UNIT | DIMENSION AND TOLERANCES | REFERENCE DOCUMENT | |
| MM | ASME Y14.5M | --- | |

**Figure 16-1: Package Information – LQFP 128 Pins**

The product(s) listed in this datasheet comply with Directive 2011/65/EU of the European Parliament and of the council of 8 June 2011 on the restriction of the use of certain hazardous substances in electrical and electronic equipment and People's Republic of China Electronic Industry Standard GB/T 26572-2011 "Requirements for concentration limits for certain hazardous substances in electronic information products (电子电器产品中限用物質的限用要求)". Hazardous Substances test report is available upon request.

http://www.solomon-systech.com