Blank Page

SSD2805C

Application Note

MIPI Master Bridge

This document contains information on a product under definition stage. Solomon Systech reserves the right to change or discontinue this product without notice.

CONTENTS

TABLES

FIGURES

1 REGISTERS

This section describes the SSD2805C registers. In the following description, the following key applies:

RO: Read Only
WO: Write Only
RW: Read/Write
RW1C: Read/Write 1 to clear (when 1 is written to the field, the bit is cleared)
RWAC: Read/Write auto clear (the bit is automatically cleared under certain condition)

Table 1-1: SSD2805C Register Summary

Offset	Name	Mnemonic	Reset Value
0xB0	Device Identification Register	DIR	0x2805
0xB1	RGB Interface Control Register 1	VICR1	0x020A
0xB2	RGB Interface Control Register 2	VICR2	0x0214
0xB3	RGB Interface Control Register 3	VICR3	0x0428
0xB4	RGB Interface Control Register 4	VICR4	0x0280
0xB5	RGB Interface Control Register 5	VICR5	0x01E0
0xB6	RGB Interface Control Register 6	VICR6	0x0004
0xB7	Configuration Register	CFGR	0x0301
0xB8	VC Control Register	VCR	0x0045
0xB9	PLL Control Register	PCR	0x0000
0xBA	PLL Configuration Register	PLCR	0x001F
0xBB	Clock Control Register	CCR	0x0003
0xBC	Packet Size Control Register 1	PSCR1	0x0000
0xBD	Packet Size Control Register 2	PSCR2	0x0000
0xBE	Packet Size Control Register 3	PSCR3	0x0100
0xBF	Generic Packet Drop Register	GPDR	0x0000
0xC0	Operation Control Register	OCR	0x0000
0xC1	Maximum Return Size Register	MRSR	0x0001
0xC2	Return Data Count Register	RDCR	0x0000
0xC3	ACK Response Register	ARSR	0x0000
0xC4	Line Control Register	LCR	0x0000
0xC5	Interrupt Control Register	ICR	0x0080
0xC6	Interrupt Status Register	ISR	0xFF06
0xC7	Error Status Register	ESR	0x0000
0xC8	RESERVED		0x0000
0xC9	Delay Adjustment Register 1	DAR1	0x0B02
0xCA	Delay Adjustment Register 2	DAR2	0x2003
0xCB	Delay Adjustment Register 3	DAR3	0x0410
0xCC	Delay Adjustment Register 4	DAR4	0x1003
0xCD	Delay Adjustment Register 5	DAR5	0x1000
0xCE	Delay Adjustment Register 6	DAR6	0x0405
0xCF	HS TX Timer Register 1	HTTR1	0x0000
0xD0	HS TX Timer Register 2	HTTR2	0x0010
0xD1	LP RX Timer Register 1	LRTR1	0x0000
0xD2	LP RX Timer Register 2	LRTR2	0x0010
0xD3	TE Status Register	TSR	0x0000
0xD4	SPI Read Register	LRR	0x00FA
0xD5	Reserved Register		0x0042
0xD6	Test Register	TR	0x0005
0xD7	Read Register	RR	0x0000

1.1 Device Identification Register

Offset Address

DIR **Device Identification Register** **0xB0**

BIT	15	14	13	12	11	10	9	8
NAME	**DIR[15:8]**							
TYPE	RO							
RESET	0x28							
BIT	7	6	5	4	3	2	1	0
NAME	**DIR[7:0]**							
TYPE	RO							
RESET	0x05							

Name	Description	Setting
DIR Bit 15-0	**Device Identification Number**	

1.2 RGB Interface Control Register 1

Offset Address

VICR1 **RGB Interface Control Register 1** **0xB1**

BIT	15	14	13	12	11	10	9	8
NAME	**VSA**							
TYPE	RW							
RESET	0x2							

BIT	7	6	5	4	3	2	1	0
NAME	**HSA**							
TYPE	RW							
RESET	0xA							

Name	Description	Setting
VSA Bit 15-8	**Vertical Sync Active Period** Specify the VSYNC active period in number of HSYNC pulses. The VSYNC active period is from the VSYNC falling edge to VSYNC rising edge. This value is used only used in Non-Burst mode with Sync Pulses.	Minimum value is 1.
HSA Bit 7-0	**Horizontal Sync Active Period** Specify the HSYNC active period in number of PCLK. The HSYNC active period is from the HSYNC falling edge to the HSYNC rising edge, This value is only used in Non-burst Mode with Sync Pulses.	Minimum value is 1.

Refer to Section 3.1 for details.

1.3 RGB Interface Control Register 2

Offset Address

VICR2 **RGB Interface Control Register 2** **0xB2**

BIT	15	14	13	12	11	10	9	8
NAME	**VBP**							
TYPE	RW							
RESET	0x2							

BIT	7	6	5	4	3	2	1	0
NAME	**HBP**							
TYPE	RW							
RESET	0x14							

Name	Description	Setting
VBP Bit 15-8	**Vertical Back Porch Period** Specify the vertical back porch period in terms of HSYNC pulses. In Burst Mode and in Non-burst Mode with Sync Pulses, the Vertical Back Porch Period is from the VSYNC rising edge to the HSYNC of the first line of active display. In Non-burst Mode with Sync Events, the Vertical Back Porch Period is from the VSYNC falling edge to the HSYNC of the first line of active display.	
HBP Bit 7-0	**Horizontal Back Porch Period** Specify the horizontal back porch period in terms of PCLK. In Non Burst Mode with Sync Pulses, the Horizontal Back Porch Period is from the HSYNC rising edge to the start of the valid display pixel. In Non Burst mode with Sync Events and Burst Mode, the Horizontal Back Porch Period is from the HSYNC falling edge to the start of the valid display pixel.	The minimum value is 1.

Refer to Section 3.1 for details.

1.4 RGB Interface Control Register 3

		Offset Address
VICR3	**RGB Interface Control Register 3**	**0xB3**

BIT	15	14	13	12	11	10	9	8
NAME	**VFP**							
TYPE	RW							
RESET	0x4							

BIT	7	6	5	4	3	2	1	0
NAME	**HFP**							
TYPE	RW							
RESET	0x28							

Name	Description	Setting
VFP Bit 15-8	**Vertical Front Porch Period** 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.	
HFP Bit 7-0	**Horizontal Front Porch Period** 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.	

Refer to Section 3.1 for details.

1.5 RGB Interface Control Register 4

Offset Address

VICR4 **RGB Interface Control Register 4** **0xB4**

BIT	15	14	13	12	11	10	9	8
NAME	**HACT[15:8]**							
TYPE	RW							
RESET	0x2							

BIT	7	6	5	4	3	2	1	0
NAME	**HACT[7:0]**							
TYPE	RW							
RESET	0x80							

Name	Description	Setting
HACT Bit 15-0	**Horizontal Active Period** Specifies the Horizontal Active Period in terms of PCLK. During this period, the DEN signal should always be high.	The minimum value is 1.

Refer to Section 3.1 for details.

1.6 RGB Interface Control Register 5

Offset Address

VICR5 **RGB Interface Control Register 5** **0xB5**

BIT	15	14	13	12	11	10	9	8
NAME	**VACT[15:8]**							
TYPE	RW							
RESET	0x1							

BIT	7	6	5	4	3	2	1	0
NAME	**VACT[7:0]**							
TYPE	RW							
RESET	0xE0							

Name	Description	Setting
VACT Bit 15-0	**Vertical Active Period** Specifies the Vertical Active Period in terms of HSYNC pulses.	The minimum value is 1.

Refer to Section 3.1 for details.

1.7 RGB Interface Control Register 6

Offset Address

VICR6 **RGB Interface Control Register 6** **0xB6**

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME		**NVD**	**BLLP**	**VCS**	**VM**		**VPF**	
TYPE		RW	RW	RW	RW		RW	
RESET		0x0	0x0	0x0	0x1		0x0	

Name	Description	Setting
Bit 15-7	***Not Used***	
NVD Bit 6	**Non Video Data Transmission Control** The SSD2805C will send non video data (written from the SPI interface) during the vertical blanking period (in Non Burst Mode) or any BLLP period (in Burst Mode). This bit selects whether HS or LP is used to send the data. 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. *Note:* ▪ *Sending data in LP mode is much slower than in HS mode. The host processor shall make sure that the duration is long enough to finish the data transfer* ▪ *The timing of HSYNC and VSYNC is not affected.*	0: Non video data will be transmitted using HS mode. 1: Non video data will be transmitted using LP mode.
BLLP Bit 5	**BLLP Control** The SSD2805C will use this field to decide whether to send Blanking Packet or enter LP mode in the BLLP period. *This field takes effect only in Non Burst Mode and when **NVD** (Register 0xB6 Bit 6) is 0.* *Note:* ▪ *Sending data in LP mode is much slower than in HS mode. The host processor shall make sure that the duration is long enough to finish the data transfer.* ▪ *The timing of HSYNC and VSYNC is not affected.* ▪ *Refer to Table 2-1 for details of Video Mode operations*	0: Blanking Packet will be sent during BLLP period. 1: LP mode will be used during BLLP period.
VCS Bit 4	**Video Clock Suspend** Specifies the clock lane behavior when there is no data to transmit in Burst Mode. *Note:*	0: The clock lane remains in HS mode, when there is no data to transmit. 1: The clock lane enters LP mode

Name	Description	Setting
	In Non Burst Mode, the clock lane will remain in HS mode all the time irrespective to the setting of this field.	when there is no data to transmit.
VM Bit 3-2	**Video Mode** Specifies the video mode the SSD2805C will use when RGB interface is selected.	00: Non Burst Mode with Sync Pulses 01: Non Burst mode with Sync Events 10: Burst Mode *11: Reserved*
VPF Bit 1-0	**Video Pixel Format** Specifies the pixel format for video mode.	00: 16bpp 01: 18bpp, packed 10: 18bpp, loosely packed 11: 24bpp

1.8 Configuration Register

Offset Address

CFGR **Configuration Register** **0xB7**

BIT	15	14	13	12	11	10	9	8
NAME	**Reserved**						**EOT**	**ECD**
TYPE	RW						RW	RW
RESET	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	RWAC
RESET	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x1

Name	Description	Setting
Bit 15-12	**Reserved**	Default value is 0000. User should not alter the value of this field
Bit 11-10	**Not Used**	
EOT Bit 9	**EOT Packet Enable** Specifies whether the SSD2805C will send out the EOT packet at the end of HS transmission or not.	0: Do not send 1: Send
ECD Bit 8	**ECC CRC Check Disable** Specifies whether SSD2805C will perform ECC and CRC checking for the packets received from the MIPI slave.	0: Enable 1: Disable
REN Bit 7	**Read Enable** Specifies whether the next operation is a write or a read operation.	0: Write operation 1: Read operation
DCS Bit 6	**DCS Enable** Specifies whether the packet to be sent is a DCS Packet or a Generic Packet. This field applies to both write and read operation.	0: Generic Packet (The packet can be any one of Generic Long Write, Generic Short Write, or Generic Read packet) 1: DCS Packet (The packet can be any one of DCS Long Write, DCS Short Write, or DCS Read Packet)
CSS Bit 5	**Clock Source Select** Selects the clock source for the PLL.	0: The clock source is TX_CLK 1: The clock source is PCLK
HCLK Bit 4	**HS Clock Disable** Controls the clock lane behavior during reverse direction communication. This field takes effect only when both **CKE** (register 0xB7 bit 1) and **VEN** (register 0xB7 bit 3)are 0.	0: HS clock is enabled 1: HS clock is disabled
VEN Bit 3	**Video Mode Enable** Controls the video mode operation when interface is RGB + SPI.	0: Video Mode is disabled 1: Video Mode is enabled

Name	Description	Setting
SLP Bit 2	**Ultra Low Power State Enable** Controls the Ultra Low Power State (ULPS) operation. *Note:* *When this field is set to 1, **HS** (register 0xB7 bit 0) will be cleared to 0 automatically.*	0: Ultra Low Power State is disabled 1: Ultra Low Power State is enabled.
CKE Bit 1	**Clock Lane Enable** Controls the mode of the clock lane when data lane enters LP mode.	0: In forward direction communication, the clock lane will enter LP mode, In reverse direction communication, the clock lane will follow the setting of **HCLK** (register 0xB7 bit 4) 1: Clock lane will enter HS mode in both forward direction and reverse direction communication.
HS Bit 0	**HS Mode** Controls whether the SSD2805C is using HS or LP mode to send data. *Note:* *This field will be cleared to 0 automatically when **SLP** (register 0xB7 bit 2) is set to 1.*	0: LP mode 1: HS mode

1.9 Virtual Channel Control Register

Offset Address

VCR **Virtual Channel Control Register** **0xB8**

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME	**VCM**		**VCE**		**VC2**		**VC1**	
TYPE	RW		RW		RW		RW	
RESET	0x1		0x0		0x1		0x1	

Name	Description	Setting
Bit 15-8	**Not Used**	
VCM Bit 7-6	**Virtual Channel ID for Maximum Return Size Packet** Specifies the Virtual Channel ID (VC ID) for the Maximum Return Size Packet sent by SSD2805C. *Note:* *The Virtual Channel for the Maximum Return Size Packet can be different from the Virtual Channel for the Data Packet(s). This field shall be set to ensure the Maximum Return Size Packet is sent to the correct Virtual Channel*	
VCE Bit 5-4	**Virtual Channel ID for EOT Packet** Specifies the Virtual Channel ID for the EOT Packet sent by SSD2805C. *Note:* *The Virtual Channel for the Maximum Return Size Packet can be different from the Virtual Channel for the Data Packet(s). This field shall be set to ensure the EOT Packet is sent to the correct Virtual Channel.*	
VC2 Bit 3-2	**Virtual Channel ID for SPI Interface** Specifies the Virtual Channel ID for the packets written in through the SPI interface, when the interface setting is RGB + SPI. *Note:* *When using RGB + SPI interface, two panels can be addressed independently by setting different Virtual Channel ID in this field and the VC1 field (register 0xB8 bit 1-0)*	
VC1 Bit 1-0	**Virtual Channel ID for RGB and MCU Interface** Specifies the Virtual Channel ID for the packets written in through the RGB interface and the MCU interface. *Note:* *When using RGB + SPI interface, two panels*	

Name	Description	Setting
	can be addressed independently by setting different Virtual Channel ID in this field and the VC2 field (register 0xB8 bit 3-2)	

1.10 PLL Control Register

		Offset Address
PCR	**PLL Control Register**	**0xB9**

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME								**PEN**
TYPE								RW
RESET								0x0

Name	Description	Setting
Bit 15-1	**Not Used**	
PEN Bit 0	**PLL Enable** Controls the PLL operation.	0: Disable 1: Enable

1.11 PLL Configuration Register

Offset Address

PLCR **PLL Configuration Register** **0xBA**

BIT	15	14	13	12	11	10	9	8
NAME	**PDIV**				**DIV**			
TYPE	RW				RW			
RESET	0x0				0x0			

BIT	7	6	5	4	3	2	1	0
NAME	**MUL**							
TYPE	RW							
RESET	0x1F							

Name	Description	Setting
PDIV Bit 15-12	**PLL Post Divider** Specifies the PLL Post Divider value P.	0x0: 1 0x1: 2 … 0xF: 16
DIV Bit 11-8	**PLL Divider** Specifies the PLL Divider value N.	0x0: 1 0x1: 2 … 0xF: 16
MUL Bit 7-0	**PLL Multiplier** Specifies the PLL Multiplier value M.	0x1: 2 0x2: 3 … 0xFF: 256 0 is not allowed in this field.

Refer to Section 3.11 for PLL setting.

1.12 Clock Control Register

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME	**SYSD**		**LPD**					
TYPE	RW		RW					
RESET	0x0		0x3					

Name	Description	Setting
Bit 15-8	**Not Used**	
SYSD Bit 7-6	**SYS_CLK Divider** Specifies the divider value for generating the SYS_CLK from the TX_CLK	Frequency setting: 00: SYS_CLK = TX_CLK 01: SYS_CLK = TX_CLK / 2 10: SYS_CLK = TX_CLK / 3 11: SYS_CLK = TX_CLK / 4
LPD Bit 5-0	**LP Clock Divider** Specifies the divider value for generating the LP mode clock from the byte clock (i.e. 1/8 of PLL clock frequency).	Frequency setting: 0x0: LP Clock = byte clock 0x1: LP Clock = byte clock / 2 … 0x3F: LP Clock = byte clock / 64

1.13 Packet Size Control Register 1 & 2

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME	**TDC[15:8]**							
TYPE	RW							
RESET	0x0							

BIT	7	6	5	4	3	2	1	0
NAME	**TDC[7:0]**							
TYPE	RW							
RESET	0x0							

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME	**TDC[31:24]**							
TYPE	RW							
RESET	0x0							

BIT	7	6	5	4	3	2	1	0
NAME	**TDC[23:16]**							
TYPE	RW							
RESET	0x0							

Name	Description	Setting
TDC Bit 31-0	**Transmit Data Count** Specifies the total number of data bytes to be transmitted by the SSD2805C in the next operation. The SSD2805C will use this value to decide the packet type to be sent out. This field is used together with **PST** (register 0xBE bit 11-0) to decide the outgoing packet length in the following two cases: ▪ Generic Long Write Packet ▪ DCS Long Write Packet with the DCS command 0x2C or 0x3C	

1.14 Packet Size Control Register 3

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME					**PST[11:8]**			
TYPE					RW			
RESET					0x1			

BIT	7	6	5	4	3	2	1	0
NAME	**PST[7:0]**							
TYPE	RW							
RESET	0x0							

Name	Description	Setting
Bit 15-12	**Not Used**	
PST Bit 11-0	**Packet Size Threshold** The threshold value for partitioning the incoming long packet data into smaller packets. The partitioning only applies to Generic Long Write Packet and DCS Long Write Packet with DCS command 0x2C or 0x3C.	Minimum value is 2. Maximum value is 0x400. *Note:* *When 16 bit interface is selected, the value in this field must be an even number. Odd number written in will be automatically truncated to an even number. For example, 0x5 will be truncated to 0x4.*

1.15 Generic Packet Drop Register

Offset Address

GPDR **Generic Packet Drop Register** **0xBF**

BIT	15	14	13	12	11	10	9	8
NAME	**GPD[15:8]**							
TYPE	WO							
RESET	0x0							

BIT	7	6	5	4	3	2	1	0
NAME	**GPD[7:0]**							
TYPE	WO							
RESET	0x0							

Name	Description	Setting
GPD Bit 15-0	**Generic Packet Drop** The host processor uses this register as a FIFO to continuously write the payload of the generic packets (Generic Short Write, Generic Long Write, and Generic Read) into SSD2805C. The SSD2805C will send the payload out using the corresponding generic packet. *Note:* *When the interface is 16 bit, the width of this field is 16 bit. When the interface is 8 bit, the width of this field is 8 bit.*	

1.16 Operation Control Register

Offset Address

OCR **Operation Control Register** **0xC0**

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME								**COP**
TYPE								RWAC
RESET								0x0

Name	Description	Setting
Bit 15-1	**Not Used**	
COP Bit 0	**Cancel Operation** When this field is set to 1, the SSD2805C will stop any further transmission after finish transmitting the current packet. The internal buffer will be cleared and the state machine will return to its initial state. After the Cancel Operation ▪ This field will be automatically reset to 0 ▪ **PO** field (register 0xC6 bit 1) of the Interrupt Status Register will be set to 1 ▪ No data in the internal buffer ▪ The host processor can start a new operation	0: NOP 1: Cancel the current operation

1.17 Maximum Return Size Register

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME	**MRS[15:8]**							
TYPE	RW							
RESET	0x0							

BIT	7	6	5	4	3	2	1	0
NAME	**MRS[7:0]**							
TYPE	RW							
RESET	0x1							

Name	Description	Setting
MRS Bit 15-0	**Maximum Return Size** Sets the maximum return size of the Read Response Packet returned by the MIPI slave. The SSD2805C will send out a Set Maximum Return Size Packet using the value in this field before every read operation to indicate to the MIPI slave about the limit of the SSD2805C. *Note:* *The host processor does not need to program the register before every read operation unless the maximum return size is changed.*	

1.18 Return Data Count Register

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME	**RDC[15:8]**							
TYPE	RO							
RESET	0x0							

BIT	7	6	5	4	3	2	1	0
NAME	**RDC[7:0]**							
TYPE	RO							
RESET	0x0							

Name	Description	Setting
RDC Bit 15-0	**Return Data Count** Reflects the number of data bytes received from the MIPI slave Read Response Packet. This field is updated by the SSD2805C.	

1.19 ACK Response Status Register

Offset Address

ARSR **ACK Response Status Register** **0xC3**

BIT	15	14	13	12	11	10	9	8
NAME	**AR[15:8]**							
TYPE	RO							
RESET	0x0							

BIT	7	6	5	4	3	2	1	0
NAME	**AR[7:0]**							
TYPE	RO							
RESET	0x0							

Name	Description	Setting
AR Bit 15-0	**ACK Response** Contains the ACK response from the MIPI slave. This field will be updated by SSD2805C when ACK with Error Report Packet is received.	

1.20 Line Control Register

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME						**FBC**	**FBT**	**FBW**
TYPE						RW	RW	RW
RESET						0x0	0x0	0x0

Name	Description	Setting
Bit 15-3	**Not Used**	
FBC Bit 2	**Force Bus Contention** Force a bus contention on the data lane when this field is set to 1. This field will be changed to 0 after the bus contention is not detected.	0: NOP 1: Drive the data lane to LP11 to force a bus contention.
FBT Bit 1	**Force BTA TE** When this field is set, a BTA will be automatically performed after any BTA so as to get the TE response from the MIPI slave.	0: No BTA after previous BTA 1: Perform automatic BTA after previous BTA
FBW Bit 0	**Force BTA After Write** When this bit is set, the BTA will be performed after a write operation. *Note:* *The bus authority will be passed to the MIPI slave after BTA. SSD2805C will not be able to send data before the bus authority is passed back.*	0: No BTA after the next write packet 1: Automatically perform BTA after the next write packet

1.21 Interrupt Control Register

Offset Address

ICR	Interrupt Control Register	0xC5

BIT	15	14	13	12	11	10	9	8
NAME	**LSEE**	**LSAE**	**LLEE**	**LLAE**	**MSEE**	**MSAE**	**MLEE**	**MLAE**
TYPE	RW	RW	RW	RW	RW	RW	RW	RW
RESET	0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0

BIT	7	6	5	4	3	2	1	0
NAME	**PLSE**	**LPTOE**	**HSTOE**		**ARRE**	**BTARE**	**POE**	**RDRE**
TYPE	RW	RW	RW		RW	RW	RW	RW
RESET	0x1	0x0	0x0		0x0	0x0	0x0	0x0

Name	Description	Setting
LSEE Bit 15	**SPI Short Buffer Empty Enable** Enables the mapping of **LSE** interrupt to the INT# pin.	0: Not enabled 1: Enabled
LSAE Bit 14	**SPI Short Buffer Available Enable** Enables the mapping of **LSA** interrupt to the INT# pin.	0: Not enabled 1: Enabled
LLEE Bit 13	**SPI Long Buffer Empty Enable** Enables the mapping of **LLE** interrupt to the INT# pin.	0: Not enabled 1: Enabled
LLAE Bit 12	**SPI Long Buffer Available Enable** Enables the mapping of **LLA** interrupt to the INT# pin.	0: Not enabled 1: Enabled
MSEE Bit 11	**MCU Short Buffer Empty Enable** Enables the mapping of **MSE** interrupt to the INT# pin.	0: Not enabled 1: Enabled
MSAE Bit 10	**MCU Short Buffer Available Enable** Enables the mapping of **MSA** interrupt to the INT# pin.	0: Not enabled 1: Enabled
MLEE Bit 9	**MCU Long Buffer Empty Enable** Enables the mapping of **MLE** interrupt to the INT# pin.	0: Not enabled 1: Enabled
MLAE Bit 8	**MCU Long Buffer Available Enable** Enables the mapping of **MLA** interrupt to the INT# pin.	0: Not enabled 1: Enabled
PLSE Bit 7	**PLL Lock Status Enable** Enables the mapping of the **PLS** interrupt to the INT# pin.	0: Not enabled 1: Enabled
LPTOE Bit 6	**LP RX Time Out Enable** Enables the mapping of the **LPTO** interrupt to the INT# pin.	0: Not enabled 1: Enabled
HSTOE Bit 5	**HS TX Time Out Enable** Enables the mapping of the **HSTO** interrupt to the INT# pin.	0: Not enabled 1: Enabled
Bit 4	**Not Used**	

Name	Description	Setting
ARRE Bit 3	**ACK Response Ready Enable** Enables the mapping of **ARR** interrupt to the INT# pin.	0: Not enabled 1: Enabled
BTARE Bit 2	**BTA Response Enable** Enables the mapping of the **BTAR** interrupt to the INT# pin.	0: Not enabled 1: Enabled
POE Bit 1	**Packet Operation Enable** Enables the mapping of the **PO** interrupt to the INT# pin.	0: Not enabled 1: Enabled
RDRE Bit 0	**Read Data Ready Enable** Enables the mapping of the **RDR** interrupt to the INT# pin.	0: Not enabled 1: Enabled

Refer to Section 1.22 for detail description of the interrupt sources.

1.22 Interrupt Status Register

ISR	Interrupt Status Register	Offset Address 0xC6

BIT	15	14	13	12	11	10	9	8
NAME	**LSE**	**LSA**	**LLE**	**LLA**	**MSE**	**MSA**	**MLE**	**MLA**
TYPE	RO	RO	RO	RO	RO	RO	RO	RO
RESET	0x1	0x1	0x1	0x1	0x1	0x1	0x1	0x1

BIT	7	6	5	4	3	2	1	0
NAME	**PLS**	**LPTO**	**HSTO**	**ATR**	**ARR**	**BTAR**	**PO**	**RDR**
TYPE	RO	RW1C	RW1C	RW1C	RW1C	RW1C	RO	ROAC
RESET	0	0	0	0	0	0x1	0x1	0

Name	Description	Setting
LSE Bit 15	**SPI Short Buffer Empty** Reflects the status of the internal short buffer of the SPI interface.	0: The short buffer is not empty. 1: The short buffer is empty.
LSA Bit 14	**SPI Short Buffer Available** Reflects the status of the internal short buffer of the SPI interface.	0: The short buffer is full. 1: The short buffer is not full.
LLE Bit 13	**SPI Long Buffer Empty** Reflects the status of the internal long buffer of the SPI interface.	0: The long buffer is not empty. 1: The long buffer is empty.
LLA Bit 12	**SPI Long Buffer Available** Reflects the status of the internal long buffer of the SPI interface.	0: The long buffer is full. 1: The long buffer is not full.
MSE Bit 11	**MCU Short Buffer Empty** Reflects the status of the internal short buffer of the MCU interface.	0: The short buffer is not empty. 1: The short buffer is empty.
MSA Bit 10	**MCU Short Buffer Available** Reflects the status of the internal short buffer of the MCU interface.	0: The short buffer is full. 1: The short buffer is not full.
MLE Bit 9	**MCU Long Buffer Empty** Reflects the status of the internal long buffer of the MCU interface.	0: The long buffer is not empty. 1: The long buffer is empty.
MLA Bit 8	**MCU Long Buffer Available** Reflects the status of the internal long buffer of the MCU interface.	0: The long buffer is full. 1: The long buffer is not full.
PLS Bit 7	**PLL Lock Status** Reflects the status of the PLL. Before the PLL is locked, the whole system is running at the reference clock input to the PLL, the host processor must access the SSD2805C registers using slow speed.	0: PLL has not been locked 1: PLL has been locked
LPTO Bit 6	**LP RX Time Out** Reflects the status of the LP RX timer. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: The LP RX timer has expired. 1: The LP RX timer has not expired.

Name	Description	Setting
HSTO Bit 5	**HS TX Time Out** Reflects the status of the HS TX timer. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: The HS TX timer has expired. 1: The HS TX timer has not expired.
ATR Bit 4	**ACK Trigger Response** Reflects whether an ACK trigger message has been received or not. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: ACK trigger message has not been received. 1: ACK trigger message has been received.
ARR Bit 3	**ACK Response Ready** Reflects whether an ACK response has been received or not. *Note:* *ACK response can be an ACK trigger message or ACK with Error Report Packet.* *This field shall be cleared by writing a "1" to the field.*	0: Response has not been received. 1: Response has been received.
BTAR Bit 2	**BTA Response** Reflects the data lane status after SSD2805C has made a BTA. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: The MIPI slave has not passed the lane authority back. 1: The MIPI slave has passed the lane authority back.
PO Bit 1	**Packet Operation** Reflects whether the SSD2805C is ready to take in more data from the host processor.	0: Not ready 1: Ready
RDR Bit 0	**Read Data Ready** Reflects whether the data from the MIPI slave is ready for read by the host processor. This field is valid only during the read operation. This field will be automatically cleared when all the received data are read out.	0: Not ready 1: Ready

1.23 Error Status Register

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME						**CRCE**	**ECCE2**	**ECCE1**
TYPE						RW1C	RW1C	RW1C
RESET						0	0	0

BIT	7	6	5	4	3	2	1	0
NAME	**LSO**	**LLO**	**MSO**	**MLO**	**VIE**	**CONT**		**VMM**
TYPE	RW1C	RW1C	RW1C	RW1C	RW1C	RO		RW1C
RESET	0	0	0	0	0	0		0

Name	Description	Setting
Bit 15-11	**Not Used**	
CRCE Bit 10	**CRC Error** Reflects the status of CRC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** (register 0xB7 bit 8) is set to 0. *Note:* *This field shall be cleared by writing a “1” to the field.*	0: No CRC error 1: At least 1 CRC error
ECCE2 Bit 9	**ECC Multi-Bit Error** Reflects the status of ECC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** (register 0xB7 bit 8) is set to 0. *Note:* *This field shall be cleared by writing a “1” to the field.*	0: No ECC Multi-Bit Error 1: At least 1 ECC Multi-Bit Error
ECCE1 Bit 8	**ECC Single-Bit Error** Reflects the status of ECC checking for the packets received from the MIPI slave. The status is valid only when the **ECD** (register 0xB bit 8) is set to 0. *Note:* *This field shall be cleared by writing a “1” to the field.*	0: No ECC Single-Bit Error 1: At least 1 ECC Single-Bit Error
LSO Bit 7	**SPI Short Buffer Overflow** Reflects the status of internal short buffer of the SPI interface. *Note:* *This field shall be cleared by writing a “1” to the field.*	0: No overflow 1: Overflow detected
LLO Bit 6	**SPI Long Buffer Overflow** Reflects the status of internal long buffer of the SPI interface. *Note:* *This field shall be cleared by writing a “1” to the*	0: No overflow 1: Overflow detected

Name	Description	Setting
	field.	
MSO Bit 5	**MCU Short Buffer Overflow** Reflects the status of internal short buffer of the MCU interface. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: No overflow 1: Overflow detected
MLO Bit 4	**MCU Long Buffer Overflow** Reflects the status of internal long buffer of the MCU interface. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: No overflow 1: Overflow detected
VIE Bit 3	**RGB Interface Error** Reflects whether there is any mismatch between the RGB interface configuration register setting and the actual RGB interface waveform. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: No mismatch 1: Mismatch has occurred
CONT Bit 2	**Contention** Reflects the status of the data lane contention detector.	0: No contention 1: Contention has occurred
Bit 1	**Not Used**	
VMM Bit 0	**VC Mismatch** Reflects whether there is a mismatch between the Virtual Channel ID transmitted by the SSD2805C and the Virtual Channel ID received from the MIPI slave. *Note:* *This field shall be cleared by writing a "1" to the field.*	0: No mismatch 1: Mismatch has occurred

1.24 Reserved Register 0xC8

Offset Address

Reserved Register 0xC8

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME				**Reserved**				
TYPE				RW				
RESET				0x0				

Name	Description	Setting
Bit 15-5	**Not Used**	
Bit 4-0	**Reserved**	Default value 00000. User should not alter the value of this field

1.25 Delay Adjustment Register 1

Offset Address

DAR1 Delay Adjustment Register 1 0xC9

BIT	15	14	13	12	11	10	9	8
NAME	**HZD**							
TYPE	RW							
RESET	0xB							

BIT	7	6	5	4	3	2	1	0
NAME	**HPD**							
TYPE	RW							
RESET	0x2							

Name	Description	Setting
HZD Bit 15-8	**HS Zero Delay** Specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{HS-ZERO}$ (HS zero delay period).	
HPD Bit 7-0	**HS Prepare Delay** Specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{HS-PREPARE}$ (HS prepare delay period)	

1.26 Delay Adjustment Register 2

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME	**CZD**							
TYPE	RW							
RESET	0x20							

BIT	7	6	5	4	3	2	1	0
NAME	**CPD**							
TYPE	RW							
RESET	0x3							

Name	Description	Setting
CZD Bit 15-8	**CLK Zero Delay** Specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{CLK-ZERO}$ (CLK zero delay period)	
CPD Bit 7-0	**CLK Prepare Delay** Specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{CLK-PREPARE}$ (CLK prepare delay period)	

1.27 Delay Adjustment Register 3

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME	**CPED**							
TYPE	RW							
RESET	0x4							

BIT	7	6	5	4	3	2	1	0
NAME	**CPTD**							
TYPE	RW							
RESET	0x10							

Name	Description	Setting
CPED Bit 15-8	**CLK Pre Delay** Specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{CLK-PER}$. (CLK pre delay period)	
CPTD Bit 7-0	**CLK Post Delay** Specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{CLK-POST}$ (CLK post delay period).	

1.28 Delay Adjustment Register 4

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

BIT	15	14	13	12	11	10	9	8
NAME	**CTD**							
TYPE	RW							
RESET	0x10							

BIT	7	6	5	4	3	2	1	0
NAME	**HTD**							
TYPE	RW							
RESET	0x3							

Name	Description	Setting
CTD Bit 15-8	**CLK Trail Delay** Specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{CLK\text{-}TRAIL}$ (CLK trail delay period).	
HTD Bit 7-0	**HS Trail Delay** – These bits specifies the number of nibble clocks (i.e. 1/4 of PLL clock frequency) for $T_{HS\text{-}TRAIL}$ (HS trail delay period).	

1.29 Delay Adjustment Register 5

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

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	0x0							

Name	Description	Setting
WUD Bit 15-0	**Wake Up Delay** The Wake Up Delay is the time to wake up the MIPI slave from Ultra Low Power State This field specifies the wake up delay period T_{WAKEUP} in terms of number of PLL cycles.	

1.30 Delay Adjustment Register 6

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

BIT	15	14	13	12	11	10	9	8
NAME					**TGO**			
TYPE					RW			
RESET					0x4			

BIT	7	6	5	4	3	2	1	0
NAME					**TGET**			
TYPE					RW			
RESET					0x5			

Name	Description	Setting
Bit 15-12	**Not Used**	
TGO Bit 11-8	**TA Go Delay** Specifies the number of T_{LPX} for TA go delay period $T_{TA\text{-}GO}$.	
Bit 7-4	**Not Used**	
TGET Bit 3-0	**TA Get Delay** Specifies the number of T_{LPX} for TA get delay period $T_{TA\text{-}GET}$.	

Refer to Section 3.4 for the readback operation.

1.31 HS TX Timer Register 1 & 2

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME	**HTT[15:8]**							
TYPE	RW							
RESET	0x00							

BIT	7	6	5	4	3	2	1	0
NAME	**HTT[7:0]**							
TYPE	RW							
RESET	0x00							

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

BIT	15	14	13	12	11	10	9	8
NAME	**HTT[31:24]**							
TYPE	RW							
RESET	0x00							

BIT	7	6	5	4	3	2	1	0
NAME	**HTT [23:16]**							
TYPE	RW							
RESET	0x10							

Name	Description	Setting
HTT Bit 31-0	**HS TX Timer** Specifies the HS TX timer timeout value in number of PLL clock. The HS TX timer will be started when the SSD2805C enters HS transmit mode. If the timer expires before HS transmission completed, the SSD2805C will ▪ set **HSTO** (register 0xC6 bit 5) to 0 to signal the time out ▪ switch to LP transmit mode to continue the transmission ▪ clear the **HS** field (register 0xB7 bit 0) to 0 *Note:* *When the SSD2805C exits from HS transmit mode, the HS TX timer will be reset.* *Software intervention is required so that the SSD2805C can go back to proper HS transmission mode.*	

1.32 LP RX Timer Register 1 & 2

Offset Address

LRTR 1 **LP RX Timer Register 1** **0xD1**

BIT	15	14	13	12	11	10	9	8
NAME	**LRT[15:8]**							
TYPE	RW							
RESET	0x00							

BIT	7	6	5	4	3	2	1	0
NAME	**LRT[7:0]**							
TYPE	RW							
RESET	0x00							

Offset Address

LRTR 2 **LP RX Timer Register 2** **0xD2**

BIT	15	14	13	12	11	10	9	8
NAME	**LRT[31:24]**							
TYPE	RW							
RESET	0x00							

BIT	7	6	5	4	3	2	1	0
NAME	**LRT[23:16]**							
TYPE	RW							
RESET	0x10							

Name	Description	Setting
LRT Bit 31-0	**LP RX Timer** Specifies the LP RX timer timeout value in number of PLL clock. The LP RX timer will be started when the SSD2805C enters LP receive mode. If the timer expires before LP receive completed, the SSD2805C will ▪ set **LPTO** (register 0xC6 bit 6) to 0 to signal the time out ▪ switch to LP transmit mode ▪ The DPHY will drive the data lane to LP11 state *Note:* *When the SSD2805C exits from LP receive mode, the LP RX timer will be reset.* *Software intervention is required so that any possible error could be cleared.*	

1.33 TE Status Register

Offset Address

TSR **TE Status Register** **0xD3**

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME								**TER**
TYPE								RW1C
RESET								0

Name	Description	Setting
Bit 15-1	**Not Used**	
TER Bit 0	**TE Response** Reflects whether a TE response has been received. When SSD2805C receives a TE response ▪ this field will be set to 1 ▪ the output TE signal will go high *Note:* *The host processor can clear this field by writing a "1" to the field. The TE signal will go low at the same time.*	0: TE response has not been received. 1: TE response has been received.

1.34 SPI Read Register

Offset Address

LRR **SPI Read Register** **0xD4**

BIT	15	14	13	12	11	10	9	8
NAME								
TYPE								
RESET								

BIT	7	6	5	4	3	2	1	0
NAME	**RRA**							
TYPE	RW							
RESET	0xFA							

Name	Description	Setting
Bit 15-8	**Not Used**	
RRA Bit 7-0	**Register Read Address** Specifies the address of the register to be read through the SPI interface when the interface is SPI 8 bit 3 wire or 8 bit 4 wire.	

1.35 Analog Control Register

Offset Address

ACR **Reserved Register** **0xD5**

BIT	15	14	13	12	11	10	9	8
NAME		**Reserved**						
TYPE		RW						
RESET		0x0						

BIT	7	6	5	4	3	2	1	0
NAME	**Reserved**							
TYPE	RW							
RESET	0x42							

Name	Description	Setting
Bit 15	**Not Used**	
Bit 14-8	**Reserved**	Default value 00000000. User should not alter the value of this field
Bit 7-0	**Reserved**	Default value 01000010. User should not alter the value of this field

1.36 Test Register

Offset Address

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

BIT	15	14	13	12	11	10	9	8
NAME								**OE**
TYPE								RW
RESET								0x0

BIT	7	6	5	4	3	2	1	0
NAME	**PNB**						**END**	**CO**
TYPE	RW						RW	RW
RESET	0x1						0	1

Name	Description	Setting
Bit 15-9	**Not Used**	
OE Bit 8	**Output Enable –** This bit controls whether the SSD2805 will output Sys_clk and Test_GPIO[2:0].	0 – Output is disabled 1 – Output is enabled
PNB Bit 7-2	**Packet Number in Blanking Period** Controls the number of packet to send during video mode blanking period.	
END Bit 1	**Endianess** Specifies the endianess of the data transmitted over the MIPI lane. Refer to Section 5 for more details.	0: Least significant byte sent first 1: Most significant byte sent first
CO Bit 0	**Color Order** Specifies the order of the color component in the pixel. Refer to Section 5 for more details.	0: RGB. R is in the higher portion of the pixel. 1 BGR. B is in the higher portion of the pixel.

1.37 Read Register

Offset Address

RR **Read Register** **0xD7**

BIT	15	14	13	12	11	10	9	8
NAME	**RD[15:8]**							
TYPE	RO							
RESET	0x0							

BIT	7	6	5	4	3	2	1	0
NAME	**RD[7:0]**							
TYPE	RO							
RESET	0x0							

Name	Description	Setting
RD Bit 15-0	**Read Data** The host processor uses this register as a FIFO to read the data returned by the MIPI slave. *Note:* ▪ *When the interface is 16 bit, the width of this field is 16 bit. When the interface is 8 bit, the width of this field is 8 bit.* ▪ *The read of this register is only valid when the **RDR** (register 0xC6 bit 0) is 1*	

1.38 DCS Command Set Register

Offset Address

ACR **Reserved Register** **0xFA**

BIT	15	14	13	12	11	10	9	8
NAME		**Reserved**						
TYPE		RW						
RESET		0x0						

BIT	7	6	5	4	3	2	1	0
NAME	**Reserved**							
TYPE	RW							
RESET	0x0							

Name	Description	Setting
	DCS command set for reading register value (SPI interface only	

2 Interface Configuration

The SSD2805C supports two interface configurations:

- MCU interface
- RGB + SPI interface.

Multiple displays can be supported in both configurations.

2.1 RGB + SPI Interface

To select this configuration, the user shall:

- set the IF_SEL pin to low
- select the desired SPI interface by setting PS[1:0]

The host processor can use this configuration to drive

- a dumb display panel
- a smart display panel
- a dumb display panel + a smart display panel

Under this configuration, the RGB interface and SPI interface are operating independently. The RGB interface is used to provide display data. The SPI interface is used to:

- program the local registers of SSD2805C
- program the registers of the dumb display panel if connected
- configure the smart display panel and send data to the smart display panel

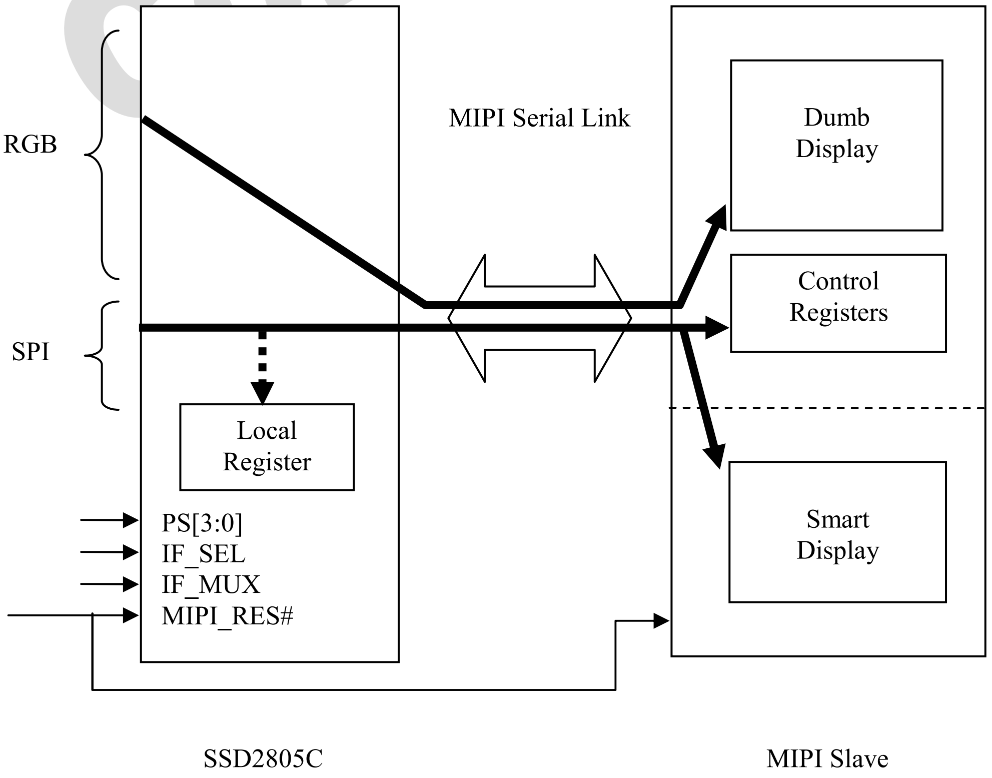

Figure 2-1: SSD2805C with RGB and SPI Interface

The Data for different destinations are separated by the VC (Virtual Channel) field of the packets. The user can program the register **VCR VC Control Register (0xB8)** to set the VC fields. When this configuration is used, the primary usage of the serial link is for sending display video data to the dumb panel. If there is a need to send non-video data through SPI interface concurrently, the SSD2805C can put them into Generic Write or DCS Write Packet and interleave them with the video packets.

If the non-burst video mode is selected, the non-video data packet will only be sent during vertical blanking period. If burst video mode is selected, the non-video data packet can be sent during both horizontal and vertical blanking period (e.g. BLLP period).

During both horizontal and vertical blanking period, the serial link can either remain in HS mode (sending blanking packet) or enter LP mode. The non-video data can also be sent in HS or LP mode. Options have been provided for whether to use HS or LP mode for the blanking period and whether to send the non-video data in HS or LP mode. The **NVD** (register 0xB6 bit 6) and **BLLP** (register 0xB6 bit 5) bits in register **VICR6** (0xB6) are provided for this purpose. Please refer to the table below for details.

Table 2-1: Operation during Video Mode BLLP Period

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. 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. 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. 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.	
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.	

2.2 MCU Interface

The host processor can use this configuration to drive multiple smart display panels. The MCU interface control signals are multiplexed with the RGB interface control signals to reduce the pin count, as the two interfaces do not operate at the same time. Two multiplexing schemes are provided. For the details please refer to Multiplexing Scheme for RGB and MCU interface of data sheet. The user needs to set the IF_SEL pin to high, set IF_MUX pin to select the desired interface multiplexing scheme and set PS[3:2] to select the desired MCU interface. When the MCU interface is not used, the CSX pin needs to be kept high.

Table 2-2: MCU Interface Control Pin Mapping

Pin Name	Description
PS[3:2]	Interface selection signal PS[3:2] is for the MCU interface

	When IF_SEL is 1 - 00: 8 bit 8080 MCU interface - 01: 16 bit 80800 MCU interface - 10: 8 bit 6800 MCU interface - 11: 16 bit 6800 MCU interface
IF_MUX	Interface multiplex selection signal - 0: Multiplex scheme 1 - 1: Multiplex scheme 2

The MCU interface is used to program the local registers of SSD2805C, configure and send display data to the smart panels in command mode. SSD2805C can drive multiple smart panels. Below are some illustrations for the use case.

The data for different destinations are separated by the VC field of the packets. The user can program the **VCR** register (0xD8) to set the VC fields. The multiple smart displays can be updated in a time multiplexing manner.

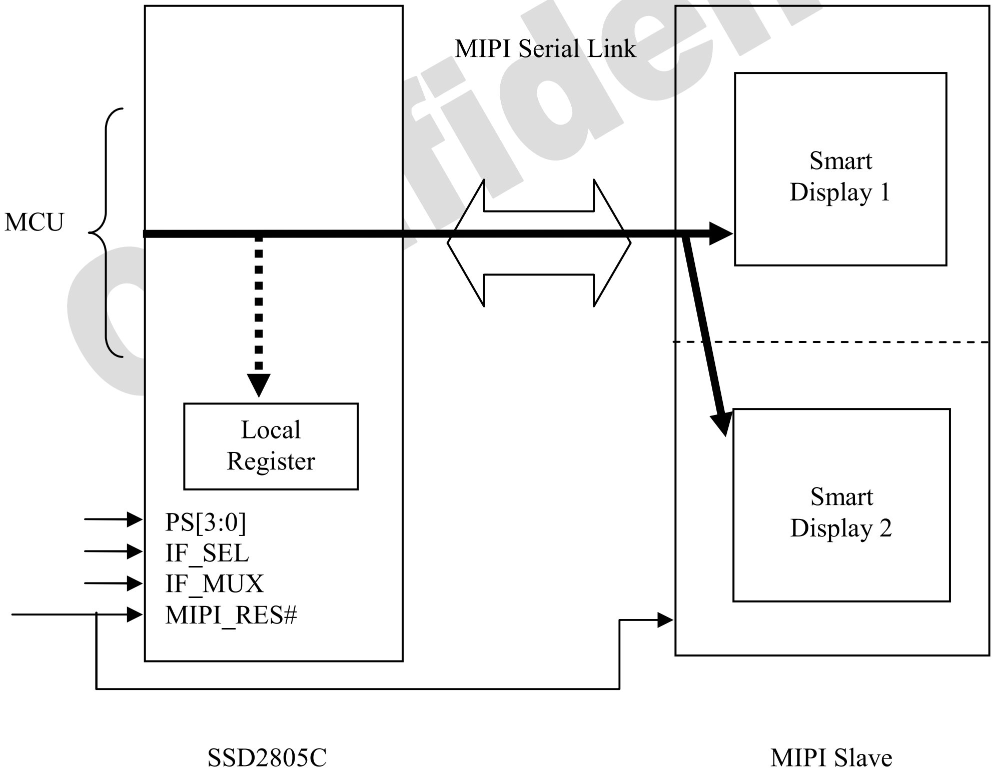

Figure 2-2: SSD2805C with MCU Interface

3 Operating Modes

3.1 Video Mode

To enable the video mode transmission, the user must set IF_SEL to 0 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. To support different bpp settings, the following data pins are used. For all cases, R should be at the upper bits and B should be at the lower bits.

- DATA[15:0] for 16 bpp.
- DATA[17:0] for 18 bpp, packed.
- DATA[17:0] for 18 bpp, loosely packed.
- DATA[24:0] for 24 bpp.

The user, first, needs to program the registers **VICR1** to **VICR6** (0xB1 to 0xB6) with correct values. The user also needs to program the **END** (register 0xD6 bit1) and **CO** (register 0xD6 bit 0) bits to 0 and 1 respectively. After programming those register fields, the user can turn on the RGB interface and enable the **VEN** (register 0xB7bit 3) to start transmission. All three video mode sequence defined in the MIPI DSI specification are supported.

In Non-Burst Mode, the **CSS** (register 0xB7 bit 5) can be set to 0 or 1. When it is set to 1 to select the PCLK as PLL reference clock and the PLL multiplication factor should be equal to the bpp value. When it is set to 0 to select the TX_CLK as PLL reference clock, the PLL multiplication factor should be set such that the serial link data rte is faster than the incoming data rate. Please refer to the table below for the PLL settings. Registers **VICR1** to **VICR6** (0xB1 to 0xB6) needs to be programmed. (**VICR1** is not used for non-burst mode with Sync Events).

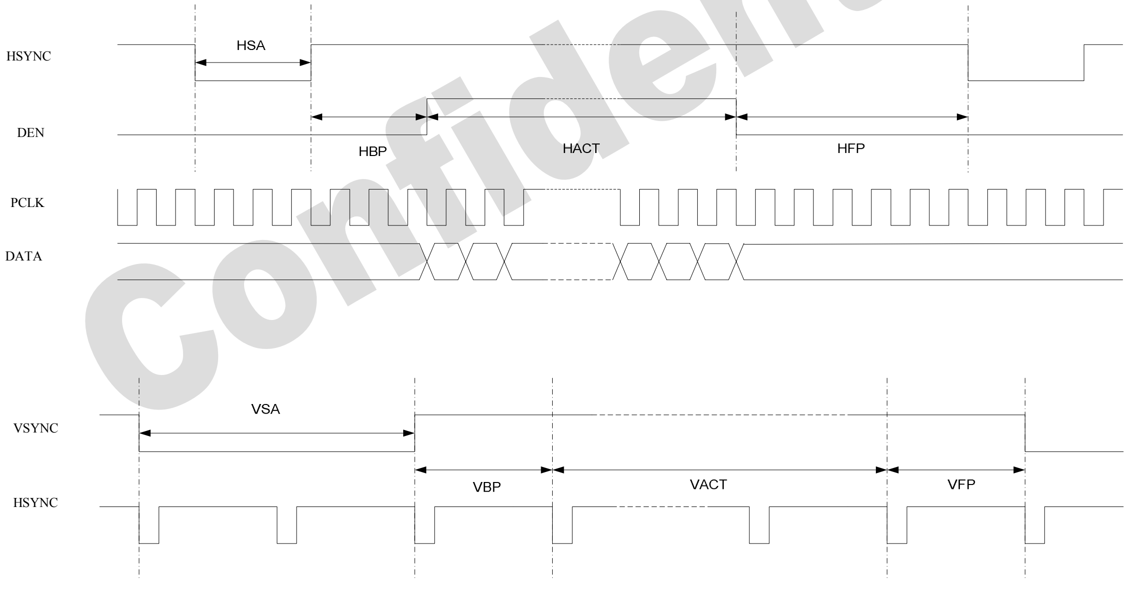

Figure 3-1: Illustration of RGB Interface Parameters for Non-burst Mode with Sync Pulses

Table 3-1: PLL Setting for Non-burst Mode (PLL reference using PCLK)

BPP (bit per pixel)	PLL Multiplication Factor		PLL Output Clock Frequency	
	1 data lane	2 data lane	1 data lane	2 data lane
16	16	8	16 x PCLK	8 x PCLK
18, packed	18	9	18 x PCLK	9 x PCLK
18, loosely packed	24	12	24 x PCLK	12 x PCLK
24	24	12	24 x PCLK	12 x PCLK

*The PLL multiplication should be half in 2 data lanes mode.

Table 3-2: PLL Setting for Non-burst Mode (PLL reference using TX_CLK)

BPP (bit per pixel)	PLL Multiplication Factor		PLL Output Clock Frequency	
	1 data lane	2 data lane	1 data lane	2 data lane
16	NA	NA	>= 16 x PCLK	>= 8 x PCLK
18, packed	NA	NA	>= 18 x PCLK	>= 9 x PCLK
18, loosely packed	NA	NA	>= 24 x PCLK	>= 12 x PCLK
24	NA	NA	>= 24 x PCLK	>= 12 x PCLK

In Burst Mode, the **CSS** (register 0xB7 bit 5) needs to be set to 0 to select TX_CLK as PLL reference clock. 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. Registers **VICR2** to **VICR6** (0xB1 to 0xB6) needs to be programmed. **VICR1** is not used for this mode. The definition of all the fields is the same as non-burst mode with Sync Events.

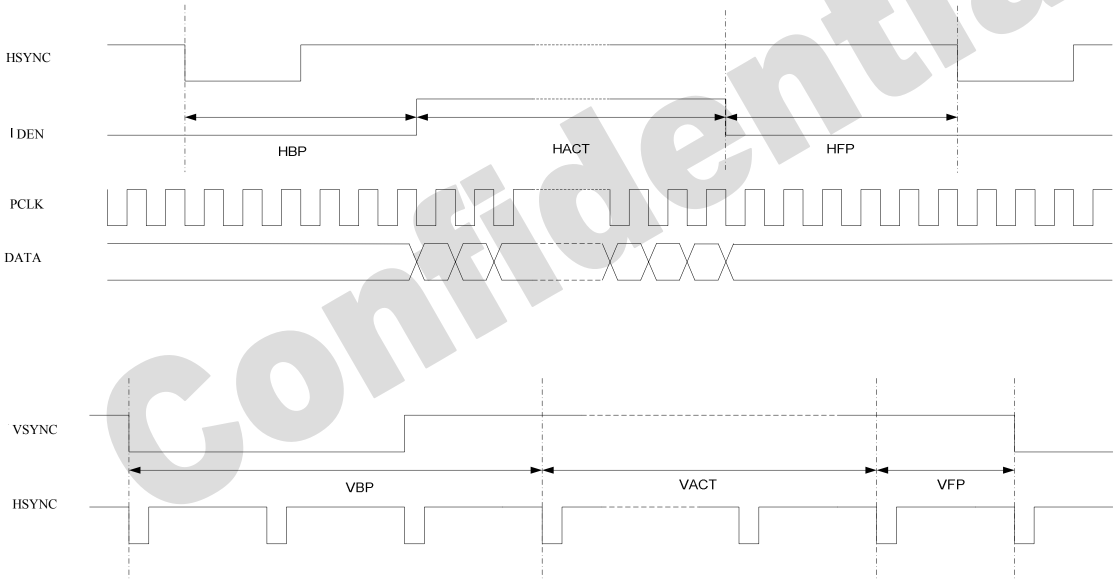

Figure 3-2: Illustration of RGB Interface Parameters for Non-burst Mode with Sync Events and Burst Mode

Table 3-3: PLL Setting for Burst Mode

BPP (bit per pixel)	PLL Multiplication Factor		PLL Output Clock Frequency	
	1 data lane	2 data lane	1 data lane	2 data lane
16	NA	NA	>= 16 x PCLK	>= 8 x PCLK
18, packed	NA	NA	>= 18 x PCLK	>= 9 x PCLK
18, loosely packed	NA	NA	>= 24 x PCLK	>= 12 x PCLK
24	NA	NA	>= 24 x PCLK	>= 12 x PCLK

*: This value should be set such that the serial link data rate is faster than incoming data rate

The SSD2805C will also monitor the status of CM and SHUT signal. When there is a change of these signals, it will send out appropriate packets. On the rising edge of CM, the CM on packet will be sent. On the falling edge of CM, the CM off packet will be sent. On the rising edge of SHUT, the Shut Down Peripheral packet will be sent. On the falling edge of SHUT, the Turn On Peripheral packet will be sent. With these packets, the MIPI slave will be able to reconstruct the RGB interface signals.

As mentioned in 2.1, the 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. Please refer to 2.1 for more details. The command mode operation through SPI interface is identical to the operation through MCU interface. The only difference is the interface type. Hence, please refer to 3.2 for more details.

3.2 Command mode

In command mode, the **CSS** (register 0xB7 bit 5) must be set to 0 to select the TX_CLK as the PLL reference clock. The PCLK pin is used as part of the MCU interface. (When SPI interface is used for command mode, the CSS setting should follow the description in 3.1) MCU interface supports both 8 bit and 16 bit data bus. To support different bus width, the following data pins are used.

- DATA[7:0] for 8 bit interface.

- DATA[15:0] for 16 bit interface.

The address range 0xB0 to 0xD9 is used for the SSD2805C local register. The user can access the registers in this range to configure and control the SSD2805C. The address range 0x00 to 0xAF and 0xDA to 0xFF is used for MIPI DCS command set and its future expansion. When the user writes data to these registers, the data will be sent over the serial link to the MIPI slave. The user can configure and control the MIPI slave through these registers.

3.3 Write Operation

To perform write operation, the user needs to set the **REN** (register 0xB7 bit 7) to 0. The SSD2805C 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 **DCS** (register 0xB7 bit 6) controls whether Generic Write Packet or DCS Write Packet will be sent out. The **VC1** (register 0xB8 bit 1-0) and **VC2** (register 0xB8 bit 3-2) fields determines the VC ID of the outgoing packets.

The SSD2805C needs to know the payload size of the outgoing packets. Hence, the user needs to program the corresponding control registers. The **TDC** field (registers 0xBC and 0xBD) indicates the total number of payload bytes.

To send a DCS Write Packet, the user needs to write the DCS parameters to the corresponding register where the register address is the DCS command. If the **TDC** (registers 0xBC and 0xBD) field is no more than 1, the SSD2805C will send out DCS Short Write Packet with the correct type. Otherwise, DCS Long Write Packet will be sent out.

To send a Generic Write Packet, the user needs to write the payload to the register **GPDR** (0xBF). If the **TDC** field is no more than 2, the SSD2805C will send out Generic Short Write Packet with the correct type. Otherwise, Generic Long Write Packet will be sent out.

Sometimes, the payload size is too big for SSD2805C to send them out in one packet. The SSD2805C will automatically partition the payload into a few smaller packets to send out. The **TDC** is used together with the **PST** (register 0xBE bit 11-0) field to determine the size of outgoing packets. The automatic partition is done differently for DCS Write Packet and Generic Write Packet.

For DCS Write Packet, the partition is only enabled if the DCS command is 0x2C or 0x3C. Otherwise, SSD2805C will not perform automatic partition. (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 (**PST**+1) bytes. The first byte is the DCS command and the following **PST** bytes are the payload. Only the last packet might contain less payload, as the total payload might not be integer multiple of **PST**. 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 Generic Write Packet, the payload will be partitioned into a few packets where the payload of each packet is **PST** (register 0xBE bit 11-0). Only the last packet might contain less payload, as the total payload might not be integer multiple of **PST** (register 0xBE bit 11-0).

For example, if the **TDC** (registers 0xBC and 0xBD) field is 200 and **PST** (register 0xBE bit 11-0) 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. This is done by setting **FBW** (register 0xC4 bit 0) to 1. The SSD2805C will automatically make a BTA after each write operation. Please refer to 3.5 for how to handle the acknowledgement received.

3.4 Read Operation

To perform read operation, the user needs to set the **REN** (register 0xB7 bit 7) to 1. The SSD2805C can issue two kinds of packets for read operation, which are Generic Read Packet, and DCS Read Packet. The bit **DCS** (register 0xB7 bit 6) controls whether Generic Read Packet or DCS Read Packet will be sent out. The **VC1** (register 0xB8 bit 1-0) and **VC2** (register 0xB8 bit 3-2) determine the VC ID of the outgoing packets.

Before the read packet is sent out, the SSD2805C 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 SSD2805C and the other is the limit of the host processor. The limit of SSD2805C is 2048 bytes of data. 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 register **MRSR** (0xC1). The user could program the **MRSR** (0xC1) before every read so that the correct value is sent through Set Maximum Return Size Packet. If the value in the **MRSR** (0xC1) is already the desired value, the user can choose not to program it. The SSD2805C will always automatically send out Set Maximum Return Size Packet using the value in **MRSR** (0xC1).

To send a DCS Read Packet, the user just needs to write the DCS command, as there is no parameter for DCS read.

To send a Generic Read Packet, the user needs to write the payload to the register **GPDR** (0xBF).

Similar to the write operation, the **TDC** (registers 0xBC and 0xBD) 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 SSD2805C will send out the correct packet type according to the **TDC** (registers 0xBC and 0xBD) value.

After sending out the read packet, the SSD2805C will automatically perform a BTA to wait for the Read Response Packet from the MIPI slave. The return data will be stored in register **RR** (0xD7). 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 register **RR** (0xD7). The user can read the data out when the **RDR** (register 0xC6 bit 0) is set to 1. After seeing **RDR** (register 0xC6 bit 0) been set to 1, the user should first read register **RDCR** which contains 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 register **RR** (0xD7). After all the return data are read out, the **RDR** (register 0xC6 bit 0) will be set to 0 by the SSD2805C.

After the **RDR** (register 0xC6 bit 0) been set to 1, the user can choose not to read the data out from register **RR** (0xD7). The user can continue performing another operation. Once the user does so, the **RDR** (register 0xC6 bit 0) will be set to 0 by the SSD2805C.

There might be Acknowledge and Error Report Packet sent by the MIPI slave at the same time. The operation of acknowledgement handling is described in 3.5.

Under certain circumstance, the MIPI slave might only send back Acknowledge and Error Report Packet without any data. Thus, the **RDR** (register 0xC6 bit 0) will not be set. Therefore, it is recommended that the user check the bit **BTAR** (register 0xC6 bit 2) first. The **BTAR** (register 0xC6 bit

2) is to indicate whether the MIPI slave has passed the bus authority back to the SSD2805C or not. Only when the **BTAR** (register 0xC6 bit 2) is 1, there might be return data. If there is no return data, the user should follow 3.5 to handle the acknowledgement.

During readback from the slave, the SSD2805C will initiate a bus-turn-around protocol, i.e. it will drive the BTA sequence and release the bus later.

The slave take control of the bus and send the data back to SSD2805C, and then initiate the same BTA sequence again to pass the bus back to master.

For each read operation, there are 2 BTA sequences, one from SSD2805C (dark blue line from Figure 3-3) and one from slave (light blue line from Figure 3-3). There is a period where both side drive 0, which is the pink region of Figure 3-3.

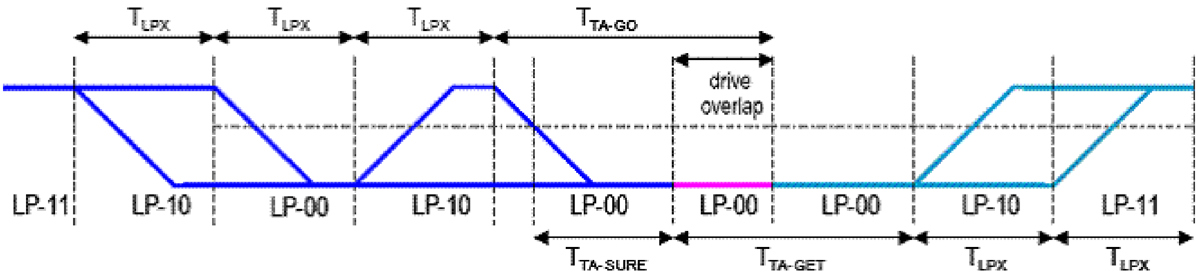

Figure 3-3 : Turnaround Procedure

User should check with the TLPX of slave spec to make sure the TLPX of SSD2805C close to the slave such that the SSD2805C will never drive beyond time when slave start driving.

3.5 Acknowledgement Operation

The SSD2805C 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** (register 0xC4 bit 0) to 1 and performing a write operation, or just performing a read operation. After the MIPI slave passes the bus authority back, the SSD2805C will set bit **BTAR** (register 0xC6 bit 2) 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, SSD2805C will set bit **ARR** (register 0xC6 bit 3) and **ATR** (register 0xC6 bit 4) to 1. **ARR** (register 0xC6 bit 3) indicates that response has been received from MIPI slave. **ATR** (register 0xC6 bit 4) indicates that the MIPI slave has reported no error with ACK trigger message. Consequently, the register **ARSR** (0xC3) 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, SSD2805C will set bit **ARR** (register 0xC6 bit 3) to 1 and **ATR** (register 0xC6 bit 4) to 0. **ARR** (register 0xC6 bit 3) indicates that response has been received from MIPI slave. **ATR** (register 0xC6 bit 4) 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** (0xC3). The user can read this register to see what error the MIPI slave has encountered.

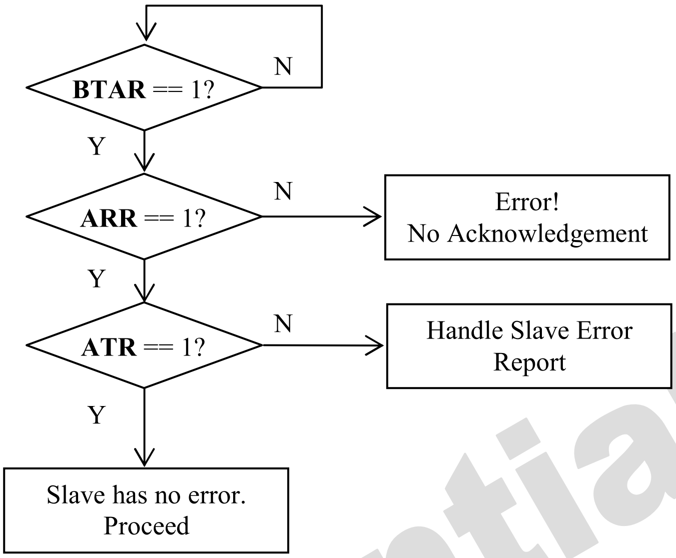

Figure 3-4: Acknowledgement Handling after Non-Read Command

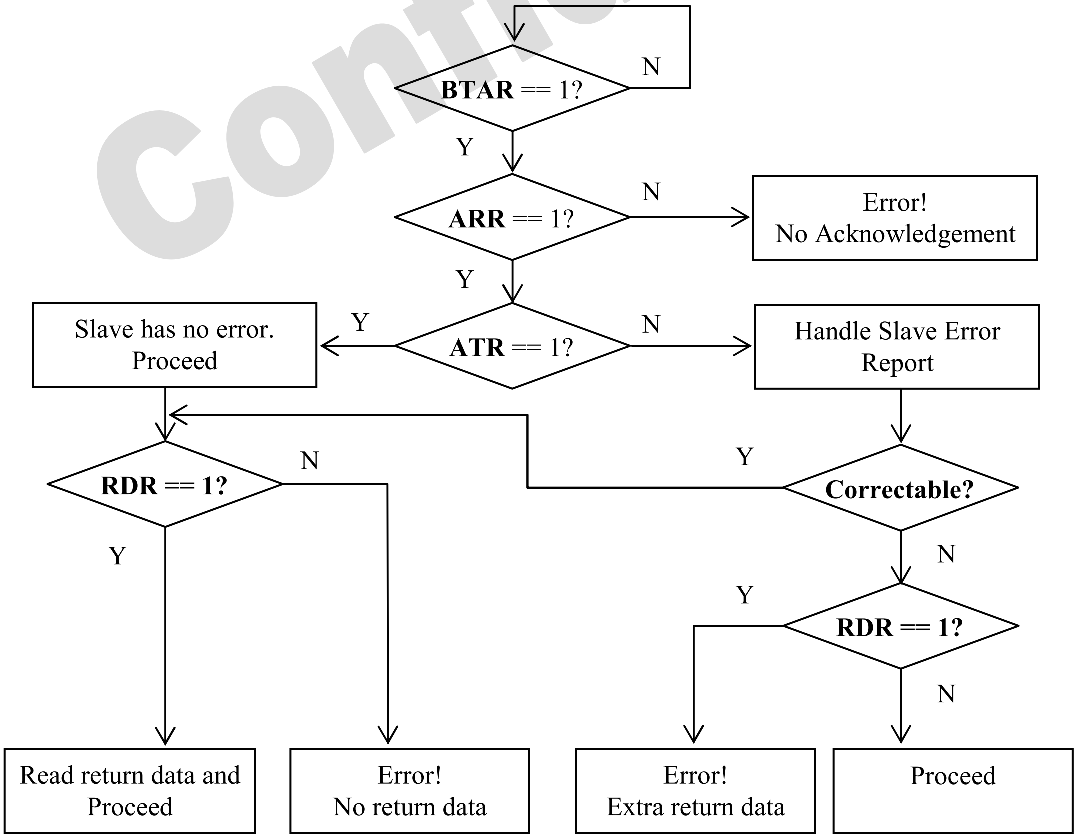

Figure 3-5: Acknowledgement Handling after Read Command

3.6 Tearing Effect (TE) Operation

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 SSD2805C by using BTA trigger message.

The TE operation can be enabled by setting bit **FBT** (register 0xC4 bit 1) and **FBW** (register 0xC4 bit 0) to 1 before writing the last command to the MIPI slave. Afterwards, the host processor can instruct the SSD2805C to send out the last command in a write packet. Since **FBW** (register 0xC4 bit 0) is 1, the SSD2805C will automatically perform a BTA after the write operation. The MIPI slave will response and pass the bus authority back. Since **FBT** (register 0xC4 bit 1) is 1, the SSD2805C 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 SSD2805C will set the TE pin to 1 to indicate that TE event has been received. DATA[16] is used as the TE pin. At the same time, bit **TER** (register 0xD3 bit 0) will be set to 1. The host processor can write 1 to this bit to clear it. Once the **TER** (register 0xD3 bit 0) is cleared, the TE signal will be set to 0.

If the MIPI slave does not send back the TE trigger message but just perform a BTA to pass the bus back, the SSD2805C 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** (register 0xC4 bit 1) 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** (register 0xC4 bit 2) to 1 to force a bus contention. After bus contention is resolved, the slave will pass the bus back to SSD2805C.

3.7 Contention Detection and Timer Operation

Two timers have been defined in SSD2805C to resolve the potential contention issue on the bus. The two timers are the HS TX timer and LP RX timer. Please see the Section 1.31 and 1.32 for details.

Whenever the SSD2805C 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.

3.8 Interrupt Operation

In SSD2805C, an interrupt signal INT# is provided to interrupt the host processor. This signal is an active low signal. When an interrupt event occurs, the signal will go low.

In SSD2805C, a number of interrupt sources can be mapped to the interrupt signal. User can control the mapping by setting the appropriate field in the **Interrupt Control Register ICR** (0xC5). When interrupt event occurs, user can determine what event has happened by reading the **Interrupt Status Register ISR** (0xC6).

3.8.1 RDR

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

3.8.2 BTAR

To indicate whether the SSD2805C has the bus authority or not. It can be used after SSD2805C makes a BTA. If the MIPI slave has returned the bus authority back to SSD2805C, the interrupt will be set to indicate so. Please note that, on power up, the bus authority is already on the SSD2805C. Hence, the SSD2805C will show that it has the bus authority.

3.8.3 ARR

To indicate whether the SSD2805C 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** (register 0xC6 bit 4)**.**
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. Please refer to 3.4 and 3.5 for the details.

3.8.4 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. Please refer to Clock and Reset Module for more details. 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** (register 0xC6 bit 7) interrupt is enabled. This is to let the user determine the programming speed before configuring the SSD2805C.

3.8.5 LPTO

To indicate that there is LP RX time out.

3.8.6 HSTO

To indicate that there is HS TX time out.

3.8.7 PO

To indicate whether the SSD2805C is ready to accept any data from the user. The SSD2805C 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 SSD2805C, 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** (registers 0xBC, 0xBD), **PST** (register 0xBE bit 11-0), and **DCS** (register 0xB7 bit 6) fields. The SSD2805C 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.

3.8.8 LSE, LSA, LLE, LLA, MSE, MSA, MLE, MLA

All these interrupts 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 SSD2805C. Otherwise, it is recommended to use the **PO** interrupt. Please refer to 3.9 for the details.

3.8.9 Interrupt Latency

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 SSD2805C. 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** (registers 0xBC, 0xBD) field, the interrupt will take a short period to reflect the correct **PO** (registers 0xC6 bit 1) 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 SSD2805C 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.

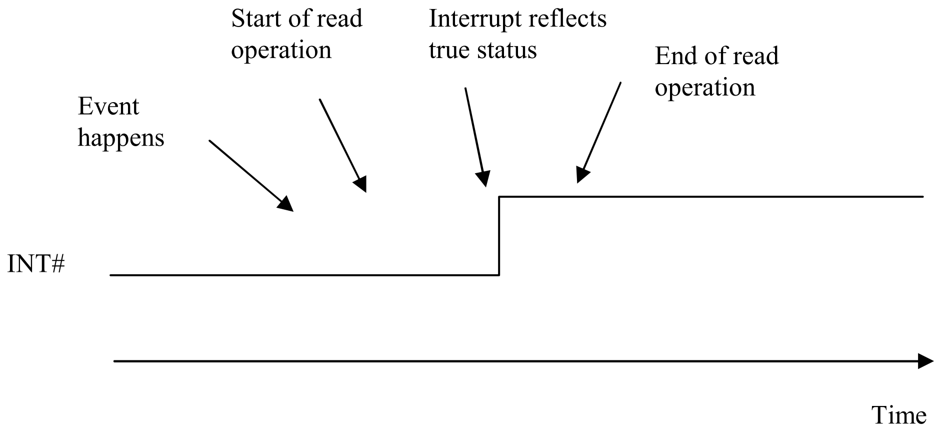

Figure 3-6: Illustration of Interrupt Latency

3.9 Internal Buffer Status

There are totally 4 data buffers inside the SSD2805C, which are MCU interface long buffer (ML), MCU interface short buffer (MS), SPI interface long buffer (LL) and SPI interface short buffer (LS).

The ML and MS buffers are used to store the data written through MCU interface when the IF_SEL is 1. They are used to store the data written through RGB interface when the IF_SEL is 0. However, since there is no flow control for the RGB interface, the status is only valid for MCU interface.

The LL and LS buffers are used to store the data written through SPI interface when the IF_SEL is 0. When the IF_SEL is 1, they are not used.

For MS and LS buffers, any packet with payload of no more than 8 bytes will be stored into them. They can store 16 and 8 such packets respectively. 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 (TDC<=8 bytes)
- DCS Long Write Packet (TDC<=7 bytes. There is one byte for DCS command.)

These two buffers are used to hold the packets which are usually used to configure the MIPI slave or the display driver.

For ML and LL buffers, any packet with payload of more than 8 bytes will be stored into them. They can store 2 such packets. The maximum size of ML and LL buffer is 2048 bytes and 320 bytes respectively. They are used to hold

- Generic Long Write Packet
- DCS Long Write Packet

In case of automatic partitioning, the packet length is determined by the **PST** (register 0xBE bit 11-0) field. It is not recommended to make the **PST** (register 0xBE bit 11-0) field so small that the packet will be put in MS or LS buffers, as it is a waste.

When the IF_SEL is 0, the user can write the data through SPI interface. With the information above, the user knows whether the next packet will be written into LL or LS buffers. Hence, the user needs to check the corresponding interrupts. The usage of the interrupts is listed below.

3.9.1 LSE

To indicate that the LS buffer is empty. Since the LS buffer can hold 8 packets, the user can write up to 8 such packets into LS buffer without needing to look at the interrupt status.

3.9.2 LSA

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

3.9.3 LLE

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

3.9.4 LLA

To indicate that the LL buffer can hold at least 1 more packet. The user can write 1 such packet into LL buffer.
When the IF_SEL is 1, the user can write the data through MCU interface. With the information above, the user knows whether the next packet will be written into ML or MS buffers. Hence, the user needs to check the corresponding interrupts. The usage of the interrupts is listed below.

3.9.5 MSE

To indicate that the MS buffer is empty. Since the MS buffer can hold 16 packets, the user can write up to 16 such packets into MS buffer without needing to look at the interrupt status.

3.9.6 MSA

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

3.9.7 MLE

To indicate that ML 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.

3.9.8 MLA

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

The 8 interrupts mentioned here can be used as flow control between the host processor and the SSD2805C. However, it requires the user to know the buffer operation well. The **PO** (register 0xC6 bit 1) 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.

3.10 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 host processor or the received data. Once a complete packet is written into the SSD2805C buffer, it will send it out through the serial link. The user can write 1 to bit **COP** at any time to cancel all the current operations. Please see 1.16 for the description.

When the SSD2805C 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 host 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 SSD2805C is in low power mode, the serial link is mainly used to send command and configuration data. If there is no data to sent, the SSD2805C will be idle in LP TX stop mode.

The user can also enter Ultra Low Power mode by writing 1 to **SLP**. Once the **SLP** is set to 1, the SSD2805C will automatically enter LP mode. If the **HS** is 1, the SSD2805C will clear the **HS** t to 0 and switch from HS to LP mode. Afterwards, the SSD2805C will issue ULPS trigger message to the MIPI slave to enter Ultra Low Power State. During this state, the clock to SSD2805C can be switched off such that the SSD2805C 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 (pleaser refer to MIPI DPHY specification). Hence, the user cannot perform any data transmission before the system exits from ULPS.

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 **RDR** to 1 to indicate that the SSD2805C is ready for read. The total number of received bytes will also be stored in **RDCR**.

After the reception is completed, the SSD2805C 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 mode.

3.11 PLL

The PLL output frequency is calculated by the equations below,

$$f_{pre} = \frac{f_{ref}}{N}$$

$$f_{VCO} = f_{pre} \times M$$

$$f_{out} = \frac{f_{VCO}}{P}$$

where the f_{ref} is the input reference clock frequency and f_{out} is the output clock frequency.

The clock frequencies need to satisfy the constraint below.

$$f_{pre} \geq 5MHz$$

$$500MHz \geq f_{VCO} \geq 225MHz$$

The value of N, M, and P are controlled in the register **PLCR**.

All the values of N, M and P can only be modified when the PLL is turned off. Hence, the sequence for modification is to turn off PLL, modify register value, turn on PLL.

4 External Interface

The SSD2805C supports three types of MCU interface,

- 6800 (Fixed E mode)
- 6800 (Clocked E mode)
- 8080

three types of SPI interface,

- 8 bit 3 wire
- 8 bit 4 wire
- 24 bit 3 wire

and RGB interfaces.

The selection is controlled by PS[3:0], IF_SEL and IF_MUX pins.

MCU interface supports both 8 bit and 16 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 bit interface, the lease significant word should be written first.

- DATA[7:0] for 8 bit interface.
- DATA[15:0] for 16 bit interface.

RGB interface supports 4 bpp settings. Below are the data pins used for each interface. For all cases, R should be at the upper bits and B should be at the lower bits.

- DATA[15:0] for 16 bpp.
- DATA[17:0] for 18 bpp, packed.
- DATA[17:0] for 18 bpp, loosely packed.
- DATA[24:0] for 24 bpp.

SPI interface supports 8 bit data bus. The least significant byte should be written first.

Below is the operation and timing diagram for each of the interfaces.

4.1 MCU 6800 (Fixed E mode) Interface

This interface consists of DATA[15:0], RWX, DCX, E and CSX. It supports both 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 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[15:0] are sampled at the rising edge of CSX. During read operation, DATA[15:0] are provided at the falling edge of CSX and the host processor should use the rising edge of CSX to sample.

Below is a diagram for illustration. Please see AC Characteristics of data sheet for the detailed waveform and timing parameters.

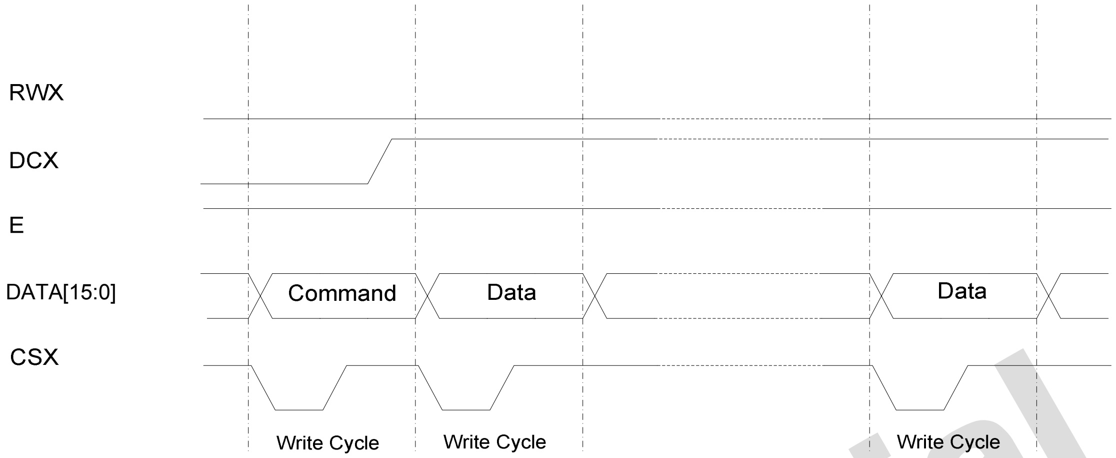

Figure 4-1: Illustration of Write Operation for MCU 6800 (Fixed E mode) Interface

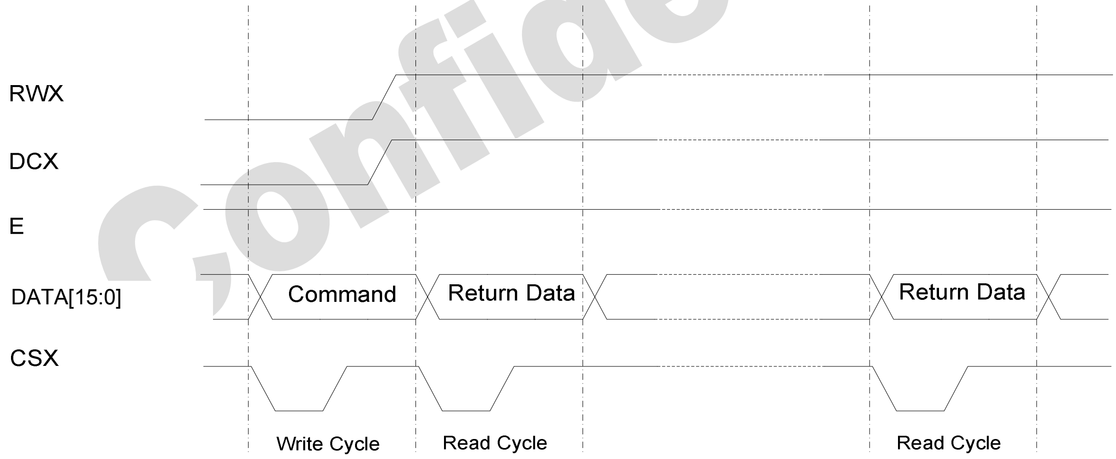

Figure 4-2: Illustration of Read Operation for MCU 6800 (Fixed E mode) Interface

In the first write cycle, only 8 bit data are written into the SSD2805C, as the command can only be 8 bit. No matter whether the interface is 8, or 16 bit, lower 8 bits are used. Please refer to the table below.

Table 4-1: MCU Interface Data Pin Mapping for Command Cycle

Interface	D23	D22	D21	D20	D19	D18	D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
16 bit									x	x	x	x	x	x	x	x	C7	C6	C5	C4	C3	C2	C1	C0
8 bit																	C7	C6	C5	C4	C3	C2	C1	C0

In the sub sequent read or write cycles, command parameters can be written into the SSD2805C. Depending on the interface width, different data pin mapping is adopted. Please refer to the table below. When the parameter is odd number of bytes and interface width is 16 bit, the last byte should be put on DATA[7:0].

Table 4-2: MCU Interface Data Pin Mapping for Parameter Cycle

Interface	Cycle	D23	D22	D21	D20	D19	D18	D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
16 bit	1st									Parameter [15:0]															
	2nd									…															
8 bit	1st																	Parameter [7:0]							
	2nd																	Parameter [15:8]							
	3rd																	…							

For certain DCS commands, the data is for LCD display. The data can be optionally written in pixel format instead of raw format. Please refer to **DFR** description for more details. In this case, the data pin mapping is given in the table below.

Table 4-3: MCU Interface Data Pin Mapping for Display Data Pixel Format

Interface	Color mode	Cycle	D23	D22	D21	D20	D19	D18	D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
16 bit	16M type1	1st									G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0
		2nd									x	x	x	x	x	x	x	x	R7	R6	R5	R4	R3	R2	R1	R0
	16M type2	1st									G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0
		2nd									R7	R6	R5	R4	R3	R2	R1	R0	x	x	x	x	x	x	x	x
	16M type3	1st									G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0
		2nd									B7	B6	B5	B4	B3	B2	B1	B0	R7	R6	R5	R4	R3	R2	R1	R0
		3rd									R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0
	262k, type1	1st									G5	G4	G3	G2	G1	G0	x	x	B5	B4	B3	B2	B1	B0	x	x
		2nd									x	x	x	x	x	x	x	x	R5	R4	R3	R2	R1	R0	x	x
	262k, type2	1st									G5	G4	G3	G2	G1	G0	x	x	B5	B4	B3	B2	B1	B0	x	x
		2nd									R5	R4	R3	R2	R1	R0	x	x	x	x	x	x	x	x	x	x
	262k, type3	1st									G5	G4	G3	G2	G1	G0	x	x	B5	B4	B3	B2	B1	B0	x	x
		2nd									B5	B4	B3	B2	B1	B0	x	x	R5	R4	R3	R2	R1	R0	x	x
		3rd									R5	R4	R3	R2	R1	R0	x	x	G5	G4	G3	G2	G1	G0	x	x
	64k										R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0

8 bit	16M	1st																	B7	B6	B5	B4	B3	B2	B1	B0
		2nd																	G7	G6	G5	G4	G3	G2	G1	G0
		3rd																	R7	R6	R5	R4	R3	R2	R1	R0
	262k	1st																	B5	B4	B3	B2	B1	B0	x	x
		2nd																	G5	G4	G3	G2	G1	G0	x	x
		3rd																	R5	R4	R3	R2	R1	R0	x	x
	64k	1st																	G2	G1	G0	B4	B3	B2	B1	B0
		2nd																	R4	R3	R2	R1	R0	G5	G4	G3

4.2 MCU 6800 (Clocked E mode) Interface

This interface consists of DATA[15:0], RWX, DCX, E and CSX. It supports both 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[15:0] are sampled at the falling edge of E. During read operation, DATA[15:0] are provided at the rising edge of E and the host processor should use the falling edge of E to sample.

Below is a diagram for illustration. Please refer to the AC Characteristics of data sheet for the detailed waveform and timing parameters.

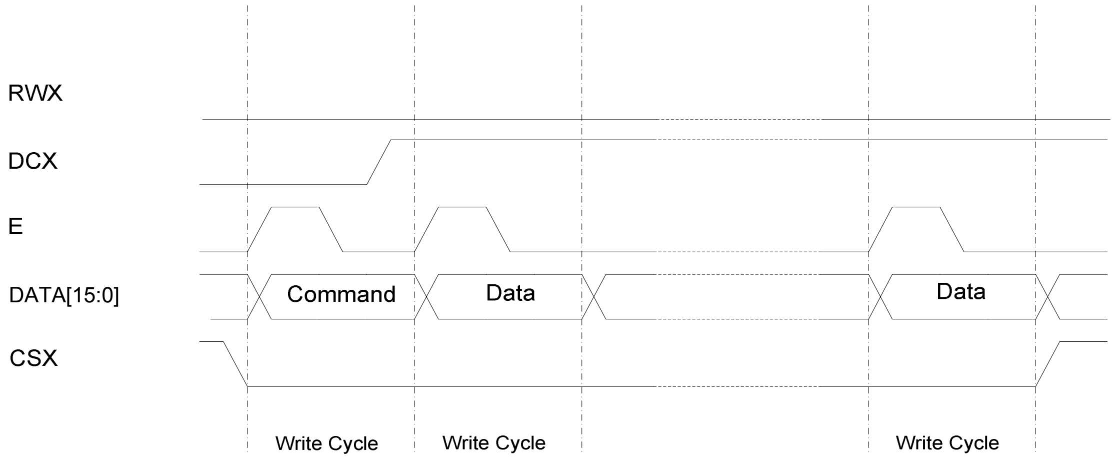

Figure 4-3: Illustration of Write Operation for MCU (Clocked E mode) 6800 Interface

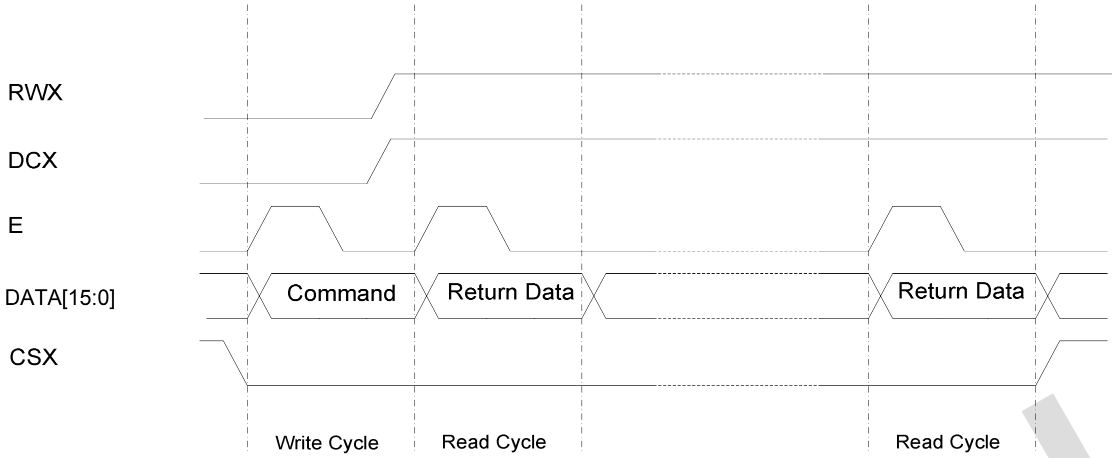

Figure 4-4: Illustration of Read Operation for MCU 6800 (Clocked E mode) Interface

4.3 MCU 8080 Interface

This interface consists of DATA[15:0], RDX, WRX, DCX, and CSX. It supports both 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[15:0] are sampled at the rising edge of WRX. During read operation, DATA[15:0] are provided at the falling edge of RDX and the host processor should use the rising edge of RDX to sample.

Below is a diagram for illustration. Please refer to the AC Characteristics of data sheet for the detailed waveform and timing parameters.

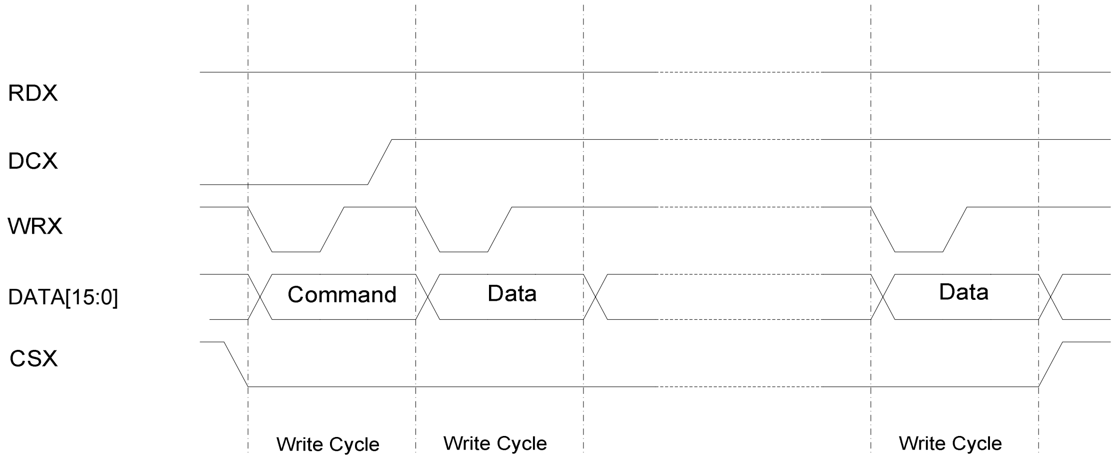

Figure 4-5: Illustration of Write Operation for MCU 8080 Interface

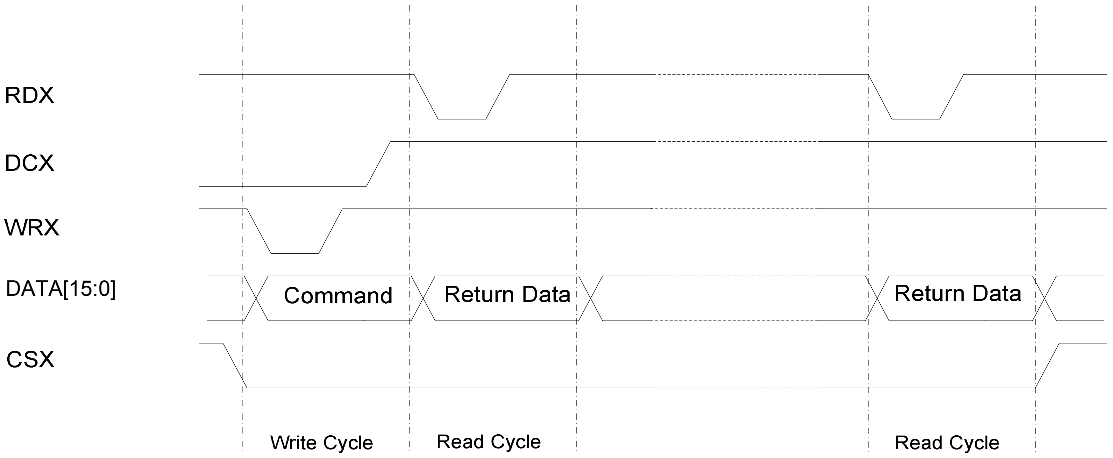

Figure 4-6: Illustration of Read Operation for MCU 8080 Interface

4.4 SPI Interface 8 bit 4 Wire

This interface consists of SD/C#, SCK, DIN, DOUT and SCSX. 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 SCSX should be driven from 1 to 0 to start an operation and from 0 to 1 to end an operation. During 1 operation, the host processor can write or read multiple bytes.

SD/C# indicates whether the operation is for data or command. When SD/C# is 1, the operation is for data. When SD/C# is 0, the operation is for command. SD/C# is sampled at every 8th rising edge of SCK during 1 operation.

During write operation, DIN will be sampled by SSD2805C at the rising edge of SCK. The first rising edge of SCK after the falling edge of SCSX 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 SD/C# is sampled at the 8th rising edge of SCK, together with bit 0 of the 8 bit data. Please see the diagram below for illustration. Optionally, the SCSX can be driven to 1 in between cycles.

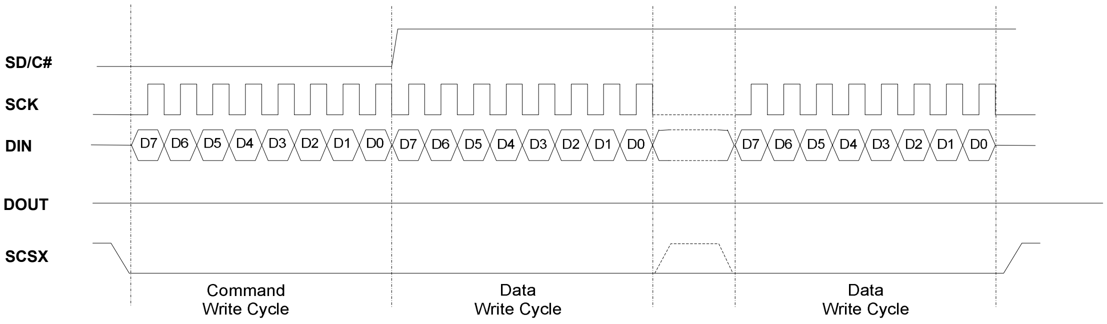
SD/C#
SCK
DIN
D7 D6 D5 D4 D3 D2 D1 D0 D7 D6 D5 D4 D3 D2 D1 D0 D7 D6 D5 D4 D3 D2 D1 D0
DOUT
SCSX
Command Write Cycle
Data Write Cycle
Data Write Cycle

Figure 4-7: Illustration of Write Operation for 8 bit 4 Wire Interface

During read operation, since there is no RWX signal to indicate whether the operation is read or write, the read operation for SPI interface needs to be handled differently from the MCU interface. After SCSX is driven low, the first cycle is always a command write cycle, which specifies the register to access. The second cycle is still a command write cycle. If the command in this cycle matches the command in register **LRR**, the SPI interface will enter read mode. The subsequent cycles will be read cycles. If the command does not match, the SPI interface will remain in write mode.

After entering the read mode, the return data is provided on DOUT, on the falling edge of SCK. The host processor should use the rising edge of SCK to sample the data. SD/C# should be driven to 1 during the read cycles. Please see the diagram below for illustration. Optionally, the SCSX can be driven to 1 in between cycles.

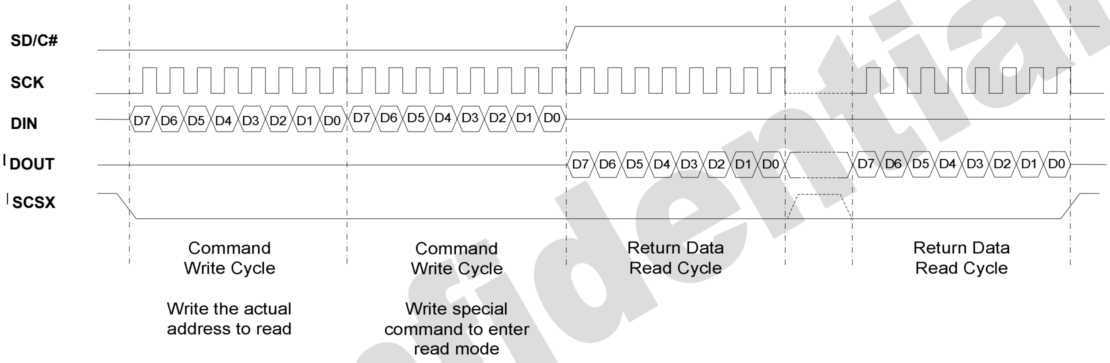
SD/C#
SCK
DIN
DOUT
SCSX
D7 D6 D5 D4 D3 D2 D1 D0 D7 D6 D5 D4 D3 D2 D1 D0
D7 D6 D5 D4 D3 D2 D1 D0 D7 D6 D5 D4 D3 D2 D1 D0
Command Write Cycle
Command Write Cycle
Return Data Read Cycle
Return Data Read Cycle
Write the actual address to read
Write special command to enter read mode

Figure 4-8: Illustration of Read Operation for 8 bit 4 Wire Interface

Please refer to the AC Characteristics of datasheet for the detailed waveform and timing diagrams.

4.5 SPI Interface 8 bit 3 Wire

This interface consists of SCK, DIN, DOUT and SCSX. 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 SCSX should be driven from 1 to 0 to start an operation and from 0 to 1 to end an operation. During 1 operation, the host processor can write or read multiple bytes.

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

During write operation, DIN will be sampled by SSD2805C at the rising edge of SCK. The first rising edge of SCK after the falling edge of SCSX samples the SDC 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 SCSX can be driven to 1 in between cycles.

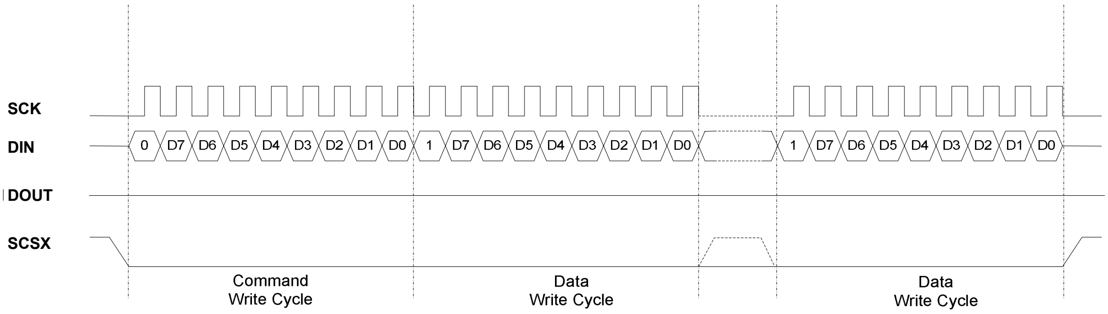

Figure 4-9: Illustration of Write Operation for 8 bit 3 Wire Interface

During read operation, since there is no RWX signal to indicate whether the operation is read or write, the read operation for SPI interface needs to be handled differently from the MCU interface. After SCSX is driven low, the first cycle is always a command write cycle, which specifies the register to access. The second cycle is still a command write cycle. If the command in this cycle matches the command in register **LRR**, the SPI interface will enter read mode. The subsequent cycles will be read cycles. If the command does not match, the SPI interface will remain in write mode.

After entering the read mode, the return data is provided on DOUT, on the falling edge of SCK. The host processor should use the rising edge of SCK to sample the data. Please note that there is no SDC bit to read out from SSD2805C. Hence, each read cycle consists of 8 bits instead of 9 bits. This is the difference between read and write cycles. Please see the diagram below for illustration. Optionally, the SCSX can be driven to 1 in between cycles.

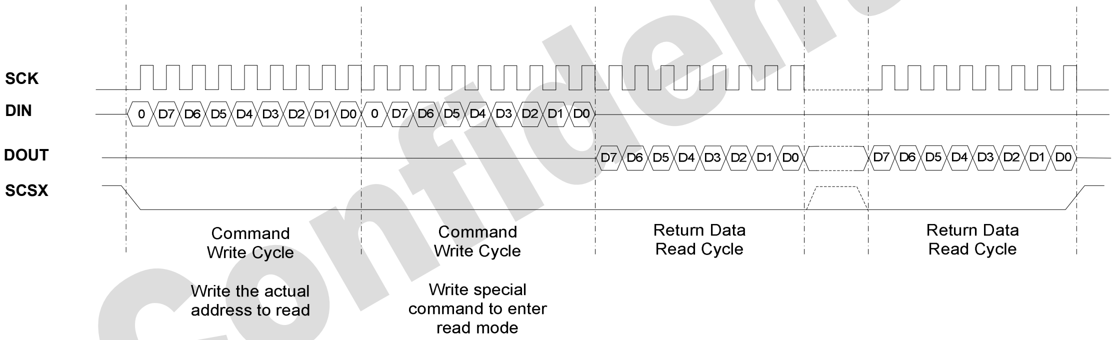

Figure 4-10: Illustration of Read Operation for 8 bit 3 Wire Interface

Please refer to the AC Characteristics of datasheet for the detailed waveform and timing diagrams.

4.6 SPI Interface 24 bit 3 Wire

This interface consists of SCK, DIN, DOUT and SCSX. 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 SCSX should be driven from 1 to 0 to start cycle and from 0 to 1 to end a cycle. During 1 operation, the host processor can have multiple write or read cycles. However, the SCSX 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 bit are the ID bit for SSD2805C, which must be 011100. If this field does not match, the cycle will not be taken in. The 7th bit is the SDC 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.

During write operation, DIN will be sampled by SSD2805C at the rising edge of SCK. Please see the diagram below for illustration. It is an example for writing data 0x1264 to register address 0x28.

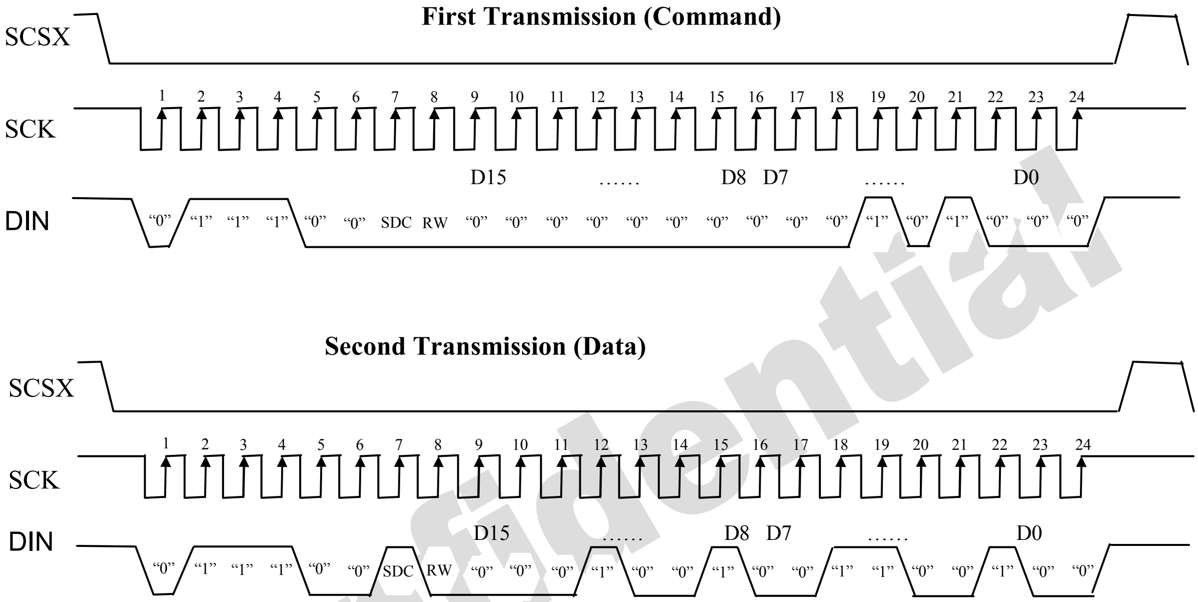
First Transmission (Command)
SCSX
SCK
DIN
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
D15 …… D8 D7 …… D0
"0" "1" "1" "1" "0" "0" SDC RW "0" "0" "0" "0" "0" "0" "0" "0" "0" "0" "1" "0" "1" "0" "0" "0"
Second Transmission (Data)
SCSX
SCK
DIN
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
D15 …… D8 D7 …… D0
"0" "1" "1" "1" "0" "0" SDC RW "0" "0" "0" "1" "0" "0" "1" "0" "0" "1" "1" "0" "0" "1" "0" "0"

Figure 4-11: Illustration of Write Operation for 24 bit 3 wire Interface

During read operation, the first 8 bit are still written by the host processor to specify whether the following 16 bit are for command or data. Afterwards, the SSD2805C will provide the return data on DOUT, on the falling edge of SCK. The host processor should use the rising edge of SCK to sample. Please see the diagram below for illustration.

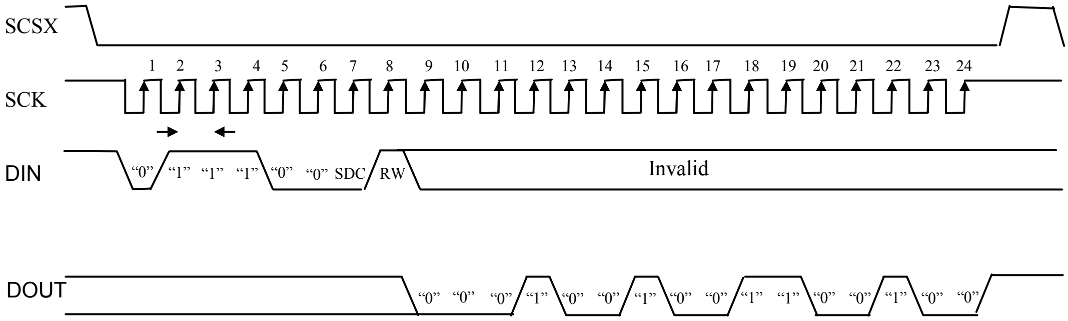

Figure 4-12: Illustration of Read Operation for 24 bit 3 Wire Interface

Please refer to the AC Characteristics of data sheet for the detailed waveform and timing diagrams.

4.7 RGB Interface

The RGB interface has already been illustrated in 3.1. Below is the pin mapping for different BPP setting. Please refer to the AC Characteristics of data sheet for detailed timing diagrams.

Table 4-4: RGB Interface Data Pin Mapping

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

5 Display Data Order

SSD2805C can output display data in different orders on the MIPI lane. Two register fields **END** (register 0xD6 bit 1) and **CO** (register 0xD6 bit 0) are provided for setting the display data order.

In Video Mode, **END** must be set to 0 and **CO** must be set to 1.

In Command Mode, the display data order are as shown below in different **END** and **CO** settings Please note that in each byte of the data, the LSB is sent first and the MSB is sent last.

- **END** = 1; **CO** = 0

16bpp

MSB		Byte 1					LSB	MSB		Byte2					LSB	MSB		Byte3					LSB	MSB		Byte4					LSB
R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0

18bpp

MSB		Byte 1					LSB	MSB		Byte2					LSB	MSB		Byte3					LSB	MSB		Byte4					LSB
R5	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B5	B4	B3	B2	B1	B0	R5	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B5	B4

24bpp

MSB		Byte 1					LSB	MSB		Byte2					LSB	MSB		Byte3					LSB	MSB		Byte4					LSB
R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0	R7	R6	R5	R4	R3	R2	R1	R0

- **END** = 0; **CO** = 0

16bpp

MSB		Byte 1					LSB	MSB		Byte2					LSB	MSB		Byte3					LSB	MSB		Byte4					LSB
G2	G1	G0	B4	B3	B2	B1	B0	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0	R4	R3	R2	R1	R0	G5	G4	G3

18bpp

MSB		Byte 1					LSB	MSB		Byte2					LSB	MSB		Byte3					LSB	MSB		Byte4					LSB
G1	G0	B5	B4	B3	B2	B1	B0	R3	R2	R1	R0	G5	G4	G3	G2	B5	B4	B3	B2	B1	B0	R5	R4	R1	R0	G5	G4	G3	G2	G1	G0

24bpp

MSB		Byte 1					LSB	MSB		Byte2					LSB	MSB		Byte3					LSB	MSB		Byte4					LSB
B7	B6	B5	B4	B3	B2	B1	B0	G7	G6	G5	G4	G3	G2	G1	G0	R7	R6	R5	R4	R3	R2	R1	R0	R7	R6	R5	R4	R3	R2	R1	R0

- **END** = 1; **CO** = 1

16bpp

MSB			Byte 1				LSB	MSB			Byte2				LSB	MSB			Byte3				LSB	MSB			Byte4				LSB
B4	B3	B2	B1	B0	G5	G4	G3	G2	G1	G0	R4	R3	R2	R1	R0	B4	B3	B2	B1	B0	G5	G4	G3	G2	G1	G0	R4	R3	R2	R1	R0

18bpp

MSB			Byte 1				LSB	MSB			Byte2				LSB	MSB			Byte3				LSB	MSB			Byte4				LSB
B5	B4	B3	B2	B1	B0	G5	G4	G3	G2	G1	G0	R5	R4	R3	R2	R1	R0	B5	B4	B3	B2	B1	B0	G5	G4	G3	G2	G1	G0	R5	R4

24bpp

MSB			Byte 1				LSB	MSB			Byte2				LSB	MSB			Byte3				LSB	MSB			Byte4				LSB
B7	B6	B5	B4	B3	B2	B1	B0	G7	G6	G5	G4	G3	G2	G1	G0	R7	R6	R5	R4	R3	R2	R1	R0	R7	R6	R5	R4	R3	R2	R1	R0

- **END** = 0; **CO** = 1

16bpp

MSB			Byte 1				LSB	MSB			Byte2				LSB	MSB			Byte3				LSB	MSB			Byte4				LSB
G2	G1	G0	R4	R3	R2	R1	R0	B4	B3	B2	B1	B0	G5	G4	G3	G2	G1	G0	R4	R3	R2	R1	R0	B4	B3	B2	B1	B0	G5	G4	G3

18bpp

MSB			Byte 1				LSB	MSB			Byte2				LSB	MSB			Byte3				LSB	MSB			Byte4				LSB
G1	G0	R5	R4	R3	R2	R1	R0	B3	B2	B1	B0	G5	G4	G3	G2	R5	R4	R3	R2	R1	R0	B5	B4	B1	B0	G5	G4	G3	G2	G1	G0

24bpp

MSB			Byte 1				LSB	MSB			Byte2				LSB	MSB			Byte3				LSB	MSB			Byte4				LSB
R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0	R7	R6	R5	R4	R3	R2	R1	R0

Solomon Systech reserves the right to make changes without notice to any products herein. Solomon Systech makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Solomon Systech assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any, and all, liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typical" must be validated for each customer application by the customer's technical experts. Solomon Systech does not convey any license under its patent rights nor the rights of others. Solomon Systech products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Solomon Systech product could create a situation where personal injury or death may occur. Should Buyer purchase or use Solomon Systech products for any such unintended or unauthorized application, Buyer shall indemnify and hold Solomon Systech and its offices, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Solomon Systech was negligent regarding the design or manufacture of the part.

All Solomon Systech Products complied with six (6) hazardous substances limitation requirement per European Union (EU) "Restriction of Hazardous Substance (RoHS) Directive (2002/95/EC)" and China standard "电子信息产品污染控制标识要求 (SJ/T11364-2006)" with control Marking Symbol . Hazardous Substances test report is available upon requested.

http://www.solomon-systech.com

Appendix: IC Revision history of SSD2805C Application Note

Version	Change Items	Effective Date
0.10	1st Release	15-Nov-05
0.20	The whole document information changed for SSD2805C	5-Aug-08
0.30	Added pin mapping for MCU and RGB interfaces.	28-Nov-08
0.40	Added 0xFA register table	16 Dec 08
0.50	Revised the name and description Bit 8 of register 0xD6	1 Apr 09
0.60	Updated section 3.4 Readback for the register setting of 0xCE	6 Jan 10
0.70	Updated the PLL multiplication for 2 lanes mode and corrected the maximum size of MS and LS to ML and LL buffer	02 Mar 10